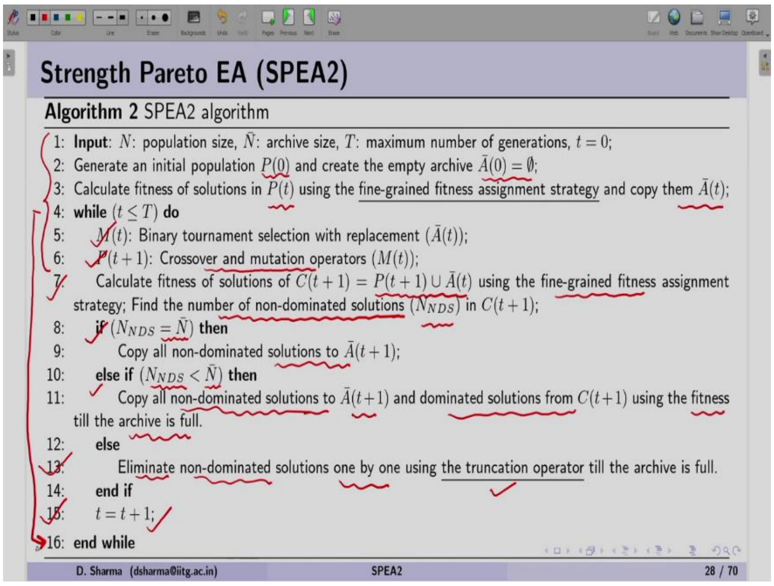
Evolutionary Computation for Single and Multi-Objective Optimization
Dr. Deepak Sharma
Department of Mechanical Engineering
Indian Institute of Technology, Guwahati

Lecture - 24
EC Techniques for Multi-Objective Optimization: SPEA2

Welcome to the part 2 of SPEA2 session. Till now we have discussed the fine grained fitness of SPEA2. As well as we have done the diversity operator that was kth nearest neighbor method.

(Refer Slide Time: 01:02)



So, if we look into the framework the generalized framework which we are following. So, for SPEA2 we have finished step from 1 to 6. In this one we started with the initial population and we kept our archive empty. Thereafter we assign the fitness to all the solution in the population and we copy all of them into the archive. Inside the while loop we perform binary tournament selection operator with replacement that created a mating pool.

In this mating pool we perform crossover and mutation using sb x crossover operator and polynomial mutation to create a another new population. So, this part we have finished in the last session now we will see how the environmental selection or the survivor stage that has been developed with SPEA2.

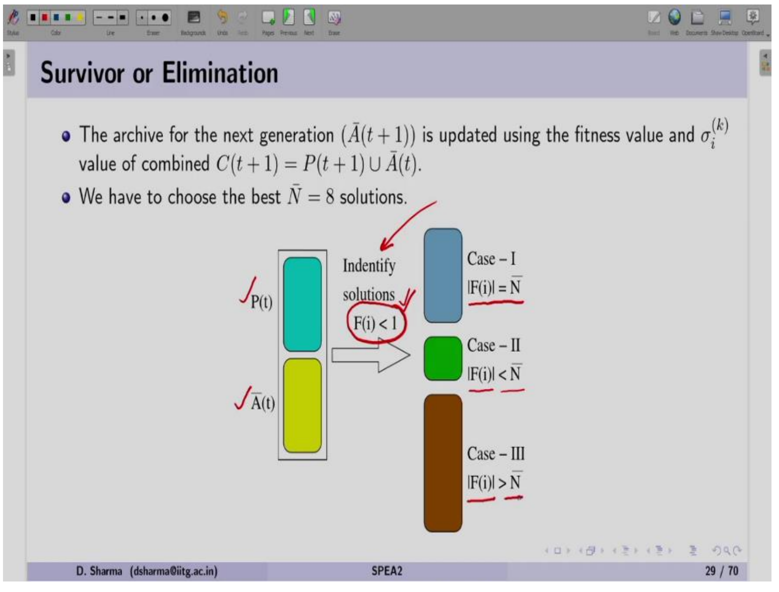
If we look into the algorithm, step 7 suggest that we have to first combine the new population and the current archive and then we will be assigning the fitness to the combine population. Thereafter we will see the number of non dominated solution which is currently represented as capital N and in the subscript you can say NDS. NDS stands for non-dominated solution.

When we are finding the number there are 3 cases. As we can see in step number 8 the number of non dominated solution can be equals to the size of archive. In this particular case, we copy all the non-dominated solution into the archive. If we have number of non dominated solution less than the size of an archive.

In this particular case, we first copy the non-dominated solutions to the archive and thereafter we are going to copy some dominated solution based on the fitness till our archive is full. In case the non-dominated solutions is more than the size of archive that is we can see in step number 13, then we have to eliminate.

Some non-dominated solutions. This we are going to eliminate one by one by using the truncation operator till the archive sizes full. In the step 15 we take an increment of 1 and we will be moving in the loop this is the while loop till our termination condition gets satisfied.

(Refer Slide Time: 04:26)

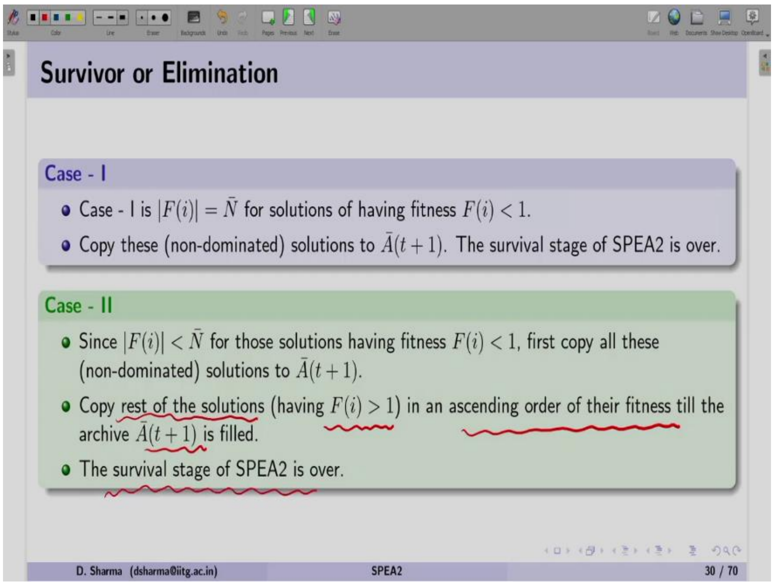


So, let us now understand how the survivor stage is developed with the SPEA2. We are going to discuss the 3 cases first and then we will take our example which we were solving in the previous session. Here first of all as we have to combine the offspring population and the current archive. So, that will be become the combine population for now.

Now, since we have to make the archive for the next generation. So, we have to fill or we have to choose the best solutions N bar solution into the archive. So, graphically if we see that we have the population, we have the archive, we are combining it and we are looking for the solution which have fitness smaller than 1. As per our earlier discussion, we know that this condition satisfies when the solution is non-dominated.

Once we have identified the non-dominated solution in the combined population so, there will be 3 cases as we discussed. First case is the number of non-dominated solution is equals to the size of archive, second case is the number of non-dominated solution is smaller than the size of archive and the third case is number of non-dominated solution is greater than the size of archive.

(Refer Slide Time: 06:08)

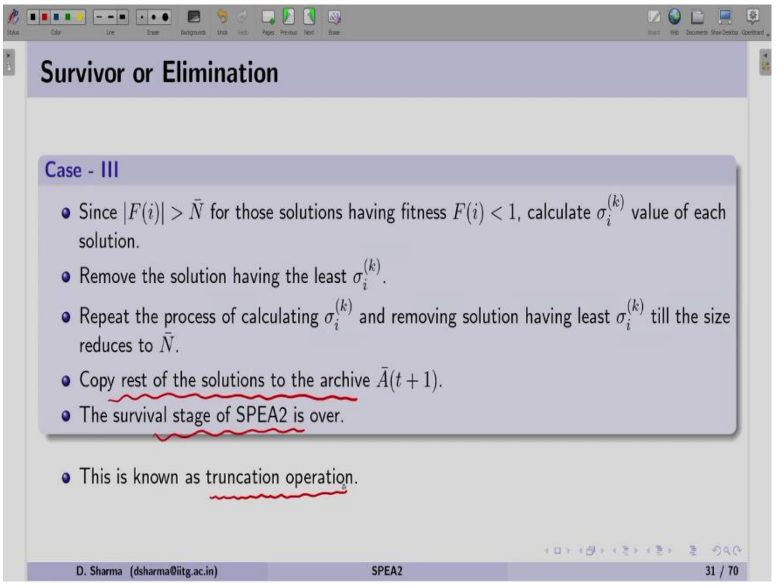


So, let us see how we can do it. So, in case I when the number is when the number of non-dominated solution is the same as the archive size. Then we copy all these non-dominated solution to the archive and at this stage the survivor stage of SPEA2 is over. The 2nd case is when the non-dominated solution is less than the size of archive.

So, in this case first of all let us copy all these non-dominated solutions to the archive. From this particular condition, it says that still there is a size left in the archive in which we can copy some solution. So, we are going to copy the solutions having fitness F i greater than 0, in an ascending order of their fitness value till the archive is full.

So, in this case some dominated solution will be copied. So, that the archive will remain full. At this stage the SPEA2 survivor stages over.

(Refer Slide Time: 07:25)

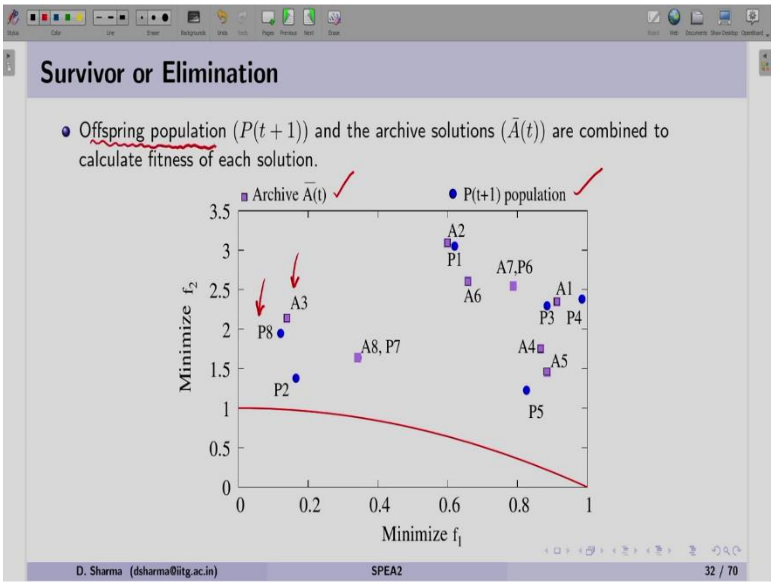


Case III is when the number of non-dominated solution is greater than the size of archive. In this case we are going to calculate sigma i k. As per our earlier discussion this sigma i k is calculate using kth nearest neighbor approach.

So, once we calculated the sigma i k we will find the solution which is having the least sigma i k value in the set of non-dominated solution. So, this particular solution will be removed and this process we will be repeating till the size reduces to the archive size. When we are removing such solutions one by one then rest of the solutions we are going to copy in the archive and that finishes the survivor stage of SPEA2.

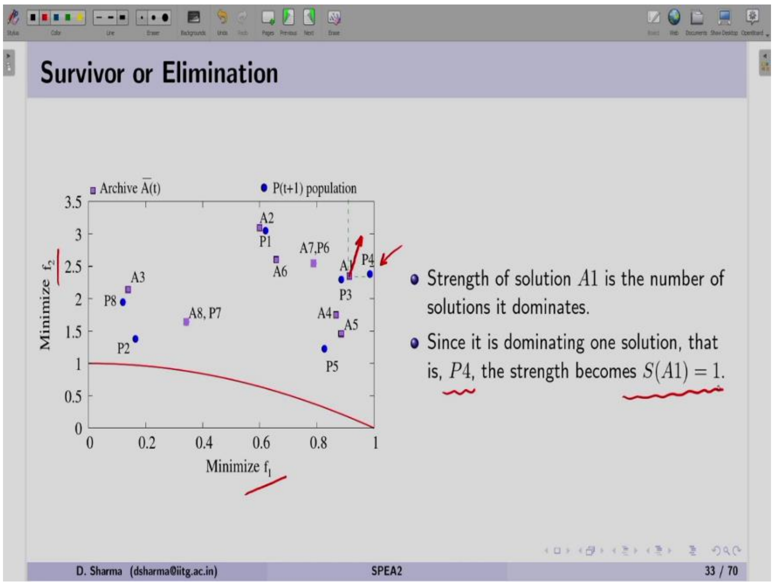
The operator which we have discussed for case III is called as truncation operator, where we are removing the solution one by one. So, basically we are removing the non-dominated solution one by one based on the sigma i k. The less value of sigma i k signifies that this particular solution is more crowded. So, therefore it should be removed. So, now we will see a that in the current example which we are solving how we can how we can update the archive.

