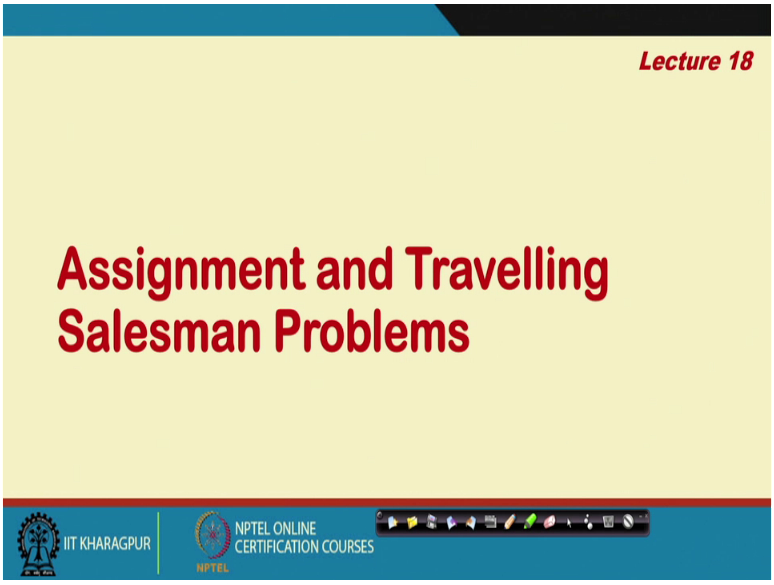
Selected Topics in Decision Modeling
Prof. Biswajit Mahanty
Department of Industrial and Systems Engineering
Indian Institute of Technology, Kharagpur

Lecture - 18
Assignment and Travelling Salesman Problem

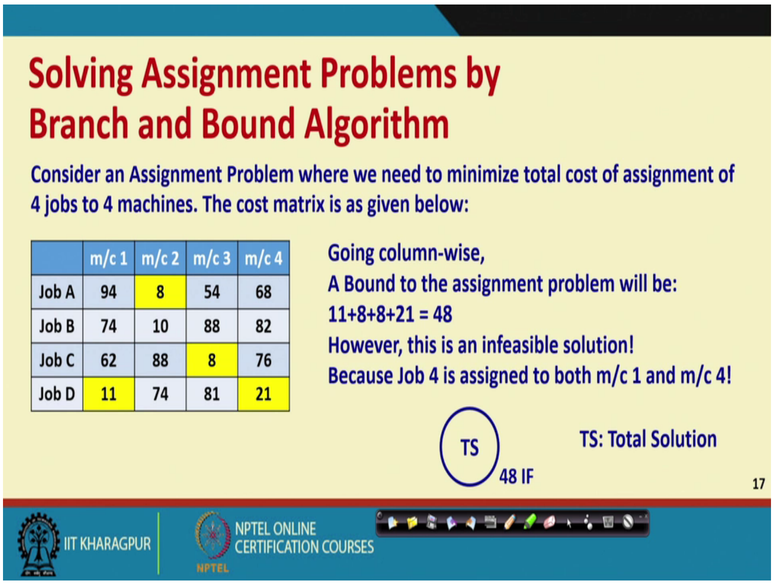
So, in our course Selected Topics in Decision Modeling; we are now in lecture 18; and here we shall discuss Assignment and Travelling Salesman Problems right.

(Refer Slide Time: 00:31)



So, assignment and travelling salesman problems.

(Refer Slide Time: 00:35)



In our previous lecture, we are in the middle of solving assignment problems by branch and bound algorithms. If you recall what I said at that point that point of time that we really do not solve assignment problems by branch and bound algorithms. There are powerful methods exact methods obviously branch and bound is also an exact method, that is the called Hungarian method. And we shall solve mainly the travelling salesman problem. You know when we solve travelling salesman problem, we shall make use of assignment problems as to find a quick bound.

At that time obviously, we are not going to solve that assignment also with branch and bound, because there are quick methods called Hungarian methods. But here is a quick example; how you can solve assignment problem using branch and bound. This will really make our understanding clear about the assignment problems.

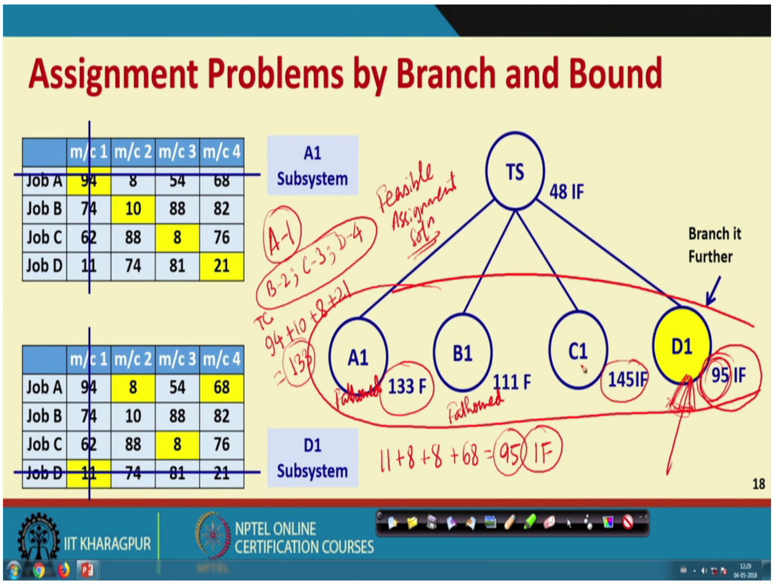
So, we have taken up these assignment problem in previous class. And we have seen how we can quickly find a bound by really seeing. See the assignment problem essentially is that there are 4 jobs and there are 4 machines. And we have to allocate these jobs to the machines in such a manner that every job is done by exactly one machine and one machine does only one job, is it all right.

