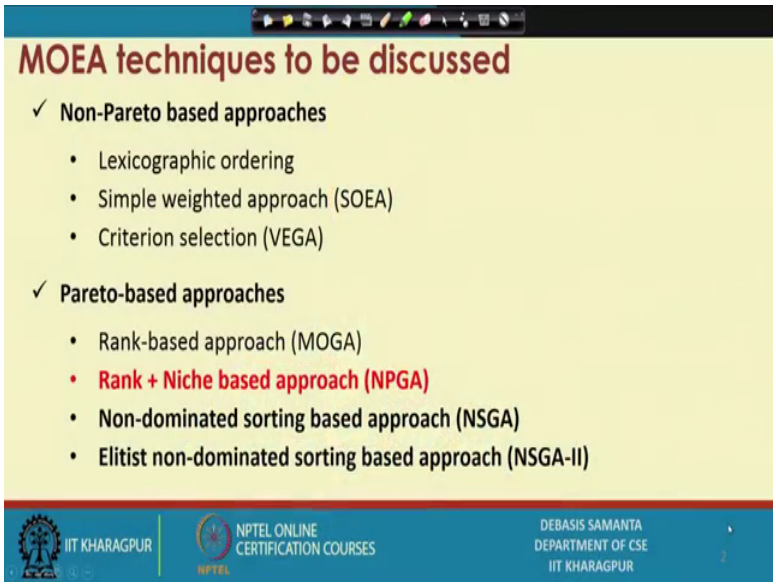
Introduction to Soft Computing
Prof. Debasis Samanta
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture – 30
Pareto- based approaches to solve MOOPs (contd.)

We are discussing Pareto based approaches to solve multi objective optimization problem. In the last class we have discussed one such approach it is called the Moga. In this lecture we learn another approach it is called NPGA, it is a short form it is full form is Niched Pareto Genetic Algorithm.

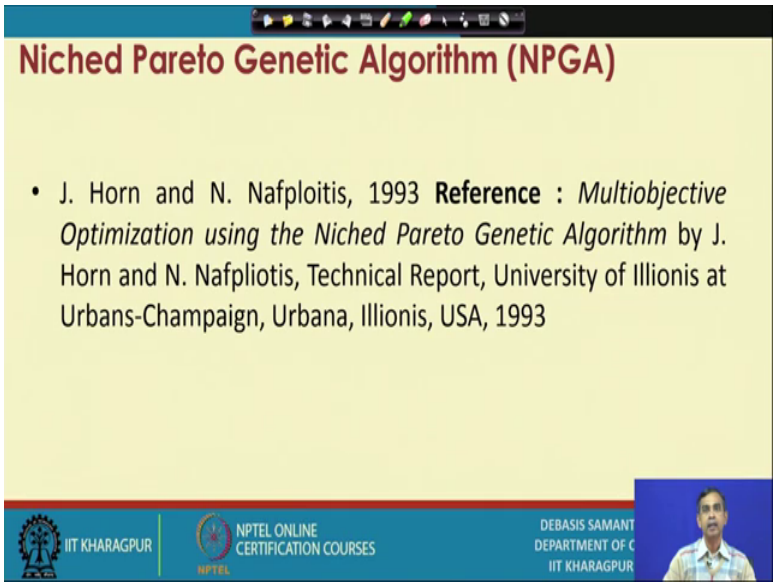
(Refer Slide Time: 00:50)



So, this algorithm is basically based on the rank plus niche calculation. Now we have learned about how to calculate the rank here we will calculate the niche count niche count essentially calculate in fact, how a solution is preferable than the other solution so, for the dominating solution is concerned.

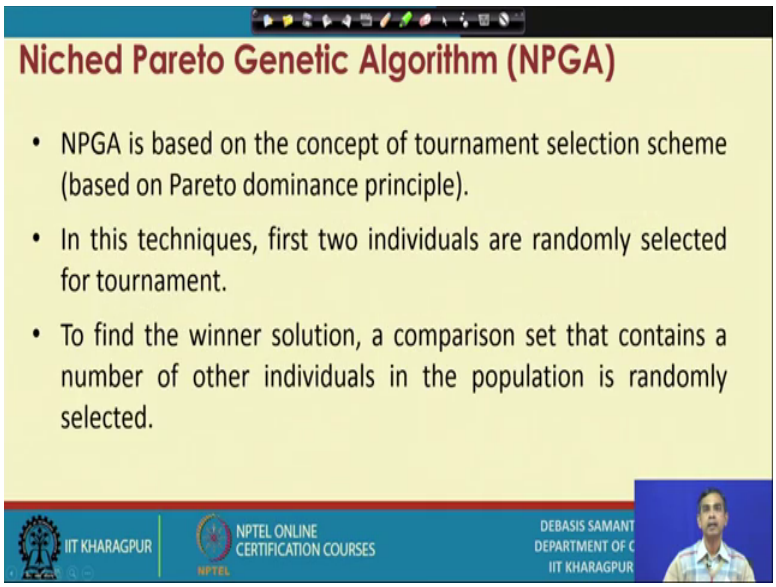
So, these are second approach second Pareto based approaches to solve multi objective optimization problem and this is called the Pareto based because it will give a number of solutions which are of Pareto found so it is the Niched Pareto Genetic Algorithm.

