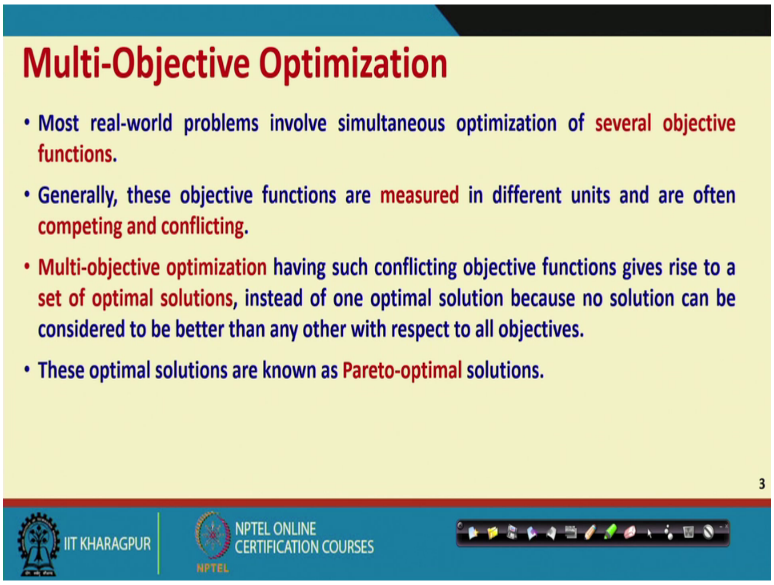
Selected Topics in Decision Modeling
Prof. Biswajit Mahanty
Department of Industrial and Systems Engineering
Indian Institute of Technology, Kharagpur

Lecture – 39
Multi-Objective Optimization

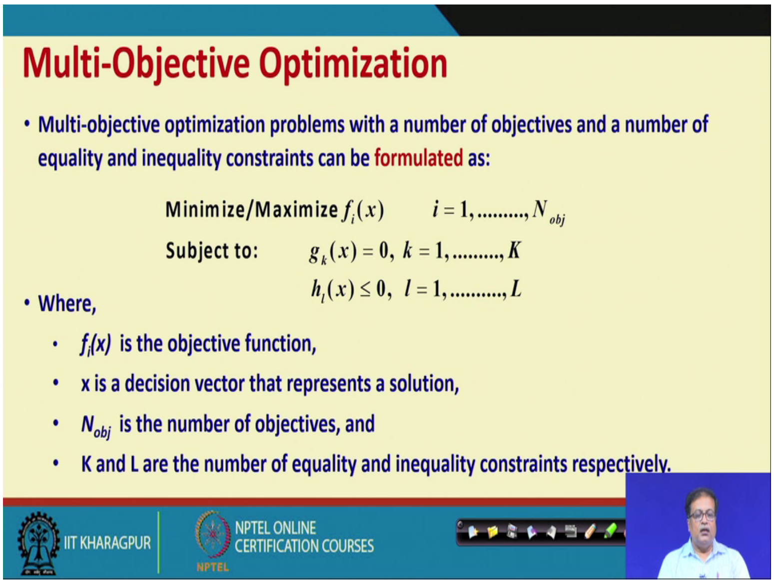
(Refer Slide Time: 00:30)



In our course Selected Topics in Decision Modeling, we are now in our 39th lecture that is multi objective optimization. Now, so far we have seen various problems with multi heuristic use starting from genetic algorithms and several other techniques, but we have not explicitly considered a multi objective optimization using multi meta heuristic methods. Now, multi objective optimization usually there will be several or a number of objective functions; the distinctness will be that they will be measured in different time units and often they would be competing and conflicting even.

Say for example, if you have to simultaneously minimize the time as well as the cost, so you may often find that while time is minimized the cost is not. And when you try to minimize cost, time is not minimized. So, under those situations, you know you are going to get a set of solutions, which can be called optimal in some sense because one optimal solution may not be available right. And all these a set of optimal solutions which cannot be considered to be better than one another with respect to all objectives. With respect to one objective we may find a particular solution to be optimal, but with regard to another objective we may find another solution to be optimal. So, what we shall do, we shall call all of these solutions as Pareto-optimal solutions, but then there is more to it there is a specific definition we shall come to it in due course of time.

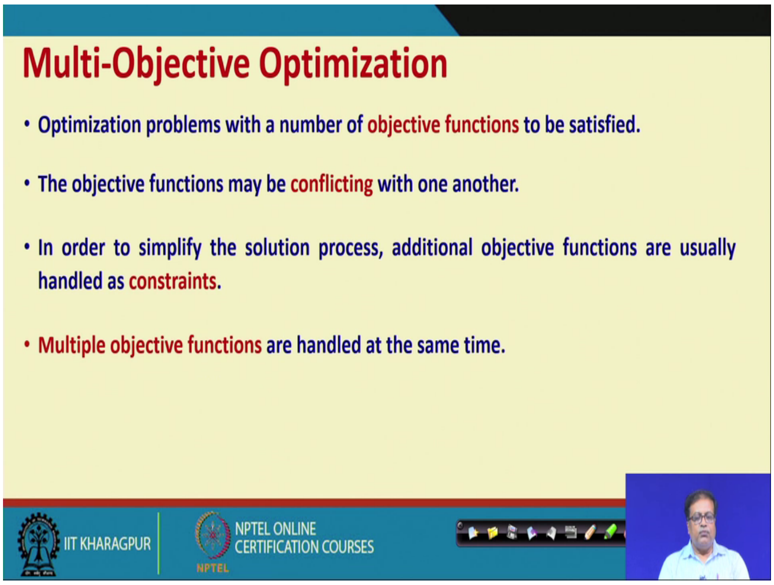
(Refer Slide Time: 02:22)



Now, multi objective optimization in a general sense, it can be thought of as and you know optimization problem where we are minimizing or maximizing a set of objective functions f i x subject to some constraints which are equality constraints g k x equal to 0, where k varies from 1 to large K. And h l x less than equal to 0 where let say that l depend l varies from 1 to large L. And N obj is the number of objectives.

