Evolutionary Computation for Single and Multi-Objective Optimization
Dr. Deepak Sharma
Department of Mechanical Engineering
Indian Institute of Technology, Guwahati

Lecture - 15
Simulations of Constraint Handling Techniques

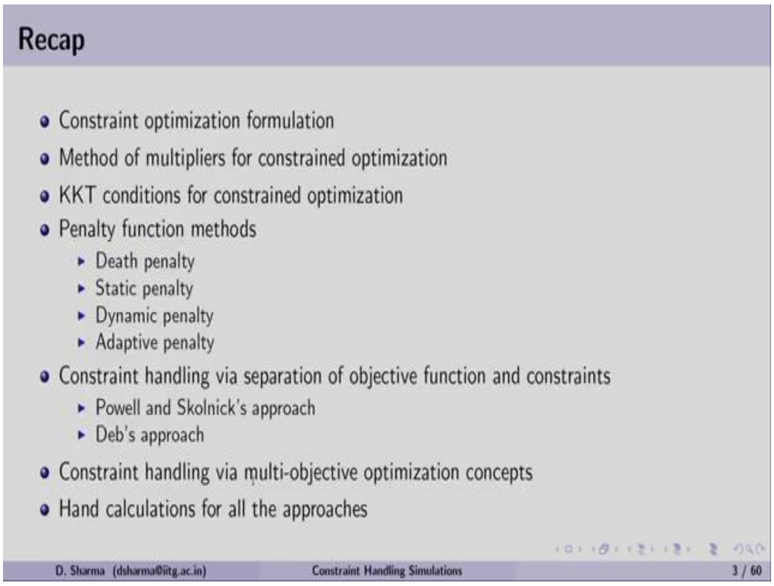
(Refer Slide Time: 00:42)

Welcome to the session on Simulation of Constraint Handling Techniques, in this particular session we will be showing you the simulation of 3 types of Constraint Handling Techniques. As we can see here we will be first going through the static penalty followed by the dynamic penalty and then we will also use Deb's Approach for handling the constraints.

These three constraint handling techniques will be tested on various kinds of a problem. As we can see here, we will be first look at the performance of these constraint handling techniques on the Constrained Himmelblau function followed by constrained Rosenbrock function, then we have bird function, then we have chosen two constraints functions such as g06 and g08 both of them are taken from the CEC 2006.

These are the competitions this is on the Congress on evolutionary computation conference 2006, these two functions we have taken. So, the first 5 functions will be the mathematical constraint optimization functions for which the optima is known to us. Thereafter we will show the performance of these techniques on the practical optimization problem such as milling process parameter optimization, welded beam design and two-bar truss design optimization.

(Refer Slide Time: 02:24)



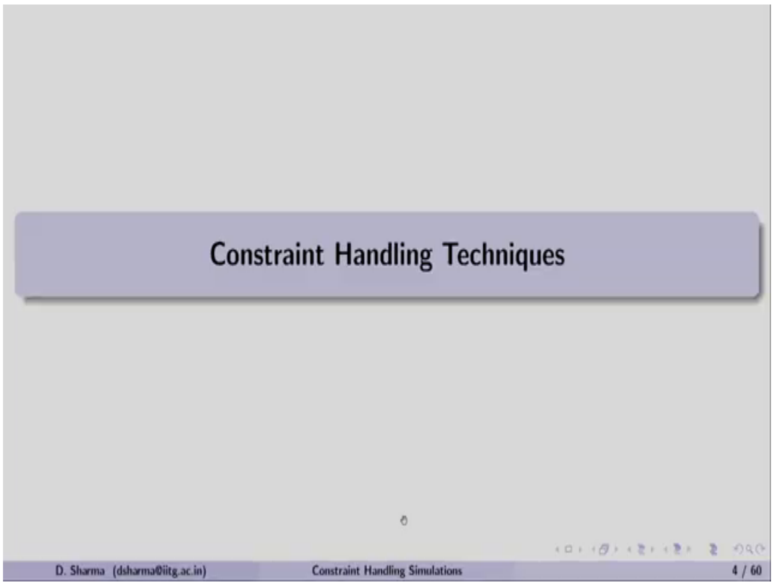
So, as a recap we have started with the constraint formulation, after that we understood the method of a multiplier from which we understood what could be the optimality condition for a constraint optimization problem.

We discussed the KKT conditions for equality and inequality constraints combined together with the objective function. Thereafter we discussed about the penalty function methods, we started with the death penalty, static penalty, dynamic penalty and adaptive penalty, so these methods we discuss.

Thereafter we discuss another way to handle the constraints that is separation of objective function and constraint. In this particular category we have gone through Powell and Skolnick's approach followed by Deb's approach.

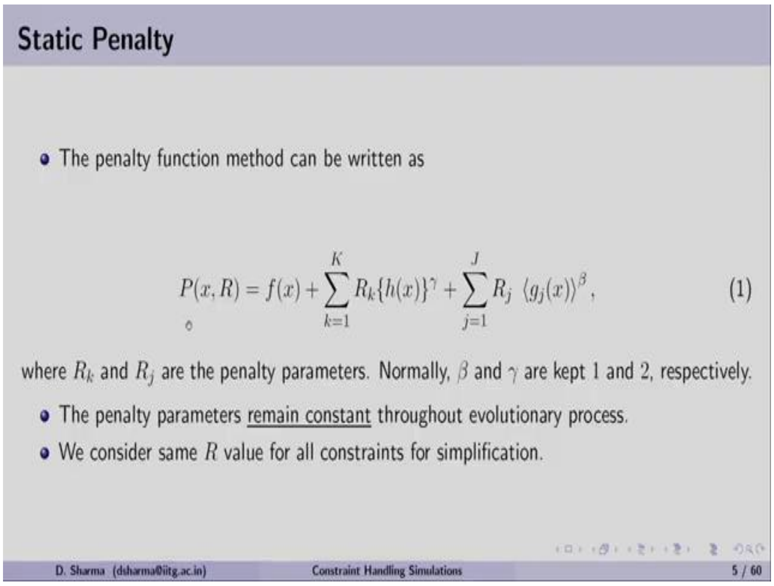
At the last we also understood that the constraint handling technique can also be done via multi objective optimization concept. For all of the above methods we have done the hand calculation for one generation. So, that we can understand how these constraint handling techniques work.

(Refer Slide Time: 03:58)



So, before we start our simulation, let us recap, let us look at the penalised function or the fitness function of the penalty of the constraint handling techniques which we have considered for the simulation.

(Refer Slide Time: 04:17)



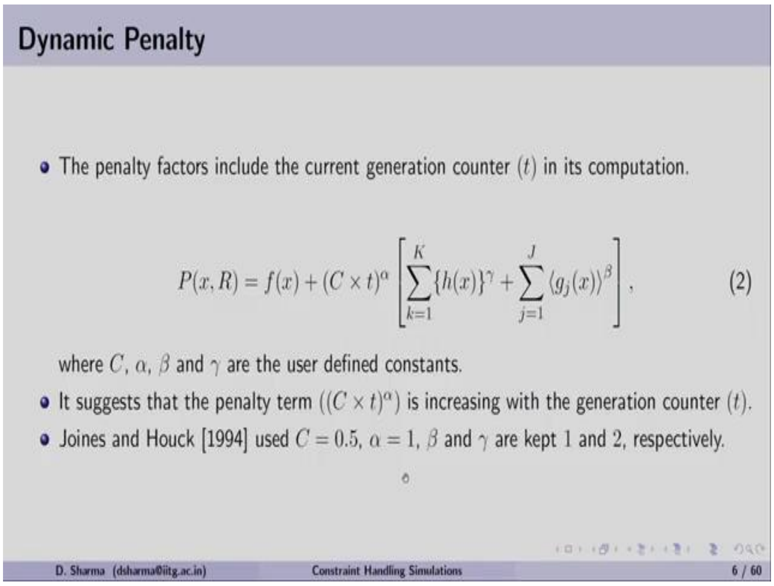
$$P(x, R) = f(x) + \sum_{k=1}^{K} R_k \{h(x)\}^{\gamma} + \sum_{j=1}^{J} R_j < g_j(x) >^{\beta},$$

We will start with the Static Penalty function method, as we can see in equation number 1, the penalised function is given as the way it is given as the objective function plus the constraint violation on the equality constraint, that is h x and the inequality constraint that is represented as g j of x.

Now in this particular penalty function method R k and R j values are different and that is keep on changing. However, when we talk about the static penalty function method, we know the penalty parameter that is R k and R j that remains constant throughout the evolutionary process. In this simulation, we consider the same value of R for all constraint for simplification.

So for example, if we have two constraints in our problem then we are keeping the same value of R for both the constraints and this value of R will remain same throughout the process.

(Refer Slide Time: 05:27)

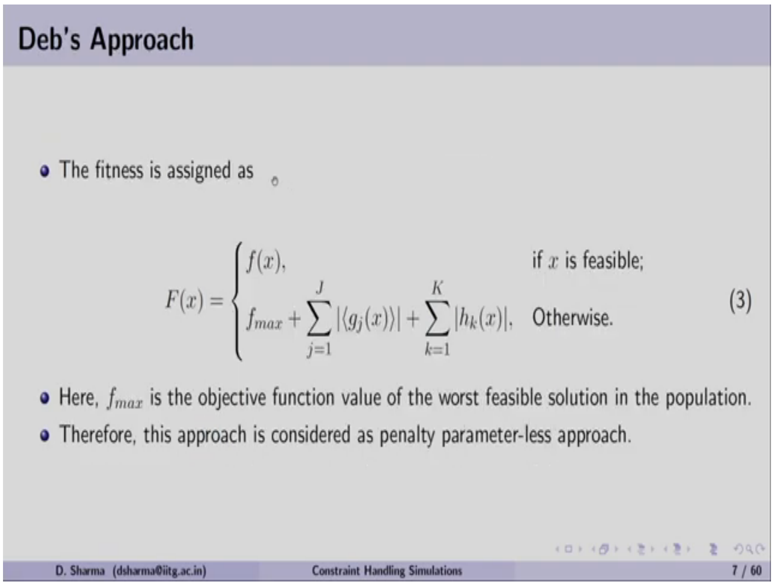


$$P(x,R) = f(x) + (C \times t)^{\alpha} \left[\sum_{k=1}^{K} \{h(x)\}^{\gamma} + \sum_{j=1}^{J} < g_j(x) >^{\beta} \right],$$

The next technique that we have taken is the dynamic penalty, as we can see in equation number 2 the penalised function is given as the objective function plus now you can see here that we have a constant C that is multiplied by t, that t is represented by the generation counter. So, this particular constant that will be increasing with the generation counter.

So, when t equals to 1 the value is small and as and when the t value is increasing this means that we are penalising the objective function more if there is a constraint violation. As we can see in equation number 2, in the inside the big bracket we have the constraint violation that will be multiplied by C into t times of alpha. We are considering C as 0.5 alpha as 1 and beta and gamma are kept 1 and 2 respectively for our simulation.

(Refer Slide Time: 06:41)



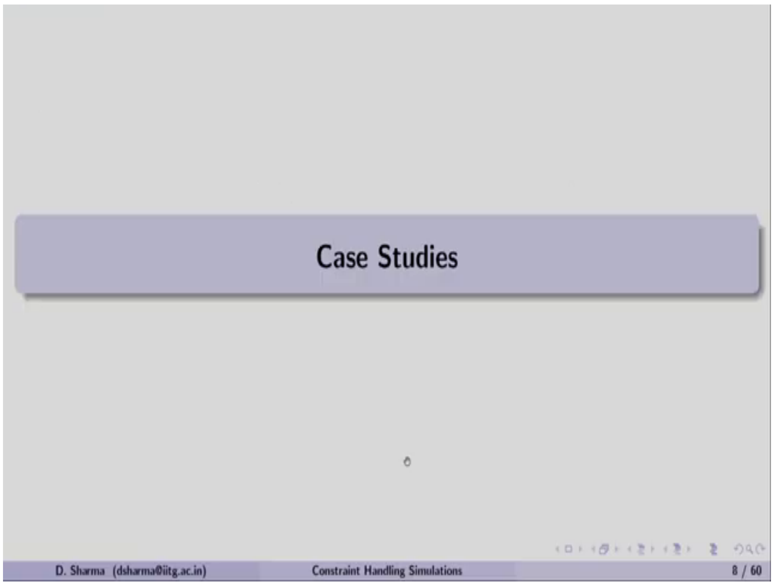
$$F(x) = \begin{cases} f(x), & \text{if x is feasible} \\ f_{max} + \sum_{j=1}^{J} |\langle g_j(x) \rangle| + \sum_{k=1}^{K} |h_k(x)|, & \text{otherwise} \end{cases}$$

Thereafter, we will be performing the simulation using Deb's approach in which the fitness of a solution say x is found as. So, if x is going to be feasible then the fitness of the solution is same as the objective function, if the solution is infeasible then we will be having f max plus the constraint violation for the inequality as well as equality constraint.

Now here f max is the objective function value of the worst feasible solution in the population. As we remember that well while we performed the hand calculation for Deb's approach. So, because of this f max which is the function value of the worst feasible solution in the population that ensures that the infeasible solution is having worse fitness than every feasible solution in the current population.

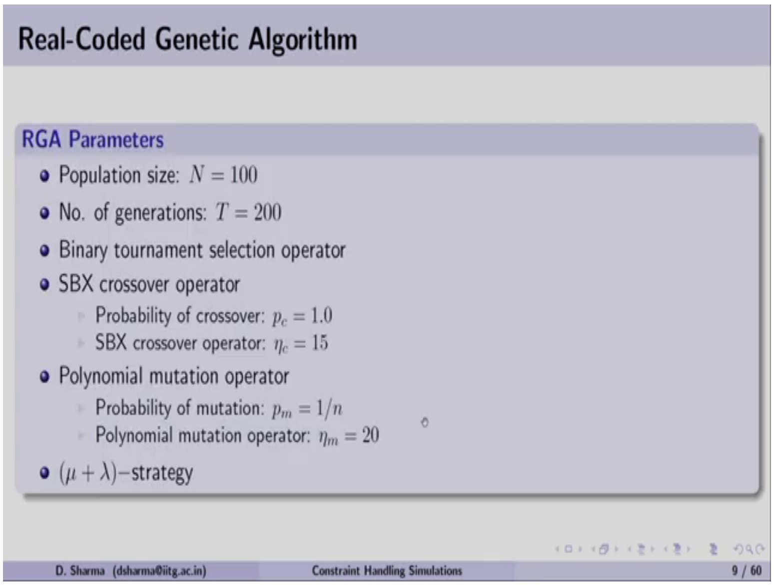
From the equation number 3 we can see that since there is no penalty term here, so this approach is also considered as penalty parameter less approach.

(Refer Slide Time: 08:05)



So, let us begin with the different kind of case studies as we discussed earlier.

(Refer Slide Time: 08:11)

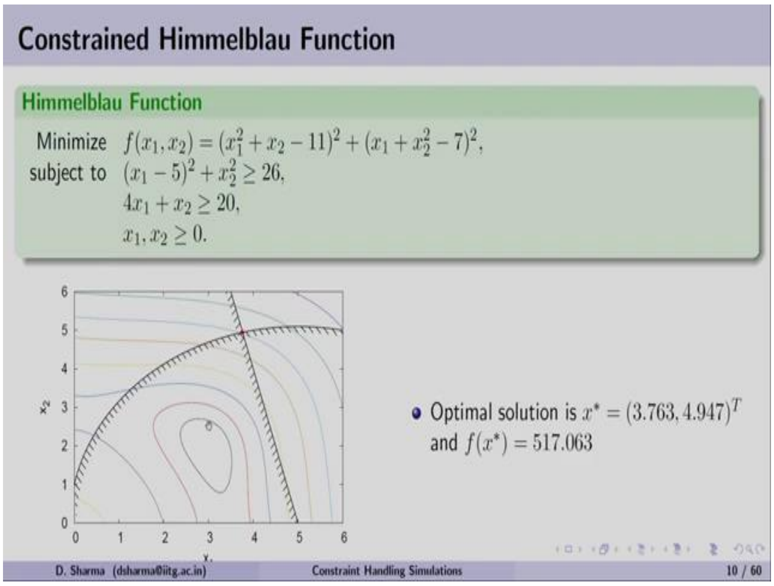


So, here we are taking the real coded genetic algorithm as our base algorithm, we will be coupling all the 3 kinds of constraint handling technique independently to run the RGA or real code genetic algorithm we have to fix the various parameters. So, for all the problems we have taken population size of 100 number of generation 200, we use binary tournament selection we used SBX crossover operator.