(Refer Slide Time: 01:35)



This is the algorithm first time reported in the year 1993 and the research paper which was published in the form of a technical report by university of Illionis at Urbans-Champaign USA, and the work proposed by J Horn and Nafploitis the title of the work is multi objective optimization using the Niched Pareto genetic algorithm. So, that is how it is called the NPGA Niched Pareto genetic algorithm.

(Refer Slide Time: 02:13)

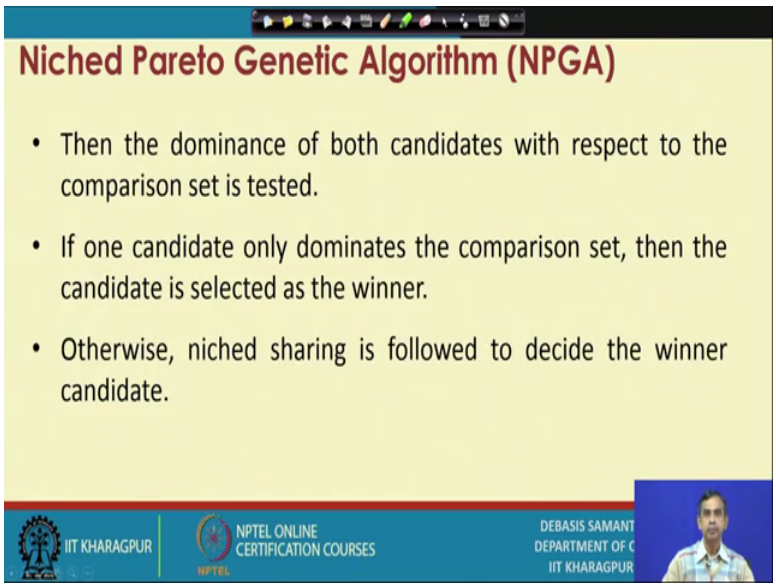


Now, the basic concept that this algorithm follows is the concept of tournament selection I told you all, the multi objective optimization problem solving basically find a different

selection strategy so that we can select the non dominated solution. So, here in this approach the selection that it follows it basically based on the tournament selection concept so it is based on the tournament selection is based on Pareto dominance concept. So, in this technique this technique is start like this in this technique we select any two solutions at random from the current generation and they are selected for that tournament.

And in the tournament there will be one winner. So, the winner will be selected and to select this winner in this approach it follows one set it is called the comparison set. This set contains a number of solutions in the current generation and the number of solutions also again selected at random so it is a completely probabilistic selection strategy because the two solutions of which we have to decide the winner also decided at random and the comparison set it is also a sub set of the current population is selected at random.

(Refer Slide Time: 03:57)



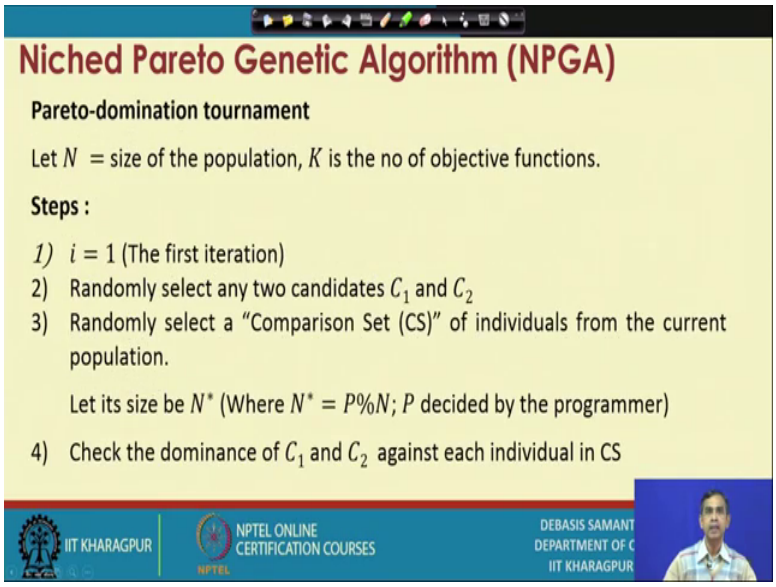
So, this is the concept of this technique now so first we have to select two candidate solutions at random and then the these two candidate sets are tested at tested with reference to the comparison set that we have selected.

Tested in the same stack if the two candidates out of which if one candidates dominates all the solution which are in the comparison set then we can say that that candidate is the winner. Otherwise if no one candidates dominates any one solution in the comparison set

then we have to calculate one count of the two solutions this is called the sharing count or also called the niched count.