But then cost are different like job A, if you want to do by machine 1 cost is very high, but if you do by machine 2 cost is very low, is it all right. So, obviously, machine 2 is a better option for job A right. So, what should be the allocation that minimizes the total cost. So, that is the assignment problem. Now, in these particular problem what we have done previously a quick bound to the minimum possible solution can be found out by simply noting the lowest values column wise or row wise any how we have done it column wise. So, in column wise you see in the 1st column the minimum is 11, 2nd column 8, 3rd column 8, and 4th column 21.

So, if you add them up then you get 48. So, this is a bound to the assignment problem that we have taken, but is it an optimal solution the answer is this is in feasible and therefore, we cannot say this is a final solution to the assignment problem. Why? Because here, look here job D is assigned to both machine 1 and machine 4; obviously, this is in feasible right.

As I said right in the beginning every job should be assign to an unique machine and vice versa. So, we have that total solution, but it is in feasible. Then, what should be do, we should then branch. So, I request you to think how do we branch this particular problem, again you can do row wise or column wise. Just think over how can you branch right. So, you can see how this is possible in our next slide.

(Refer Slide Time: 04:23)



What we have done. Look here the 1 that is machine 1 has to be put either to A or to B or to C or to D. There should cannot be any other option, is it not it should be one of the 4. So, those could be our branching, that is either A to 1 or B to 1 or C to 1 or D to 1. Are they mutually exclusive, yes; are they collectively exhaustive, yes, because 1 has to be assign either to A or to B or to C or to D. So, branching really fulfills our criteria of mutual exclusivity and collective exhaustibility.

I have been said that how do I find bounds for each allocation. Let us look at A 1. So, you see A 2 1 is already done. So, when I take this A 1 branch A 2 1 is already done. So, you have cut it off, because once A 2 1 is done neither A nor 1 can be allotted to anyone else, is it all right. So, that is what we have done, we have cutoff you know machine 1 and job A.

So, now, we have left to with this small matrices these B, C, D and 2, 3, 4. Again if you find the bound, how do we find. See machine 2 then can be allotted to 2, because out of these numbers, 10 is the lowest; out of these numbers, 8 is lowest; out of these number, 21 is lowest.

So, the highlighted portions would be our allotment right. So, in fact, let us write down that allotment also the allotment then will be A to 1 which is already known; B to 2; C to 3; and D to 4. And you know is it a unique assignment? Yes. Is it feasible then? It is a feasible assignment solution. But what is the total cost in that case, it will be 94 plus 10 plus 8 plus 21, so this is going to be the total cost.

How much is this, 104, 112, 133 all right. So, this is 133 and it is a feasible solution. So, you look here that is written there that is we have 133 feasible solution we get if we do A to 1. So, A to 1 subsistence is 133 feasible. See the TS when we had done that is a total system, at that time we have not taken out A or 1, so that is a difference between the subsystem A 1 and the total system.

Another example is given here that is, D 1 subsystem. So, what is D 1 subsystem? In the D 1 subsystem, you know we have first of all you know D to 1, so obviously D and 1, they are removed. In the remaining matrix, the allocation is A to 2 because that is a minimum, C to 3 and again A to 4.

So, again look here A is allotted to both 2 and 4. So, this solution is not feasible. So, value 11 plus 8 plus 8 plus 68. How much is that? 11 plus 8 plus 8 plus 68. How much, please calculate? 27 plus 68, 95. But is it feasible? No, it is infeasible. So, 95 in feasible is written there all right.

Now, similarly I am not showing the details, you can also calculate the bound for B 1 and C 1. They are found respectively 111 and 145. So, what is to be seen now, that look here we have now found bounds of the solution 133, 111, 145 and 95. Now, one thing to note, see the tree is a network. Out of this network, these are my N nodes.

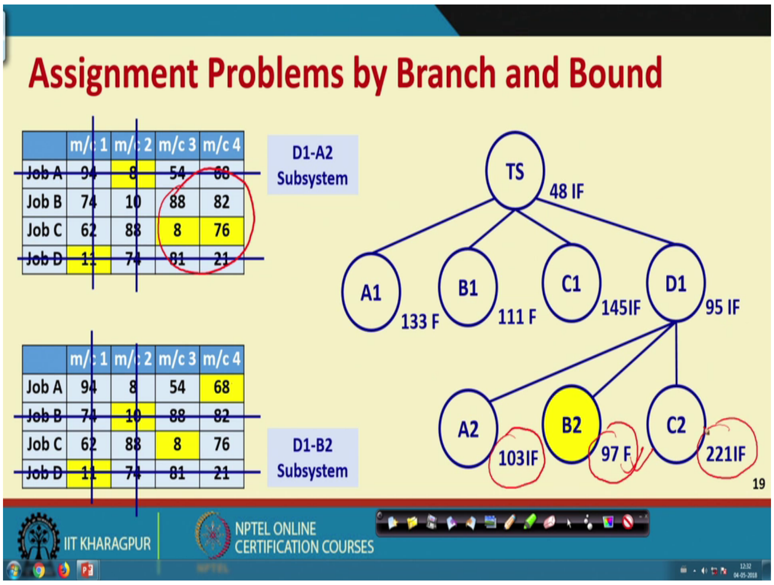
You see what are some N nodes, the N nodes are those which are not further branched right that means, they are at the end of the you know the tree. So, out of all the N nodes, if you compare and then find out the lowest in case of minimization and highest in case of maximization, and then find the bound.

And if you find it feasible, we found the optimal answer right. So, which one is the lowest here? The lowest is here right. The lowest is here that means, these is a candidate for further branching. Why? Because no point branching A 1 further, because anyway we have feasible solution, so these A 1 is fathomed; B 1 is also fathomed right. A 1 is fathomed. Why, because the base solution is found. B 1 is also fathomed, because the feasible solution is found. C 1 is in feasible; D 1 is also in feasible.