So, in order to perform crossover operator the probability of crossover is 1 and the there is a user defined parameter called eta c that is kept at 15, polynomial mutation is used for mutating.

The solution the probability of mutation is kept as 1 by n and as we remember n is the number of variable in a given problem. Polynomial mutation having the user defined parameter called eta m and the value is 20, we use mu plus lambda strategy with real coded genetic algorithm.

(Refer Slide Time: 09:27)



$$Minimize \quad f(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$$

$$subject\ to \quad (x_1 - 5)^2 + x_2^2 \geq 26,$$

$$4x_1 + x_2 \geq 20,$$

$$x_1, x_2 \geq 0.$$

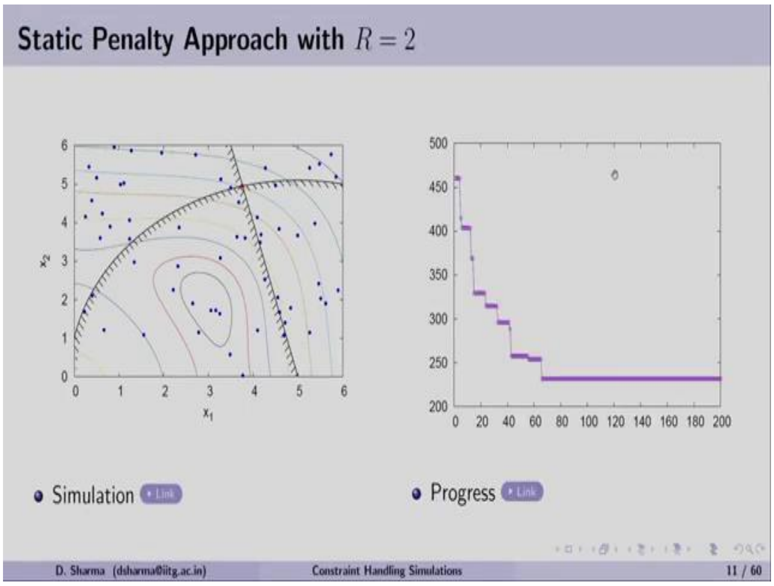
So, let us begin our simulation for the first problem that is Himmelblau Function. As we know this Himmelblau function is a unconstrained problem, however we included these two constraints into the problem.

So, that we can have the constraint Himmelblau function we have assumed that x 1 and x 2 a both of them are greater than 0. Now if we look at the figures on the left hand side in this particular figure we can see for an unconstrained Himmelblau function the optimize lying at 3 2 3 is x 1 and x 2 is 2.

However because of these two constraints the optima is move to the intersection of these two constraint as we can see as a red dot here. Now the area above this red dot that is the right top corner, so this particular area is feasible and rest of the area is the infeasible for the given problem.

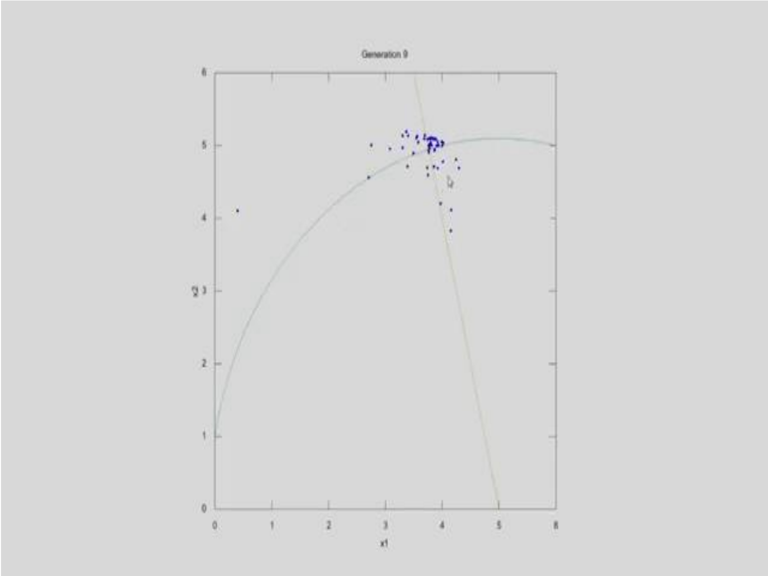
Now, as we can understand from this problem that because of these two constraints the original optimal solution for an unconstrained problem is change to something else. On the right hand side we can see that the optimum solution is given, it is note that this optimal solution is shown for 3 decimal places of accuracy.

(Refer Slide Time: 11:15)



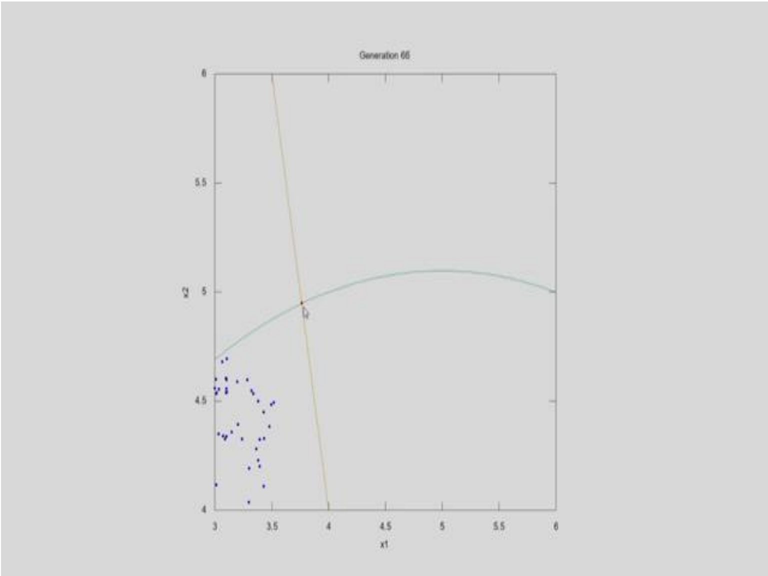
So, let us begin our simulation here we are starting with a static penalty approach where the value of R is 2. So, we are purposely keeping a value of R small here to see the effect of the R here. In the figure we can see that the solution that is the 100 number of individual or solutions that are generated randomly. So, these are the initial share solutions that are generated randomly. Now let us see how this particular method will solve the problem.

(Refer Slide Time: 11:54)



Now, as you can see 2 constraints are shown red dot is also shown.

(Refer Slide Time: 11:58)

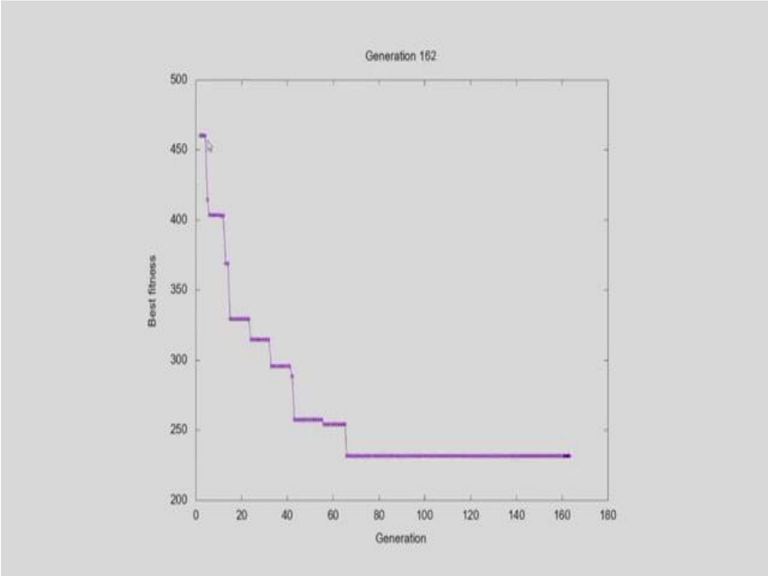


So, red dot is an our optimum solution, but somewhere close to 22 generation the blue dots these are the current solution in the population. So, these solutions are they have left the, our optimum solution and they are moving towards some other solution.

It is only because we have kept the value of R is 2 which is small meaning that we are not penalizing our constraint violation, enough that will be penalizing our objective function. As we remember in our previous slide the optima.

So, the objective function at optima is close to 500, but because R is taken as a small value even the infeasible solutions having less fitness value than the optimum solution. And therefore these solutions they have approach to the optimum solution, but after some generation these solutions have left the optimum solution.

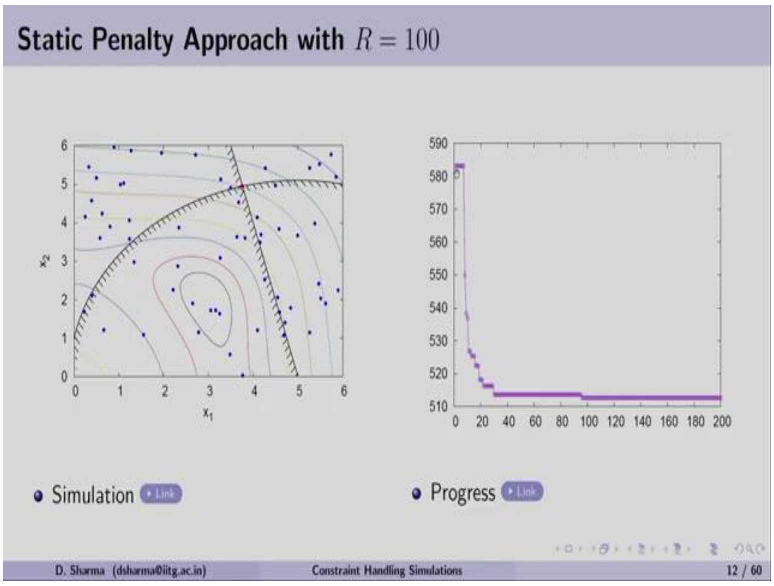
(Refer Slide Time: 13:19)



Let us look at the progress now here we can see that the fitness started close to 460 and then the best fitness is keep on reducing.

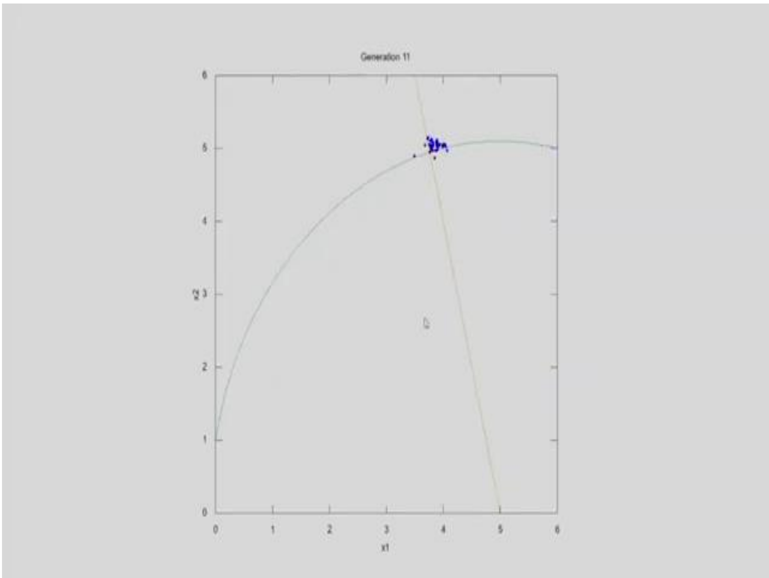
And we know that the function value at the optima is more than 500, since one of the solution which we will see later that it was an infeasible solution. But even after adding the constraint violation with small value of penalties R equals to 2, this particular penalty function method is unable to keep the optimum solution and has left the solution and find the some other optimal solution.

(Refer Slide Time: 14:04)



Since we know that R value is small, now we have kept R equals to 100. Now it is a big value for this particular big value of R meaning that if there is a constraint violation we are penalising more the objective function with respect to the static penalty method. So, we are starting with the same set of a solution for R equals to 100 and let us see how these solutions will find the optimal solution for us.

(Refer Slide Time: 14:38)

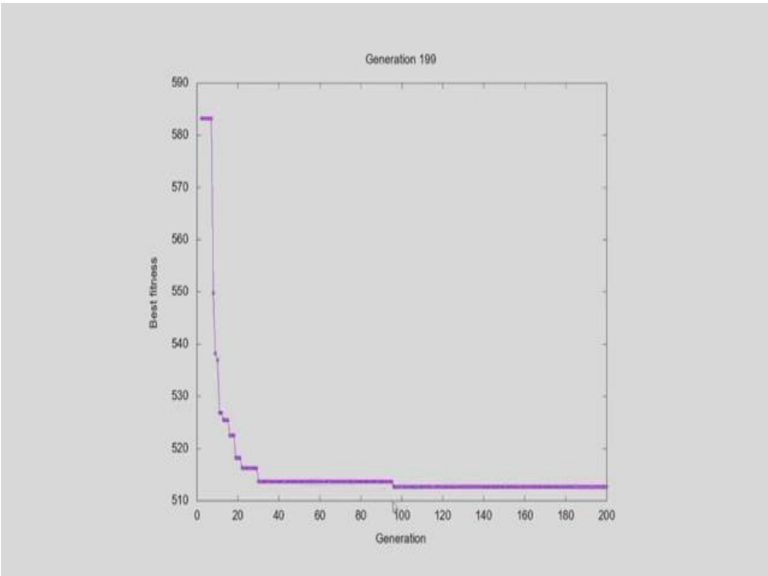


Now, as we can see the solutions has moved to the feasible region. Now, they are moving towards the optimum and they have found some they have found this red dot. Apart from that there are other regions where the solutions are converged, it is only because the function value the optimum function value at this point is still 5 more than 500 by keeping R equals to 100 is not sufficient that will keep all the solution at the optimal solution.

Still it is 100 generation and let us see how the solution whether they will stick to the optimum solution or they will converge to the another solution. Still we can see that one of the solution is stick to the optimal solution of the given problem; however, it still the other solutions are moving in a direction where they are finding a better fitness value. Let us look at the progress now.

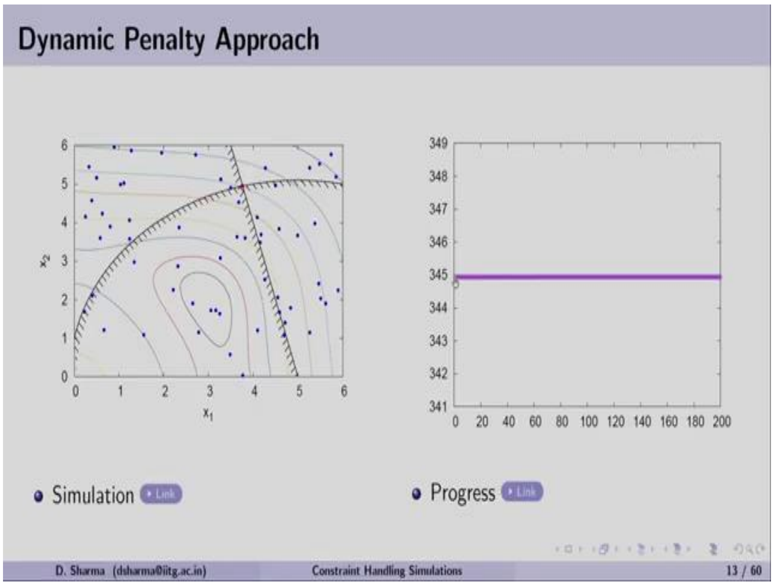
Now from this particular plot we can see that the fitness is starting somewhere close to 585 and then it has improved here and let us see the simulation now.