So, based on this niche count the solutions which are having higher niche count will be selected for the winner so this way we can repeat the procedure for say N number of times say so that we can get the next generation and all the solutions in the next generation is basically the Pareto solution at the moment. So, this is the idea actually.

(Refer Slide Time: 05:19)



Now we can we can state this idea more formally in the form of a pseudo code and we will discuss this pseudo code here. So, suppose N, N is the size of the population and the problem which multi objective problem whether the multi objective problem which we are going to solve has K number of objective functions now. So, let N be the size of the population and K is the number of objective functions at their in our multi objective optimization problem so it is basically a iterative steps.

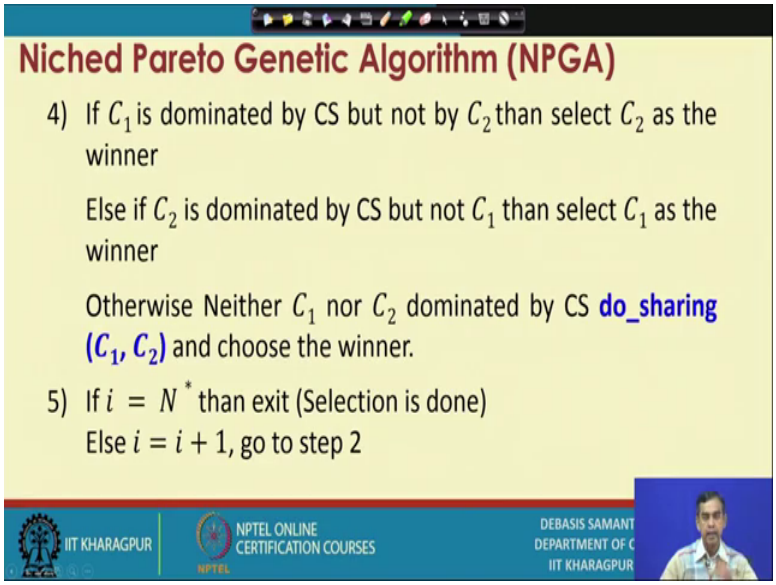
So, we will start with the first iteration let i equals to 1 it is basically tagged the number of iteration that it contains. Now the first step as I told you so that we have to select any two solutions randomly from the current population let the two solutions are denoted as C 1 and C 2 they are the candidate solutions.

And then we have to select randomly a set it is a sub set of the current population we denote this set as a comparison set and the size of the set is decided by the programmer

let the size of the set be N star where N star is P percentage of the current P percentage of the current population where the P is decided by the programmer. Usually it may be 10 percent or it is a 15 percent or 20 percent like this.

So, the depends on how fast you want to terminate it or how more exhaustively you want to search it depends on. So, if the P is very high then termination will be at a higher rate and if it is P is low it is a slow algorithms, but it will be better solution at the time anyways. So, so N star be the size of the comparison set at any moment. Then the next task is basically here is the tournaments selection should be applied here. So, we have to check the dominance of C 1 and C 2 against all the solutions are there in the comparison set. So, now so check the dominance, now let us see how the dominance can be checked.

(Refer Slide Time: 07:55)



It is basically so here so dominance solution is basically following the concept of domination that we know that mean with respect to so there are 3 2 criterias could do conditions rather if C 1 satisfy all the conditions with respect to any solutions which belongs to the comparison set then we can say that C 1 dominates any solutions there in the CS.

So, if the s solutions is C 1 is dominated by CS, but not by C 2 that means, C 1 is not inferior whereas, C 2 is inferior then select C 2 as the winner. So, in that case C 2 because C 1 is dominated by one solution so C 1 is not a good one so in that case a solute

take the solution C 2 on the other hand if C 2 is dominated by CS, but not C 1 then select C 1 as the winner.

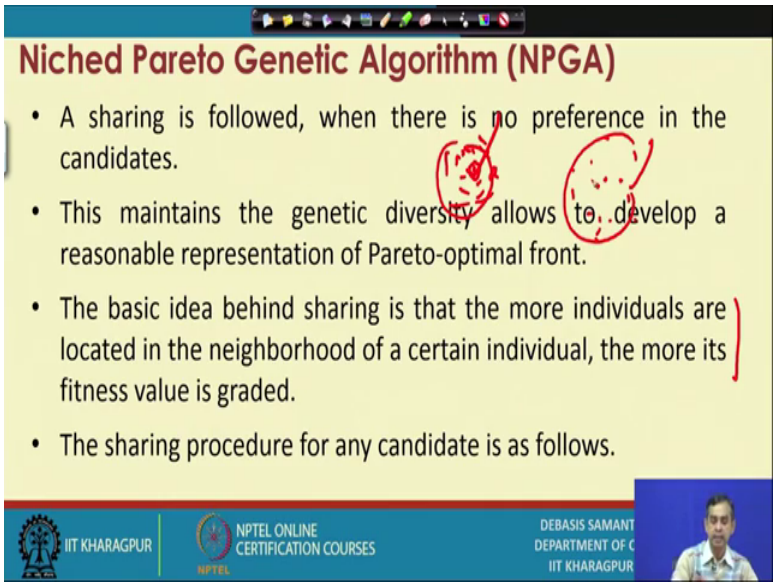
So, the two conditions if anyone is dominates then that can be selected as the winner. Now, there may be one case where neither C 1 nor C 2 dominated by CS.