So, we they are not fathomed, you can branching it further. Which one will you choose C 1 or D 1? Obviously D 1, because that value is lower right. So, if you if you branch C 1, obviously we are not going to get any value better than 145, it will be worse than 145. So, since we already have 95 here, we can take it further.

Question is look already there is 111; which is feasible. So, anything if you get feasible out of C 1 later on by branching it further obviously we are going to get something which is worse than 145. What value is that, because that will be always lower than higher than 111 so that means, it is there is no need to really go further into C 1, is it all right. So, these kind of logic you must use that which one should be we should branch further and which one we should not right; So, having said that we need to know branch D 1 further.

(Refer Slide Time: 11:53)



	m/c 1	m/c 2	m/c 3	m/c 4
Job A	94	8	54	68
Job B	74	10	88	82
Job C	62	88	8	76
Job D	11	74	81	21

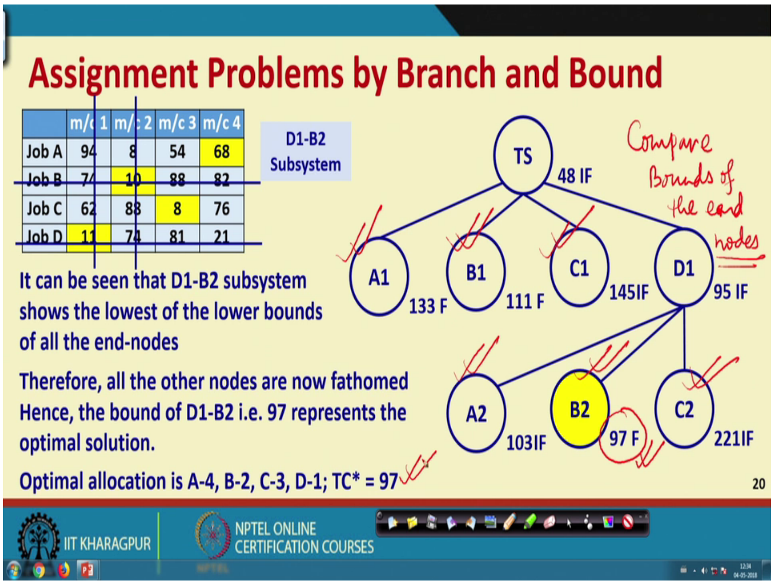
	m/c 1	m/c 2	m/c 3	m/c 4
Job A	94	8	54	68
Job B	74	10	88	82
Job C	62	88	8	76
Job D	11	74	81	21

So, let us see how it goes. So, look here now we have taken out the A 1 and D 1 and then you know it could be A 2; it could B 2; it could be C 2. So, this is the D 1, A 2 that means D 1 and A 2 this portion D 1 and A 2, then D 1 and A 2 then B 2 and D 1 and C 2. So, two examples are shown here. If you see D 1, A 2, then these two helps to be cut, because D 1 is cut that means, D is cut then 1 is cut, then A 2, A is cut and 2 is cut. So, when all those are cut you know you have only a matrix available that is this part right. So, only a small portion is remaining.

Now, again 8 and 76 they are lower. If you do column wise, but that is an infeasible solution; the total comes 11 plus 8 plus 8 plus 76 that becomes 103. And it is an infeasible solution. So, look here that is what is written there. If I go to B 2, then we see that after cancelling out, you know D 1, which was our earlier one, and then B 2 then we get these portion which is 54, 68, 8 and 76. The bounds are 8 and 68.

So, what should be the value? The value should be then 11 plus 10 plus 8 plus 68. This becomes 97 and it is a feasible solution, is it all right. So, we get a feasible solution here, and C 2 we may calculate, we find; it is 221 infeasible. So, how do all of this compare now? Let us look at in our next slide then how entire thing compares.

(Refer Slide Time: 13:58)

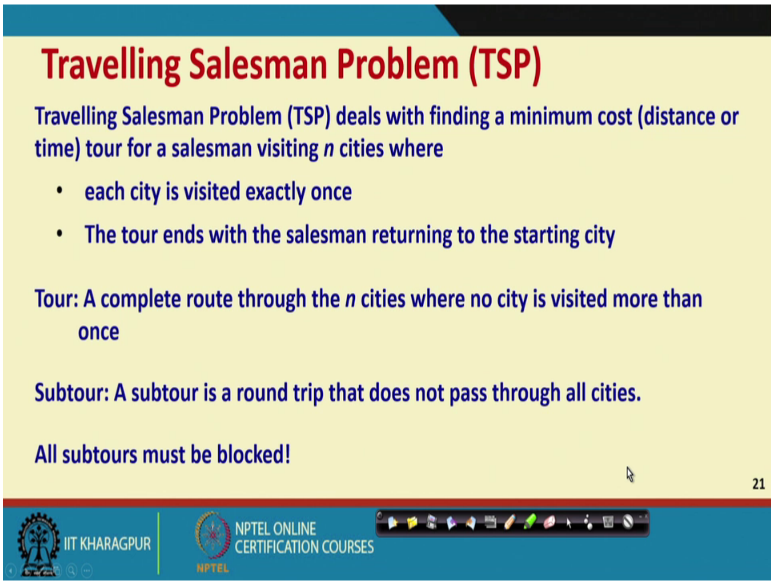


So, what are the N nodes? Let us take those N nodes right. Let us take all the N nodes that we have found so far. So, A 1 this is 1; B 1 this is N node; C 1 it is another N node; A 2, B 2 and C 2, so all are taked. Say D 1 and TS they are not N nodes, because they are you know branched further. Now, compare bounds of the N nodes, compare the bounds. Which one is the best possible value? Which bound is the best from a minimization problem point of view?