(Refer Slide Time: 16:02)



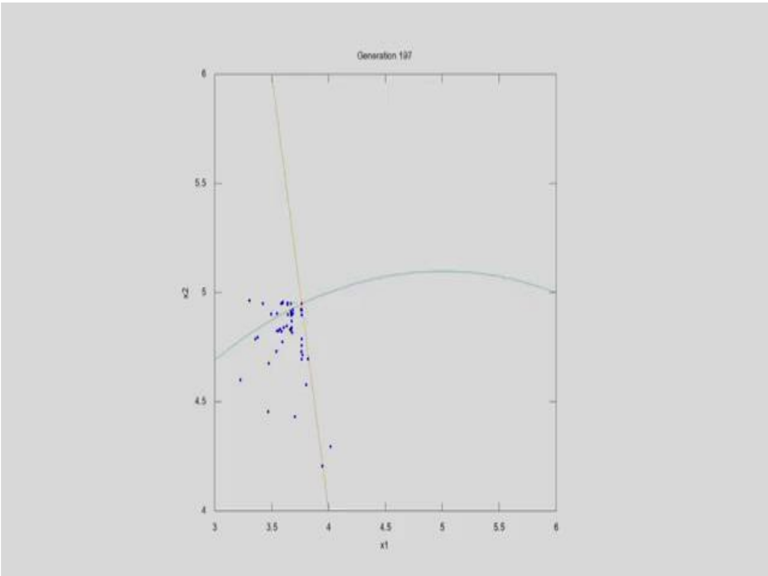
Here we can see the fitness started with the large value, it is keep on reducing and then even after 30 generation there is no change in the function value. But close to more than 90 generation they the, these this function found the, another solution which is better than the optimum solution. But we know that that solution is infeasible solution.

(Refer Slide Time: 16:37)



Now, let us start the Dynamic Penalty Approach. Now in this particular approach we are again starting with the same set of solutions.

(Refer Slide Time: 16:52)

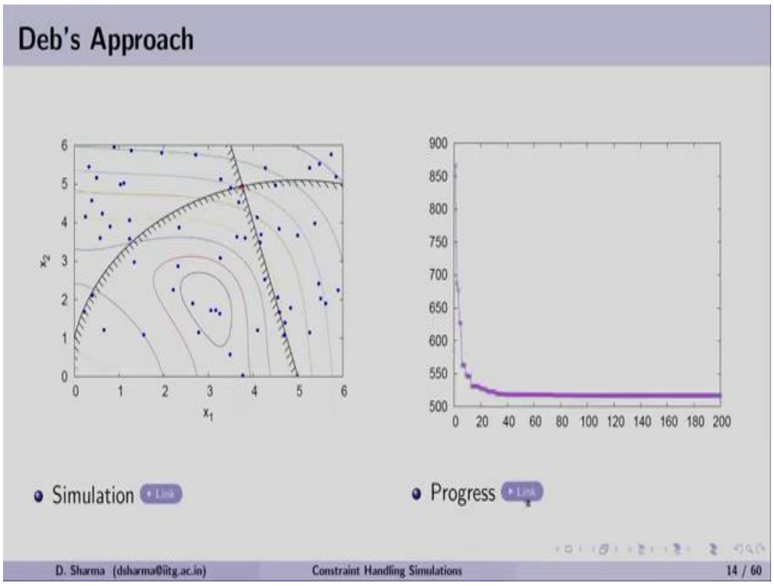


So, let us see how this dynamic penalty approach will help RGA to find the optimal solution. As we can see many of the solution in the feasible region some solutions have converged to the optimum solution now and let us see how the other solution will move. Now, here all these solutions they are almost fixed their position because, there is no solution which has better fitness. But somehow close to 90 generation, the solutions which are converge to the optimum they have left the position.

It is only because there is another solution which will be which will be showing a better fitness, it is only because the values of c and alpha we have chosen, that is not sufficient for the given problem that will penalize our objective function maximally. So, as we can see it has the optimum solution is already left.

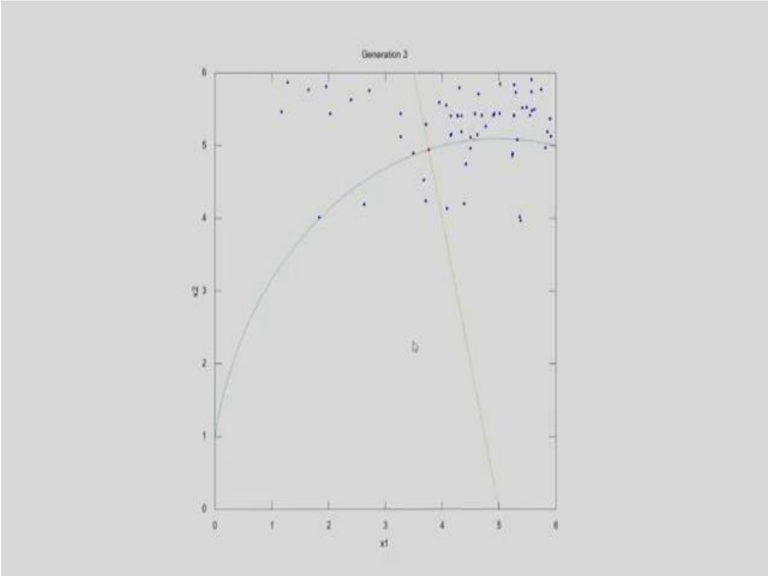
Now here it is very interesting that when this particular simulation was started a one particular solution showed the fitness of 345 and we will see later that this solution is infeasible. But because we are not penalizing the constraints with the sufficient value and therefore this infeasible solution is showing better fitness than the optimal solution which is close to 500 a value.

(Refer Slide Time: 18:31)



Now come to the Deb’s Approach, here we are again starting with the same set of input same set of initial population here. Let us see how this particular constraint handling technique with RGA solve the given problem.

(Refer Slide Time: 18:49)

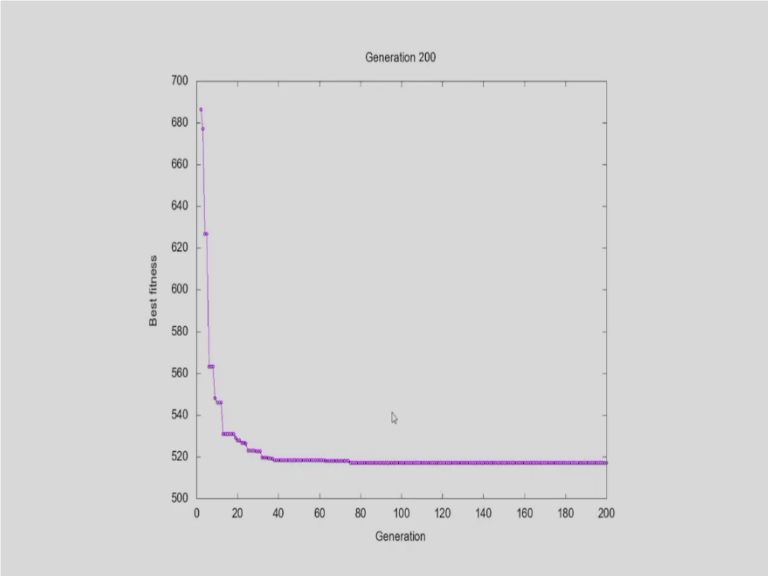


Now, as we can see the solutions has all the solutions are converged to the feasible region. Now inside the feasible region, these solutions are moving towards the optimal solution. So, somewhere close to 30 generation we are already converged to the optimum solution and rest of the solutions are always are already converged here.

Now we have to see whether the Deb's constraint handling technique is also suffering with the same problem as we have found with the penalty function methods. Here in which case the population left the optimum solution and converge to the sum and other infeasible solution.

So, we have to see here that these solutions are moving in which direction and as per our theoretical understanding we know that the feasible solution the infeasible solution is always having worst fitness than the feasible solution. So therefore, as soon as the solutions are converged to this feasible and optimum solution there is these solutions will never go to the other infeasible solution as per their as per their fitness assignment. Now, let us look into the progress of Deb's approach on the Himmelblau function here.

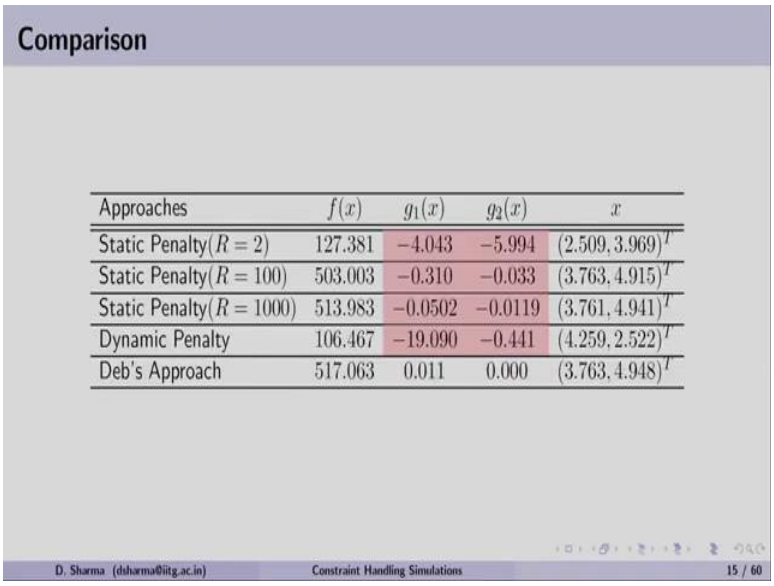
(Refer Slide Time: 20:20)



.

Here we can see that it is started somewhere close to 690 and we can see just after 30 generation we are very close to the optimum solution and then finally close to 72 generations we have reached to the optimum solution and all the solutions are converged to the optimum solution here.

(Refer Slide Time: 20:48)



Approaches	$f(x)$	$g_1(x)$	$g_2(x)$	x
Static Penalty($R = 2$)	127.381	−4.043	−5.994	$(2.509, 3.969)^T$
Static Penalty($R = 100$)	503.003	−0.310	−0.033	$(3.763, 4.915)^T$
Static Penalty($R = 1000$)	513.983	−0.0502	−0.0119	$(3.761, 4.941)^T$
Dynamic Penalty	106.467	−19.090	−0.441	$(4.259, 2.522)^T$
Deb's Approach	517.063	0.011	0.000	$(3.763, 4.948)^T$

Now, here the simulations which we have performed those are little confusing, why because in penalty function methods we have left the optimum solution. So, let us look at what is the best solution they are generating. So, with the help of this RGA using static penalty R equals to 2 we can see the function value is 127 which is quite less than the optimum solution. However, this particular solution is infeasible because both of the constraints are not satisfied. So, when we are using a very small value of R say 2.

So, this constraint violation is not adding much value into the objective function and that is why this static penalty was unable to solve this problem. Looking at R equals to 100 we have increased the value of a penalty here, we can see that the, this the solution is still infeasible.

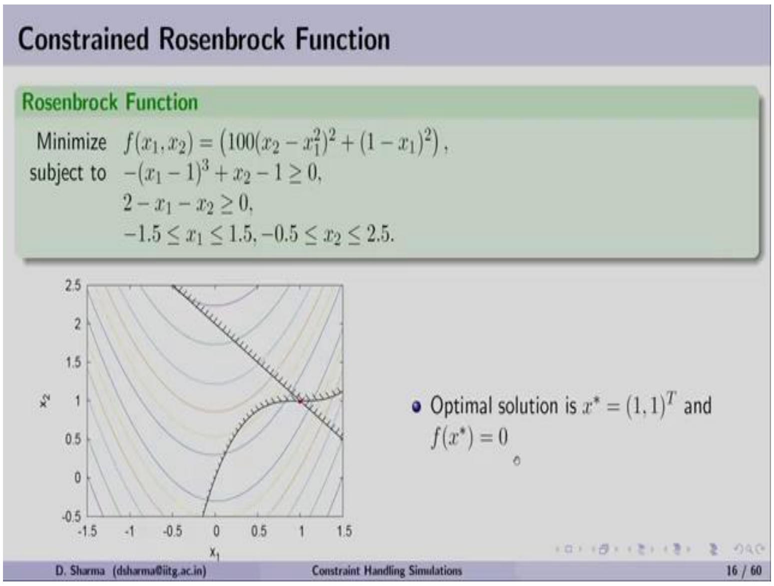
But looking at the constraint violation this solution is still close to the optimum solution. Since the objective function it is varying in the range of 500. So we change the value of R from 100 to 1000 we perform the simulation using real coded ga. Now you can see that although both the constraints are not satisfied.

But their constraint violation shows that we are close to the optimum solution and that is evident from their x value from the last column. Dynamic penalty has the same problem that the value of a c and alpha is not are not sufficient and that is why this dynamic penalty converge to the infeasible solution and the constraint violation is still large. But because we are not penalizing sufficiently the dynamic penalty is unable to find the optimum solution for the given problem. Now the Deb’s approach if we locate look at it.

Now, the fitness of the fitness assignment of a solution in such a way that the feasible solution is always having better fitness than any of the infeasible solution. So, we found that this optimum solution is found by the Deb’s approach, both the constraints are satisfied and we have reached to the optimum solution as we can see on the, at the last column of the table.

So, here what we can see that although if we can keep our if you want to use the penalty function value function method. We have to use those value of R c or alpha according to the problem any arbitrary value may not be able to solve the problem that is evident from the given example. Now let us see the behavior of these constraint handling techniques for the other problems.

(Refer Slide Time: 24:00)



$$Minimize \quad f(x_1, x_2) = (100(x_2) - x_1^2)^2 + (1 - x_1)^2),$$

$$subject\ to\ -(x_1 - 1)^3 + x_2 - 1 \geq 0,$$

$$2 - x_1 - x_2 \geq 0,$$

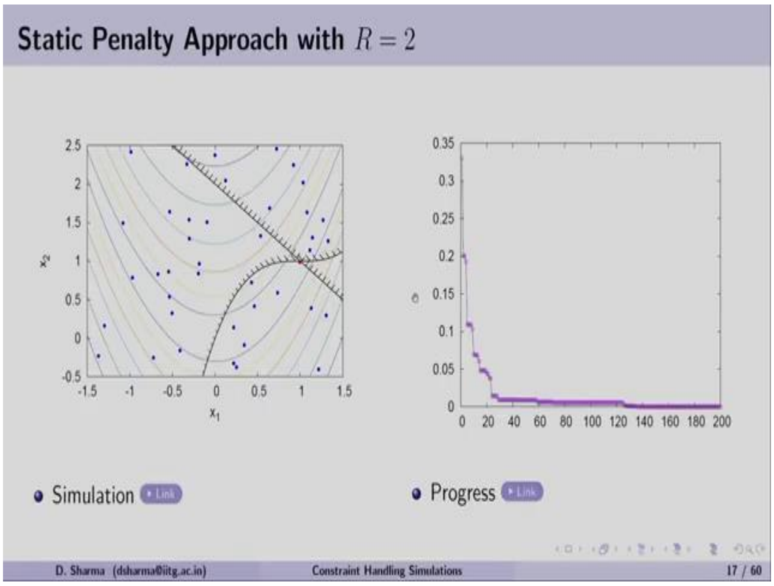
$$-1.5 \leq x_1 \leq 1.5, -0.5 \leq x_2 \leq 2.5.$$

Now, we will start with the Rosenbrock problem. So, we have taken only 2 variable Rosenbrock problem, as we remember this Rosenbrock problem was a unconstrained problem. We have introduced 2 constraints into the problem to make a constrained optimization problem here the value of x 1, so the range for x one and x 2 are given here.