So, in this case in this case we have to follow one procedure it is called the sharing count calculation. So, we term this procedure as do sharing between C 1 and C 2. Now do sharing procedure I will discuss in details later on so this based on this do sharing procedure it will return either C 1 or C 2 as the winner and then we will select finally, the winners from there. Now this step the step that we have discussed right now has to be repeated until ok.

We can select N star until we select all the so it is basically until you selects N dash N dash is basically size of the mating pool. So, here N dash is also a parameter it is decided by the programmer; that means, the how so N dash is basically denotes at any instant what will be the size of Pareto found like anyway.

So, it is basically we can consider N dash is the size of the mating pool that needs to be considered to generate the population at the next generation. So, this is a repeative or iterative step we have to follow it and then we will find N dash number of solutions for the mating pool. Now so, this now we will discuss about the method what is called the do sharing method.

(Refer Slide Time: 10:44)



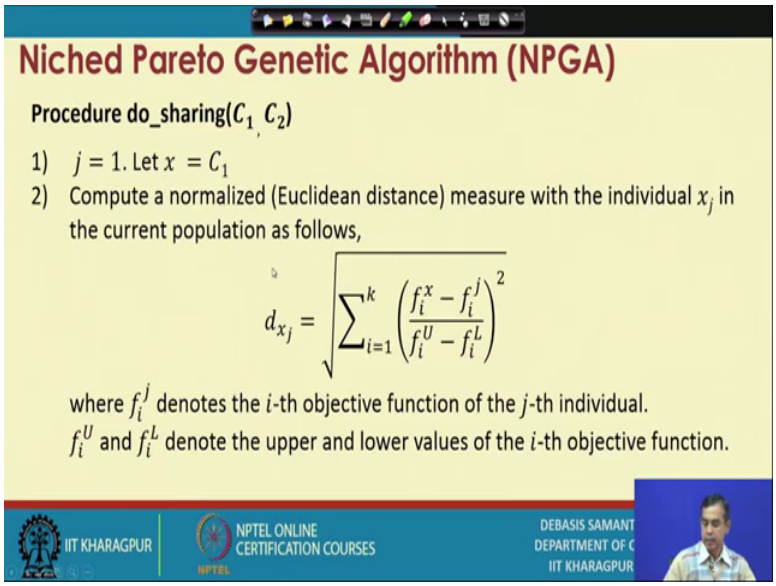
Now, so, here basically do sharing is a process of sharing count it is called the also niche count so and so basically we have to follow this sharing procedure when we see that out of the randomly selected two candidates no one is the winner and the guest the main idea about this do sharing calculation is to maintain a good population diversity which allows to develop a reasonable representation of Pareto optimal front.

So, it is basically the idea about this one; that means, how the population diversity can be maintained. Now here the basic idea behind the sharing is that the more individuals are located in the neighborhood of a certain individual the more its fitness value is regarded so that is why the thing is that.

So, basic idea is that if this is the one solution and it has a number of solutions near about this one. So, its niche count will be high on the other hand this is the one solution which has which has the only fewer solution then its niche count will be low.

So, depending on this if it has very high niche count then we will select this solution compared to this solution as a preferable so for the Pareto optimal solution because it represents many of the solution then which represented by this one. So, this way we will select the then the these way we can count the sharing count or niche count and then we can select the winner solution from there. Now let us see how the niche count can be calculated and what is the method that is followed there.

(Refer Slide Time: 12:27)



So, here the idea it is that so this is the current solution and there is a all the other solutions which is there then you have to we have to find the niche count like so we have to find that from this solution what is the similarity from the other solution this similarity is measured by means of Euclidean norms the distance between this solution to this solution.

So distance between the two vectors we can say so this is the solution this current solution is the one solution and this is the another solution which is there is the another solution. So, distance from one solution to any other solution is basically a similarity measure by means of Euclidean norm.

So, the idea is that from the current solution right we have to calculate the distance from all other solution which is there in the solution in the population. So, here idea it is like this so it it is again iterative process because we have to count the distance or distance from the current solution to this solution let we have to count the sharing count of C 1.

So, we will repeat it the same procedure will be repeated for the C 2 calculation also. So, this is the Euclidean distance that you have to calculate which basically measures the individual any one individual say x j where j is starting with 1 x 1 and another is C 1.