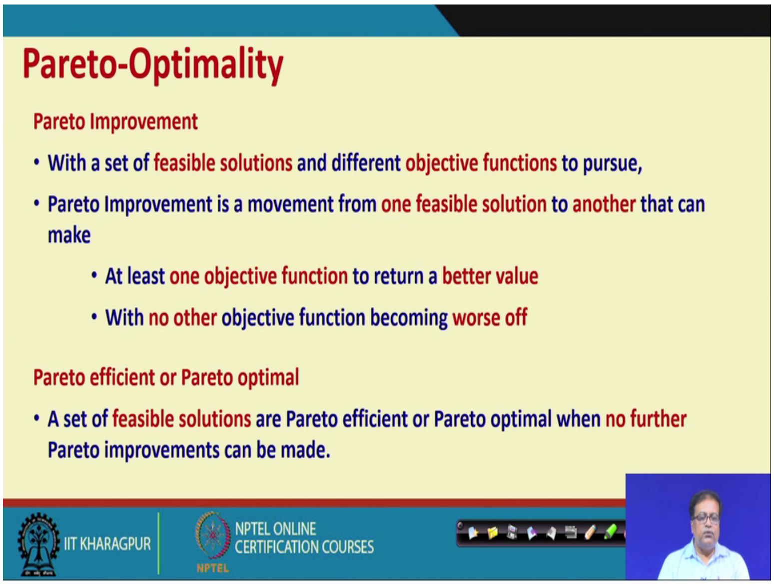
So, you see this could be in the most general form set of objective functions which could be minimize or maximize, and there are both equality and less than equal to 0 constraints, is alright, so that is the general form of multi objective optimization problem.

(Refer Slide Time: 03:25)



So, as I said the objective functions may be conflicting with one another. And if you want to simplify the solution process, sometimes what happens you can you know treat some of the objective functions as constraints, but then you know treating function as objective function and treating or converting it to a constraint is not exactly same thing. So, you know the solution obtain will be may be satisfying those constraints, but cannot be called as optimal with respect to all of these objective functions, is it alright.

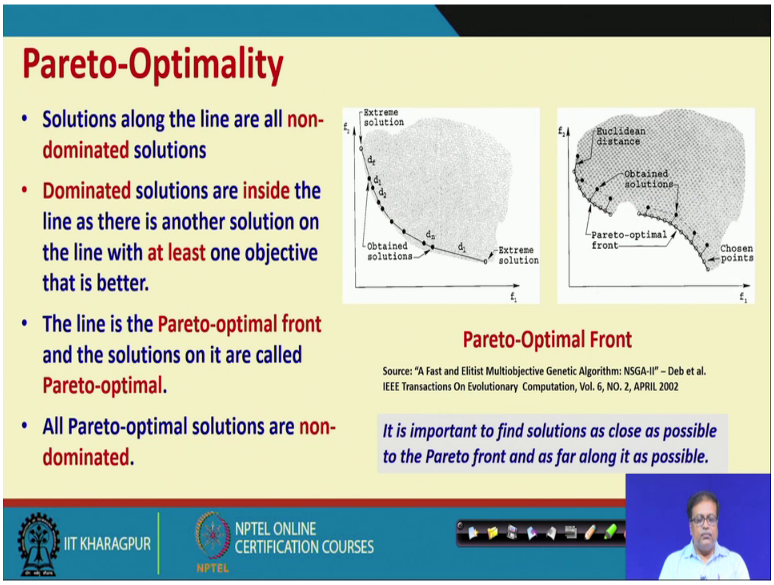
(Refer Slide Time: 04:09)



So, how do we carry out the multi objective optimization that we shall come little later; first let us understand: what is the basic concept of Pareto-optimality. Let us assume that we have a set of feasible solutions is it all right. And they are different objective functions a set of different objective functions, the first concept that what is Pareto improvement right. So, if I take two such feasible solutions, then Pareto what you call improvement can be defined in these way that at least 1 objective function is returning a better value with no other objective function becoming worse off, is it all right.

So, supposing we have two feasible solutions with regard to one objective function, the first one is better; but with respect to any other objective function, the first one is not any you know worse off than the others. So, we may say that going for the first one instead of the others is a kind of Pareto improvement is all right. Now, consider you are doing Pareto improvement and finally, you are left with a set of feasible solutions where no further Pareto improvement can be made, then those set of feasible solutions can be called as the Pareto efficient or Pareto-optimal solutions is alright, so that is how to define Pareto-optimality.