(Refer Slide Time: 09:14)



Now, if we look into the figure here we have combined both the population that is the archive and the population which we get after crossover and mutation. Here the notation A stands for the archive member and the notation P stands for the population which is created after crossover and mutation. So, therefore we are representing or referring this particular population as an offspring population.

(Refer Slide Time: 09:51)

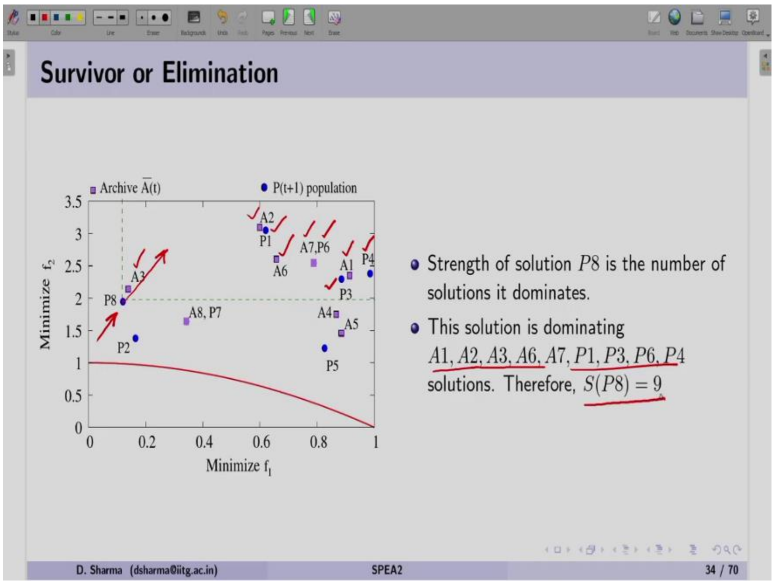


Now, we know that if we want to assign a fitness to any solution in the combine population, we have to first calculate the strength of a solution. As an example I have taken few solutions to understand the how we can calculate the strength followed by the raw fitness.

If we look into the figure now in this particular figure let us choose solution A1, in this particular solution A1 in order to calculate the strength of this solution we have to find the number of solution it dominates.

We know that since it is a minimization problem. So, both the objective are minimization as well as it is a 2 objective problem. So, if we take A1 as our reference and if we look into the first quadrant we can see how many solution are dominated by A1. Looking into the figure we can see that the solution P4 is dominated by A1. Since A1 is dominating one solution the strength of A1 is 1.

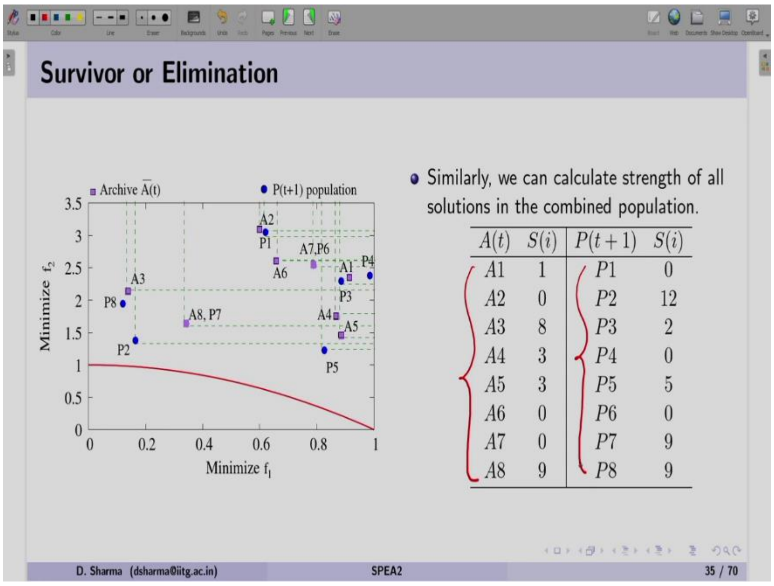
(Refer Slide Time: 11:16)



Let us take another case, in this particular case we have chosen the solution P8. Now if we are going to calculate the strength we can see that if I take solution 8. And if we look into the first quadrant with respect to the solution P8 solution number A3, A2, P1, A 6, A7, P6, A1, P3 and P4 these are the solutions which are dominated by P8.

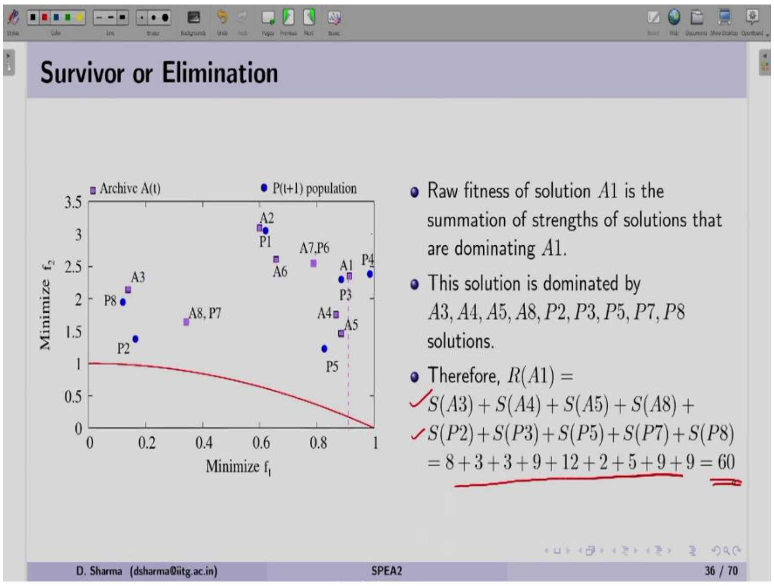
So, if we can see the complete list on the right-hand side and if we count the number of solutions here. So, the strength of P8 will become 9 because it is dominating 9 solution.

(Refer Slide Time: 12:12)



Similarly, we can calculate strength of all the solution in the combined population. So, if we see the figure on the left hand side. So, the first quadrant of each solution are shown and by looking into the first quadrant we can find that the strength of the members in the combined population that are given in the table.

(Refer Slide Time: 12:48)



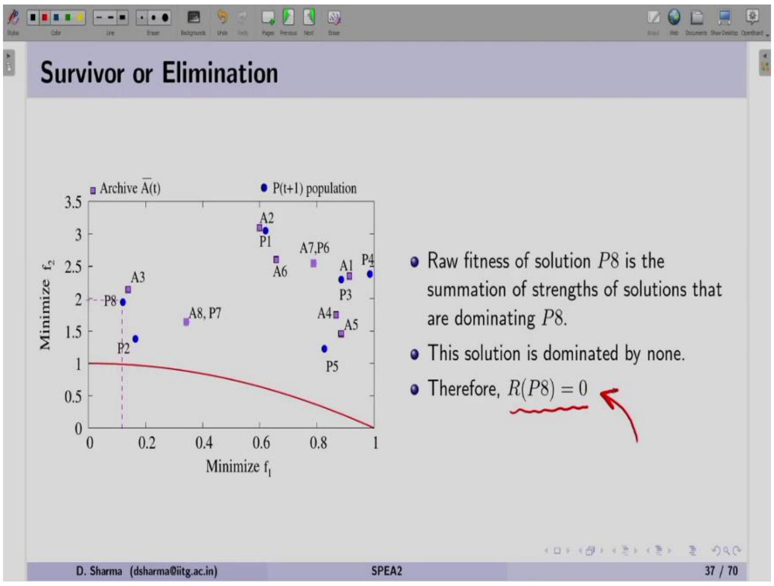
Once the strengths are given or calculated the next step is we have to find the raw fitness of each solution. As per our earlier discussion raw fitness is the summation of the strength of those solutions which are dominating the given solution.

So, we will see few examples here of the raw fitness. If we take solution A1 in this case since it is to objective minimization problem and we take A1 as our reference and if we look into the third quadrant with respect to solution A1 the solutions which are lying in the third quadrant we can see that A3, P 8, P2 is 8, P7 similarly P3, A4, A5 and P5.

So, these are the solutions which are dominating A1, meaning that once we get the list of solutions which are dominating A1 we have to take the summation of the strength of all such solutions. So, here we have written the strength of those solutions by adding them we get the strength of solution A1 is 60.

So, we remember that if the raw fitness of any solution is a large number meaning that it is dominated by many solution and therefore, it is not a good solution and the number 60 for solution A1 represents the same. Let us take the another solution to find out the raw fitness.

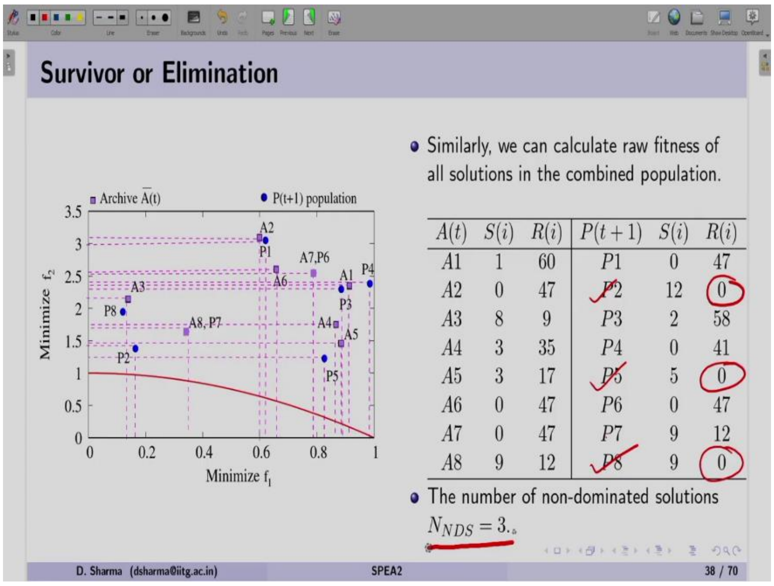
(Refer Slide Time: 14:48)



Here we are choosing say solution P8 and looking at the figure on the left-hand side and if we take P8 and with respect to this solution. If we look into the third quadrant for the minimization problem we can see that there is no solution which is dominating P8. Meaning that the raw fitness of P8 will become 0. Moreover we also know that if any solution is having raw fitness equals to 0 that solution will become the non-dominated solution in the current set of current set of solution or the population.

If we are going to follow the same procedure for all the solution we can calculate the raw fitness for each and every solution.

(Refer Slide Time: 15:48)

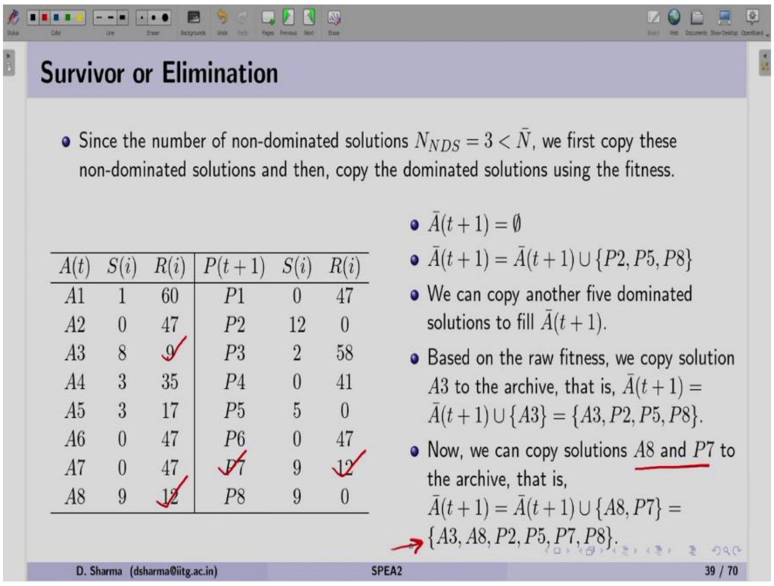


So, here we can see in the figure that, we have drawn the third quadrant for each and every solution. And accordingly when we are going to make a summation of the strength of those solutions we can find the raw fitness. So, the table on the right-hand side shows the solution it is strength as well as it is raw fitness.

Similarly, for the population it is strength and the raw fitness. It is noted that we are calculating the strength and the raw fitness of a combined population. From this particular table we can see that population the population member P2, P5 and P8. They have a raw fitness equals to 0, meaning that these solutions are going to be the non-dominated solution.

So, the number of non-dominated solution in the current set is 3. Now we are in a position to apply the survivor stage of SPEA2; because we know how many are the non-dominated solution.