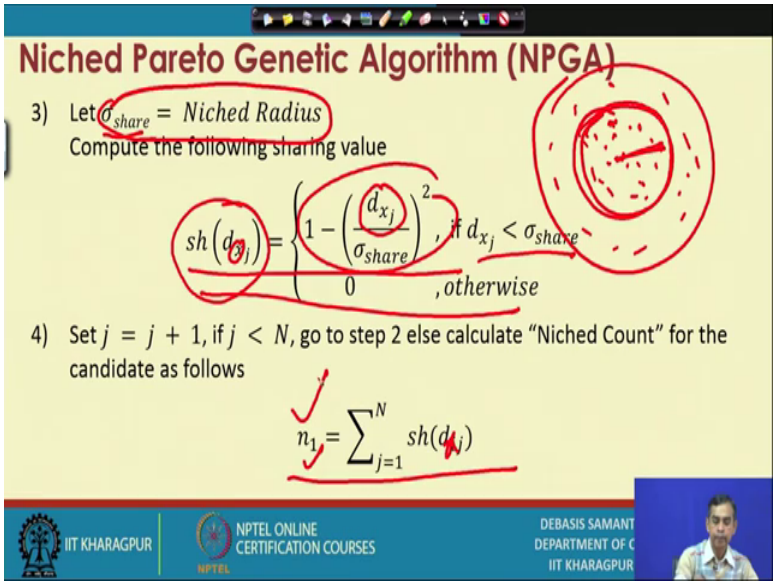
So, it is basically the distance between the currents x and the j-th solution. So, we denote it as a d x j. So, this is basically similarity between the two solution the current solution

and any solution which is there. So, j 1 then we have to repeat for all other solutions so it is like this and for the j equal to 1 any one solutions they are other than the C 1.

Let us see what is the idea about it so basically for each solution we have the objective function for the i-th vector. So, i equals to 1 to k so for each come function we have to calculate it is and this is the the i-th function value for the x solution and this is the i-th function value for the j-th solution and here also we consider f U and f L at the lower the upper limit and lower limit with respect to the i-th function.

So, once we know this things then we will be able to calculate it is the basically Euclidean distance form actually. So, it will it will return what is the similarity or the distance from the solution x to any j-th solution. So, so this way we will be able to calculate this d x j from the current solution to any other solutions which is there in the current population.

(Refer Slide Time: 15:27)



So, one the d x j is calculated then we will be able to compute the sharing count and to calculate the sharing count we have to decide one parameter this parameter is called the C sigma parameter. C sigma parameter is also called the niched radius. Now this C share basically depends by it is decided it is determined by the programmer.

So, the idea it is like this so if this is the solution then it is basically the area out of which a or within which we want to calculate the niche count so it is one sigma shear this is the

radius like if we take the higher value then it is the higher one. So, the idea is that if we take lower value only few solutions are will be under confidence if we take the higher value then a large number of solutions will be under the confidence.

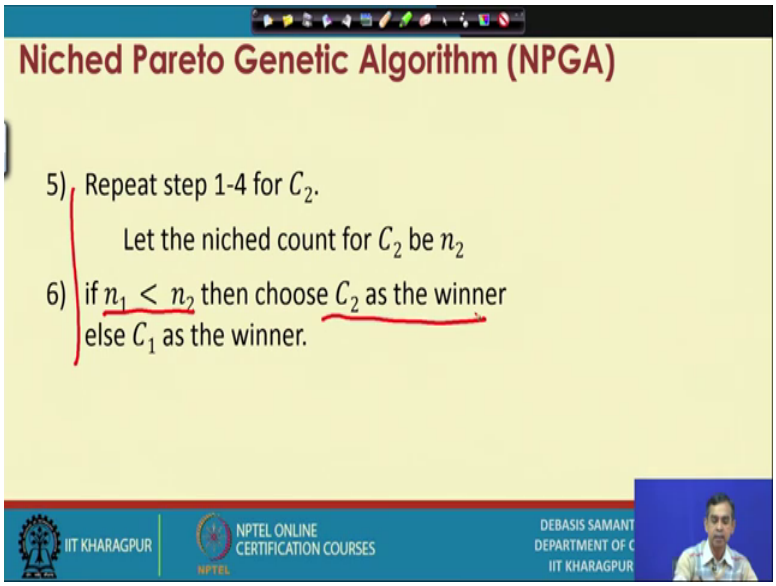
Obviously, it is here the cost of competition because for if it is smaller one then calculation is less if it is larger one then calculation is higher. Now once we decide the sigma share so this is the niched radius then we will be able to calculate the sharing count.

The formula for caring the size is here if d x j with less than sigma share that mean if any solution within this radius then thus it basically give a contribution this contribution is like this one minus d x j we have already calculated earlier from the current solution x to any j-th solution and divided by C sigma so it is basically a normalized value and then so it is like this one.

And if it is 0 if any solution which is beyond or outside the niched radius, so it is 0, so this way it will count only the sharing values or niched values within this solution value. So, this way the from the current solution x to any solution j we shall be able to calculate the sharing count or it is called the niche sharing count.