It is this one 97. See is a lowest, and it is a feasible solution. So, look here if I branch any one of them the other N nodes further, we are not going to get anything better than the anything value that we already have. Since, this is what it is it automatically therefore, is this is our best possible value, that is 97 feasible. Is it all right? So, that is how we solve and then the optimal allocation is the highlighted one that is A to 4; B to 2; C to 3 and D to 1 and TS is 97. I hope it is cleared now right.

So, that is about solving assignment problem by branch and bound, but as I said we are not going to solve assignment problems by branch and bound branch and bound method. We shall solve by Hungarian method. When we solve what is known as the Travelling Salesman Problem. So, let us some move over. And see the travelling sales man problem.

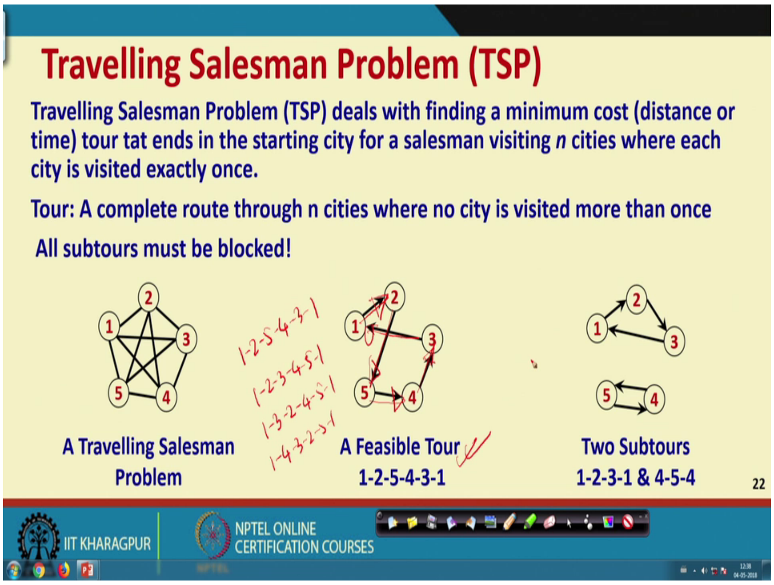
(Refer Slide Time: 16:18)



The travelling sales man problem deals with finding a minimum cost, distance or time, tour for a salesman visiting n cities. Where each city is visited exactly once, and tour ends with the salesman returning to the starting city. So, he starts from a city goes to every other city exactly once, comes back to the starting city, and you know what should be his route? Considering that you know he wants to move the minimum distance or minimum time or in minimum possible cost that is what we need to find out.

So, a tour is a complete route through the n cities where no city is visited more than once. But then there could be various subtours, a subtour is a round trip that does not pass through all the cities. So, the solution of Travelling Salesman Problem really requires that we block all the subtours. And this is not going to be easy, because there could be so many subtours in a particular you know problem that we consider. Let us see what are they.

(Refer Slide Time: 17:47)



So, here is an example you know there are five cities, and all these five cities are connected by parts. You know you can see you can go from 1 to 2; 1 to 3; 1 to 4; 1 to 5. Obviously, they all paths may not be available all the time, is it all right. But then this is a complete thing that you shown here. Question is how do you go from 1 to you know 2 to 3 to 4 to 5 and come back to 1, you know, so that the total cost or total time or total distance is minimized or total profit maximize also, that could be another case as well.

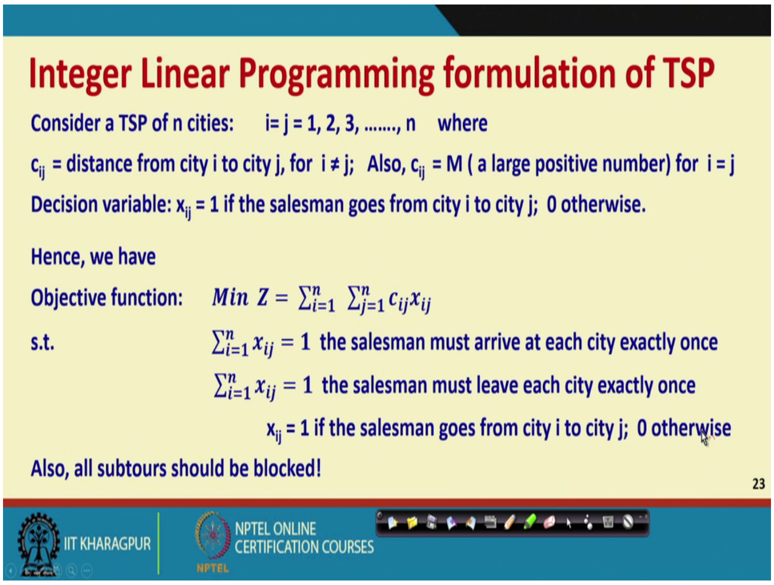
So, look here, here is an example away feasible tour that is 1 to 2, you know we have 1 to 2, then 2 to5, 5 to 4, 4 to 3, 3 to 1 right, so this is a feasible tour. You can also think or obviously, 1 to 2 to 3 to 4 to 5 to 1 possible. You can go from 1 to 3 to 2 to 4 to 5 to 1; or 1 to 4 to 3 to 2 to 5 to 1. You know, 1 to 2 to 5 to 4 to 3 to 1. So, you see there could be various combinations that are possible, and such combinations are going to increase as a problem size increases is it all right.

So, if we have let us say tens city travelling salesman problem, it could be that much more complicated. But then you know the problem is really not complex, because we have so many tours to compare that itself is not such a big problem. What is a problem really is how to block those subtours.