(Refer Slide Time: 17:15)



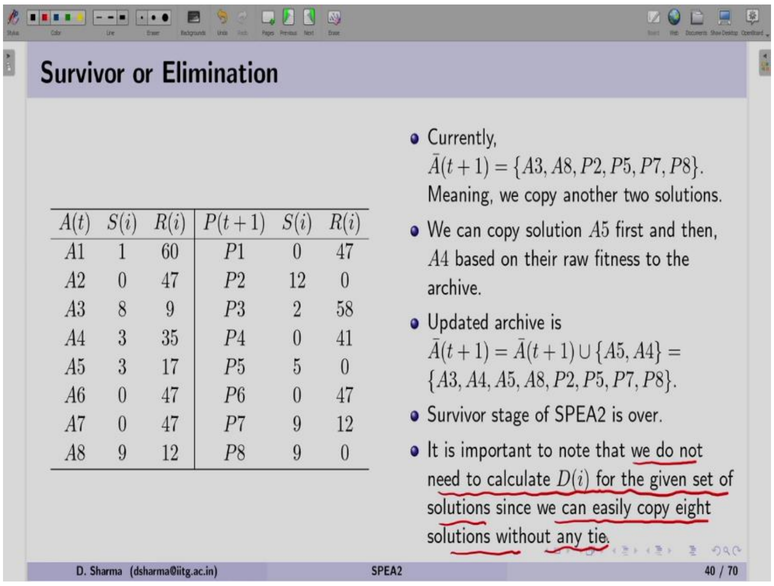
So, in the current set we since we have number of non-dominated solution 3 which is less than N bar which is the size of an archive. So, the first step is let us copy all these non-dominated solution and thereafter we are going to copy dominated solution using the fitness.

So, initially the archive for the next generation is kept empty and thereafter we are going to copy the non-dominated solution. Once these non-dominated solutions are copied so we know that currently the size is 8 so we can copy another 5 dominated solution. Now if we look at the raw fitness of this solution.

So, solution P2, P5 and P8 are copied. Now we will see which solution has the least fitness. From the given table we can see the solution 3 has the least fitness raw fitness. So, therefore, A3 is going to be copied in the archive. So, the updated archive will include A3 as we can see on the right-hand side.

Now, we still we have a space for 4 and we have copied 4 solution. So, after copying three. So, the next best fitness we can see with solution A8 and a P7 that is 12. Now since we have a space so we are going to copy these two solution into the archive. And now this is the current archive which has copied 3 plus 1 plus 2 basically 6 solutions.

(Refer Slide Time: 19:16)



Since we have copied 6 still we have a space of A2 solution to fill the archive completely. As of now the solutions having the fitness the raw fitness of a 12 are copied. So, if we look into the next best fitness we have solution A5 having a fitness of 17 and so A5 and then followed by A4. So, since these two solutions can be accommodated.

So, we are going to add them into the archive population archive and these are the solutions which are copied into the archive for the next generation. Since the archive is already full so we can say the survivor stage of SPEA2 is over. Now it is important to note that we do not need to calculate D i for the given set of solutions. Since we can easily copy 8 solutions without any tie.

It means that when we are calculating the fitness of any solution. So, the fitness is made of the raw fitness plus the diversity D i. In the current scenario we can see that even without calculating the diversity D i we can copy the 8 solution without any tie.

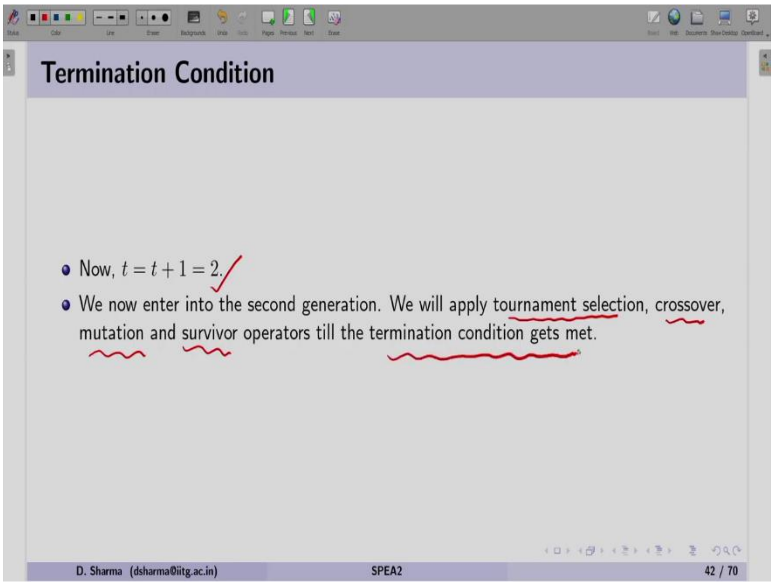
However, if any scenario comes where there is a tie and we do not know how many which solution we have to copy in order to fill the archive. Then we have to find out this D i and then we are going to copy the solution based on the fitness. So, that finishes the survivor stage of SPEA2.

(Refer Slide Time: 21:30)

And if we look into the figure now here you can see that we have we started with the initial archive as shown in the left-hand side figure and thereafter after 1 generation we can see the updated archive. On the same scale of f 2 and f 1 we can see that the solution after 1 generation they started moving towards the pareto optimal front and that is what we are expecting.

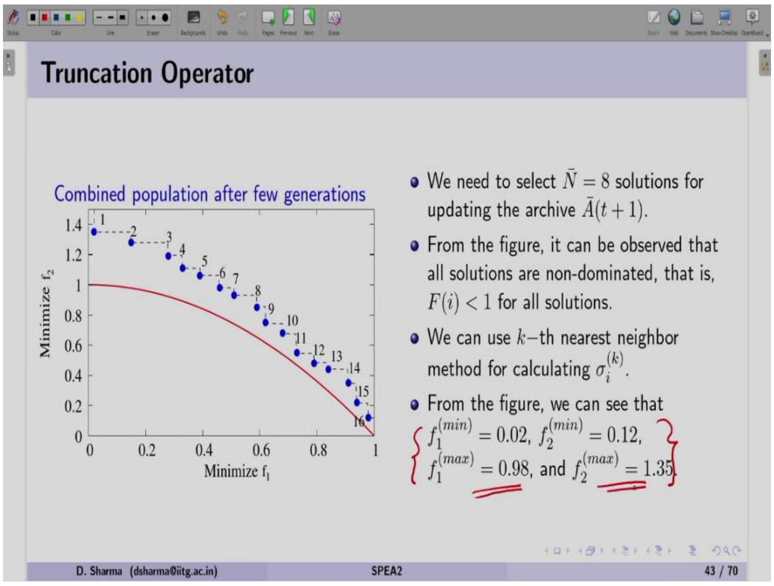
That generation by generation the archive will be updated and the non-dominated solution will be emphasized. And these solutions will be moving towards the pareto front and finally, converges on the front. Once the survivor stage is over we increase the counter by 1.

(Refer Slide Time: 22:30)



So, we can see here that we increase the counter by 1. So, we get t is equal to t plus 1 2 and still we are in the while loop of generation. So, in the 2nd generation again we are going to apply the same set of operators such as tournament selection operator, crossover and mutation and then finally, the survivor stage we are going to apply till the termination condition get satisfied.

(Refer Slide Time: 22:59)



Now, in the current example which we have taken the population was evolved in such a way that we can see that the case 2 where the number of non-dominated solution was less than the archive size was encounter. So, in that case we copied the non-dominated solution followed by the dominated solution in order to fill the archive. But what we expect is when we are running SPEA2 for further generation then we may realize that there are many non-dominated solution in the combined population.

So, at that particular stage we have to copy N bar equal to 8 solution for the given problem. So, as a representation basis we have taken one more example in order to understand the case 3 when the number of non-dominated solution is more than the archive size. We can see from the slide here that we will be understanding the truncation operator for case 3.

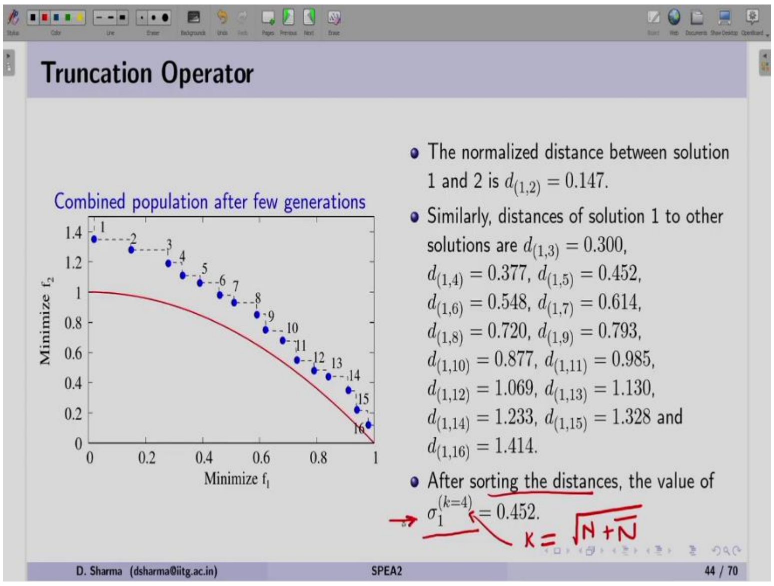
So, the in the figure which is shown in the left-hand side here we assume that this is the combined population after few generations. So, here we have taken say all 16 solutions as you can see in this particular figure. Now in the current situation as we know we have to choose the best N bar solution which is 8 for updating the archive.

So, from the figure it can be observed that all the solutions are the non-dominated solution. So, here as a special representation we have shown this particular dashed line to show that all these solutions are the non-dominated solution. So, since the number of non-dominated solution currently is 16 which is greater than the N bar. So, we are going to use the truncation operator which is the case III.

In this case we are going to use k-th nearest neighbor approach and in this method we will be calculating the sigma i k value. So, let us see how it is working. So, first of all we have to find out the minimum and maximum in both the objective. It is only because we want to calculate the distance in the normalized objective space.

So, from this figure f 1 min and f 1 max are shown are written here. Similarly f 2 min and f 2 max are given here.

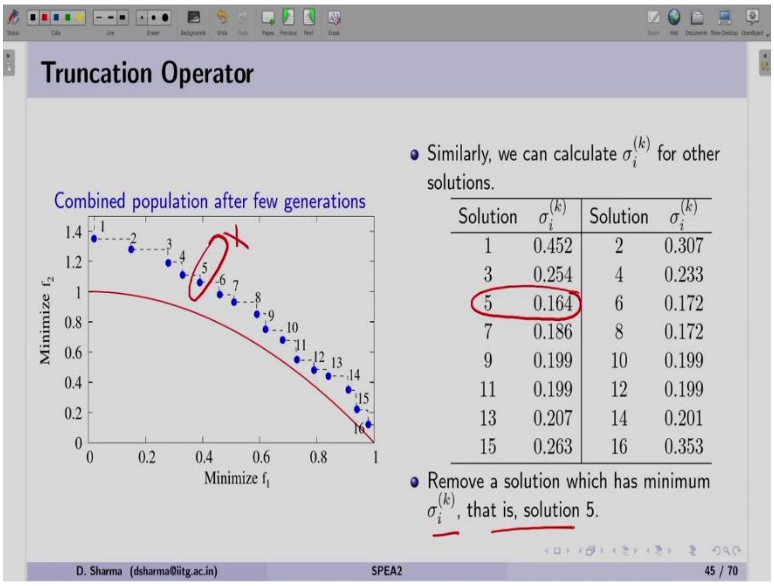
(Refer Slide Time: 26:09)



Now, let us calculate the normalized distance between solutions 1 and 2. When we are going to calculate this distance is coming out to be 0.147. So, the distance means we are talking about the Euclidean distance in the objective space. Similarly, I can find the distance of a solution 1 with the other solution. So, the representation such as d 1, 3 meaning the distance between 1 and 3.

Similarly, 1 and a 4, 1, 5 etcetera. Now after sorting the distance we can find the k-th element. So, sigma 1 k equals to 4. So, we remember that this k, k-th element we find with the rule that k is equal to under root of N plus N bar. N is the population size and N bar is the archive size. So, after sorting we get the sigma 1 k value

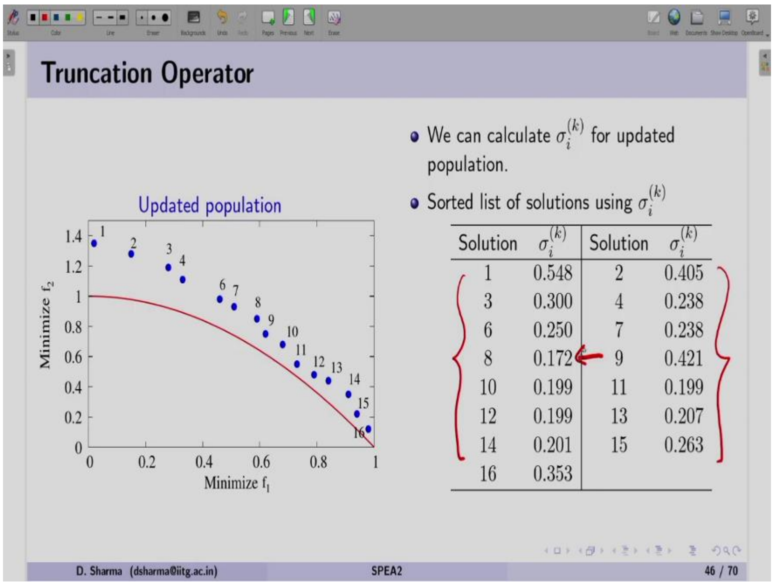
(Refer Slide Time: 27:20)



Similarly, if we are going to calculate sigma i k values for other solutions. So, we have tabulated all these values. So, for all 16 solutions sigma i k values are given. As per the truncation operator we have to find which is having the minimum sigma i k value. So, from this particular table we can see that the solution 5 has the minimum sigma i k value and therefore, it should be removed.