(Refer Slide Time: 05:57)



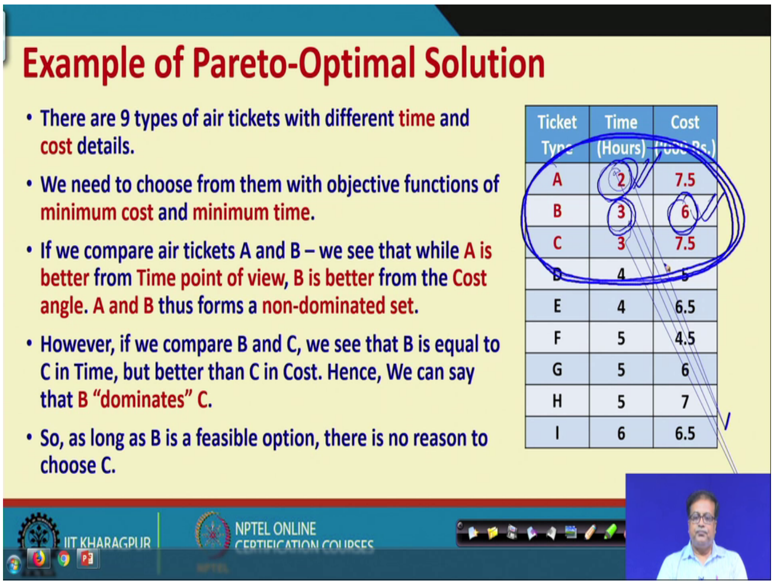
So, here you can see that the solutions along the line. So, you see if you if you just look at these particular plot, then you may find that in this plot there are a series of solutions all of them are feasible solutions is alright. And out of all those feasible solutions, if I find the solutions on the boundary is it not, you know these are the obtained solutions usually they constitute a Pareto-optimal front. And the solutions on the Pareto-optimal front are usually Pareto-optimal.

But then the Pareto-optimal front shape need not be same all the time you know there could be different shapes. For example, you know here the shapes are different is it not. There would be different shapes, but suppose when the shapes are different you know you may have difficulty in interpreting the solutions. So, all these things are there and the solutions which are not on the Pareto front, you can say that those are dominated solutions right. The dominated solutions usually will be inside the line because if I compare a solution on the optimal front you know you may find that that there is at least one solution which will be better than the solution which is you know better than the solution which is inside.

So, let us see, what exactly we are talking about. Suppose this is a point A, which is a solution, which is inside. Now, with regard to f 1, in these direction, and f 2, these are the values right. Now, you can see, at least one solution. Suppose let us take these point, call it B. The B is definitely better than A, with regard to both the objectives. So, we can say that B dominates A right. So, A e becomes dominated solutions, and because A is a dominated solution, it cannot be part of the Pareto-optimal front.

So, in that sense, all the solutions which will form our Pareto-optimal solution, they can be called as non-dominated is it all right. And it should be important to find solutions as close as possible to the Pareto front, and as far along it as possible is it alright, so that is the basic idea of the Pareto-optimality. But it will be more clear, if we take an example.

(Refer Slide Time: 08:55)

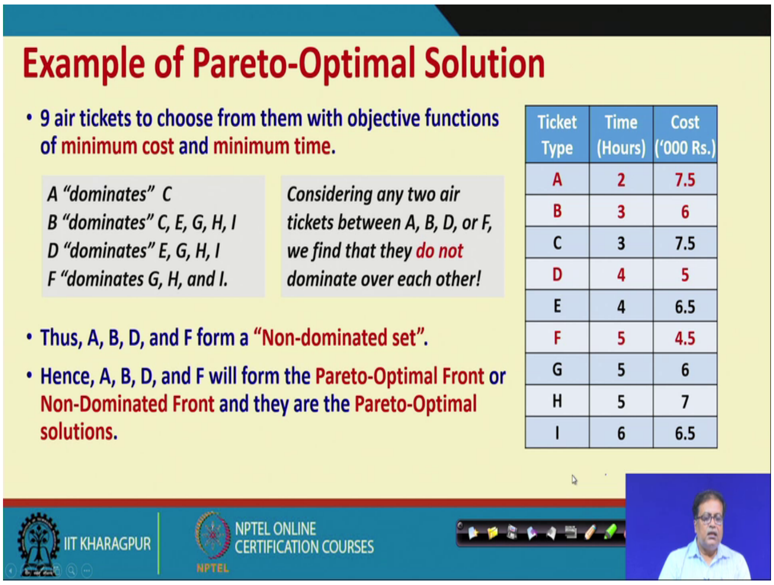


ah Let us look at these particular example. Suppose, there are 9 different options in front of us to buy air tickets right, which are A to I. Now, in each option, you know if we travel, it takes certain hours of time, and certain 1000 of rupees of cost. For example, if you go by the ticket type A, then it will take 2 hours to travel, but cost will be 7500 rupees is it alright. And our objective function is to minimize cost and minimize time, both at the same time.

So, let us take three solutions. Three solutions all are feasible, let us call them A, B, and C right. So, supposing we consider A, B, and C these three solutions, which are highlighted in red, and a let us compare them. Compare A with B, does A dominate B. Look at carefully, A win with regard to time, you know A takes 2 hours, but B takes 3 hours, so which one is better, A is better. With regard to cost again, you know which one is better, B is better. So, you can see that you cannot say that A is better than B, in all respects, A is better than B in time, but B is better than A in cost.

But, if you compare A with C, very interesting thing found that A is better in time, A is not worse off in cost. So, you can say then A dominate C. Compare B with C, B and C, you know B is equal in time with C, but cost is less. So, in that sense, you know also say, B dominate C is it all right. So, what about A and B, you can say that A and B, they form a non-dominated set is it all right. So, in that sense, A and B will be candidates for the Pareto-optimal solution is it all right. So, these are some of the things, which you know we should understand very clearly to have our understanding of the solutions for multi-objective optimization.