And then once we know the sharing count for all for with respect to the current solution and with other solution we will be able to compute the niche count. So, niche count is denoted by n 1 it is basically summation of all the sharing count from the x with respect to the any other solution here. So, if j equals to 1 to N then we will be able to calculate so d in to x j from x j solution from the current solution to any other j-th solution. So, this basically gives you the niche count n 1 for the current solution C 1.

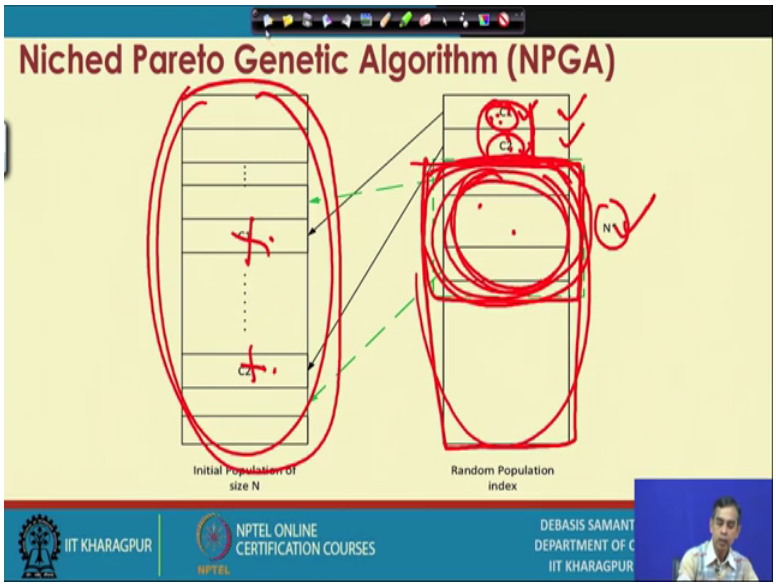
(Refer Slide Time: 18:20)



Now the same procedure can be repeated for the candidates solution C 2 and we will be able to calculate another niched count let it be denoted by n 2. So, these are the two niched count n 1 for C 1 and n 2 for C 2 then we can select the candidate based on this niche count which is shown here.

So, if n 1; if n 1 is less than n 2 then choose C 2 as the winner otherwise we should we select C one. So, this way if we select at random any two candidate solution then any one either one will be selected as a Pareto optimal solution, but this solution will be decided by the tournament selection this is the idea about the tournament selection that it follows in case of niched Pareto genetic algorithm.

(Refer Slide Time: 19:11)



Now, here is the pictorial description to illustrate the same idea. So, this basically represents the current population at any instant at any iteration. Then we have to select at random any two solution C 1 and C 2 and this solutions are listed here. So, this is another random selection.

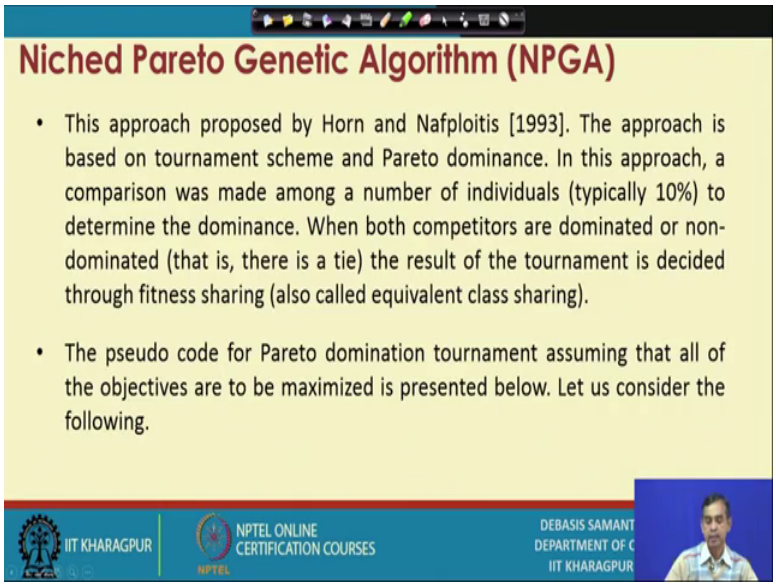
Next the P percentage of this current population let this be N star is selected then all the this then this then the number of solutions which belongs to the number of solution number of N star solution number of N star solution will be selected from the current population except C 1 and C two because it has been selected that means, out of this things we have to select N star number from the remaining and that also will be selected at random so it is in that sense completely random on.

So, let this is basically N star number of subsets from this sets which is basically called the so this set is called the comparison set as we have discussed once the comparison set is known then we have to see if C 1 and out of this C 2 if C 1 is dominated. So, if C 1 dominates all other solutions here in this comparison set then we can say C 1 is the winner.