So, here if we look into the figure on the left-hand side this particular solution 5 should be remove from the current set.

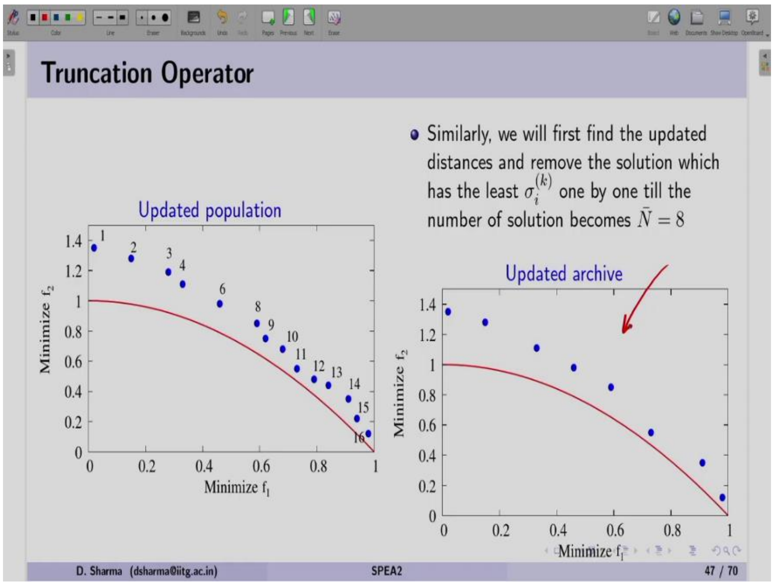
(Refer Slide Time: 28:11)



If we remove it, we can see the updated figure right now. In this figure you can see the solution 5 is missing. Now since the solution 5 is removed we have to again calculate the sigma i k value for the updated population.

So, it is an important point that as soon as we are deleting any solution, we have to recalculate sigma i k value for the updated population given in the figure. So, let us this these are the again sorted list of the sigma i k value for the rest of the 15 solution. From this particular table we can see the minimum value is corresponding to solution number 8 that has to be removed. So, we are going to remove the solution number 8.

(Refer Slide Time: 29:10)

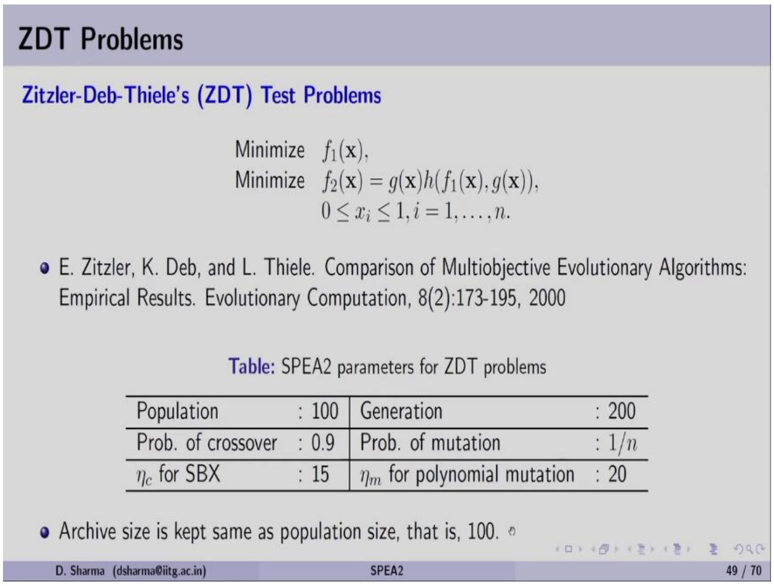


So, what we can expect? At the last is when we will be removing these solutions one by one. So, we are going to get the diverse set of solutions here that will be updating our archive. So, here these are the diverse solutions which we can see after removing the solution one by one using the sigma i k value.

Now, we will be looking at the simulations of SPEA2. So, till now we have done the hand calculation for one generation of SPEA2. We also took a case of truncation method in which we are removing the solution one by one when the number of non-dominated solution is more than the archive size.

Now, we will be testing this SPEA2 on the set of multi objective optimization problem under simulation.

(Refer Slide Time: 30:18)



$$Minimize \quad f_1(x)$$

$$f_2(x) = g(x)h\big(f_1(x), g(x)\big)$$

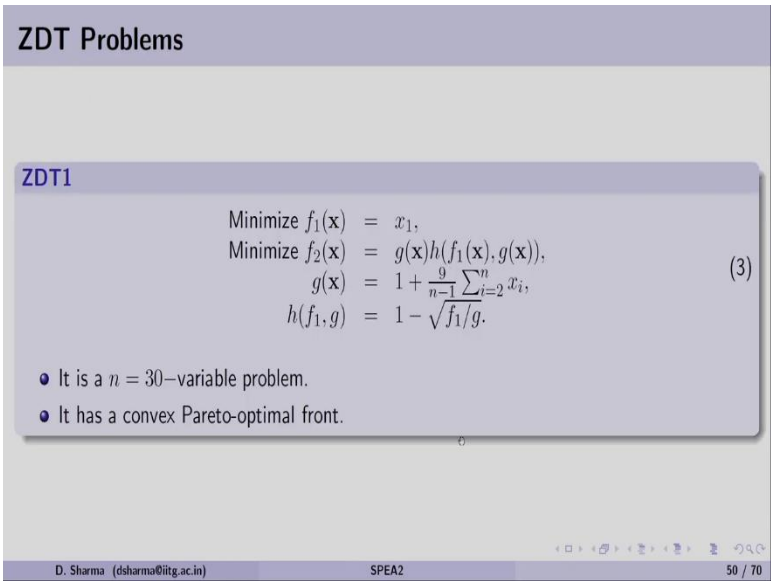
$$0 \leq x_i \leq q\ 1, i = 1, \ldots, n.$$

So, we will see the simulations now. We will take the first case first problem set as a ZDT problems. These problem says that we will be minimizing the f 1 objective and the f 2 objective will be written in terms of g and the function h which is again made of f 1 and g. All the variables are lying between 0 to 1.

So, the details of the ZDT problems can be found in the given paper. The parameters which we have set for ZDT problem we have taken population size 100. If we are going to run for 200 generation crossover probability 0.9, mutation probability 1 by n eta c 15, eta m 20 and since we have the archive in SPEA 2.

So, we are keeping the same archive size as the population. So, currently it is 100.

(Refer Slide Time: 31:24)



$$Minimize \quad f_1(x) = x_1$$

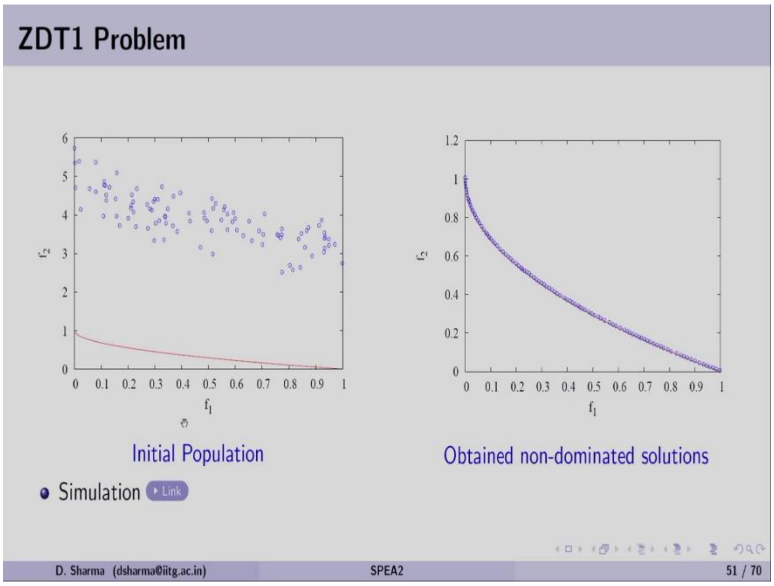
$$Minimize \ f_2(x) = g(x)h(f_1(x), g(x))$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$$

$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}}$$

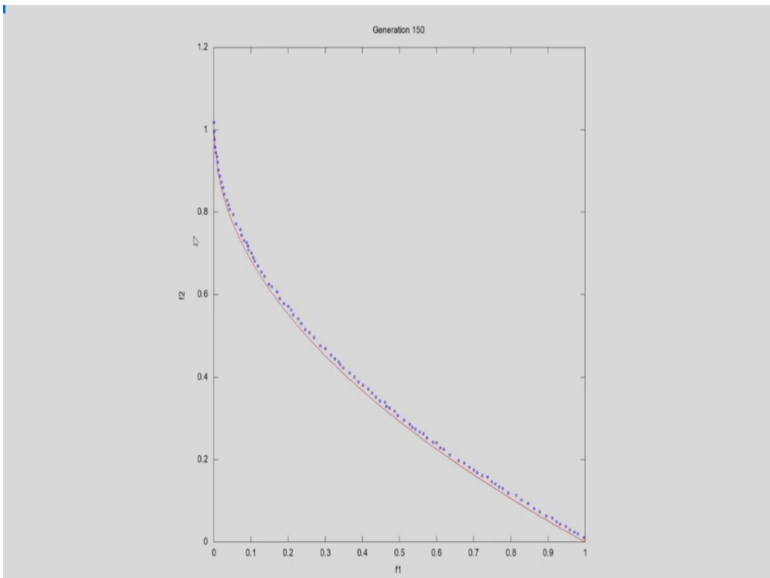
Let us begin with the first problem which is known as ZDT1 problem. Here the f 1 objective is x 1 and the f 2 objective which is made of the function g and h are given here. This is a 30 variable problem and since it is a mathematical problem. So, we know where is the pareto optimal front and it is a convex pareto front.

(Refer Slide Time: 31:55)



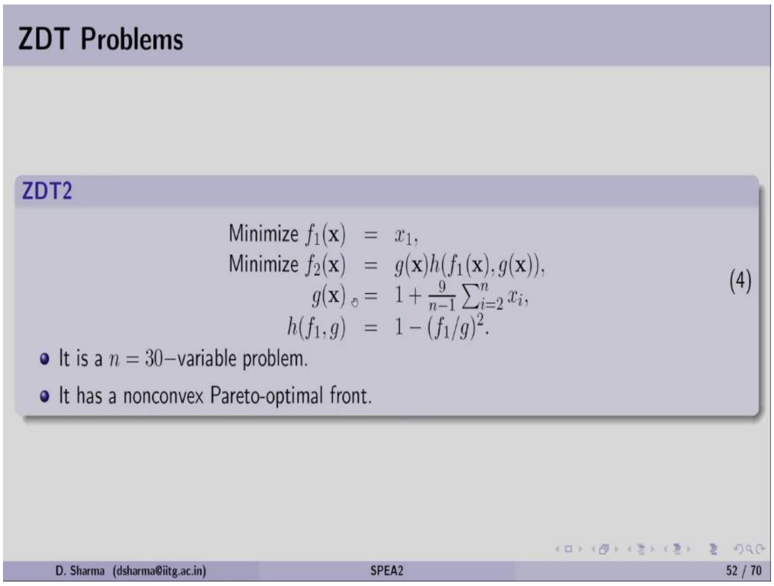
We can see this is the initial population and the red line represents the pareto front for the given problem. So, the solutions are distributed and they are far from the front and after 200 generations we can see the solutions are well converged and there is a diversity as well among the solutions that are converged on the pareto front.

(Refer Slide Time: 32:23)



Let us look into the simulation now. Here we can see these solutions are moving towards the front. And when they are close we can see the solutions have distributed nicely over the Pareto front which is shown in the red line.