(Refer Slide Time: 11:43)

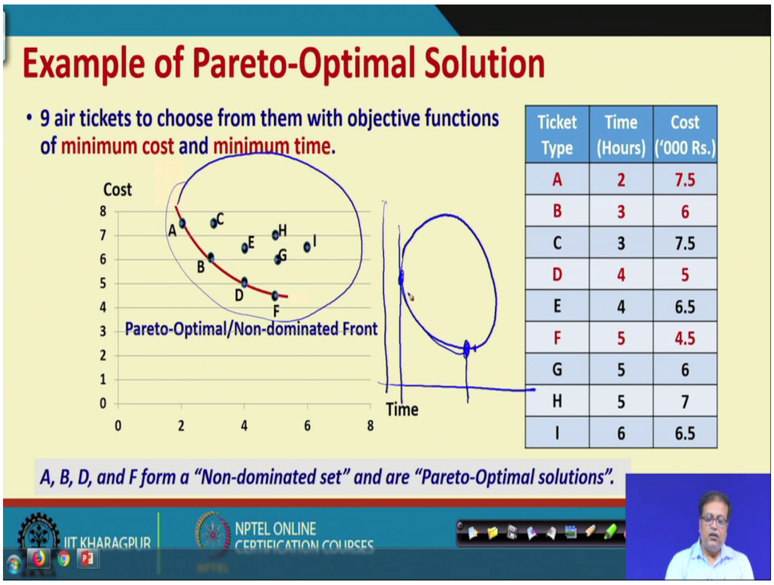


Now, moving further, you can see that A dominate C that is fine, A dominate C. But, A does not dominate any other, because A is the lowest in time that is fine, but cost also it is probably the highest. So, C it dominates, because C cost is also 7.5, but every other feasible solution has a lower cost. So, A does not dominate any other, but A dominate C. The B dominates C, B dominates E is it all right, B dominates E, and B dominates G, and B dominates H, and B dominates I right. D dominates E, D dominates G, D dominates H, and D dominates I. Why, because look here, D is 4 5, and you know all those other ones that is E, the time you know when it comes to D between D and E, you can see both are same time, but cost is lower in D. And with regard to the G, H, I, D is better in both respects right

Similarly, F, G, H, I, if I compare, the F you know is equal to G in time, but better in cost. And F and H again time is same, but cost is less in F. And F is better than I in both time and cost is it alright. So, this is about the dominates. So, but then, the remaining ones that is A, B, D, and F, if you compare them in pairs, you know interesting thing that we shall find is that no clear winner can be found, no clear winner is it alright.

So, if I see A and B, we find that you know no clear winner. If you see A and D, again D is better in cost, A is better in time. A and F again, you know similar kind of thing. So, we find that A, B, D, F, they form a non-dominated set. And therefore, our conclusion could be A, B, D, and F, they will form a Pareto-optimal front or a non-dominated front, and they will be our Pareto-optimal solution is it alright. So, they will together form a set of Pareto-optimal solution right.

(Refer Slide Time: 14:36)



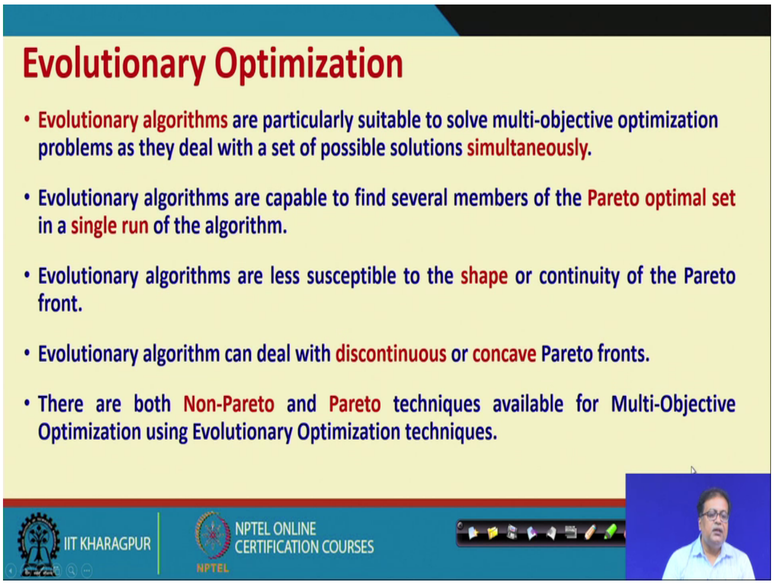
Now, further, if we plot them. Now, look here if we plot all of these A, B, C, D, E, F, G, H, I, then clearly can you see that while all the solutions are here, this is all the solutions. Now, out of all the solutions, this A, B, D, F you know they are in the Pareto-optimal front or the non-dominated front is it alright. And interesting thing could be on a graphical way that you know if I draw a 90 degree line on the curve, a 90 degree tangent.

So, supposing these are all the set of solutions. So, these are all the set of solutions, and these are my axis. And if I draw a 90 degree tangent right, a 90 degree tangent, so something like it will just touch here. So, you know it will be somewhere here, it won't go any beyond, because then you know the slope changes. So, slope should be within these, so somewhere here, so that portion can be the Pareto-optimal solutions.