On the figure that is the figure on the left hand side we can see that because of these constraint the optima is lying at the intersection of these two constraint and the region just below this particular point that is the feasible region and rest of the region is the infeasible region.

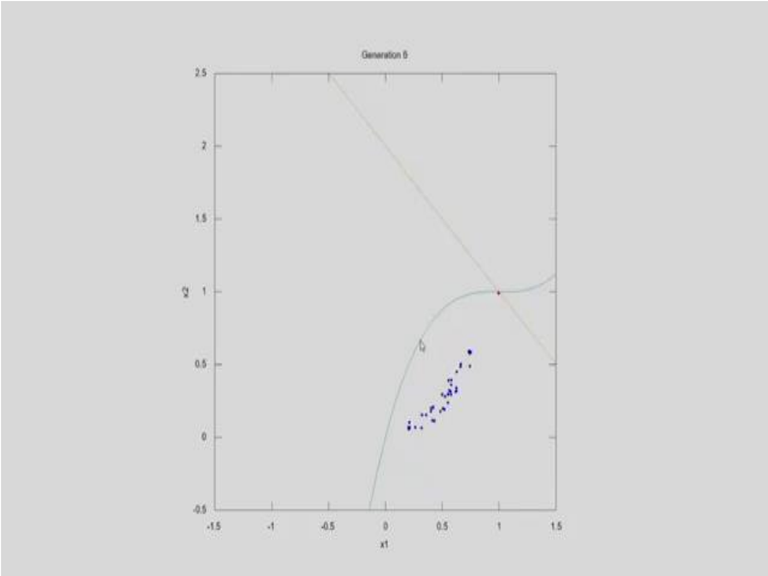
The optima for the Rosenbrock function is 1 and a 1 and the function value at it is 0. Now what you can see that the Himmelblau function was having the function value 500, but here it is the function value 0.

(Refer Slide Time: 25:03)



So, let us see how these functions constraint handling techniques will solve this particular problem. So, we are starting with static penalty approach with R equals to 2, now we have generated the initial population within the range of x 1 and x 2. Let us see how we are going to solve this problem here.

(Refer Slide Time: 25:26)

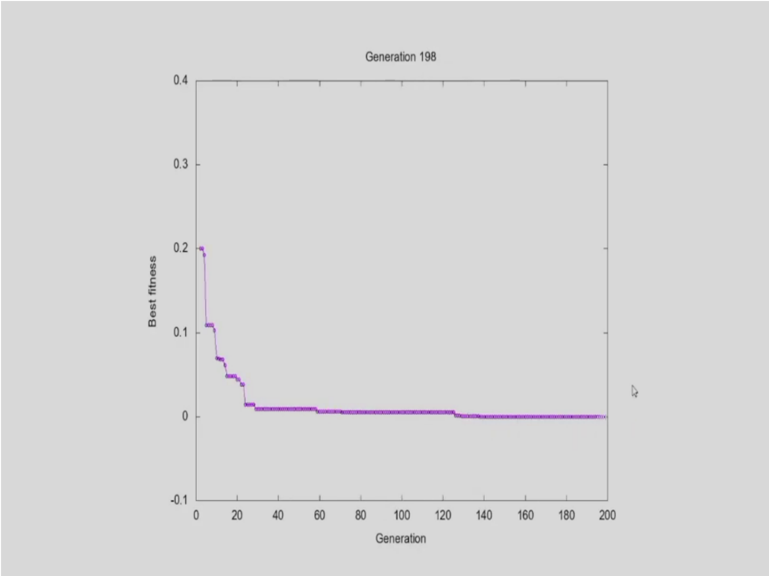


So, as we can see the solutions are converged into the feasible region now and inside this feasible region these solutions are now moving towards the red dot, which is the optimum solution for the given problem. Moreover, we know that for the Rosenbrock problem there are some local optima's as well in between and that is why the solutions are getting stuck.

Now, we have to see that since the solutions are in the feasible region, whether the penalty function static penalty function with R equals to 2 is able to generate the optimum solution or not. See close to 130 generation some solutions are generated close to the optimum solution and with the iteration these solutions are now moving towards the optimum solution.

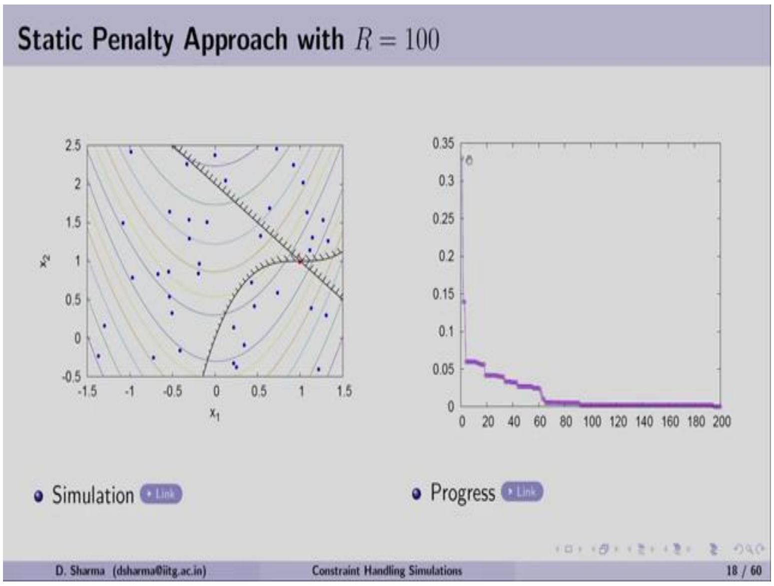
So let us wait till 200 generations are over, still we are very close to the global optimum solution. Let us look at the progress of the solution or the best fitness.

(Refer Slide Time: 26:40)



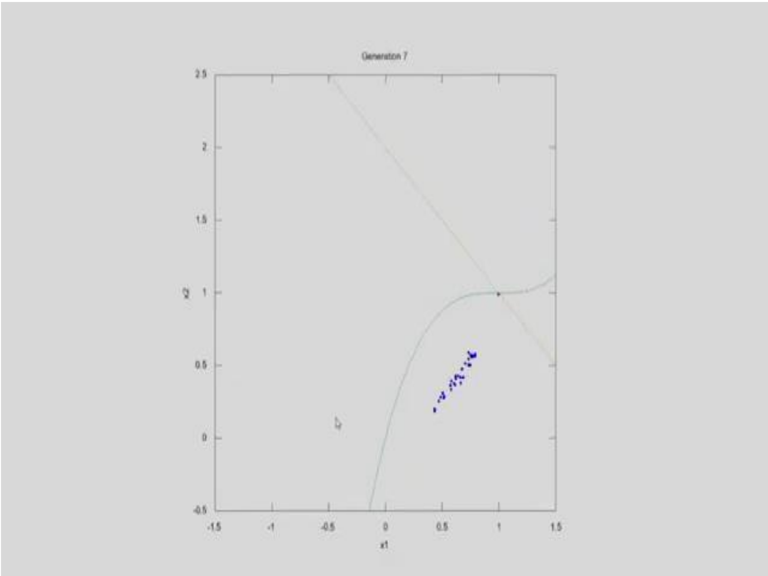
It is started from 0.2 and as we know that the best fitness is 0. Now as we can see that there is a little improvement after 55 generation, but still we have not reached to the optimum solution. But just after 120 generation the solutions are converge to the optimal solution here.

(Refer Slide Time: 27:02)



Now, suppose for the given problem if we increase the value of the static penalty from R equals to 2 to R equals to 100 we will see the performance of this static penalty function. Here we are starting with the same initial population and let us see how these simulations are how these solutions are moving here.

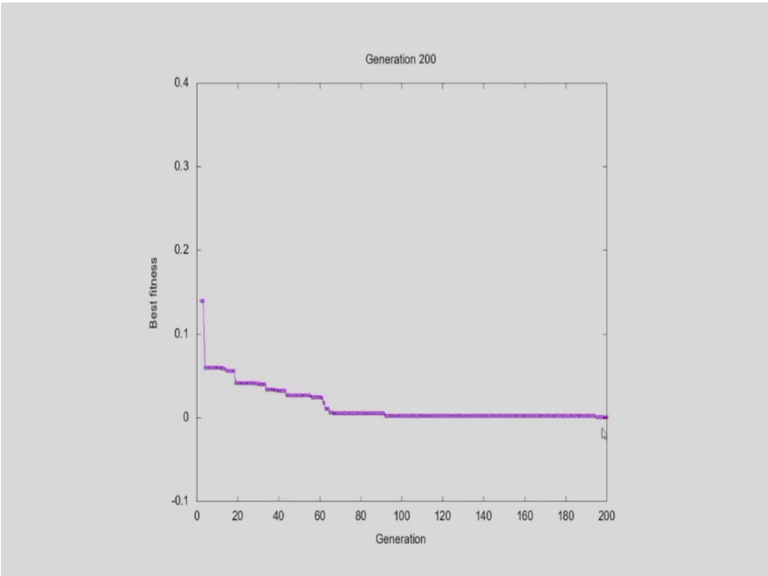
(Refer Slide Time: 27:31)



Here the solutions are already in the feasible region and slowly and slowly they are moving towards the optimum. Since there are local optimum solution, so population so the members are stuck there and we are still moving towards the optimum solution. Since it is less than 100 iteration we have to see whether the larger penalty value R equals to 100 can find the optimum solution for the given optimization problem. So, still we have little far as we can see that the all the solutions are converged to the same solutions and the red dot is little far from the these blue dots are little far from the optimum solution that is given in the red dot.

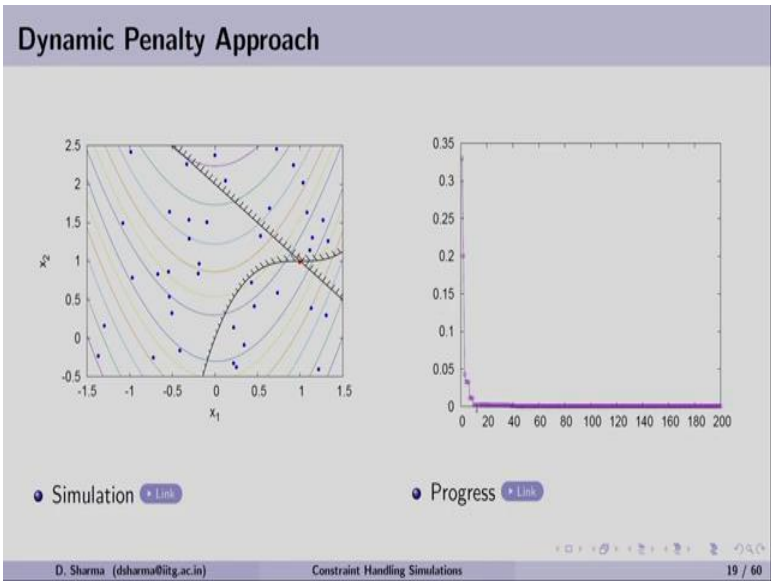
So, still we have not converged to the optimum solution and just at the last iteration in the last generations the solutions are converged. Now we can see here from the progress plot here it is started from somewhere 0.32 and then the fitness is keep on improving. As we can see that the algorithm was unable to generate the good solution or the solution close to the optimum and just at the last just before 200 generation, we found some solution converge to the optimum solution.

(Refer Slide Time: 29:03)



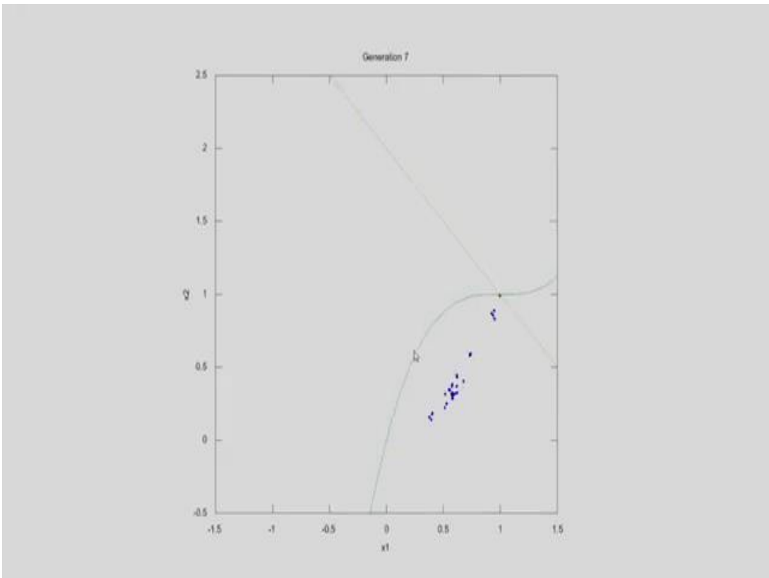
So, let us see the progress here. So, it is started with the value and then this fitness is keep on improving. Now just after the 60 generation or close to 90 generation we are close to the optimum solution, but not converged to the optimum solution and we have to see that whether at the last we are able to converge to the optimum solution or not. Because that happened just before 200 generation yes, so there is a little improvement just before 100 generation.

(Refer Slide Time: 29:47)



We will solve the same problem now using Dynamic Penalty Approach here, we are starting with the same set of solutions.

(Refer Slide Time: 29:59)



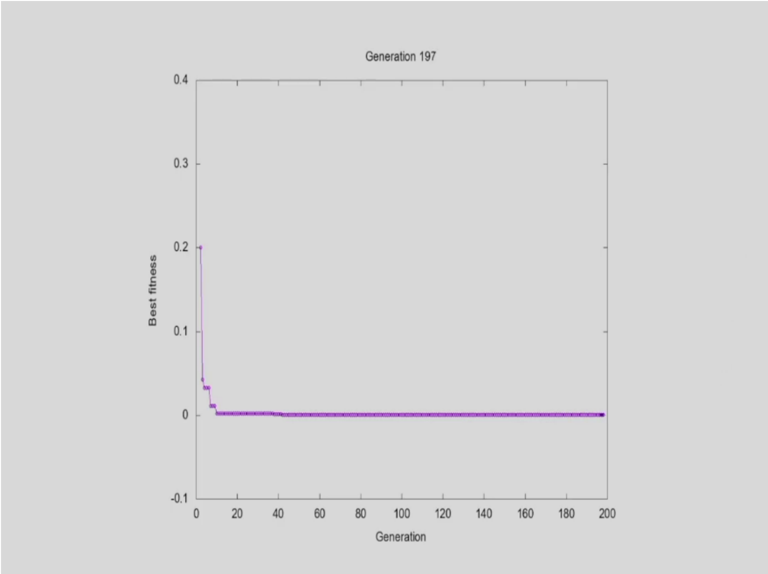
Now, and let us see how this penalty function approach or a method can solve this particular problem. So, solutions are already in the feasible region, we are close to the optimum solution in less than 20 generation and now we have to see whether the dynamic penalty can help us to solve the problem. Although as soon as we are in the feasible region the objective function value is not penalised here.

But the solutions which we are showing that are following mu plus lambda strategy, meaning that these crossover and mutation they can be generating the solutions if they are not good they are not included into the population.