So, supposing we have you know 1 to 2 to 3 and comeback to 1. And so and then 4 to 5 to 4 you see they are not allowed, they are all subtours. So, can you think of all different subtours that you can think right. So, you see 1 to 2 to 1; 1 to 3 to 1; 1 to 4 to 1; 1 to 5 to

1; 2 to 3 to 2, you know 2 to 4 to 2; 2 to 5 to 2 etcetera, etcetera. These are all size 2. What about size 3? 1 to 2 to 3 to 1; 1 to 2 to 4 to 1; 1 to 2 to 5 to 1, and 2 to 3 to 4 to 2. You see similarly there are subtours of size 1, size 2, subtours of size 3, subtours of size 4, and all of them has to be blocked right that is a non-trivial task.

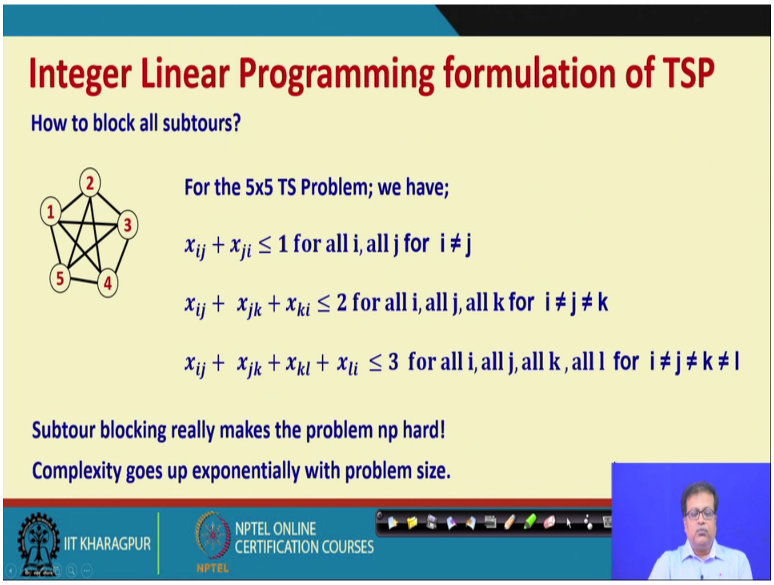
(Refer Slide Time: 21:05)



So, let us look at then some of these what we have right. So, before that so all those subtours have to be blocked right. So, let us look at the formulation the considered a TSP of n cities i equal to j equal to 1, 2, 3 to n. c i j is a distance or cost whatever you say from city i to city j, and c i j equal to M for i equal to j, because there is no question of going from 1 to 1 or 2 to 2 etcetera. So, they should be blocked. And therefore, we can put M in 1 to 1; 1 to 2; 1 to 3. M is a large positive number, anything you add to it small numbers or deduct from M, M remains same right.

So, therefore, as you can see the objective function is minimize Z equal to sigma i equal to 1 to n; sigma j equal to 1 to n; c i j x i j sum over i x i j will be 1, and sum over j not i they should be j really. So, and sum over j you know x i j will be also 1. So, therefore that means that you know the salesman must arrive at each city exactly once, and sales man must leave each city exactly once, so that keeps those two conditions. And x i j could be 0 or 1. If it is 1, that means, it goes from city i to city j, and 0 means you know it does not; and all subtours must be blocked so that is a formulation.

(Refer Slide Time: 22:48)



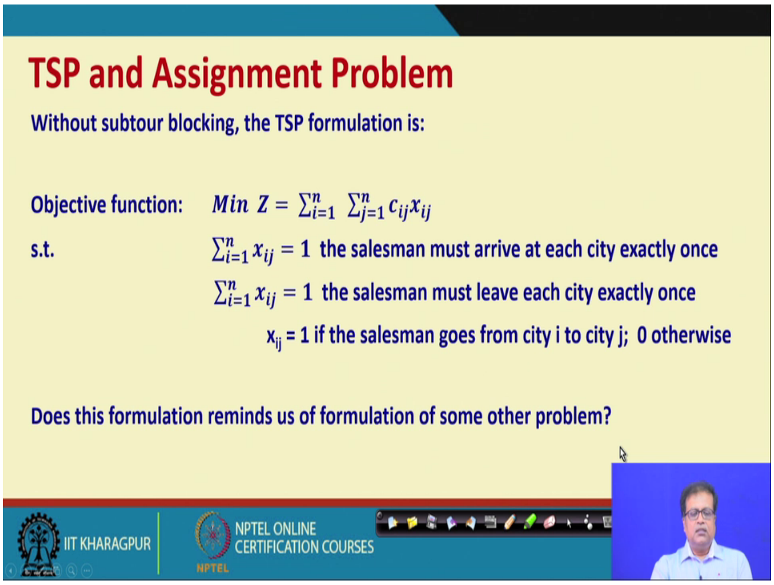
And these formulation you know so here is that how to block the subtours that has been shown. So, for the 5 5 TS problem; x i j plus x j i should be less than equal to 1. What does it mean that if x i j equal to 1 then x j i should be 0. You know if x j i equal to 1; then x i j should be 0, for all i for all j i not equal to j. x i j plus x j k plus x k i should be less than equal to 2 right, because if 1 to 2 is there, 2 to 3 is there, then 3 to 1 cannot be there all right.

Similarly, x i j plus x j k plus x k l plus x l i should be less than equal to 3 that means, if 1 2 is there, 2 3 is there, and 3 4 is there, 4 to 1 should not be there right. Only three of them should be available.

So, all these equation, all these constraints are to be written to really make the problem formulation. So, you can see these really makes the problem np hard. What does it mean, complexity goes up exponentially with problem size right. So, solving 10 by 10 problem is like solving 10, 9 by 9 problems right so that is sort of complexity increase really happens. So, it is not that you cannot formulate an np for a travelling salesman problem you can do it, but solving will be enormously difficult. So, we have to find some methods.

The branch and bound method works well for some size, but beyond a certain size even branch and bound will fail. Then we have to use heuristic methods, meta heuristic methods which we shall discuss in due course of time right. Let us now see how branch and bound can be made use of for solving the travelling salesman problem right.