Anyway it will not be possible to do through geometry, because particularly if these are the cases, where only two sets of solutions are there objective functions, but there could be multiple objective functions, instead of simple time and cost. So, here it is only time and cost. But, suppose it is 3rd dimension, then it will be a 3-D figure right, or there could be 4th dimensions, 5th dimensions etcetera.

So, we have to have a kind of methods, which can be called as non-dominated sorting. So, what is non-dominated sorting, we shall see a little later. So, we have understood that how exactly we find the non-dominated set, and how do I find the Pareto-optimal solutions. So, with that knowledge let us go further.

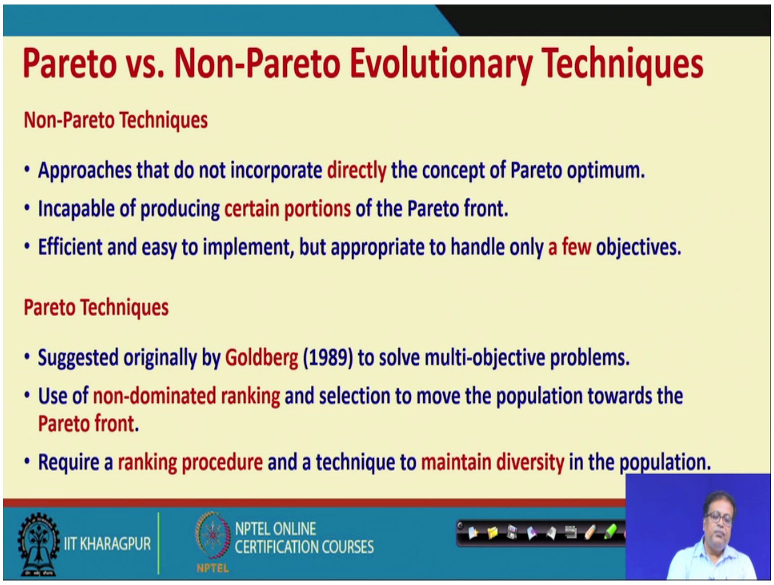
(Refer Slide Time: 16:54)



And let us see what are the some methods by which we can do the you know solve multi optimization problem through meta heuristic techniques. So, we can do, what is known as the evolutionary optimization techniques. The evolutionary optimization kind of methods, there are several methods we have already studied starting from genetic algorithm to simulated annealing to the, you know the PSO methods, and all that all of them in a short can be called as evolutionary algorithms.

So, evolutionary algorithms are capable to find the several members of the Pareto-optimal set in a single run right. And there also less acceptable to the shape or continuity, it is a convex or concave, and it handles the concave once or discontinuous one very nicely. And there are both Non-Pareto and Pareto methods for multi-objective optimization using evolutionary optimization techniques right.

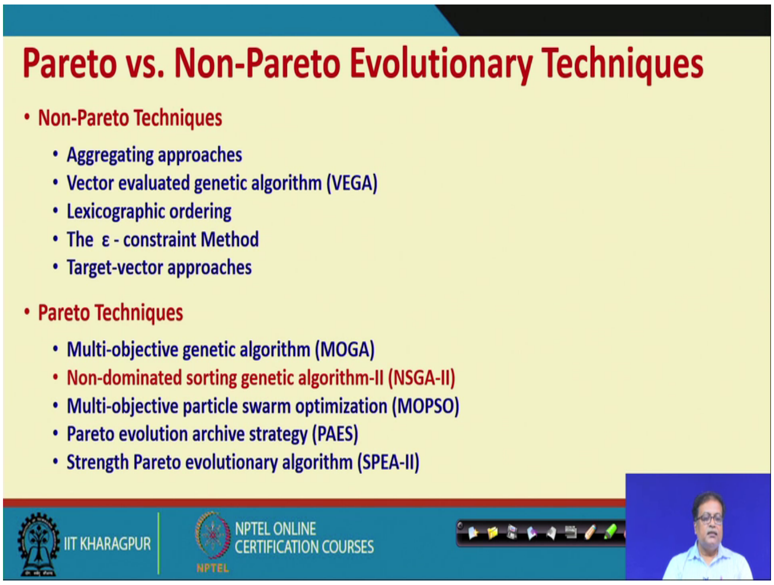
(Refer Slide Time: 17:59)



The Non-Pareto methods they do not incorporate directly the concept of Pareto optimum. And therefore, it can produced cannot produce certain portions of Pareto front right. Easy to implement, but cannot handle large number of objectives.