Now if suppose C 1 is not dominated by any solution then we have to check that whether C 2 is C 2 dominates any solutions are there. So, if C 2 dominates all the solutions which are there in the comparison set, but not C 1 then we will select C 2. Now if C 1 and C 2 neither one dominates any solution there then we have to calculate the niched count for

both C 1 and C 2 and based on this niched count we will select one solution. So, this is the idea about the NPGA algorithm the niched Pareto genetic algorithm and then it will calculate the solution.

(Refer Slide Time: 21:31)



And this step to be followed for N dash number N dash is the number of mating pool right and from the this mating pool we will be able to produce the reproduction. The reproduction procedures and then make selection of mating pair are the same it is usual that is there in the conventional genetic algorithm.

Now so what we have learnt here is that then NPA the niched Pareto algorithm approach is basically is a different it follows a different selection strategy and this selection strategy is basically based on the concept of tournament selections. And in the tournament selection we consider the Pareto dominance concept.

So, it is basically Pareto dominance based tournament selection tournament scheme as the selection scheme in niched Pareto genetic algorithm. So, there are few conventions. So, the first convention is that in this approach we a comparison set was first considered or if generated with a number of solutions and typically it is 10 percent of the current populations solutions and when both competitors are dominated or non dominated so there may be tie then it basically the niche count this thing.

So, we have to resolve the tie actually; that means, if C 1 and C 2 neither dominated no one dominates the comparison set then we have to follow a tie and this tie is basically resolved by the niche count calculation. Now in the slides I have mentioned some quote, but this quote is not so much important only the basic concept how it works that is important.

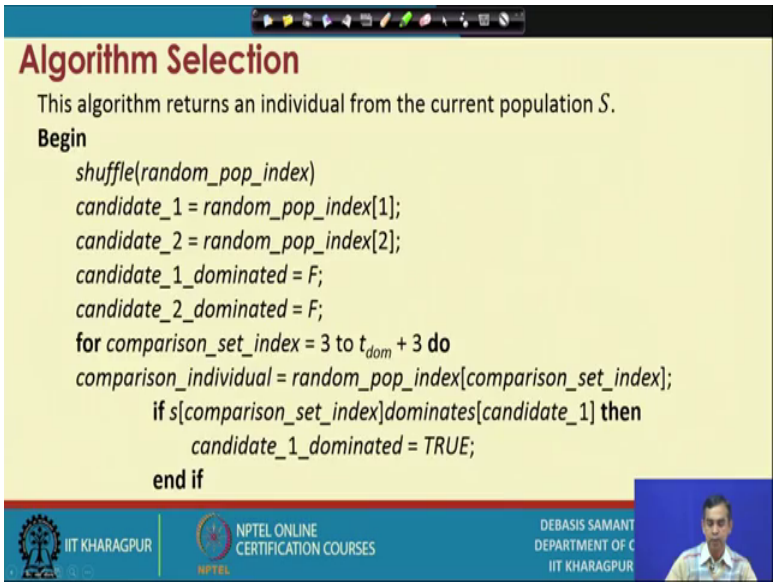
Now in this algorithm also we have to consider few more parameters sigma share so this is the one parameter should be chosen very carefully because if we select this parameter not so accurately or properly then it may leads to unwanted termination or termination with local optimum.

So, this risk is there and then; obviously, how much of the size of the comparison set also needs to be considered and usually this can be considered by means of empirical observation, empirical study. Then we have to start with different values of N star value and then you can select it there is no other way of course, other than this one.

(Refer Slide Time: 24:12)

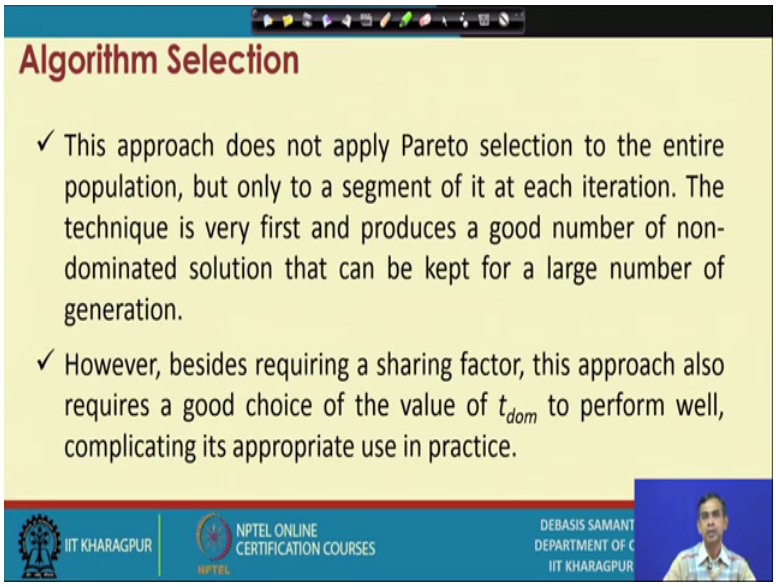
So, this is the idea about it and then that that is the code actually that code I do not want to discuss it here.