So, we are still close to the optimum solution and we have to see whether these solutions will converge to the optimum solution or not. So, here close to 160 generation still we are little far and we have to see whether we are able to converge or not now at the last.

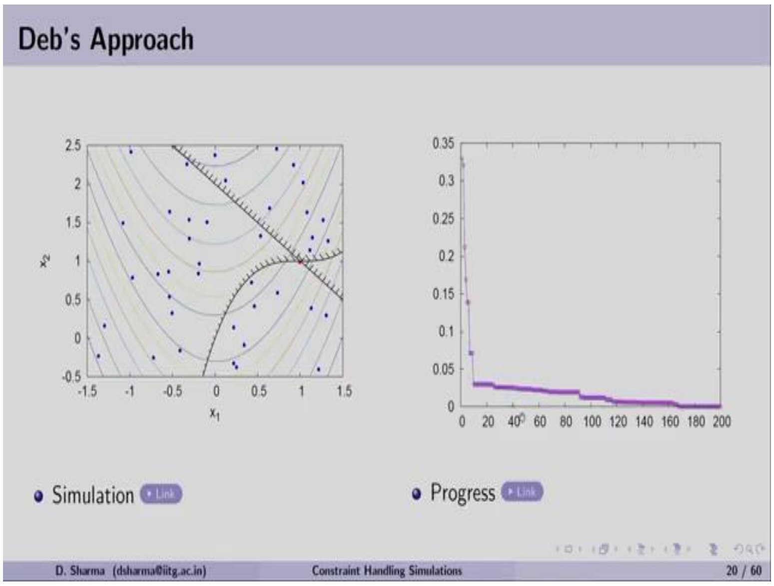
So, still this particular method is unable to converge to the optimum solution, let us see the progress now. Now as we can see that there is a drastic improvement in the fitness value and then the population or the members are unable to improve it and we converge close to the optimum solution. So, let us see the progress now here.

(Refer Slide Time: 31:29)



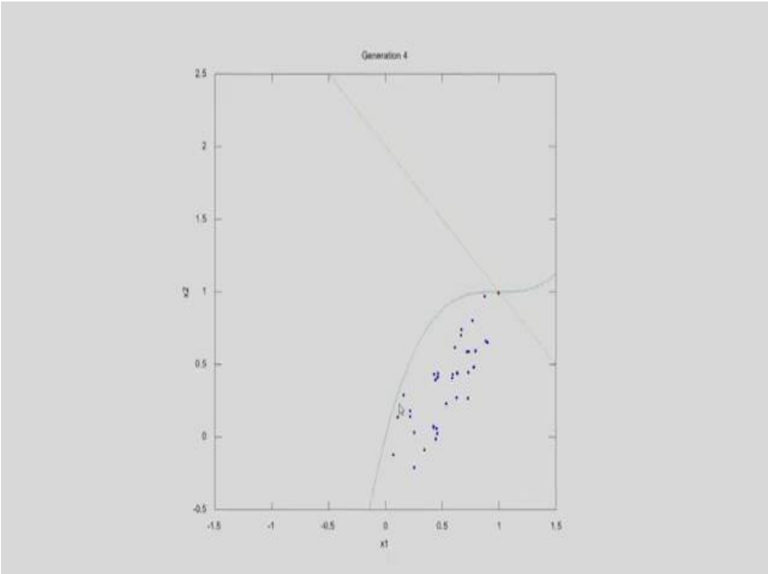
So, here we can see that we are very close to the fitness of 100 here and since we are very close and there were no solution new solution that was generated on the optimum solution. So, this is was the best fitness given by this particular method.

(Refer Slide Time: 31:51)



At last we have Deb's approach here, now in this Deb's approach we are starting with the same initial population and we have to see whether this approach is able to solve the problem.

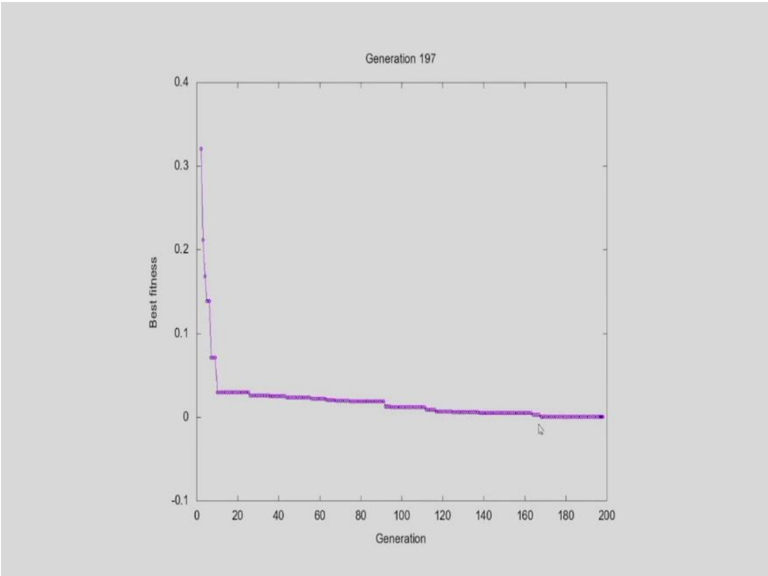
(Refer Slide Time: 32:02)



Now we can see the solutions are quickly converge to the feasible region. Now, these solutions are slowly moving towards the optimum solution and this is evident from the previous simulations that the since there are local optimum solutions so these solutions. So the population members are getting stuck to the local optimum and we have to see whether we can find the optimum solution using the Deb's approach. So, the half of the generations are already over and still we are little far. But every time after few generation new solutions are generated that are closer to the optimum solution.

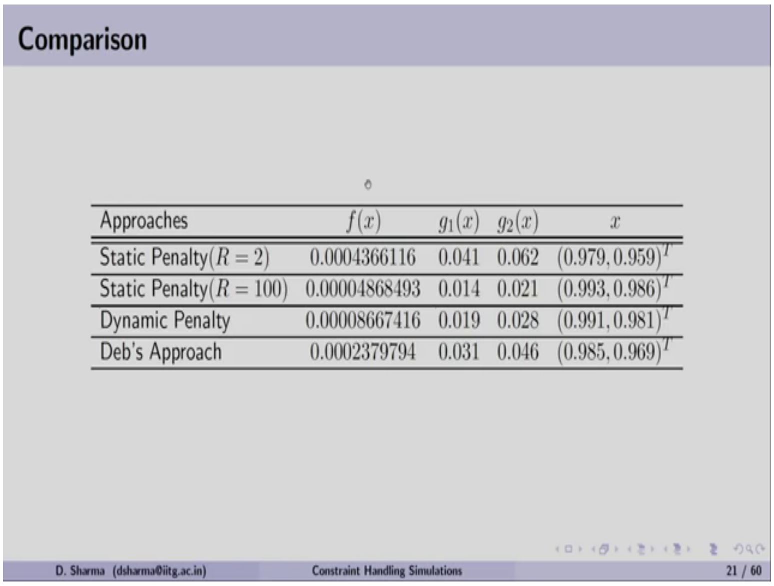
So, we have to see the progress now the solutions are very close to the optimum in 170 generations and we have to see whether we can find the optimum solution or not. Yes so we can see that it is not exactly on the optimum solution, but it was very close to the optima.

(Refer Slide Time: 33:12)



Let us see the progress, now we started somewhere more than 0.3 fitness function fitness value of the best fitness and then which is keep on improving and still after 120 generations we are not converged to the 0.But as soon as after 165 generation we have converged to the optimum or very close to the optimum solution with Deb's approach a penalty constraint handling approach.

(Refer Slide Time: 33:44)

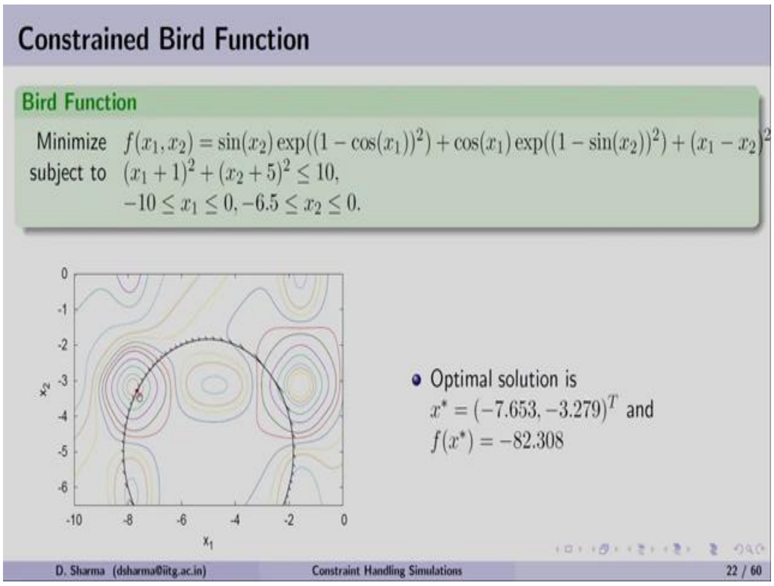


Approaches	$f(x)$	$g_1(x)$	$g_2(x)$	x
Static Penalty($R=2$)	0.0004366116	0.041	0.062	$(0.979, 0.959)^T$
Static Penalty($R=100$)	0.00004868493	0.014	0.021	$(0.993, 0.986)^T$
Dynamic Penalty	0.00008667416	0.019	0.028	$(0.991, 0.981)^T$
Deb's Approach	0.0002379794	0.031	0.046	$(0.985, 0.969)^T$

Now, let us compare the solutions, as you can see whether it is a static penalty with R equals to 2 or it is a static penalty with R equals to 100, dynamic penalty or Deb's approach all of these approaches find the feasible solution as you can see both the constraints are satisfied.

Now, if we look at the function value up to 3 decimal places all of them are giving 0 value, but after that there are certain changes. Now looking at these values we can say that all these methods have found the near optimum solution for the given problem.

(Refer Slide Time: 34:28)



$$Minimize \ \ f(x_1, x_2)$$

$$= sin(x_2)\ exp\left(\left(1 - \ cos(x_1)\right)^2\right) + \ \ cos(x_1)\, exp\left(\left(1 - \ sin(x_2)\right)^2\right)$$

$$+ \ (x_1 - \ x_2)^2,$$

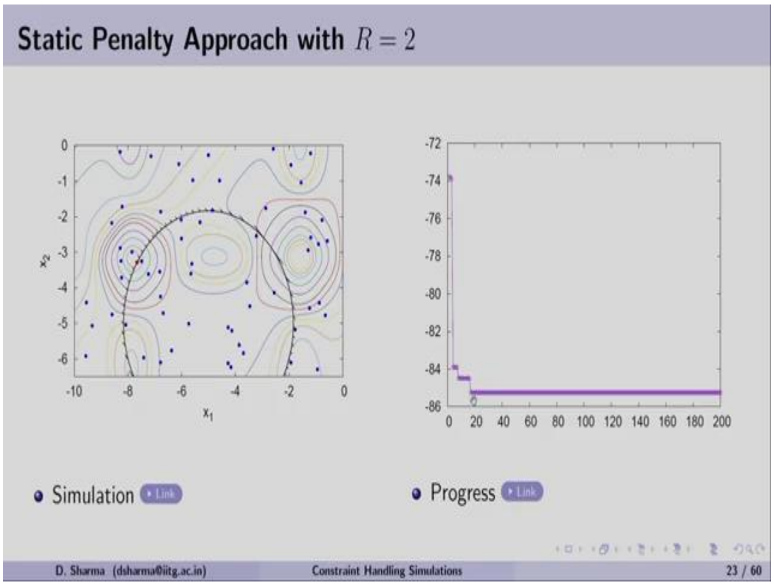
$$subject\ to\ (x_1 + 1)^2 + \ (x_2 + 5)^2 \leq \ 10,$$

$$-10 \leq \ x_1 \leq \ 0, -6.5 \ \leq \ x\_2 \leq \ 0$$

Now, we will start with the Bird Function, this bird function is also an unconstrained problem. Now we can see that we want to minimize the function it include sin exponential cos as well as the square term. So, this is going to be having a little complex problem to solve, to make it constrained we have included just 1 constraint and this particular constraint now we have x 1 and x 2 this is lying between minus 10 to 0 and minus 6.5 to 0.

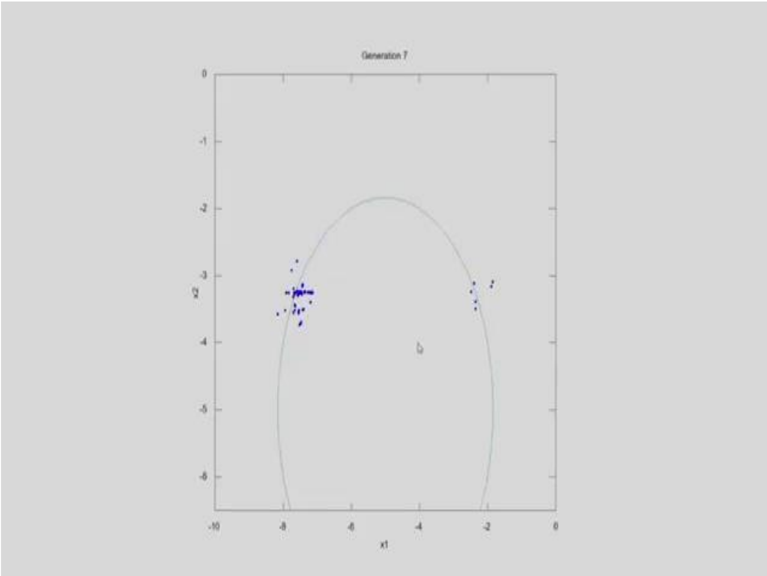
Now here for the given problem we have the, we have the contours of the objective functions as well as we have the constraint. Since we have just one constraint the optimal solution is lying. As you can see in the plot the red dot represents the optimum solution for the given problem. Now the right hand side the optimum solution is given having a fitness value of minus 82.308 this optimum solution is having the accuracy up to 3 decimal places.

(Refer Slide Time: 35:36)



We are again performing the simulation for R equals to 2 with the static penalty approach and the initial population is given here. So, let us see how this penalty approach static penalty approach can solve this problem.

(Refer Slide Time: 35:50)

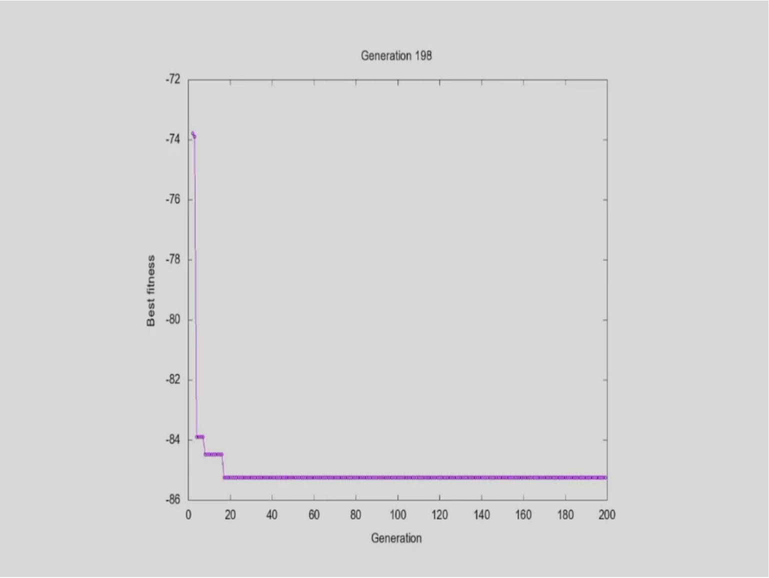


So, as we can see the solutions are converged to the feasible region. Now here some solutions are feasible some solutions are infeasible, but the optimum solution is already found. Since the penalty is less as we can see the solutions which are converged to the optimum solution they have left that position and they started converging to the other area. But we have to see how finally this particular population will be converging to the optimal solution.