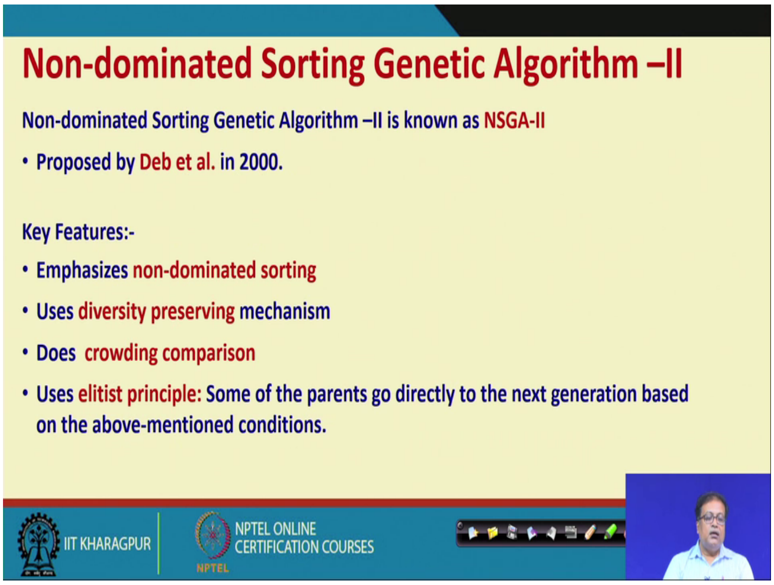
The Pareto techniques on the other hand, initiated by Goldberg; Goldberg has got a you know he was a direct student of Holland, who first talks about the genetic algorithm or meta heuristic techniques. And Goldberg his book on genetic algorithm is also very very famous to solve the multi-objective problems using the Pareto techniques. So, usually it uses non-dominated ranking and selection to move the population towards the Pareto front right. And the two procedures are followed, one is a ranking procedure, and another is a technique to maintain diversity right. So, these are the two techniques that are followed.

(Refer Slide Time: 19:05)



So, here are some Non-Pareto techniques, aggregating approaches, vector evaluated GA, lexicographic ordering, e-constant method, and target vector approaches. And some of the Pareto techniques include the multi-objective GA, non-dominated sorting GA, particularly NSGA-II that we shall discuss that is why it is highlighted, then multi objective PSO, Pareto evolution archive strategy, strength Pareto evolutionary algorithm is all right. So, these are some of the Pareto techniques that we shall have.

(Refer Slide Time: 20:02)



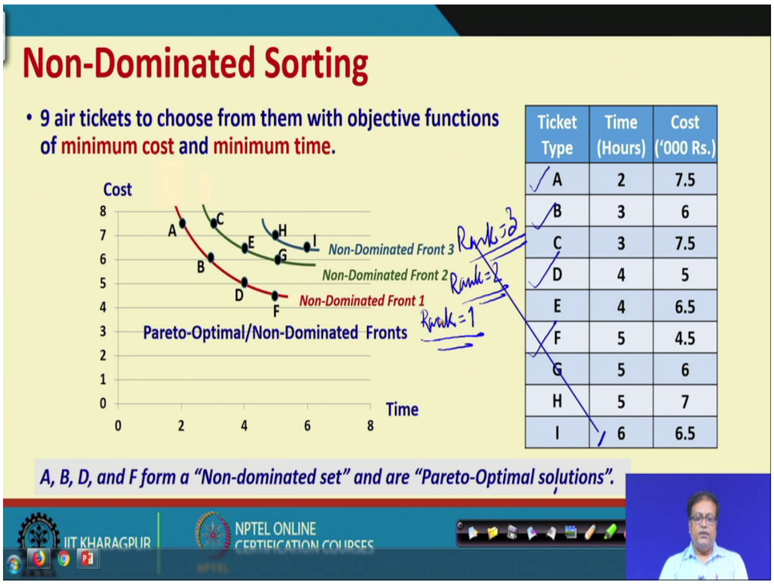
Now, specifically we are going to discuss what is known as the non-dominated sorting genetic algorithm II or the NSGA-II. In the subsequent lecture, we shall give an example. But, here also I shall take a simple example, the in fact the example that we have already taken, how exactly the NSGA-II the broad idea is can be explained through that simple example.

Now, (Refer Time: 20:19) or Deb et al, there are other authors, he proposed these NSGA-II, in the year 2000 that paper. In that you know specifically if you see the NSGA-II or the non-dominated sorting are the name suggests, it emphasizes non-dominated sorting emphasizes the diversity preserving mechanism. If you recall, you know while discussing meta heuristics, I have specifically told that any meta heuristic method has to have two things, one is there should be selection pressure, and there should be the population diversity.

The selection pressure, so that the best chromosomes are selected. The population diversity, see to it that the entire portion of the population is well represented. Why, because so that search can be really multi pointed. Why, otherwise there is a chance of you know localizing or ending up at a local optimum solution. It does a crowding comparison; we will explain it later, in order to achieve this. And finally, it also uses the elitist principle; some of the parents go directly into the next generation based on above-mentioned conditions.

So, exactly how NSGA-II is applied, the exact method of meta heuristics, specifically the elitism and others can be understood in the next lecture. But, in this particular lecture, we shall specifically try to understand the non-dominated sorting, and the diversity preserving through the crowding comparison, or crowding calculations crowding distance calculation.

(Refer Slide Time: 20:44)



So, what is non dominated sorting. See the non-dominated sorting very important example is that we have already seen in these particular example of you know choosing air tickets with minimum cost and minimum time, we had those options. And from those options of time and cost, we have seen earlier that the A, then B, then D, and F, they have together, they form the Pareto front is it alright. So, we call them you know the line that we get by joining them as the non-dominated front 1 right. And we use non dominated rank, so maybe we can say that this A, B, D, F, they are non-dominated rank is 1 right. So, I call it, non-dominated front 1.