(Refer Slide Time: 24:18)



Code you can follow it and you can just understand the concept it is there.

(Refer Slide Time: 24:23)



Now, here so the selection algorithm is the main or critical one a step in this approach and here as we can note that this approach does not apply Pareto selection to the entire population. So, this is the one what is called the criticism against this solution that it only consider the subset of solution and that is why I told you that there is an issue that how we can decide this subset.

Because if we do not decide this subset judiciously then also it may not give the good result or it may not terminate in a finite time so this is the one what is called the criticism that this solution is apart from that it does not consider Pareto selection to this. This means that it has certain chance that you will that into the local optima.

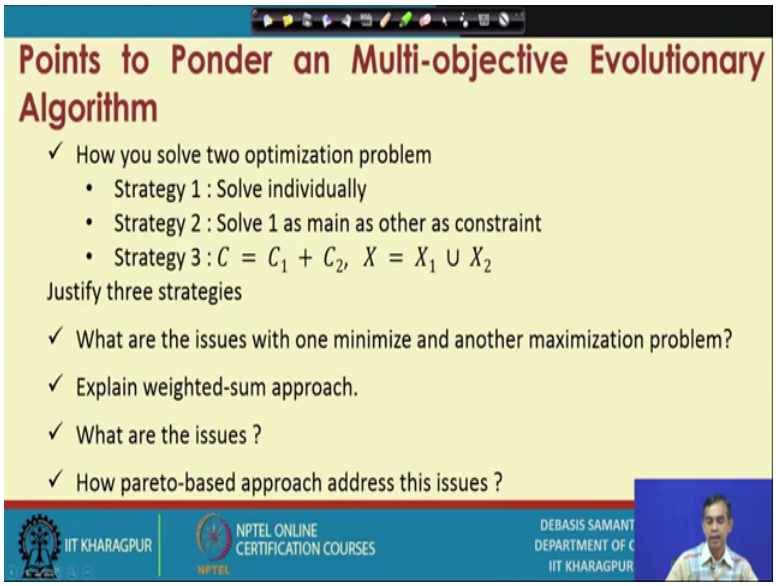
However, these technique is very fast and produce a large number of non terminated solution that can be kept for a large number of generation. So, this basically can be considered there are many fine I will discuss another one idea about it. So, idea is that sometimes we have to consider 2 or more approaches to solve the multi objective that is called the hybrid approach.

For example, using so this approach NPGA as it is first in compared to the other Pareto based solution. So, you can use it to generate one solution which are basically solutions or the Pareto front and then another solution another approach which are may be relatively little bit high in competition demand can be followed but only based on the Pareto optimal form that we have discussed here.

So, in that in this case the NPGA can be considered with a very large comparison set large means not necessary 10 percent it is so maybe say 75 percent and then the comparison then we can select the Pareto front and then based on this Pareto front as the solutions for the genetic algorithm so you can follow some other multi objective optimization problem so approach solving a MOOP's approach like say Moga and then we can select finally, the Pareto optimal front.

So, this is the idea that can be followed. Basically the hybrid approach means two different approaches can be followed that approach can be both of the approach can be Pareto based approach or one (Refer Time: 27:11) Pareto based approach and another Pareto approach like this.

(Refer Slide Time: 27:19)



So, this is the thing if your time is possible then we will be able to adopt this things otherwise it is very difficult to follow this one. Now before conclusion just I want to have some discussion about it as I told you hybrid approach like. So, there are many strategies one is that non Pareto based approach what we have discussed earlier.

So, solve individually; that means, maybe say bigger approach or maybe say some other approach or soya approach is like this then or may be lexica figuring also you can consider and then we can solve one as the main and other is the constant like.

So, it is like it is there in the approach that we have discussed in Lexico (Refer Time: 28:01) actually and then we can follow the two solution combine and then the resultant solution can be obtained. And also we have understood about if one solution is to be minimized and other to be maximized.

How they can be converted uniformly all the minimization problem or maximization problem it is not an issue because we have to calculate the dominance relation between the two solution they are they can be all minimized or can be a mixed type minimization and maximization, but that condition can be checked simple a programming solution is required.

And also we have discussed about weighted sum approach that is they are in the soya and we understood that this soya is very fast and then no need to do anything else only to

calculate some a priori knowledge like what is the weight values for the all the functions there and then that also can be followed and then the solution that can be obtained from the soya approach can be used for the other Pareto based approach that is a hybrid combination of this one.

And Pareto based approach are the best solution compared to the non Pareto based approach; however Pareto based approach are computationally expensive compared to the non Pareto based approach. So, if the competition time needs to be adjusted then we have to think some other strategy there in the solution ok. So, this is the algorithm Pareto based algorithm that we have discussed about niched Pareto based genetic algorithm and there are two the most advanced Pareto based solution this is called the NSGA algorithm and NSGA two also there. We will discuss all this thing in the next slide.

Thank you very much.