(Refer Slide Time: 32:42)



$$Minimize \quad f_1(x) = x_1$$

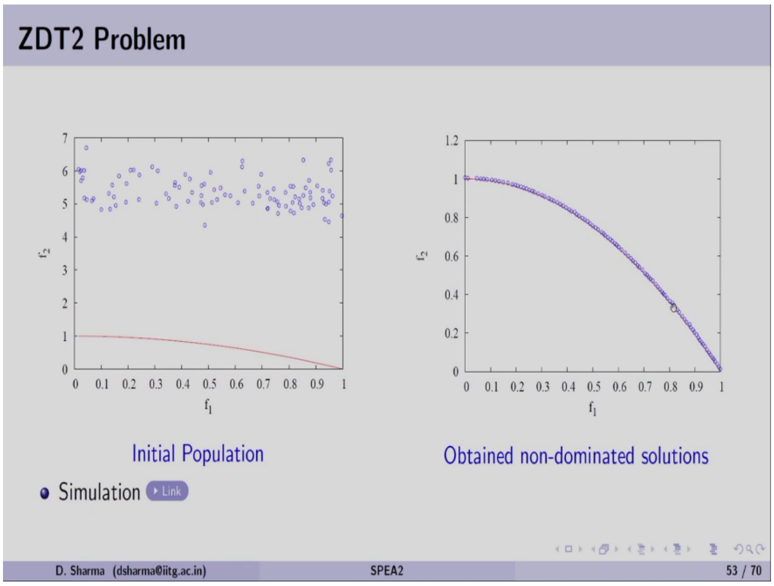
$$Minimize \quad f_2(x) = g(x)h(f_1(x), g(x))$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$$

$$h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^2$$

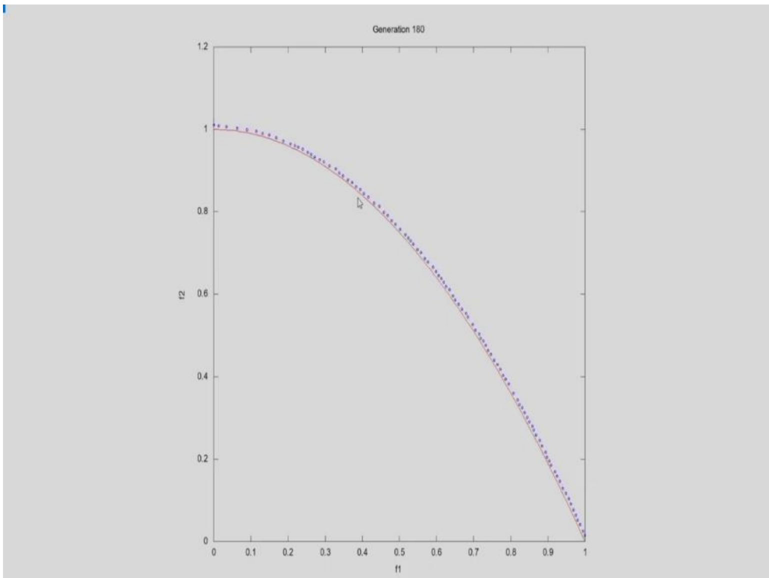
Now, we will go to the another problem which is called ZDT2 problem. In this case we can see that the form of g has been it has been the same, but the h is different and because of this h the f 2 and f 1 nature is going to be different. And for the given problem we are going to get the non-convex pareto optimal front this problem is also a 30-variable problem.

(Refer Slide Time: 33:15)



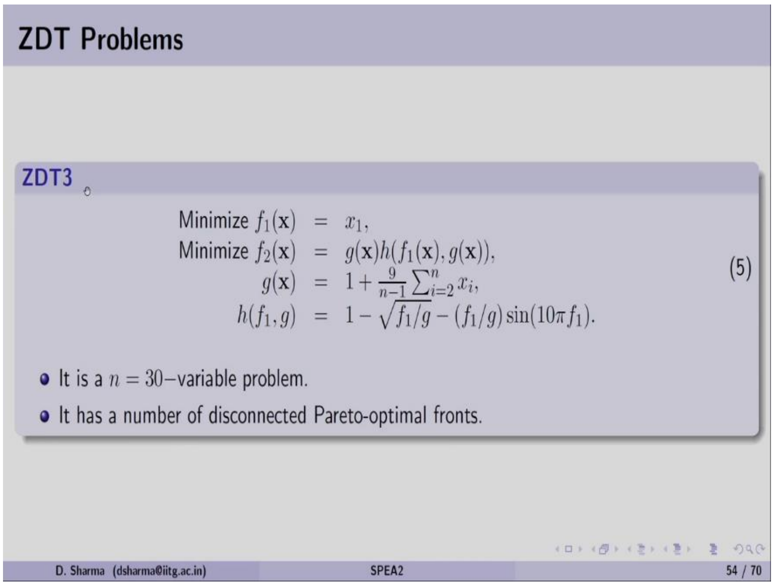
So, as we can see the initial population is started a little far from the pareto front and after running the algorithm for 200 generation the solutions are converged. So, we can see that SPEA2 can very well solve the problem which has non-convex pareto optimal front. Let us see the simulation for the same problem.

(Refer Slide Time: 33:41)



So, here we can see the solutions are kind of at the 2 corners. But when the solutions are moving towards the front they are nicely distributed and slowly and slowly they are moving towards the pareto optimal front which is shown in the red line.

(Refer Slide Time: 34:00)



$$Minimize \quad f_1\,(x) \;=\; x_1$$

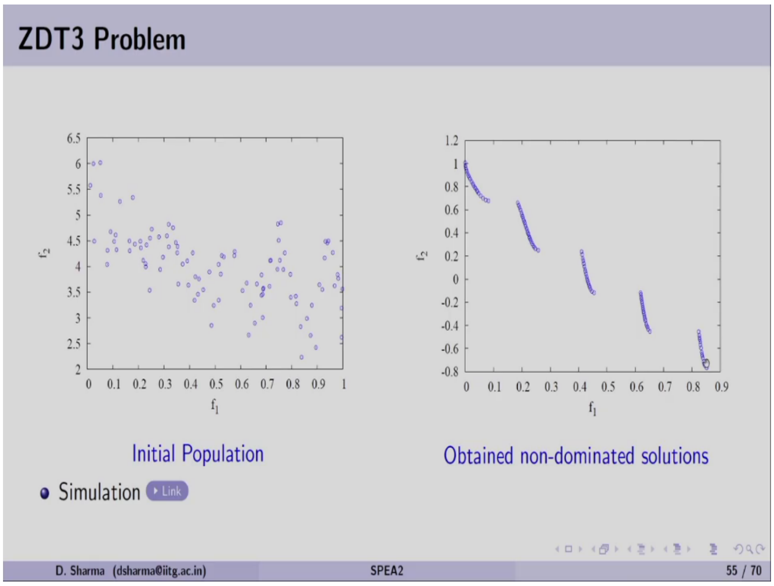
$$Minimize \quad f_2\,(x) \;=\; g(x)h(f_1(x), g(x))$$

$$g(x) \;=\; 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$$

$$h(f_1, g) \;=\; 1 \;-\; \sqrt{\frac{f_1}{g}} \;-\; \left(\frac{f_1}{g}\right) sin\;(10\pi f_1)\;.$$

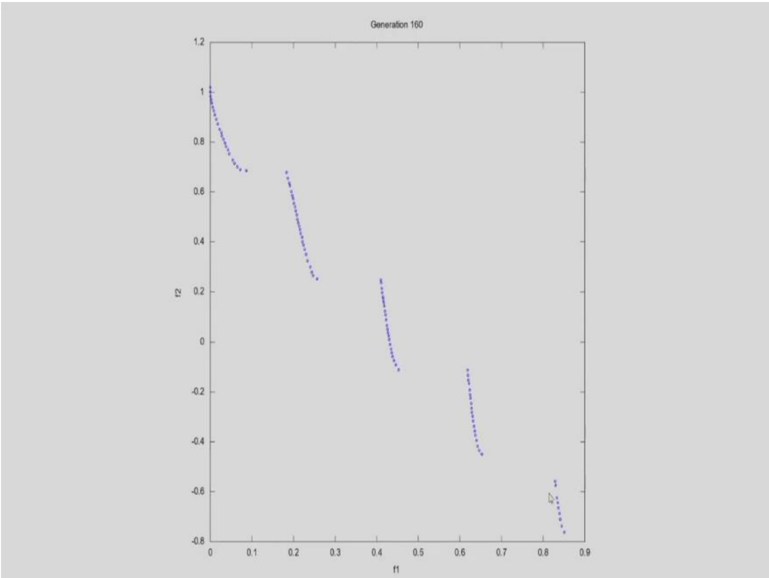
Moving to the 3rd class of a problem which is ZDT3 here again the form of f 1, f 2 and g are the same. But h has been changed and in this case since the sin x function is included. So, we can expect some disconnected pareto optimal front for the given problem. This problem is also solved for 30 variable problem.

(Refer Slide Time: 34:26)



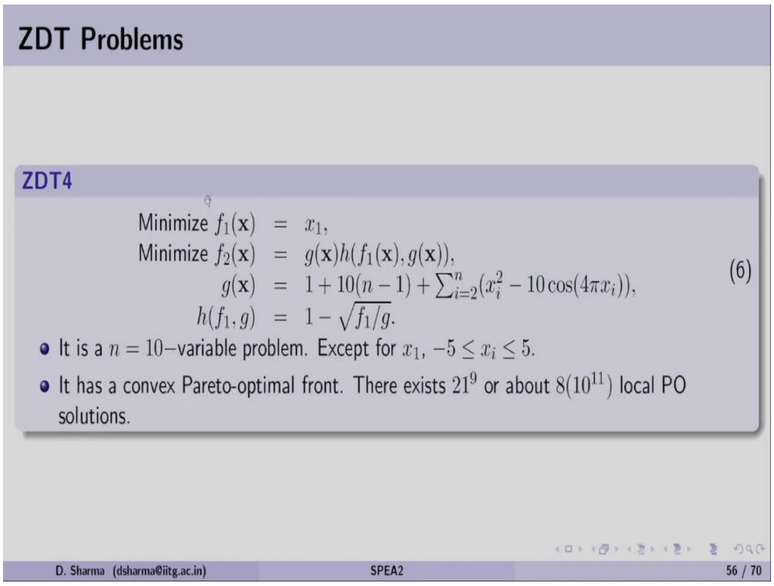
And here the solutions are distributed as you can see the range of f 2 is from 6.5 to 2. After 200 generation the solutions are distributed on the disconnected front and the range of f 2. We can see from 1 to minus 0.8 and f 1 is lying between 0 to 0.9. If we see how these solutions are moving towards the pareto front.

(Refer Slide Time: 34:58)



Now, we can see initial at the beginning itself these solutions are distributed in the disconnected front and slowly and slowly they are moving towards the pareto front. And at the bottom we can see we are going to get all the 5 disconnected front for the given problem.

(Refer Slide Time: 35:18)



$$Minimize \quad f_1\,(x) \; = \; x_1$$

$$Minimize \quad f_2\,(x) \; = \; g(x)h(f_1(x), g(x))$$

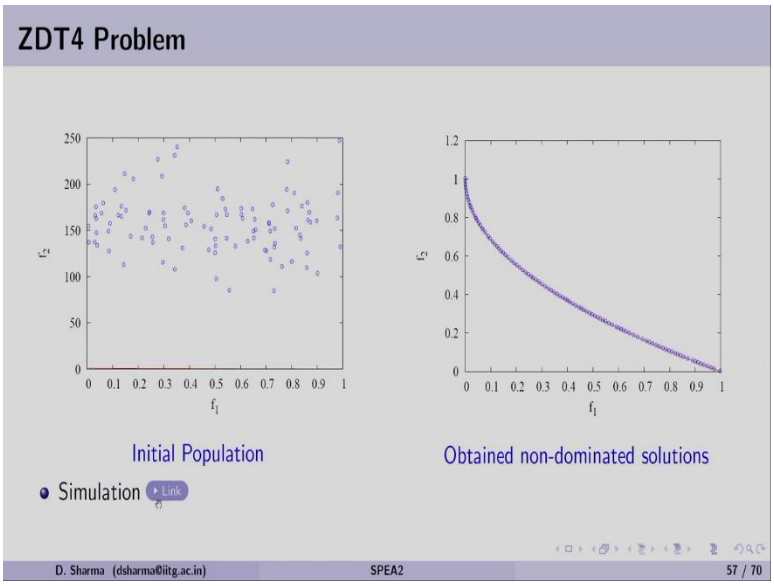
$$g(x) \; = \; 1 + 10(n-1) \, + \sum_{i=2}^{n}(x_i^2 \; - \; 10\,cos(4\pi\, x_i))$$

$$h(f_1, g) \; = \; 1 \; - \sqrt{\frac{f_1}{g}}.$$

After ZDT3 we have ZDT4 problem in the ZDT4 problem we can see the form of g is changed and h is the same as ZDT1. In this case we will be solving a problem with a 10 variable. Here except for x 1 meaning that x 1 will be lying between 0 to 1 and the rest of the solutions will be lying between minus 5 to 5.