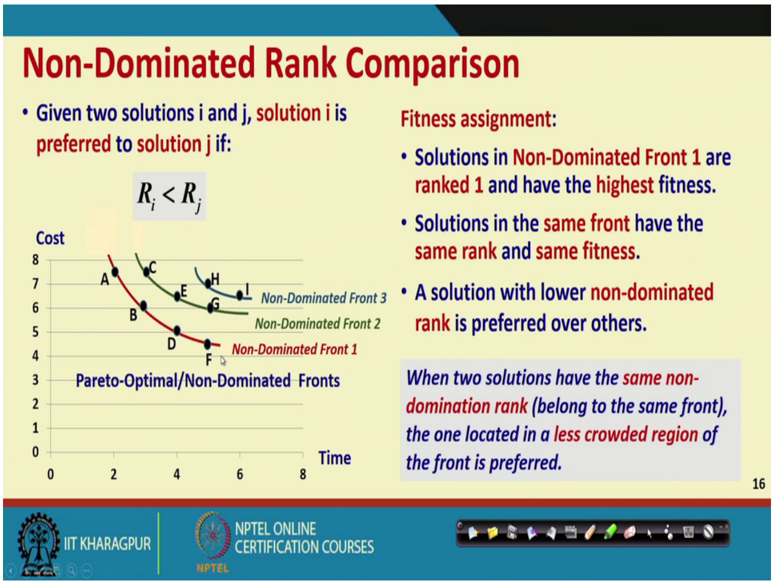
Now, leave out A, B, D, F, so we have the five left C, E, G, H, I right. Suppose we do the same procedure once again right, C, E, then G, and H, I you see that if I compare between C and E, you know they do not dominate each other, again C and G; they do not dominate each other. But, then they dominate over H and I, at least some of them dominate over H and I. So, H and I goes out, and C, E, G again forms a non-dominated front leaving out A, B, D, F is it alright. So, they will be now our non-dominated front 2, which therefore, their rank will be at this stage will be the rank will be 2 right. So, at this stage, their rank will be 2.

Now, leaving them, the last two that is H and I, again you see you know they do not dominate over each other. So, they front the non-dominated front 3, they will be forming.

So, for them, the rank will be 3 right, so that is what we shall get. We have rank equal to 1 for A, B, D, F, rank equal 2 for C, E, G, and rank equal to 3 for H and I is it alright.

So, by non-dominated sorting, the method will be explained through an example in the next lecture. But here we can see that A, B, D, F that the front 1, and their rank will be 1, so that means, for you know going from one population to the next, A, B, D, F will have first priority, because they are of higher rank, rank 1. And the C, E, G will be next priority, their rank is 2. And H and I, their rank will be 3. So, this is the first thing that is called the non-dominated sorting. And non-dominated sorting can be employed to really find out the ranks of the solutions, non-dominated ranks of the solutions right, so that is the first thing.

(Refer Slide Time: 25:57)

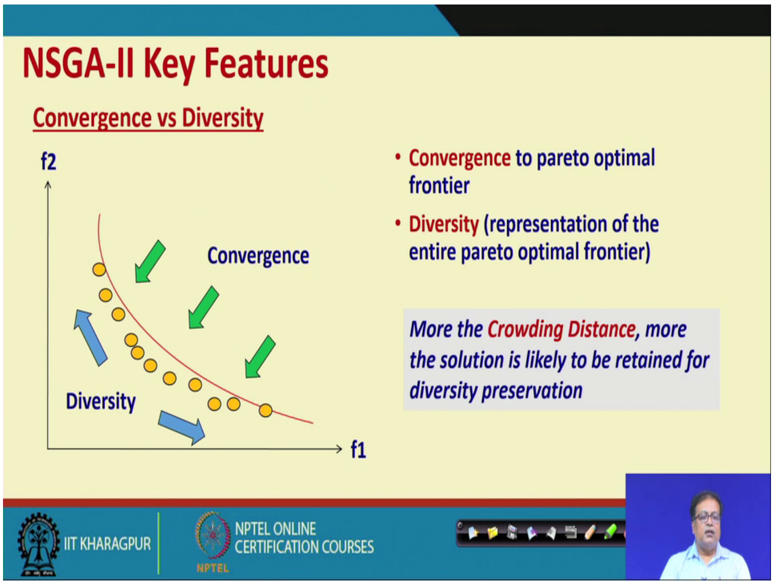


So, now the same thing is explained here that given two solutions, i and j, solution i is preferred to solutions j, if R i is less than R j, less here means that 1 is less than 2. So, all those with non-dominated rank 1, there in the front the first front, they will be more preferred than others whose ranks are higher, higher in the sense, rank 2, 3 etcetera. So, solutions in non-dominated front 1 are ranked 1, and have the highest fitness. Solution in the same front, have the same rank and same fitness. And a solution with lower non dominated rank is preferred over others.

So, when two solutions have the same non dominated rank that is belong to the same front, the one located in a less crowded region of the front is preferred. So, this is the next question. Suppose I have to choose 3 out of ABDF, so which three should I choose is it ok. Look here all of them are having the same rank, is it all right. So, the question is they should be well represented. And how do I well represent them that means, we have to ensure a population diversity. So, how do you ensure population diversity we should try to get solutions from all regions of the you know front, so which basically means that out of ABDF, suppose B and D are closed together then one of them should be ok.