(Refer Slide Time: 25:04)

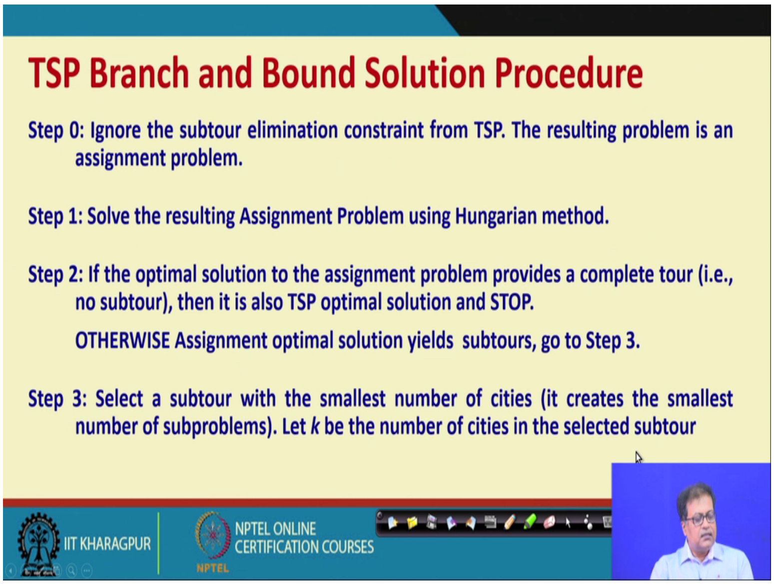


So, let us see how it is. So, this is how again we formulate. So, the total formulation is minimize Z equal to you know sigma i sigma j c i j x i j some over i x i j equal to 1 some over j x i j equal to 1 and x i j equal to 0 or 1. So, does these formulation remains as a formulation of some other problem right, think over. This is nothing but the assignment problem right.

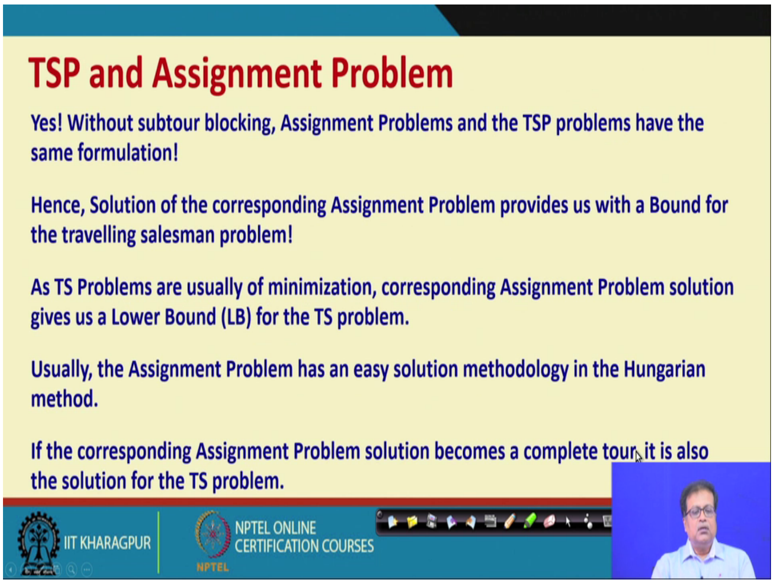
What happens in assignment problem, if you really take out the subtour blocking, then the remaining formulation is exactly same as that of the assignment problem right. So, that means, supposing there is no subtours the assignment solution is a complete tour itself, then the assignment solution itself is travelling salesman problem, but then the assignment solution need not be right.

Giving a solution which is you know also that TSP solution, it need not. Because the assignment solution may lead to a TSP solution which is having subtours right, so that is where the trouble is. But anyhow let us go ahead and see what exactly it means right.

(Refer Slide Time: 26:40)



(Refer Slide Time: 26:42)



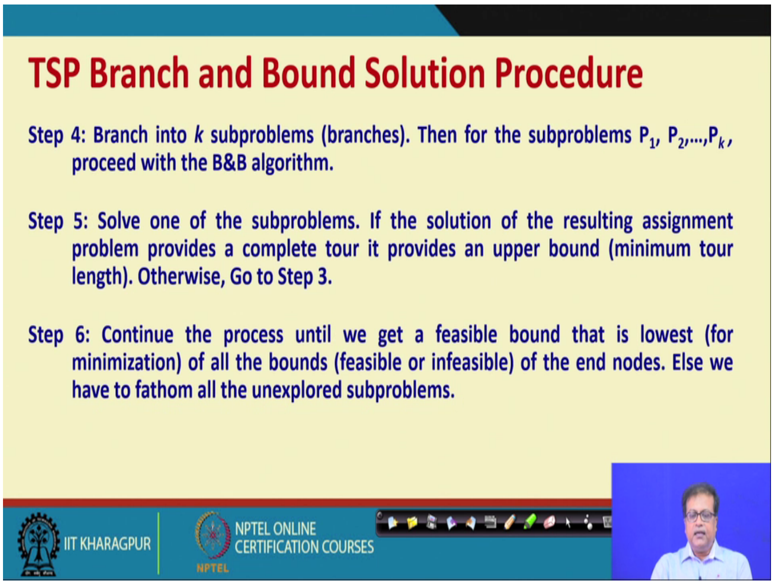
So, that is you know yes without subtour blocking, assignment problems, and the TSP problems have the same formulation. Hence, solution of the corresponding assignments problem provides us with a bound for the travelling salesman problem. So, as TS problems are usually a minimization corresponding assignment problem solution gives us a lower bound for the TS problem.

Usually the assignment problem has an easy solution methodology that is Hungarian method right. So, if the corresponding assignment problem solution becomes a complete tour it is also the solution for the TS problem. So, I already said that let us go further.