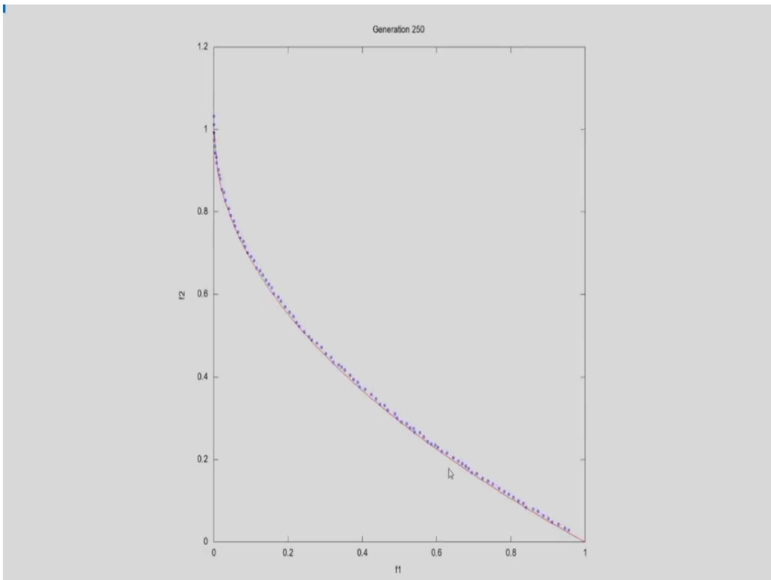
The problem which we are going to solve has a convex pareto optimal front and there will be lot of local pareto optimal solution. Such as 21 to the power 9 or about 8 into 10 to the power 11 local pareto optimal solutions.

(Refer Slide Time: 36:11)



The initial population for ZDT1 can be seen that it is started in f 2 around 250 to 100 and f 1 is in the range of 0 to 1 always. And after running this particular algorithm for many generations. So, we can see that the solutions are well converged to the pareto front.

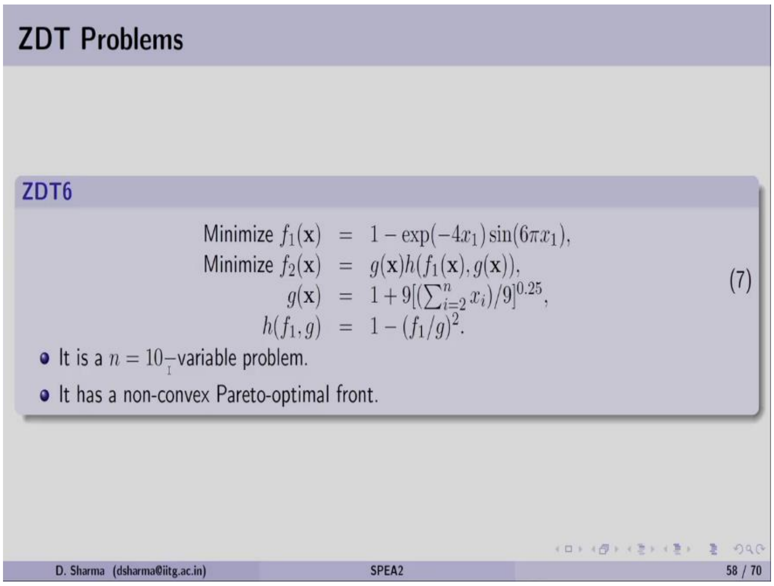
(Refer Slide Time: 36:39)



So, we can see how SPEA2 is performing on the given problem as we can see the solutions are move to the one of the corner and as soon as these solutions are moving towards the pareto front. So, these solutions get distributed along the front and slowly and slowly these solutions will be converged on the pareto front.

For the given problem since it is complex we have run SPEA2 for 300 generations to get the pareto to get the solutions converged on the pareto optimal front.

(Refer Slide Time: 37:18)



$$Minimize \quad f_1(x) = 1 - exp(-4x_1)sin(6\pi x_1)$$

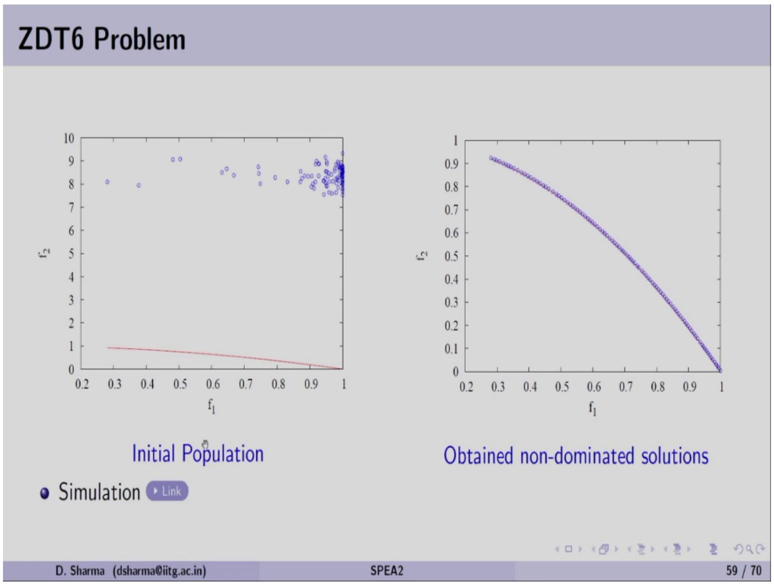
$$Minimize \quad f_2(x) = g(x)h(f_1(x), g(x))$$

$$g(x) = 1 + 9\left[\frac{(\sum_{i=2}^{n} x_i)}{9}\right]^{0.25}$$

$$h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^2.$$

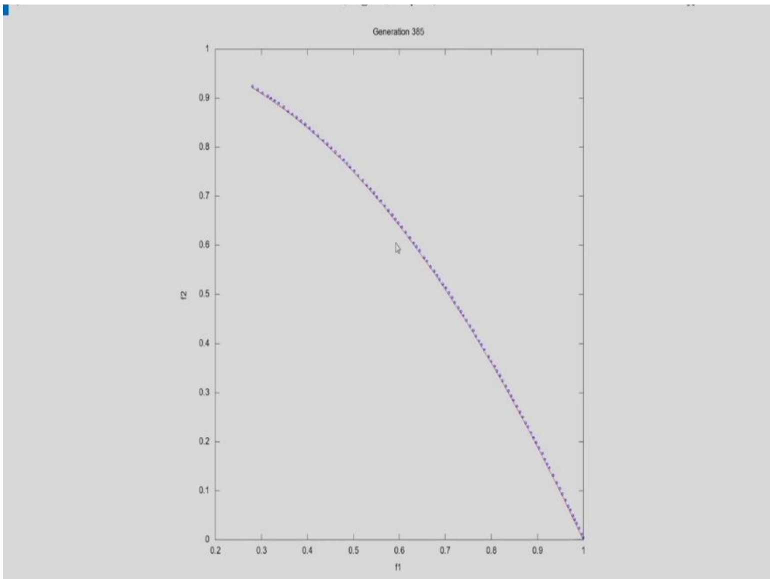
The last problem which we will be solving in ZDT problems that is called ZDT6. Now in the ZDT6 the form of f 1 has been changed. Similarly, we can see there is a change in g function as well as the h function. Since h is similar to ZDT2 the pareto optimal front will be non-convex we are going to solve this particular problem for 10 number of variables.

(Refer Slide Time: 37:50)



So, as we can see the range of f 2 the initial population is generated close to 8, 9 and f 1 value between 0 from 0.2 to 1 and most of the solutions are generated on the one side of the objective space. And after running this SPEA2 for many generations as we can see that the solutions are converged on the pareto front. If we see the simulation of SPEA2 for the given problem.

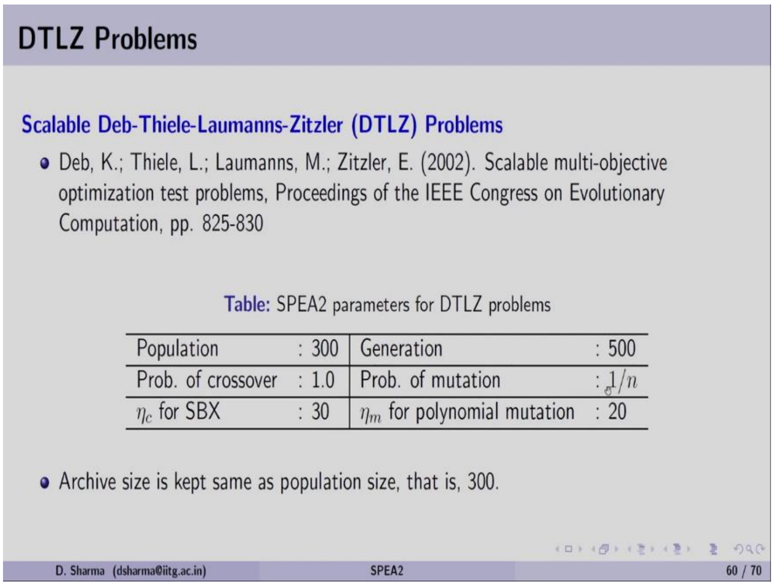
(Refer Slide Time: 38:23)



So, we can see that the solutions are actually converged on the corner of this objective space as soon as these solutions are converging towards the front we can see the there is a good diversity and since the problem is difficult.

So, we are running for more number of generation and we can see slowly and slowly these solutions have converged on the pareto front. So, in this case we have run SPEA2 for 400 generation to get the solution on the pareto optimal front.

(Refer Slide Time: 39:02)

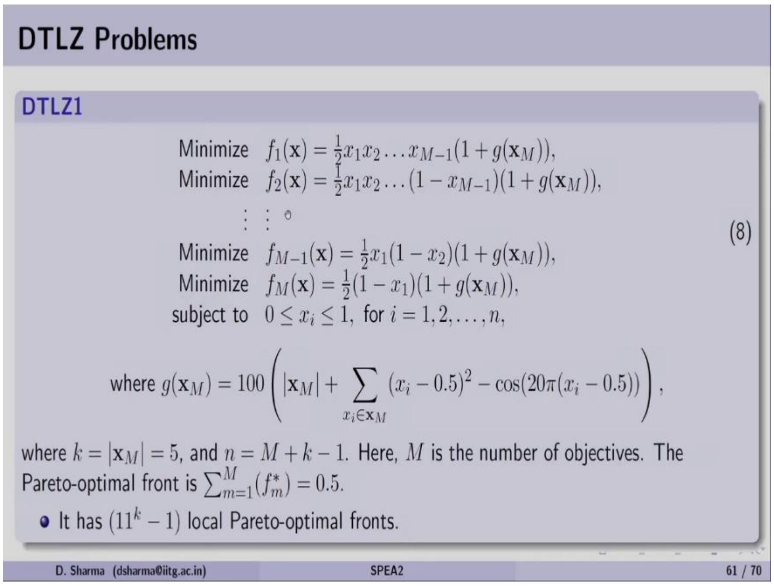


After the class of ZDT problems as we have understood, that these ZDT problems are 2 objective problems and some problems are easy to solve, some problems are non-convex, some problem have lot of local pareto optimum solutions and. So, on, but these ZDT problems are always 2 objective problem.

So, the another class of problem on which we will see the performance of SPEA2 is the DTLZ problem. These DTLZ problems are the scalable problem meaning that we can solve for any number of objective function. So, let us look into the DTLZ problem now. So, these problems are proposed by the 4th 4 authors and from their surname it came the name came out as a DTLZ problem.

The parameters which we have chosen to check the performance of SPEA2. The population size is 300, generation is 500, crossover probability we kept it 1, probability of mutation is 1 by n, eta c 30, eta m 20. And here also we are keeping the archives size same as the population size which is 300.

(Refer Slide Time: 40:26)



$$Minimize \qquad f_1(x) = \frac{1}{2}x_1x_2 \ldots x_{M-1}(1+g(x_M))$$

$$Minimize \qquad f_2(x) = x_1x_2 \ldots (1-x_{M-1})(1+g(x_M))$$

$$\vdots \, \vdots$$

$$Minimize \qquad f_{(M-1)}(x) = \frac{1}{2}(1-x_2)(1+g(x_M))$$

$$Minimize \qquad f_M(x) = \frac{1}{2}(1-x_1)(1+g(x_M))$$

$$subject\ to \qquad 0 \leq x_i \leq 1, \qquad for\ i = 1,2, \ldots, n,$$

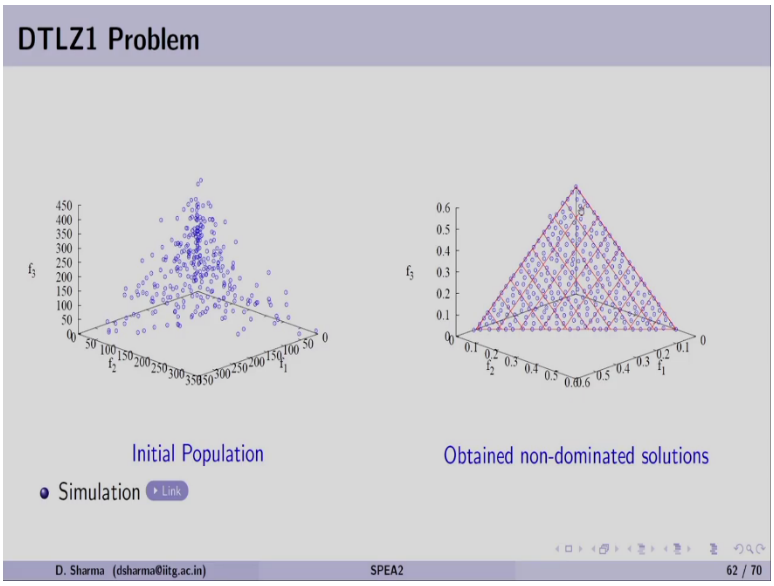
$$where \qquad g(x_M) = 100\left(|x_M| + \sum_{x_i \in x_M} (x_i - 0.5)^2 - cos\big(20\pi\,(x_i - 0.5)\big)\right),$$

Let us see the DTL1 first since the problem is scalable. So, we can see that how we can write objective from f 1 to f M. So, capital M represents the number of objective. In this DTLZ1 all the variables are lying between 0 to 1. If we look into the form we have g function which is written in the various objective function.