Now, here we can see that the solutions have already left the red dot, they started moving in the invisible region. It is only because what we understood from the previous simulation the R value which is 2 right now is not sufficient that will be penalizing the infeasible solution much, so that we can differentiate feasible and infeasible solution.

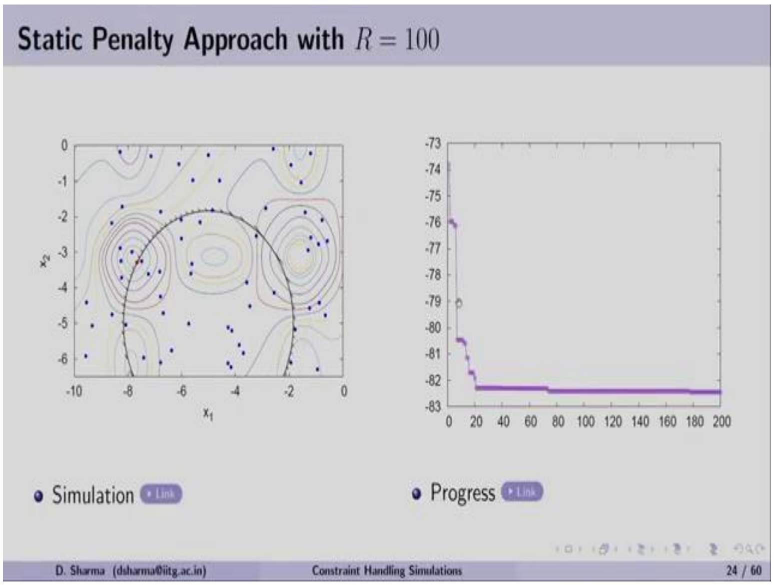
Let us look at the fitness as you can see that close to 20 generation some solution which is having a better fitness was found and we know that this particular solution is having this solution is a infeasible solution.

(Refer Slide Time: 37:09)



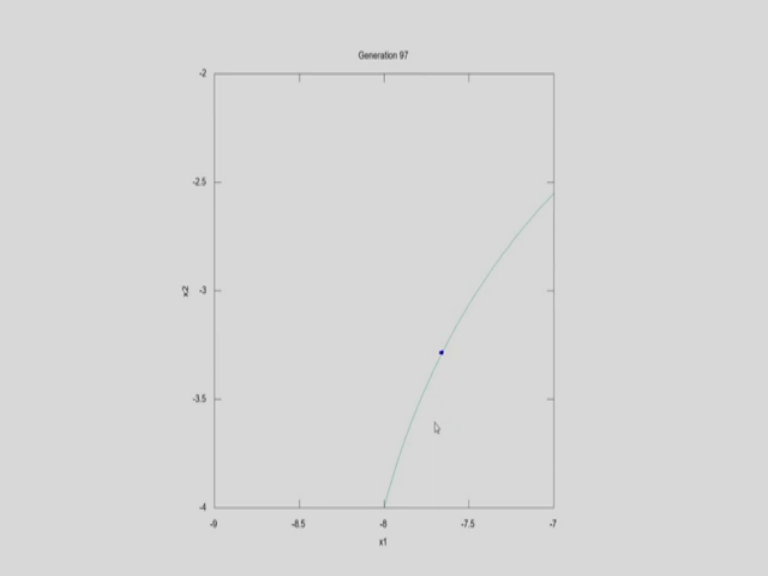
Looking at the progress here close to 16 generation that particular infeasible solution was found, since R equals to 2 is a small value. So, that infeasible solution was not penalised sufficiently and that is why the population members left the optimum solution.

(Refer Slide Time: 37:31)



Let us see with R equals to 100, now since we have increased the penalty R equals to 100 a large value.

(Refer Slide Time: 37:46)

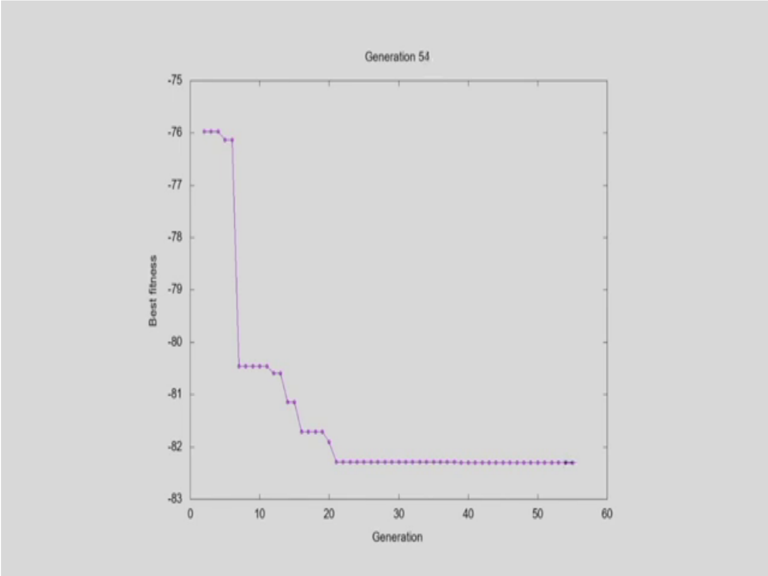


Let us see how this static penalty method is going to solve this particular problem. Again all the solutions are converged into the feasible region and now solutions are moving towards the optimum solution.

So, close to 25 some solutions are already converged to the optimum and we have to see whether R equals to 100 is still sufficient that will retain the optimum solution or it will leave to some another infeasible solution. Yes now you can see some solution are started moving away from the optimum solution going into the infeasible region.

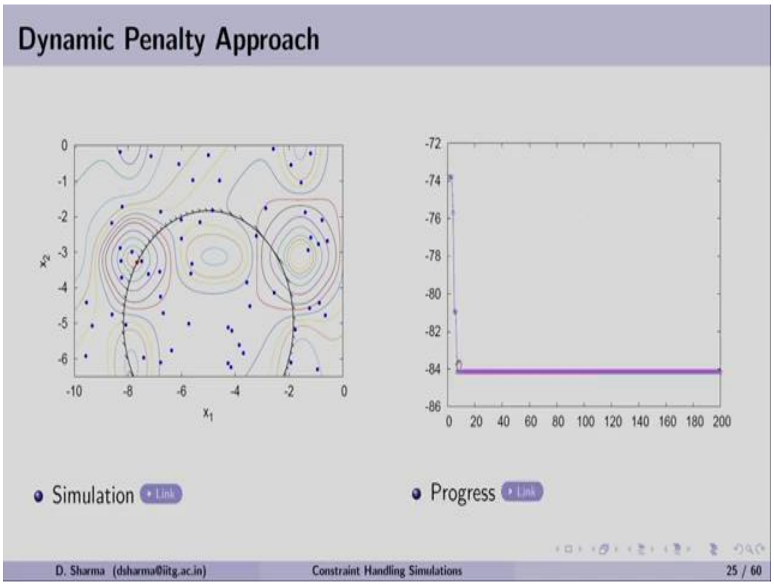
And we have to see up to 200 generation whether some solution will completely leave the optimum solution or not. So, here still we can see that the solution are stick into the same region here which is on the red dot and since there is no movement it seems that there is no other solution which is having a better fitness. So, as we can see even after 200 generations some solutions have converged to the optimum solution.

(Refer Slide Time: 38:58)

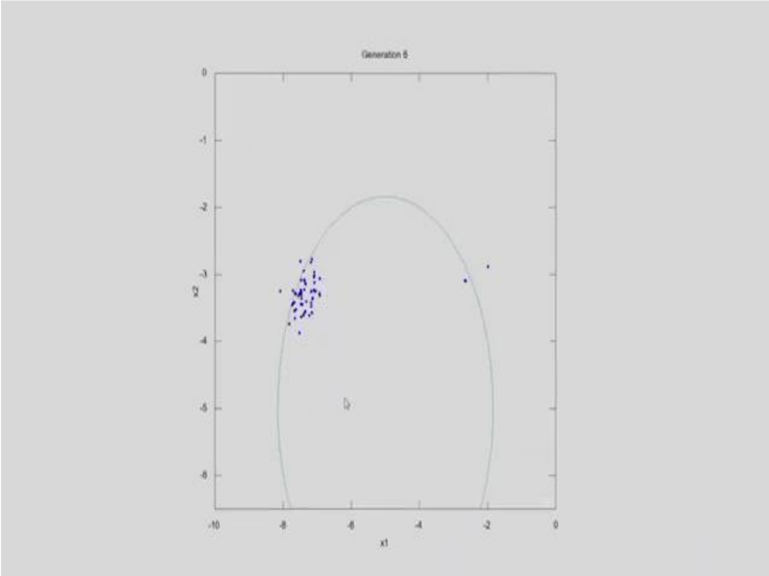


Now, let us see the progress it is started somewhere from minus 76 and then it started converging. Now we know that up to 20 generation we have already found the optimum solution, it seems that R equal to 100 is not sufficient. Now because of if this algorithm or this technique has found another solution which is having better fitness than the optimum solution and that solution is a, infeasible solution that we will see later.

(Refer Slide Time: 39:26)



(Refer Slide Time: 39:34)

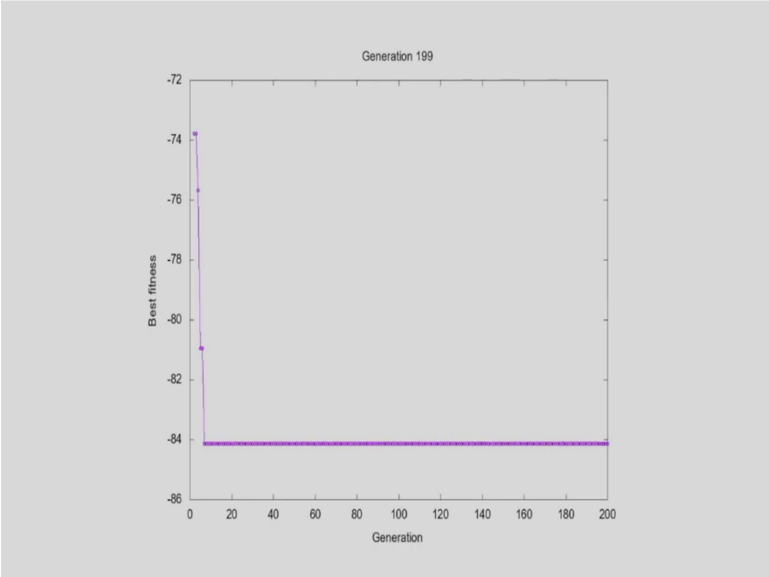


Now, we are showing the simulation for dynamic penalty approach. We are again starting with the same initial population and let us see how these solutions will be moving. Again the solutions are converge into the feasible region, they are again moving towards the optimum solution. But there are other solutions as we can see on the on the other side of the green line, that these solutions are still there and the other solution are also moving in this region.

It is only because we are not penalising the infeasible solution sufficiently. We have to see that even after 200 generation that this particular technique will leave the optimum solution and find the in feasible solution as the best solution for the given problem. So, here there are certain solution there are many solutions which are infeasible and they must be showing better fitness than the optimal solution. Now the population is more or less stuck and we can see that the solutions.

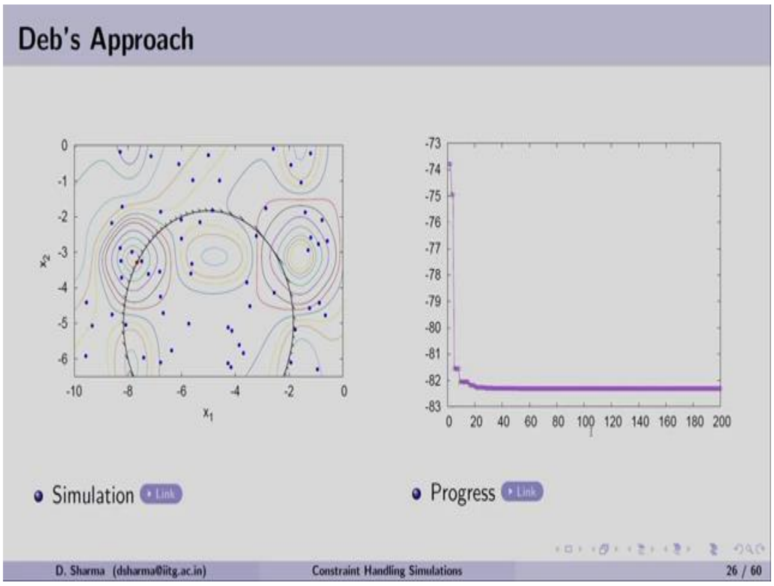
So, there are some solution at the optima, but the other solutions are at the infeasible region. Now we can see there is a; there is a improvement in the solution, but there is an infeasible solution which is showing the best fitness here and that is why even before 10 generation there is no change in the best fitness value.

(Refer Slide Time: 41:01)

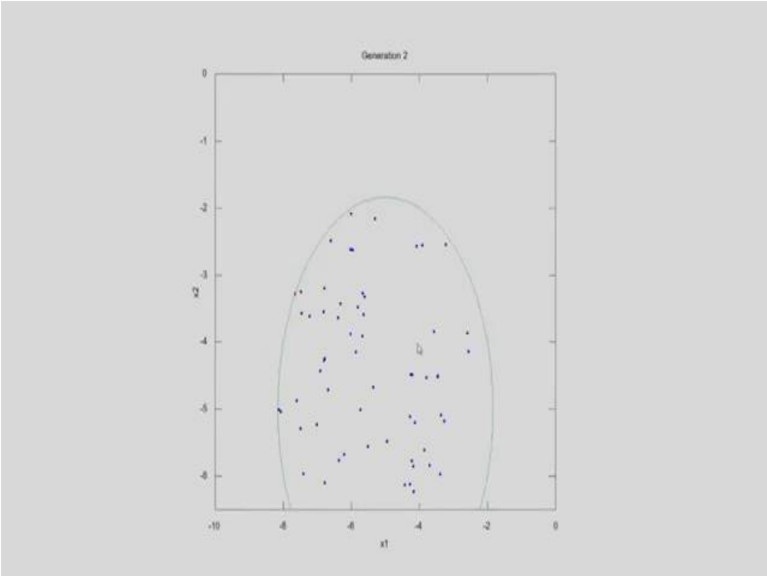


So, as we can see the best just after 5 6 generation or 7 generation this infeasible solution is the is showing the best fitness, meaning that this penalty function approach is unable to differentiate the optimum or the feasible or infeasible solution.

(Refer Slide Time: 41:21)



(Refer Slide Time: 41:29)

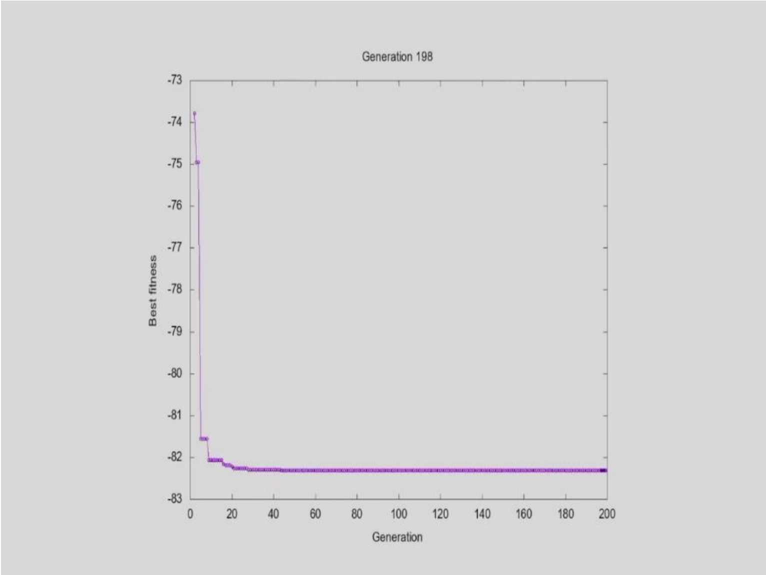


Now come to the Deb's approach now, we are again starting with the same set of solutions, all solutions now you can see that they are converged to the feasible area and the solution now started moving towards the optimum solution. And as we remember that the feasible solution is always having better fitness than the infeasible solution.