So, what is the methodology? Methodology should be therefore, will be like this step 0; ignore the subtour elimination resulting problem is assignment. Solve the assignment problem using Hungarian method. If the optimal solution provides a complete tour that is no subtour, then it is also TSP optimal solution stop.

Otherwise the assignment optimal solution yield subtours right then go to step 3. Step 3, select a subtour is the smallest number of cities; it creates the smallest number of subproblems. Let k be the number of cities in the selected subtours. So, you see the solution gives a subtour. Now, there will be usually 2 or 3 subtours. So, take the smallest subtour right.

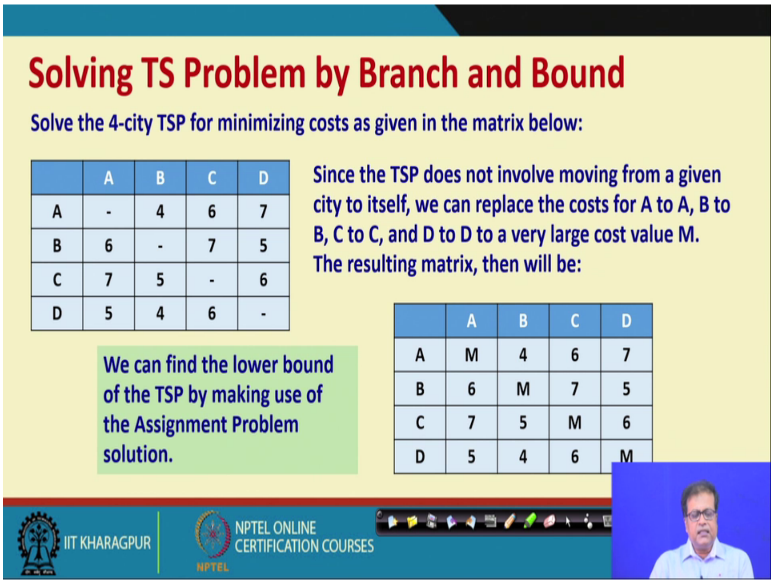
(Refer Slide Time: 28:24)



Take the smallest subtour and then you know branch it into k subproblems. If this subtour has got k legs, then make k sub problems right that is how the branching. Now, each sub problem again can be solved by a assignment algorithm, is it all right, and then follow the methodology.

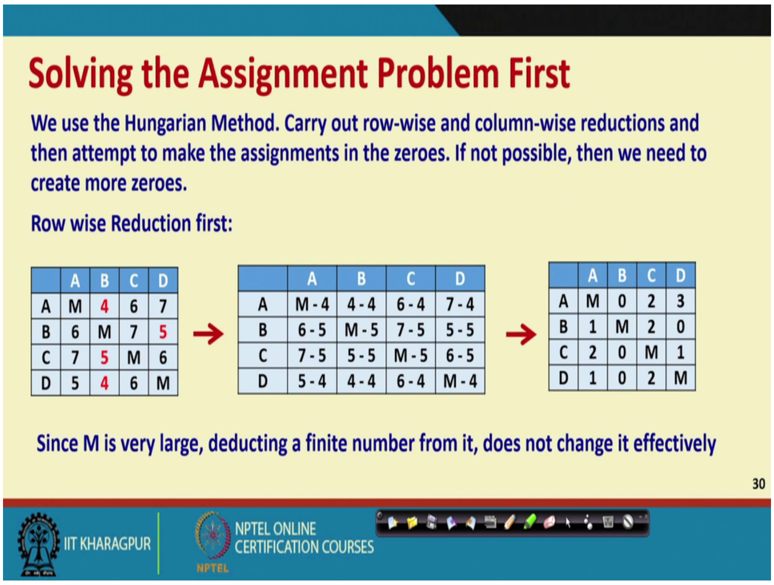
We know that that find out, you know those bounds and see all the N nodes and the 1 that gives us the best possible solution will be our solution right. Continue the process until we get a feasible bound that is lowest for minimization of all the bounds of the N nodes. But if you cannot find that then we have to fathom all the unexplored subproblems. So, this how that TSP can be solved by branch and bound procedure.

(Refer Slide Time: 29:19)



Let us take up the problem. So, here is a 4-city TSP for minimizing costs as given in the matrix below right. Since, the TSP does not involve moving from a given city to itself. So, we can replace all the corners as M, is it all right. So, M is a very high value M and anything you deduct or add small numbers M remain same right. So, this is how is our the matrix the TS problem matrix, and you can solve this by assignment problem to find the bounds.

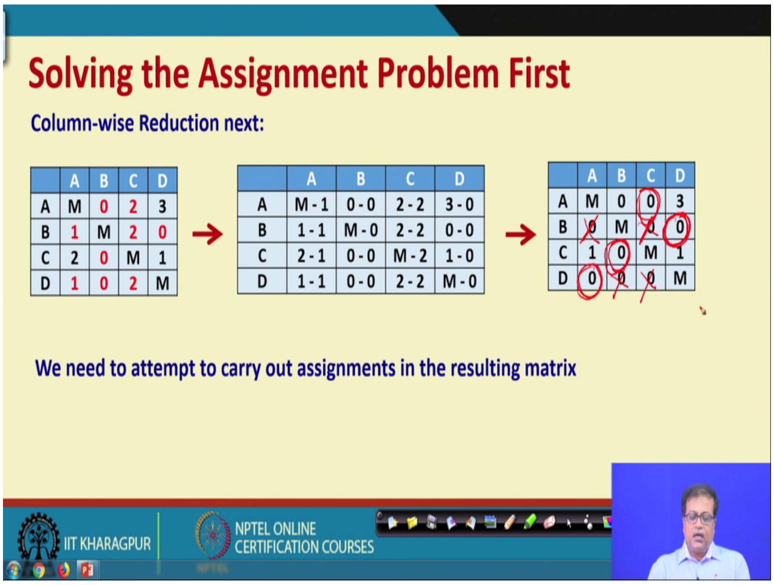
(Refer Slide Time: 29:59)



So, let us solve by assignment problems. So, what exactly we do in the Hungarian method, we do the row wise and column wise reductions right. And after those reductions, we find the zeroes; and then the zeroes we try to allocate that means we make our locations in those zeroes. But if we cannot, that means, you do not succeed to making all the allocations, then we have to create more zeroes right. Let us see how it goes.