So, here we can see the g function is written which has x i 0.5 whole square and some cos term. For solving DTLZ1 problem we have taken a parameter k which is the cardinality of x M equals to 5. So, the number of variable for DTLZ problem will become M plus k minus 1, M is the number of objective, k is 5 and minus 1.

Now here the pareto optimal front for DTLZ1 is a plane and the and the difficulty or the characteristic of a DTLZ1 problem is it has 11 to the power k minus 1 local pareto optimal fronts.

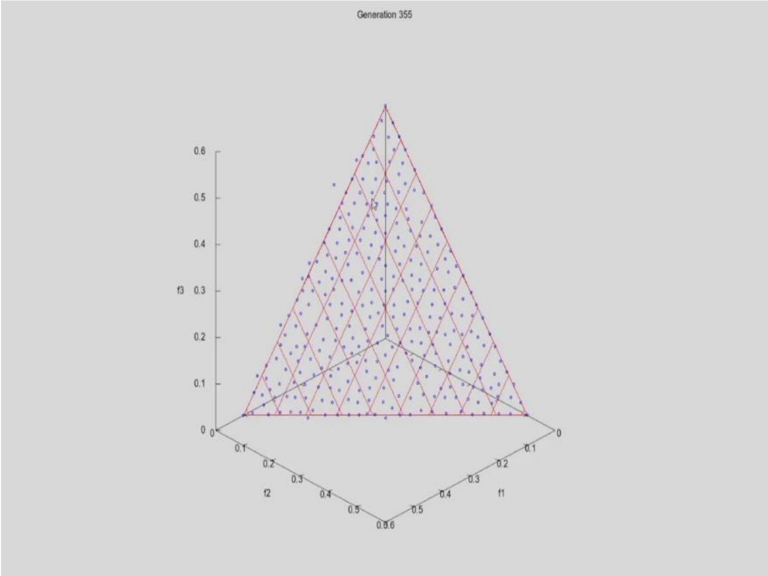
(Refer Slide Time: 41:45)



We can see the simulation now. So, fist we will see the initial population now we can see the range of f 1, f 2 and f 3. So, these solutions are starting very far from the pareto optimal surface. Since it is a 3-objective problem. So, we are we call it as a pareto optimal surface.

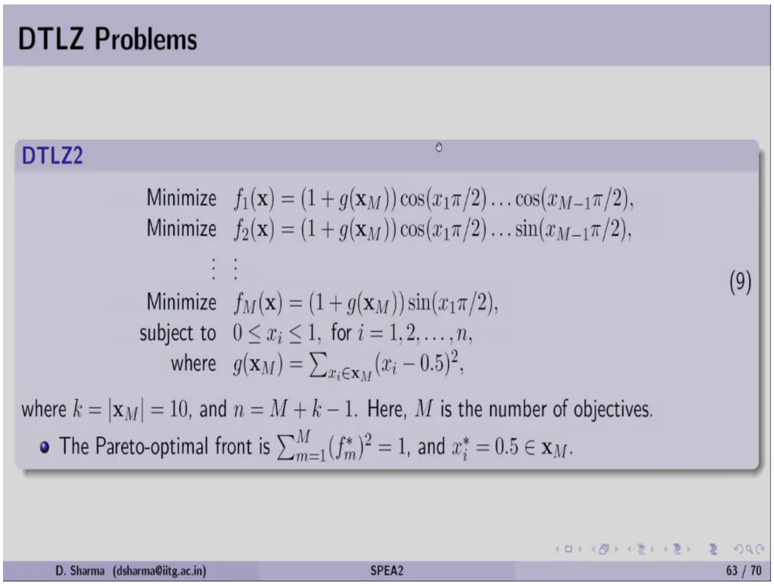
And after running for 500 generation, we can see on the right-hand side figure that all these solutions are converged to the pareto surface as well as we have a good diversity among the solution. So, we will see the simulation of SPEA2 for DTLZ1.

(Refer Slide Time: 42:29)



So, we can see that the solutions are converging towards the front which are quite far and close to 140 generation. Some solutions are converged and the rest of the solutions thereafter converged on the pareto surface. Now here these solutions are keep on changing their position it is only because we have the truncation method or operator to keep the diverse solution on the pareto surface.

(Refer Slide Time: 43:03)



$$Minimize \qquad f_1(x) = (1 + g(x_M))cos\,(x_1\pi\,/2) \;\ldots\; cos\,(x_{M-1}\,\pi\,/2),$$

$$Minimize \qquad f_2(x) = (1 + g(x_M)\;)cos(x_1\pi\,/2) \ldots sin(x_{M-1}\,\pi\,/2)$$

$$\vdots \quad \vdots$$

$$Minimize \qquad f_M(x) \quad = \ (1 + g(x_M))sin(x_1\pi\ /2)\ )$$

$$subject\ to \qquad 0 \leq\ x_i \leq\ 1, \qquad for\ i = 1,2, \dots\ , n,$$

$$where, \qquad g(x_M) \ = \sum_{(x_i\, \in\, x_M)} (x_i - 0.5)^\wedge 2$$

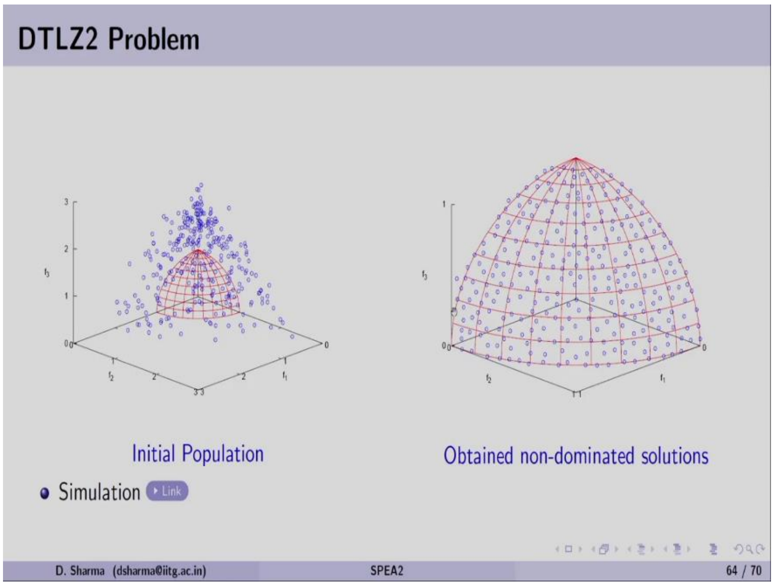
$$where\ k = |x_M| = 10, \qquad and \quad n = M + k - 1$$

$$\sum_{m=1}^{M} (f_m^*)^2 \ = 1, \qquad and \qquad x_i^* \ = \ 0.5 \ \in \ x_M$$

The next class of problem is DTLZ2. Here the form of DTLZ2 is different from DTLZ1 it involves cos and sin functions. Similarly, we have this x i the all variables will be lying between 0 to 1 and the g function as given here. For solving DTLZ2 problem we have taken k equals to 10 the number of variable will be M plus k minus 1, meaning that if we take 3 number of objective.

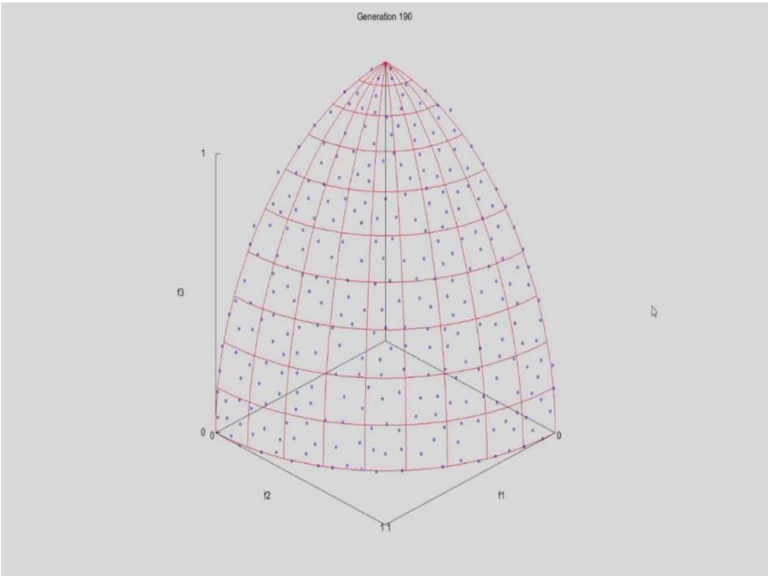
So, 3 plus 10 minus 1 means the number of variable will become 12. Here the pareto optimal front will be f m square for all the objectives and in this case this is going to be the quarter of the sphere.

(Refer Slide Time: 43:56)



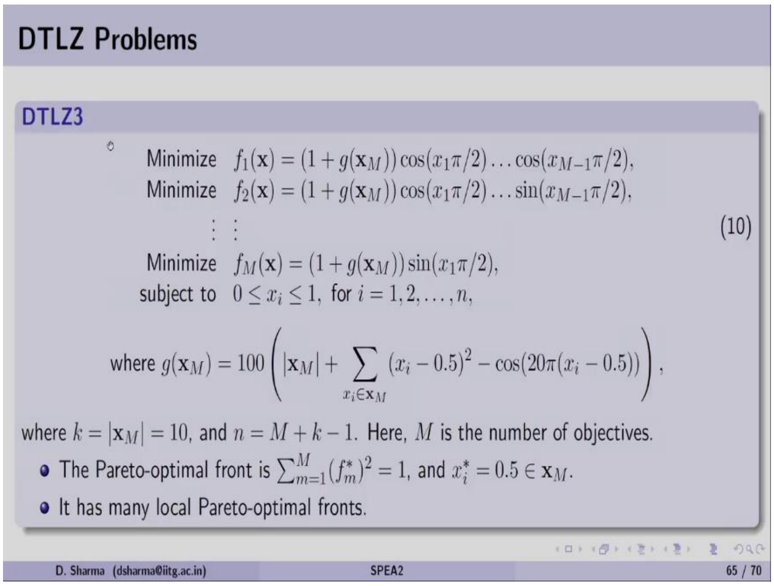
We can see the initial population is generated which is quite far and after running SPEA2 the solutions are converged on the pareto surface which we can see on the right-hand side and they are also diverse.

(Refer Slide Time: 44:16)



We will see the simulation for the given problem. So, we can see within 30 generations the solutions are converged to the pareto front and they keep on changing their position because the truncation method is selecting the solutions or we can say the truncation method deleting those solutions which are crowded. Since we have run SPEA2 for 500 we can see that the solutions are converged on the pareto front.