So, this Deb's approach should not leave the optimum solution once it is found and till now it is evident from the simulation that all the population members are converge to the red dot as we can see here. And now we have to see it should not move and we have to see our simulation till 200 generations. So, more or less this these solutions are they are not moving at all and they have converged to the optimum solution.

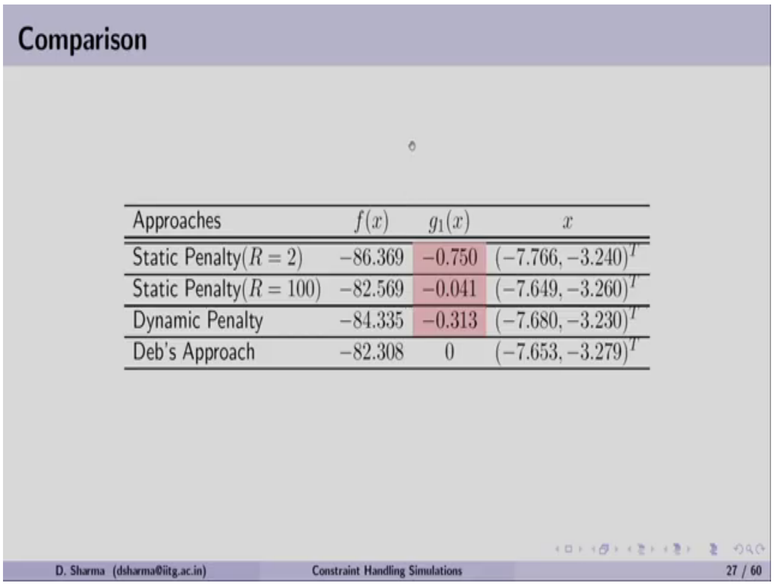
So, let us wait for 200 generation now. So, close to 170 generation is still this the population members are the same and now close to 200 as well there is no improvement in the solution, it is only because we have found the optimum solution. Looking at the progress the best fitness versus iteration, we can see here that just before just after 20 generations say 25 the optima solution was found.

(Refer Slide Time: 42:56)



So, let us look at the progress now. So, as we can see the progress is still they are improving and somewhere close to 28 generation we have reach to the optimum solution and therefore the best fitness remains the same.

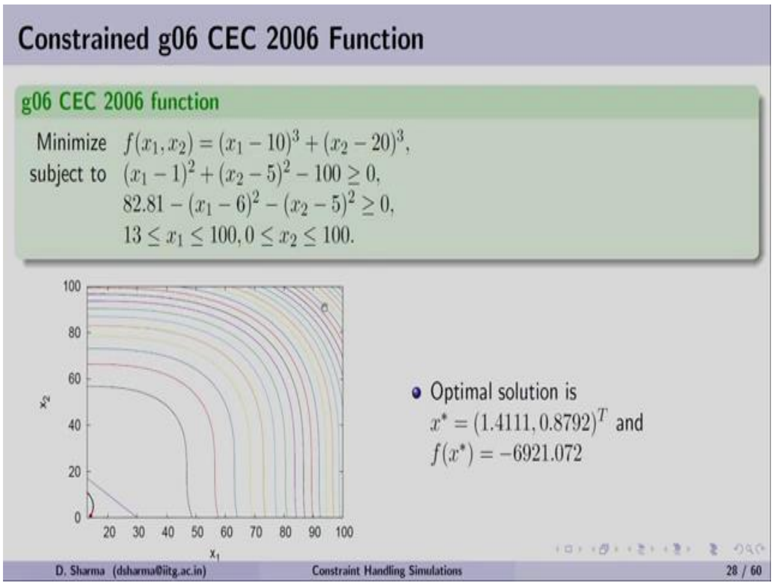
(Refer Slide Time: 43:19)



Now, let us compare all these methods, now we have the function value. So, as we can see the static penalty with R equals to 2 the constraint g 1. So, this is only the one constraint we have it is not satisfied, although it is infeasible solution this R equals to 2 was unable to find the feasible solution for us.

Similarly, when we have R equals to100 solution is close to the optimum as we can see the constraint violation is very small, but it is an infeasible solution. For dynamic penalty as well we have not penalised or the parameters which we have taken here that are not sufficient to penalize the objective function. And look at the Deb's approach the constraint violation is 0, this means that up to 3 decimal places it is 0, it is a feasible solution and this approach gave us the optimum solution for the given problem.

(Refer Slide Time: 44:24)



$$Minimize \quad f(x_1, x_2) = (x_1 - 10)^3 + (x_2 - 20)^3,$$

$$subject\ to \quad (x_1 - 5)^2 + (x_2 - 5)^2 - 100 \geq 0,$$

$$82.81 - (x_1 - 6)^2 - (x_2 - 5)^2 \geq 0,$$

$$13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100.$$

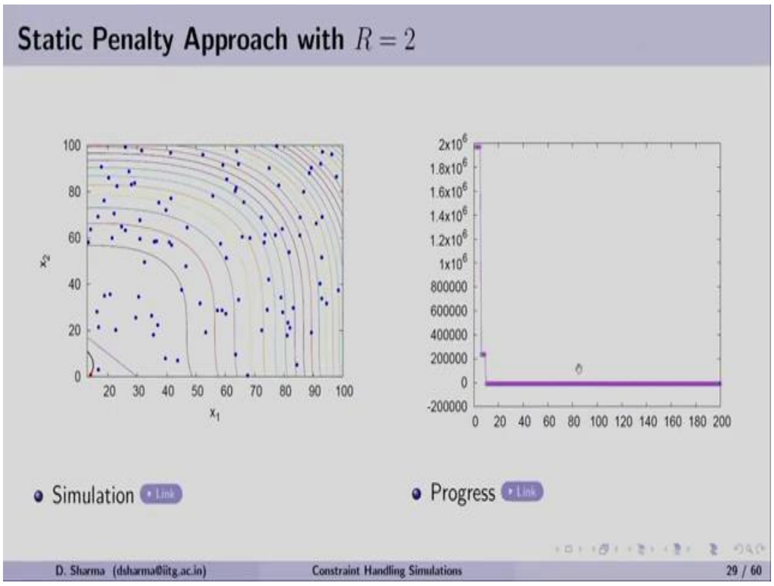
Come to the another mathematical constraint optimization problem, this problem g06 this has been taken from the competition that was performed in the Congress on evolutionary competition conference this is in 2006. These mathematical problems are relatively difficult to solve; however, since we know the optimum solution. So, we can always see the progress of our algorithm as well as our constraint handling techniques. So, let us look at this g06 problem.

Now, this particular problem the objective function is a cubic function and this particular objective function which we want to minimize it is subjected to so these are the 2 equations equation of the circle. So, here you can see minus 100 greater than 0 and in another equation it is 82.81 and then another equation of the circle greater than 0.

So, meaning that the feasible region will be inside these two rings of the circle. The value of x 1 and x 2 are given they are varying from 0 13 to 100 and x 2 will be varying from 0 to 100.

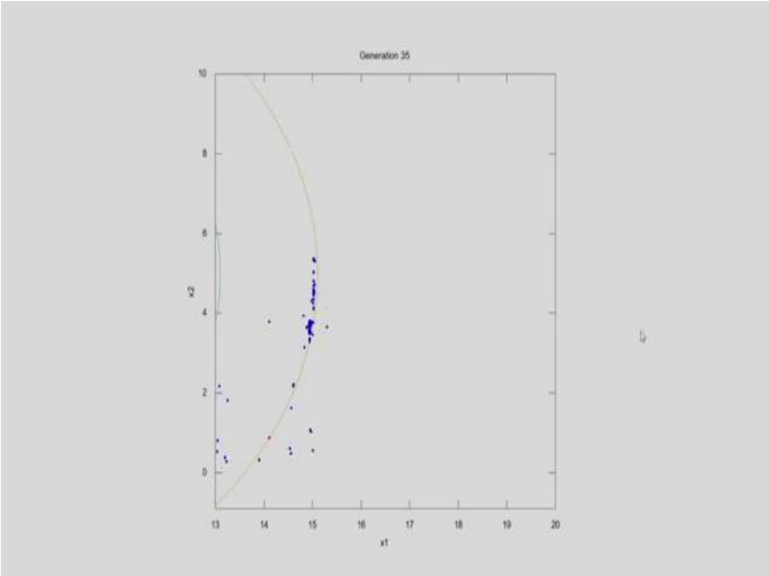
What is the difficult part here is that, since the range of x 1 and x 2 is large. So, solution has to move from very far place to the feasible region and find the optimum solution. Now we can see a red dot that is the optimum solution for the given problem, on the right hand side, the optimum solution is given and the function value is also given.

(Refer Slide Time: 46:09)



This is the initial population that we are starting with the static penalty approach with R equals to 2. So, let us see the simulation for this static penalty approach for a small value of a R.

(Refer Slide Time: 46:22)

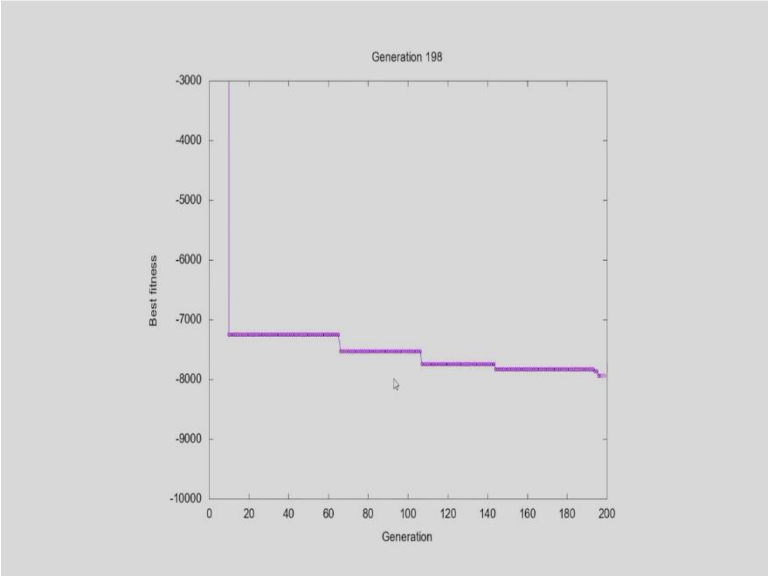


Now, these solutions they are very far from the optima. Now, they are moving slowly and slowly all the solutions are moving towards the optimum solutions and in the feasible region and let us see whether this approach can find the optimum solution for the given problem or a not.

Now these solutions as we can see that they are start moving towards the optimum solution, but still there are some infeasible solutions that are retained in the population, it is only because we are not penalising the infeasible solutions sufficiently. So, that we can differentiate feasible and infeasible solution and similarly we can find the optimum solution for the given problem.

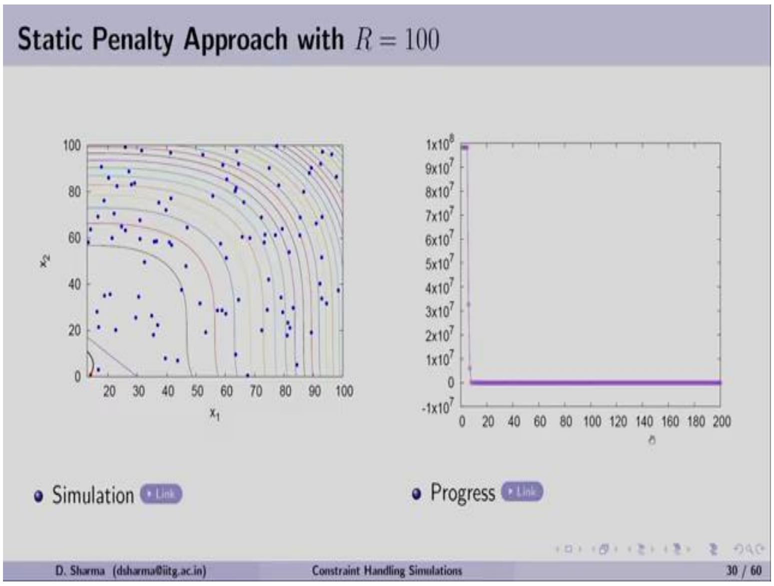
So, still solutions are around the optimum solution, but nobody has converged to the optimum. But these solutions are started moving in another direction and we know that in the given population we have feasible and infeasible solution. But have not found the optimum solution. Look at the progress it is started with a very large value which is 2 into 10 to the power 6 and it is reducing to a very small value.

(Refer Slide Time: 47:46)



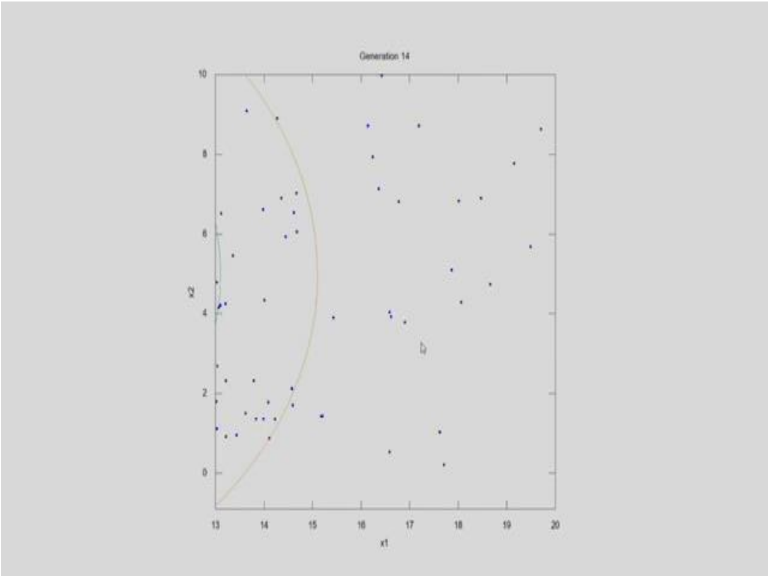
So, let us see the progress. Now, now as we can see it is started with a large value if we just focus into the small area. So, somewhere close to 10 generation. So, the fitness value from 10 to the power 6 is reduced to less than 7000 and the fitness value is keep on improving. But as we know that this function the this particular approach with R equals to 2 is unable to give us the optimum solution.

(Refer Slide Time: 48:15)



So, let us try the another simulation with R equals to 100 which is a very big that that is a large value and we can see whether this large value is still sufficient to solve our problem or not.