So, this is our original matrix. So, these are the row wise smallest numbers 4, 5, 5, and 4. So, what I do, I deduct those numbers from every number, and then I get the resulting matrix. So, M, 0, 2, 3 say I have deducted 4, so M, 0, 2, 3. So, why similar method 1, M, 2, 0, 2, 0, M, 1, 1, 0, 2, M, so M minus 4 becomes M, because any how it still remains a very large number that means A to A should be blocked really so that is the row wise reduction.

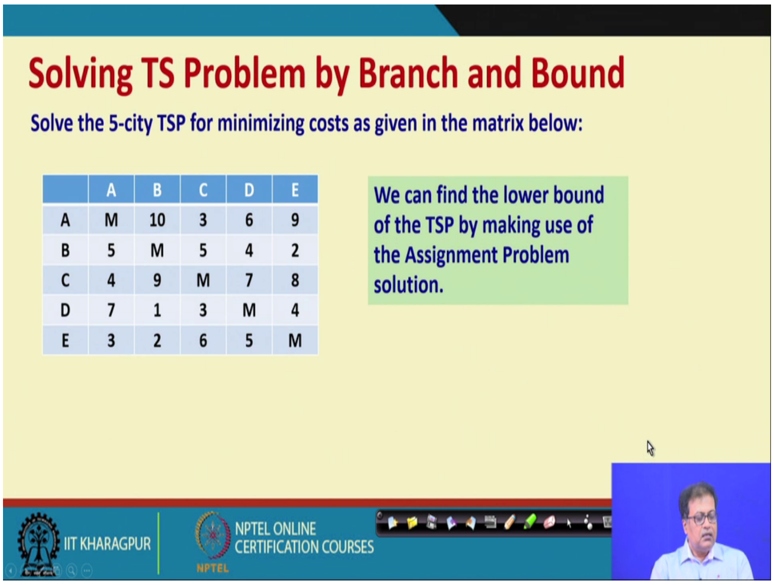
(Refer Slide Time: 31:14)



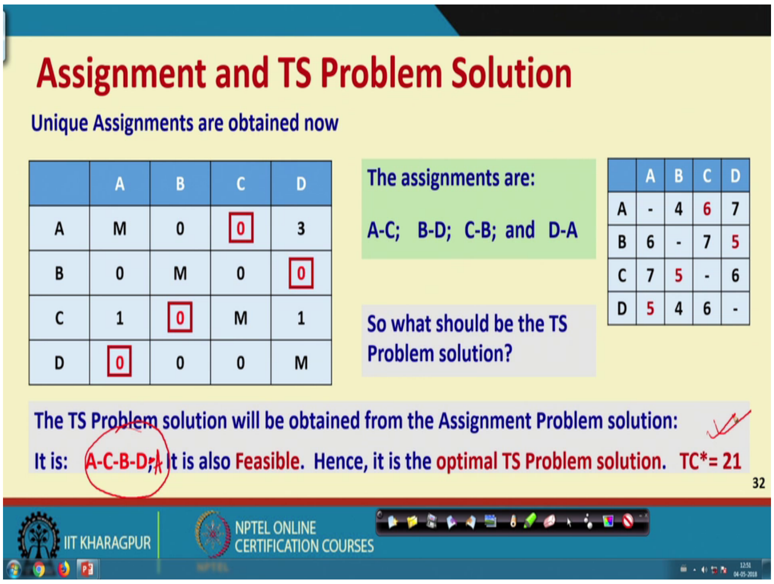
Then we do the column wise reduction. So, again we see these are the smallest number column wise. Now, if we if we do those you know, deductions then this is our resulting matrix. It is M 0 1 0, 0 M 0 0, 0 0 M 0, 3 0 1 M right. What we have to do now, we have to now carryout assignments. How to carry assignments, you know assign in the unique zeroes.

So, look this is an unique zeroes, so some assignments can be done here. But then these 0 gone right; so these 0 is gone also. Moment these two zeroes are gone then these becomes an unique 0s. So, we make some assignments that means, these two zeroes are gone. So, now these becomes unique, and these becomes unique. So, look here this is how we make the assignments right so that is shown in our next slide that is yeah.

(Refer Slide Time: 32:14)



(Refer Slide Time: 32:15)



So, this is our next slide. So, all those you know assignment have been made and that is A to C; B to D; C to B; and D to A so and then the total value is comes out to be 21 is it all right. So, what this is the assignment solution. So, what is the TS solution? We see A to C; C to B; B to D; and D to A. So, TS problem solution will be obtained from the assignment solution, and that should be A to C to B to D sorry to A right, that A is missing.

So, A to C to B to D to D to A, and it is also feasible, because we get a complete tour right. So, hence it is the optimal TS problem solution also. And what is the TC that is 21.

So, we all lucky here we solve the assignment problem, and we find the assignment solution is also the TS problem solution. So, we get the optimal solution for the TS problem right. So, this is the simple problem, really with there is no need to really go for branch and bound, the optimal solution for the assignment problem becomes the optimal solution for the TS problem, and you know with there is no need to branch or further bounds to be found, is it all right.

The bound for the total problem the TS total solution is 21 right so and that itself is feasible so that is a TS problem solution. But we may not be lucky like, these all the time right we shall take more complicated problems in our next lecture right.

Thank you very much.