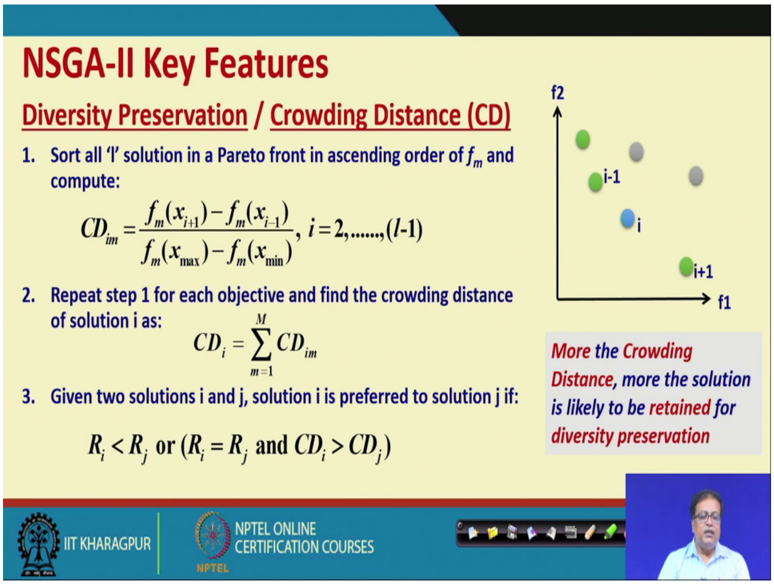
So, if suppose B and D are in a crowded region, their chances are reduced; whereas, any particular solution which is in the less crowded region, they should be preferred, that is the essential idea of the crowding comparison or obtaining what is known as a crowding n distance.

(Refer Slide Time: 27:59)



So, two things are important one is convergence. So, basically obtaining the Pareto-optimal front; And second is diversity; see that the solutions are representative of all parts of the population and that is done by looking at the crowding distance. And more the crowding distance, more the solution is likely to be retained for diversity preservation.

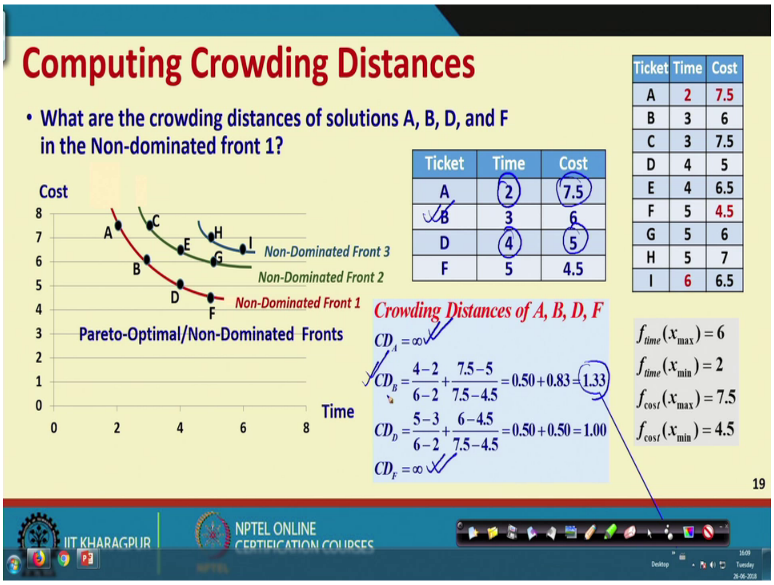
(Refer Slide Time: 28:31)



So, how do we calculate the crowding, the crowding can be calculate as calculated as you know this way that is that the for every objective function, so f is an objective function value the f m x i plus 1. So, this is the crowding distance for i right. So, the suppose we sort them according to the objective function, then what is the crowding distance of i.

Look at the two nearby ones that is i plus 1 and i minus i plus 1. So, the difference of the objective function value f m x i plus 1 minus f m x minus 1 divided by the highest function value minus the lowest function value this is with regard to one objective. And then sum over all objectives, is it all right. So, more the crowding distance more is likely to be retained, so that is how the crowding distance is calculated. What is exactly it is we will understand through an example.

(Refer Slide Time: 29:43)



So, let us see that particular example that is the same problem that we have taken. Say these are the four ABDF, and we have to find out their crowding distances according to the formula. So, look here with regard to the time, what is the minimum and maximum. So, you see with regard to the time maximum is 6 and minimum is 2. With regard to the cost maximum is 7.5, and minimum is 4.5. Now, the crowding distance of A B D F, now we know that the crowding distance of A will be infinity. Why, because you see A has no neighbor on the other side is it all right. So, since there is those extreme you know feasible solution, for them we assign a crowding distance of infinity. So, in that sense A and F will have infinite crowding distances.

What about B, the B the neighbors are A and D so, look here A and D the values are 4 and 2. So, 4 minus 2 minus 6 minus 2, those are the maximum values plus you know for all objective this is for one objective that is time. And with regard to cost, what it is, the 7.5 minus 4.5 that is the denominator and then numerator will be that value of A and D, so 7.5 minus 5. So, you see these are the values to compare. So, 2 and 4 and 7.5 and 5 to be obtained for B, is all right. So, 4 minus 2, 6 minus 2, 7.5 minus 5 by 7.5 minus 4.5, so it comes to 1.33 that is the crowding distance for B.

And what is the crowding distance for D. For D the values the neighbors are B and F. So, 5 minus 3 and 6 minus 4.5 divided by the extreme values; so, it becomes 1, is it all right. So, now suppose we have to choose any three, which three should we choose; obviously, A because it is infinity; obviously, F because it is infinity. And out of B and D, which one is higher; it should be B because the crowding distance is higher that means, the B is in a less crowded region, is it alright. So, that is how the crowding distance is to be brought into for making a choice of the solution which will go from one generation to the other generation right. So, that is how you know we understood certain part of the Pareto-optimality and then basic idea of the NSGA-2, is it alright.

So, with that we will stop here.

Thank you for your patient hearing.