(Refer Slide Time: 44:47)



$$Minimize \qquad f_1(x) \;=\; (1+g(x_M))cos\,(x_1\pi\,/2)\;\;\ldots\;\;cos\,(x_{M-1}\,\pi\,/2),$$

$$Minimize \qquad f_2(x) \;=\; (1+g(x_M)\;)cos(x_1\pi\,/2)\ldots sin(x_{M-1}\,\pi\,/2)$$

$$\vdots\;\vdots$$

$$Minimize \qquad f_M(x) \;=\; (1+g(x_M))sin(x_1\pi\,/2)\,)$$

$$subject\ to \qquad 0\le\; x_i \le\; 1, \qquad for\ i = 1,2,\ldots,n,$$

$$where, \quad g(x_M) \;=\; 100\;(|x_M| \;+\; \sum_{x_i\in x_M}(x_i-0.5)\hat{\ }2 \;-\; \cos(20\pi\,(x_i \;-\; 0.5)))\quad ,$$

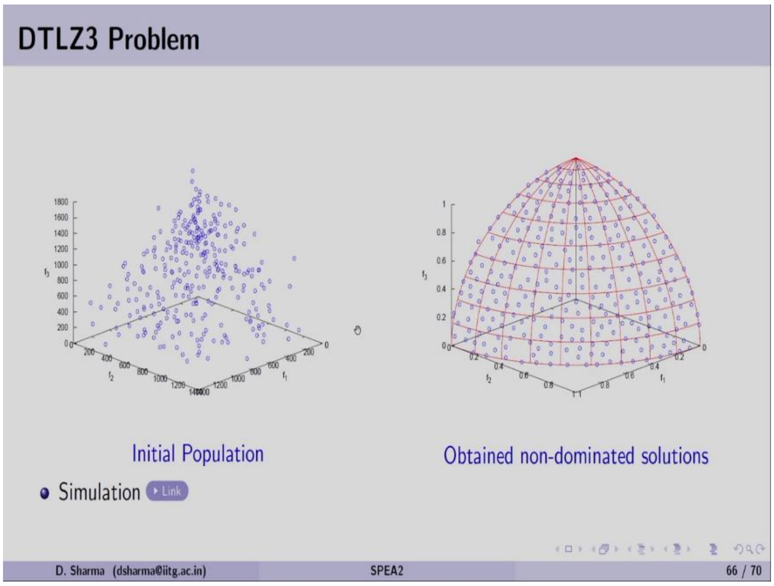
$$where \quad k = |x_M| = 10, \qquad and \qquad n = M+k-1$$

$$\sum_{m=1}^{M}(f_m^*)^2 = 1, \qquad and \qquad x_i^* \;=\; 0.5 \;\in\; x_M$$

The 3rd the another class of problem is called DTLZ3. The form of f 1, f 2, f M again has been changed still it has cos and sin functions and the g function which is used with these f 1, f 2 till f M, the form has also been changed.

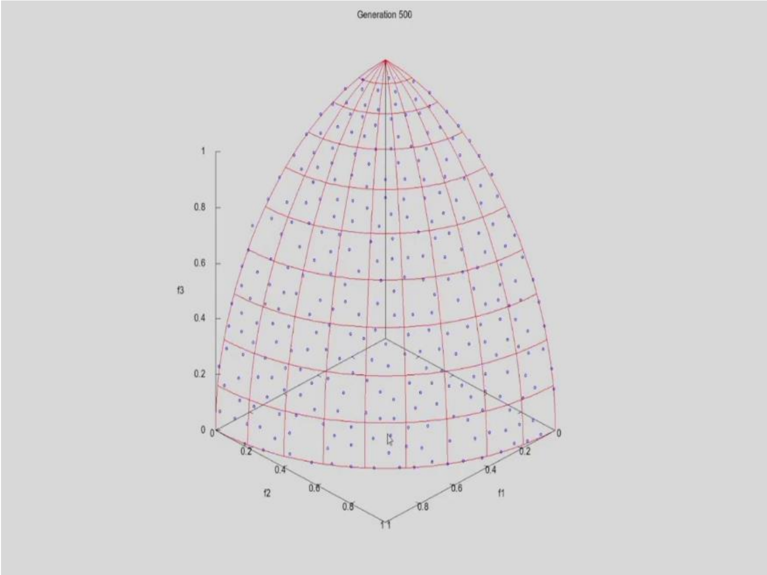
In this case by solving DTLZ3 problem we are taking k equals to 10, number of variable is equals to M plus k minus 1 which is the same as in our previous problems. And the if we look at the pareto front it is going to be the same as DTLZ3. However, this particular problem is going to have many local pareto optimal fronts.

(Refer Slide Time: 45:35)



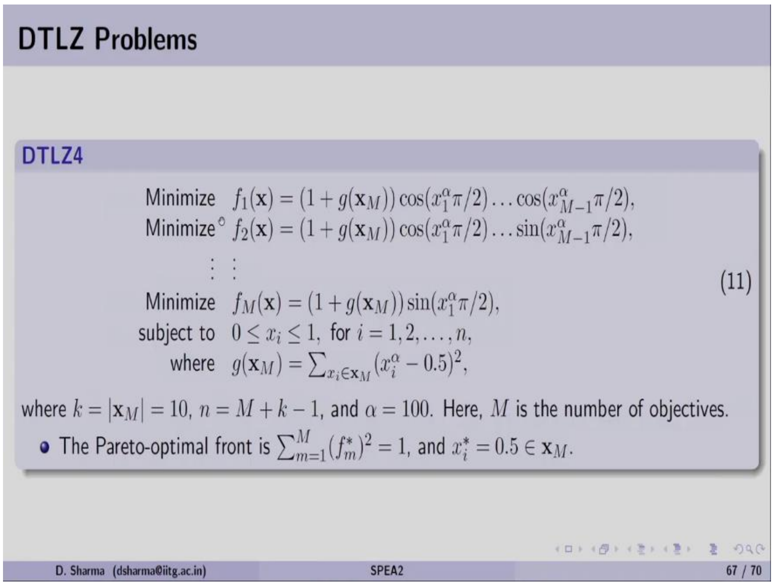
This is the initial population which we can see on the left-hand side. If we look at the f 1, f 2 and f 3 values we can see that these solutions are very far from the pareto surface. And after running this SPEA2 algorithm for 500 generation the solutions have well converged on the pareto surface and there is a enough diversity to represent the surface.

(Refer Slide Time: 46:04)



Let us see the simulation now. We can see these solutions are keep on improving and moving towards the pareto surface and after few generations these solutions will be trying to converge on the pareto surface. So, close to 360 generations. These solutions are now converged and we can see that not only convergence there is a good diversity among the solutions which are converged on the pareto surface.

(Refer Slide Time: 46:37)



$$Minimize \quad f_1(x) = \left(1 + g(x_M)\right) cos(x_1^\alpha \pi /2) \ldots cos(x_{M-1}^\alpha \pi /2)$$

$$Minimize \quad \& f_2(x) = \left(1 + g(x_M)\right) cos(x_1^{\alpha}\pi /2) \ldots sin(x_{M-1}^{\alpha} \ \pi /2)$$

$$Minimize \quad \& f_M(x) = \left(1 + g(x_M)\right) sin(x_1^{\alpha}\pi /2)$$

$$subject\ to \quad 0 \leq x_i \leq 1, \quad for \quad i = 1,2, \ldots ,n$$

$$where \quad g(x_M) = \sum_{(x_i \in x_M)} (x_i^{\alpha} - 0.5)^{\wedge}2,$$

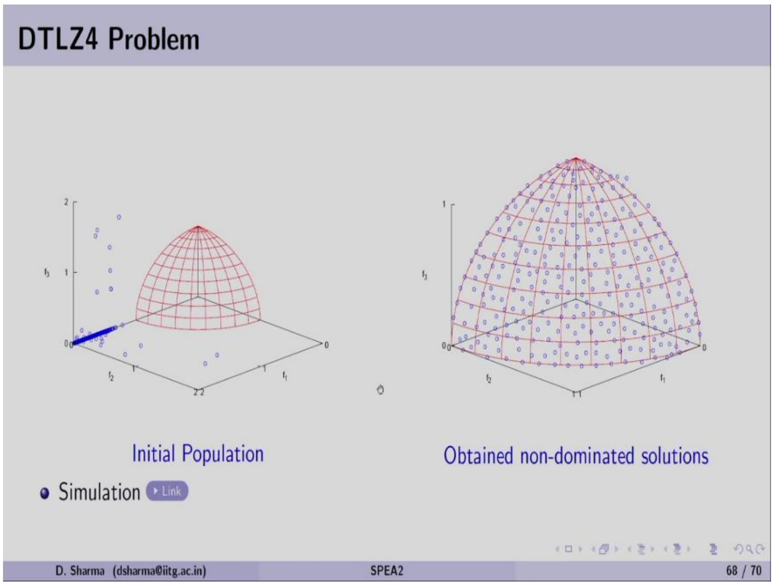
$$where\ k = |x_M| = 10, n = M + k - 1, \quad and\ \alpha = 100$$

$$\sum_{m=1}^{M} (f_m^*)^2 = 1, and\ x_i^* = 0.5 \in x_M$$

Come to the last problem which we have taken in this particular session this is DTLZ4. In this DTLZ4 problem we can see the form of f 1, f 2, f m that have been changed. Now although these are cos and sin function, but inside it the x i to the power alpha. So, the alpha has been introduced here and the variables are going to be lying between 0 to 1 and g M form is also changed.

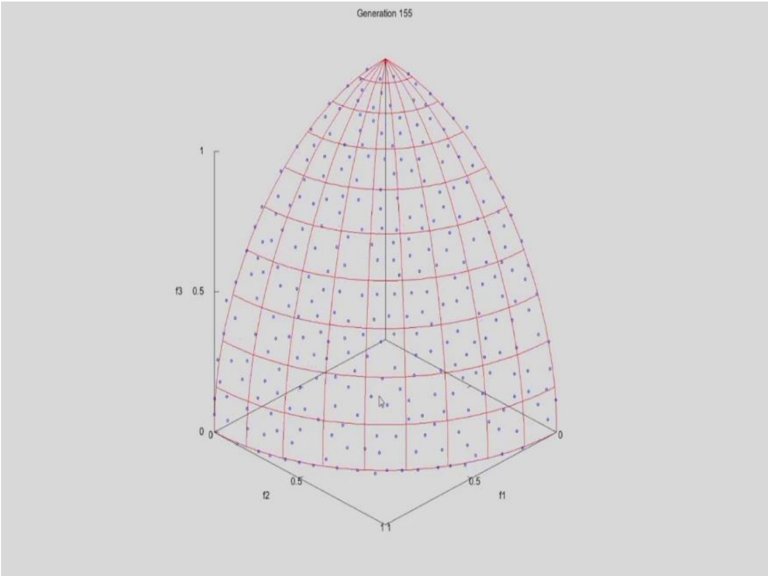
For solving this DTLZ4 problem we have taken k equals to 10 number of variable as the same which is M plus k minus 1. Alpha has been taken 100. The pareto optimal front for the given problem is the same as DTLZ2 and we are going to see how SPEA2 is going to perform.

(Refer Slide Time: 47:30)



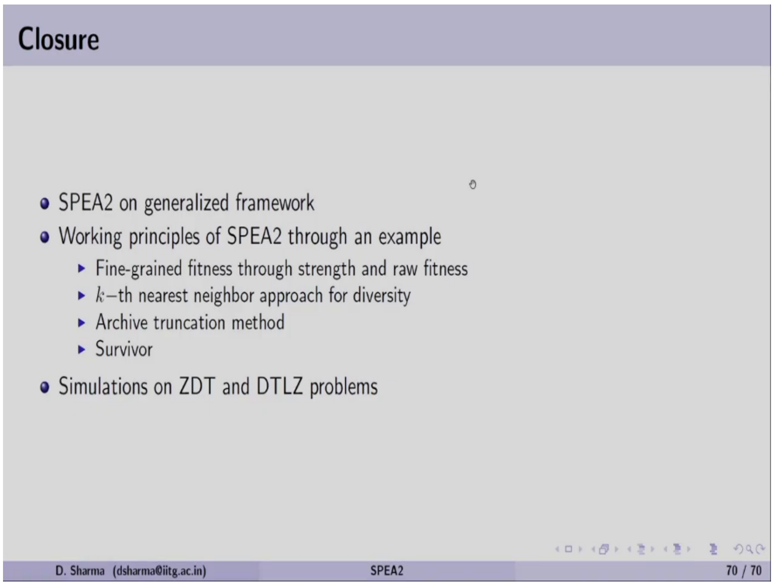
The initial population as we can see it is generated. On the one side of the objective space and after running this algorithm for 500 generation we can see that the solutions are converged on the pareto surface. We will see this simulation now.

(Refer Slide Time: 47:55)



Now, we can see the quickly within 50 generation these solutions are converged to the pareto surface. So, initially this solution is started from the 1 of the corners of the objective space, but as soon as these solution converge to the front they the solution get distributed on the on the pareto optimal surface.

(Refer Slide Time: 48:19)



So, from in this session which we have dedicated to SPEA2. We have come to the closer of this session. In this particular session we started understanding the SPEA2 on our generalized framework. On this framework since SPEA2 uses different kind of ranking as well as a diversity.

We understood SPEA2 through a working example. So, in this session we discussed about the fine-grained fitness through strength and raw fitness. We also did hand calculation for k-th nearest neighbor approach for diversity.

We also discussed about the archive truncation method and we discussed the survivor stage. The SPEA2 the original version incorporates the SBX crossover operator and mutation operator for generating the offspring population. As well as the binary tournament selection was used that is with replacement. In order to see the performance of SPEA2 we tested on the 2 test problems which are ZDT problem set and the DTLZ problem set.

From the simulations we can see that when the problem has the non-convex front convex front it is connected front or there are many local pareto optimal solutions are front. In those different situations SPEA2 was able to converge in all of the solutions and there was a good diversity among the solution. So, with this understanding on SPEA2, I conclude the session on this algorithm which is called SPEA2.

Thank you very much.