(Refer Slide Time: 48:29)

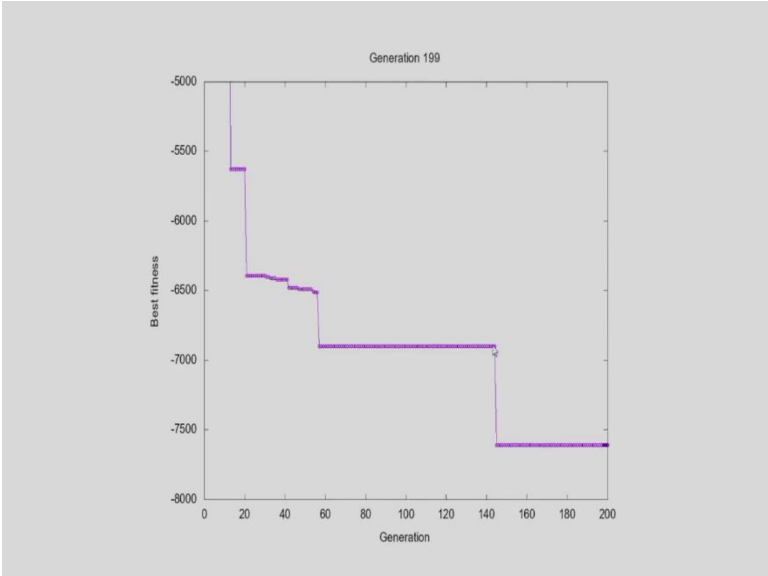


So, we are starting with the same set of initial population. Now the members are moving towards the feasible region and once they are in the feasible region they are also started moving towards the optimum solution that is shown in the red colour. Now these we all the solutions are now converged into the close area we have one solution although it is infeasible.

But close to the optimum solution and we have to see whether this particular method or the approach with R equals to 100 is able to find the optimum solution or not. it seems that there is a one infeasible or some infeasible solutions are there, the solution may not converge we have to see that see the solutions are start converging to the another infeasible region..

But not converging to the optimum solution. Meaning that R equals to 100 is still not sufficient for the for this particular problem that can differentiate infeasible and feasible solution. Let us look at the progress, so the y axis we have a best fitness and x axis is the number of generations it is started with a 10 to the power 8 which is a very large value and it has gone to the small value.

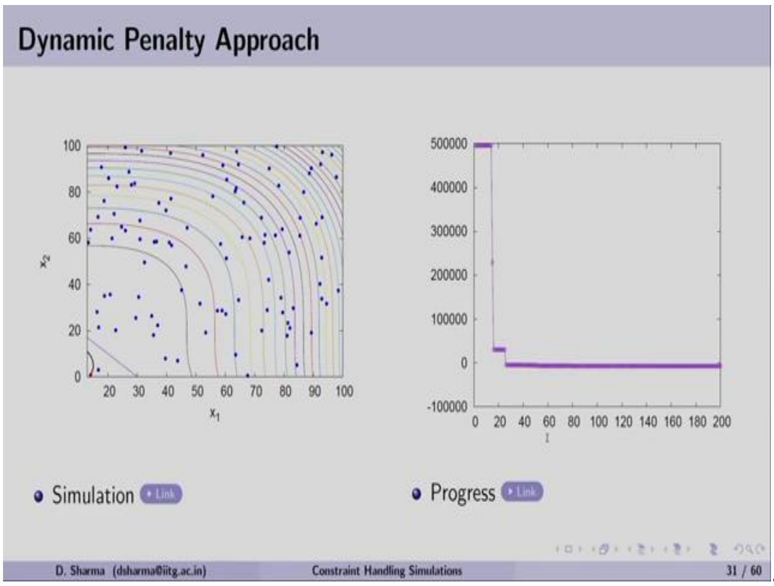
(Refer Slide Time: 49:59)



So, let us see the simulation now. In this case it is started with 10 to the power 8 in the focused. So, we have reduced the y axis just to focus how the progress is how the best fitness is progressing here.

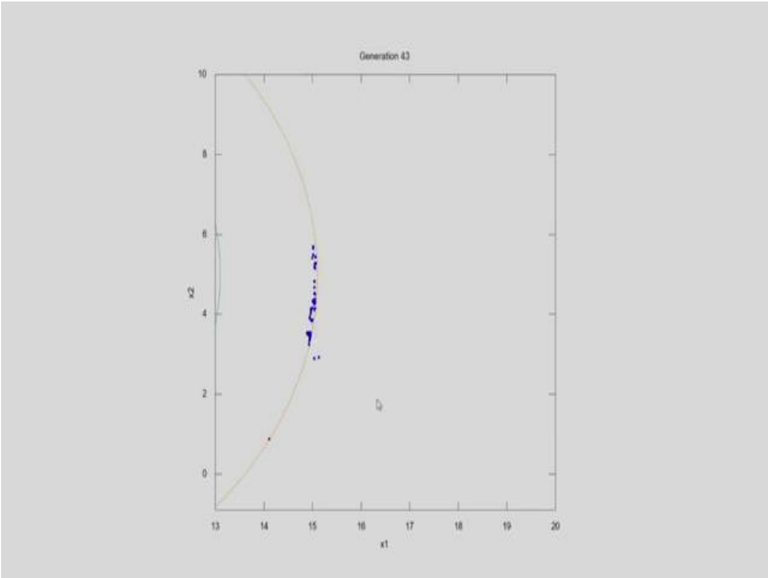
Now, we can see that somewhere close to 140 generation, this particular approach find another solution which is which is infeasible. But the penalty function method is showing less fitness as compared to the objective function value.

(Refer Slide Time: 50:35)



Now, we will see the we will see the simulation for dynamic penalty approach here.

(Refer Slide Time: 50:42)



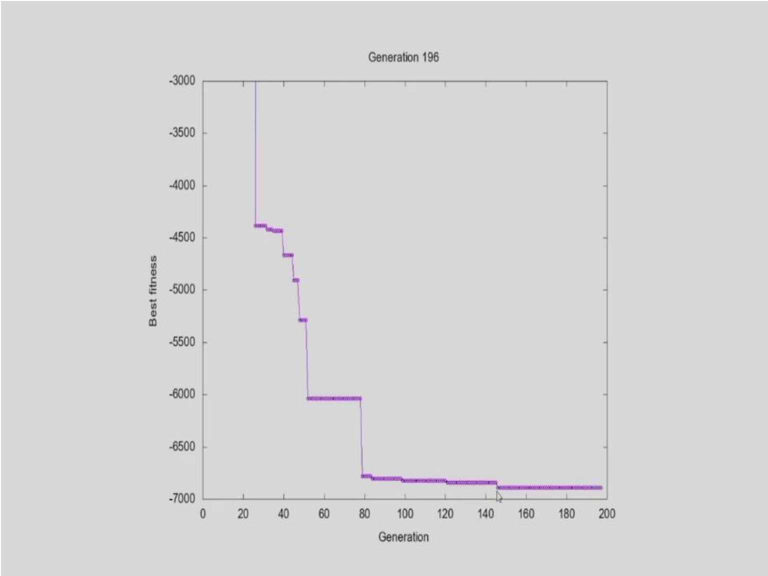
Now, here the solutions are again moving slowly and slowly as we know the value of c and alpha that we have chosen may not be sufficient and that is why the solutions are slowly converging into the feasible region.

Now we have to see that the dynamic penalty is still suffering the same problem as the static penalty, because the value of c and alpha should be sufficient to differentiate feasible and infeasible solution. Now we can see that there are some solutions which are quite close to the optimum.

Now, we have to find whether this dynamic penalty can find the optimum solution or not. But still we can see that the solution are still moving and we are close to the optimum, but somewhere in between we have this infeasible solution.

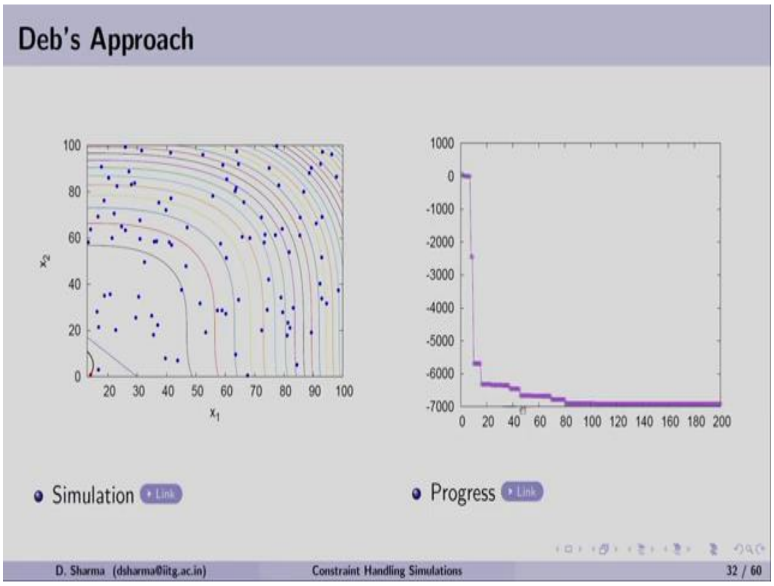
Now solution are start moving towards this infeasible solution, meaning that the value of c and alpha are not sufficient to penalize these infeasible solution. Now let us look at the progress although we know that this method has not found the optimum solution, we will see the simulation now.

(Refer Slide Time: 51:55)



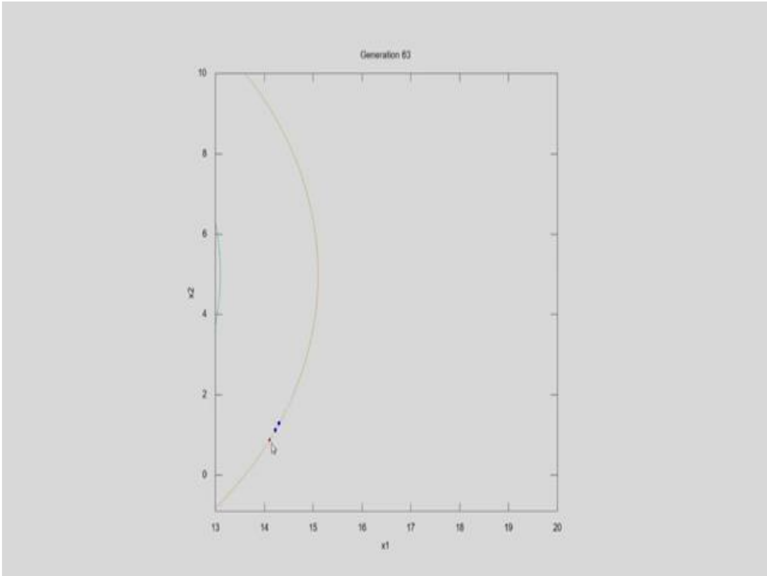
So, it is started with the very large value and now it is start moving towards reducing the objective function value. Now as you can see close to 120 generation we have are very close to the optimum solution. But afterwards the infeasible solution which was showing the better fitness that was where the solutions were on the other members who are started moving converging towards that solution.

(Refer Slide Time: 52:31)



Come to the last approach that is the Deb's approach. In this particular approach we have to see whether this approach is able to find the solution, when we are starting with the same initial solutions.

(Refer Slide Time: 52:41)

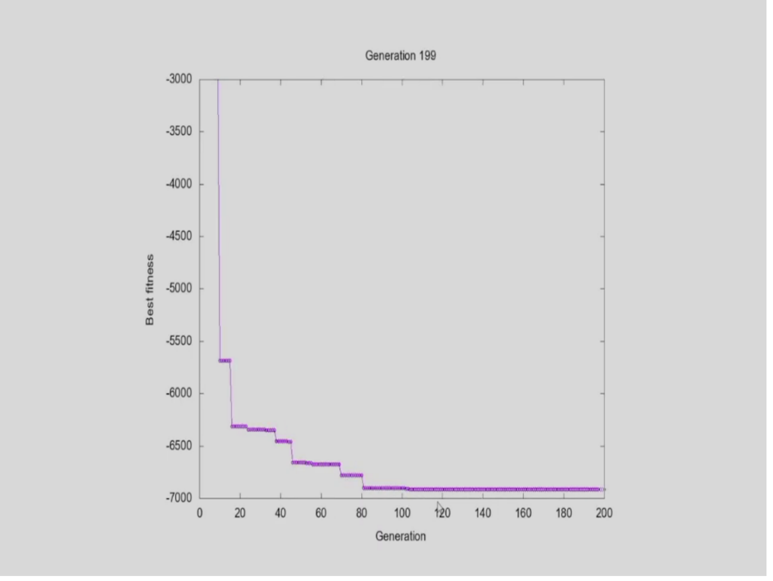


Now, as we can see quickly all the solutions are converge to the feasible region. Now along the constraint boundary these solutions as you can see these solutions are moving towards the optimum solution. Still we are little far from the optimum and we similar to the previous simulation.

We are not going to get any infeasible solution and now close to 100 close to 80 generations we have reached to the optimum solution. We have to see whether Deb's constraint handling approach with RGA is we will be finding another solution better than the optimum solution, we have to see the simulation now. So, 150 generations are already over and all members are converged to the optimum solution.

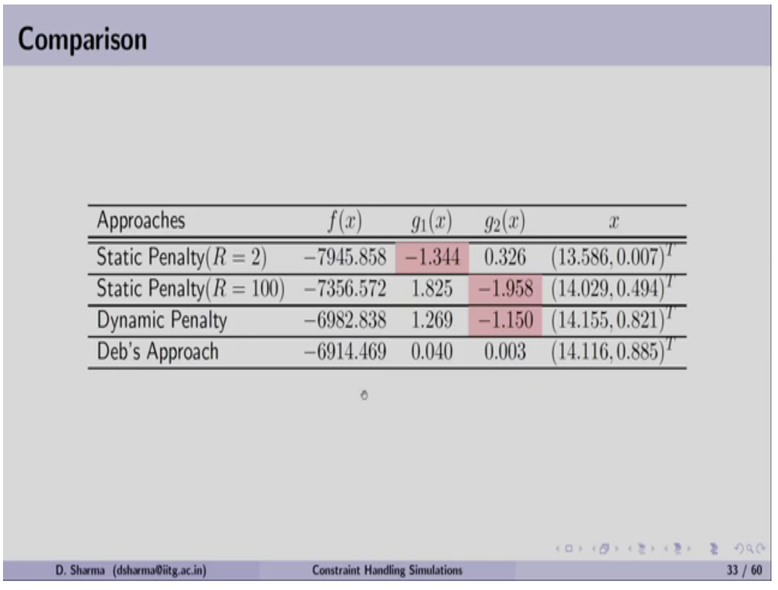
So, we have to see up to 200 generation whether the solutions are converged or not. So, as we can see with Deb's approach we are able to find the optimum solution for the given problem. Now look at the fitness it is started close to 0 and then it started improving it let us see the progress now.

(Refer Slide Time: 54:00)



So, it started somewhere close to 12 value and then after 10 generation there are improvement and close to 70 and 80 generation. The fitness is keep on improving and then after 100 and 3 generation the this approach was able to find the optimum solution with real coded g.

(Refer Slide Time: 54:26)



Approaches	$f(x)$	$g_1(x)$	$g_2(x)$	x
Static Penalty($R = 2$)	-7945.858	-1.344	0.326	$(13.586, 0.007)^T$
Static Penalty($R = 100$)	-7356.572	1.825	-1.958	$(14.029, 0.494)^T$
Dynamic Penalty	-6982.838	1.269	-1.150	$(14.155, 0.821)^T$
Deb's Approach	-6914.469	0.040	0.003	$(14.116, 0.885)^T$

So, let us compare these solutions now, the solutions which we have obtained using static penalty R equals to 2. Now here g 1 constraint is not satisfied which is an infeasible solution.

So, it found the fitness as we can see when we are minimizing this function values is better than the optimum solution. But the solution is infeasible meaning that we are not penalising our infeasible solution sufficiently. Similarly we have a static penalty R equals to 100 in this case g 2 is not satisfied and that is why we are getting a solution having a better fitness than the optimum, but it is an infeasible solution.

Similarly, the dynamic penalty also give us the infeasible solution for us because g 2 is not satisfied. If we look at the Deb's approach it is able to find the optimum solution and we can see that g 2 the constraint g 2 is active and the solution which we get is the optimum solution for the given problem.