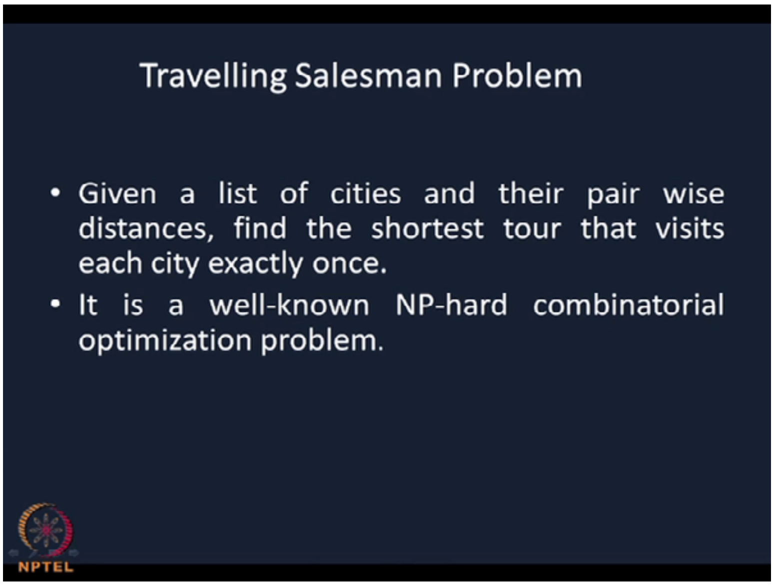
Optimization
Prof. A. Goswami
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture - 20
Travelling Salesman Problem

Today we are going to discuss the travelling salesman problem. In travelling salesman problem, what happens is basically you have n cities. Suppose a traveller is there, a salesman is there. Salesman will start from his home and after that the salesman will visit each and every city, and ultimately comeback to his home only. So, this is the travelling salesman problem and we want to find out is the shortest tour by which the time will be minimum.

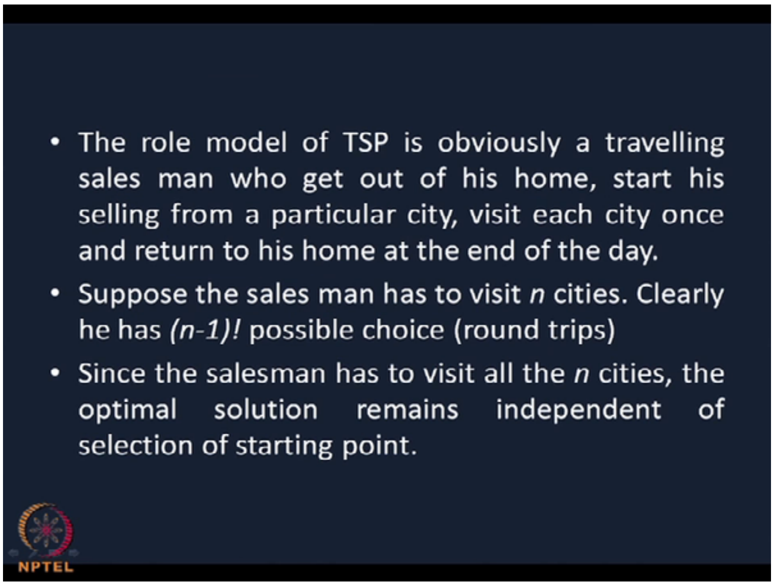
(Refer Slide Time: 00:50)



Let us see by in general given a list of cities and their pair wise distances what we have to find. We have to find out the shortest tour that visits each city exactly once. Please note this thing that each city is visited exactly once. We will see that it will enter one city only once and it will exit one city only once and not more than once. Also, we will come back to where we have started.

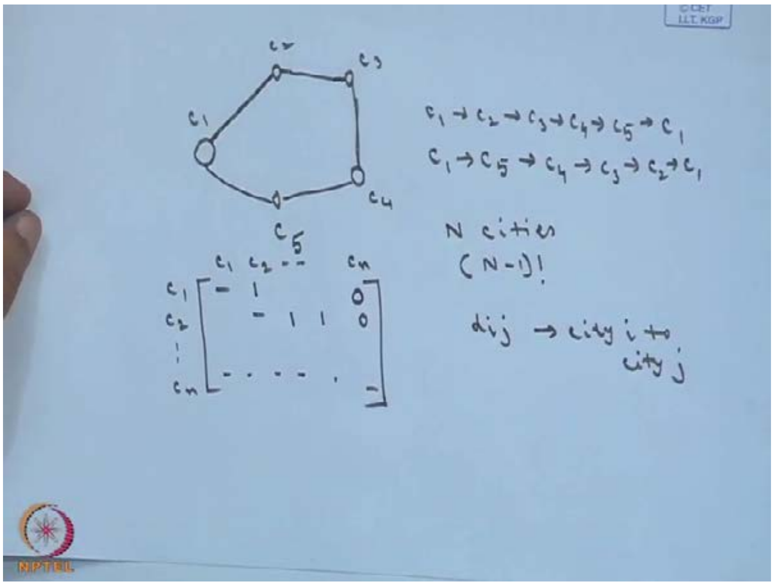
Next, if you see you are telling that it is a well known NP hard combinatorial optimization problem. Now, what do you mean by NP hard optimization problem? On NP hard problem, basically NP hard means the problem which can be solved in non-deterministic polynomial time. Usually if we can solve a problem in the polynomial time, its fine, but there are problems just like travelling salesman problem which cannot be solved in non-deterministic. The polynomial time for this reason we call it as the NP hard problem. We will see afterwards that the travelling salesman problem is the NP complete problem and that problem can be solved in the polynomial time. Then, all the NP complete problems also can be solved in the polynomial time.

(Refer Slide Time: 02:17)



So, this is NP hard problem. That means, we can obtain the approximate solution. Therefore, the role model of TSP as I was telling you obviously is a travelling salesman who comes out from his home, and start his selling from a particular city and visit each city once and return to his home at the end of the day again.

(Refer Slide Time: 02:37)

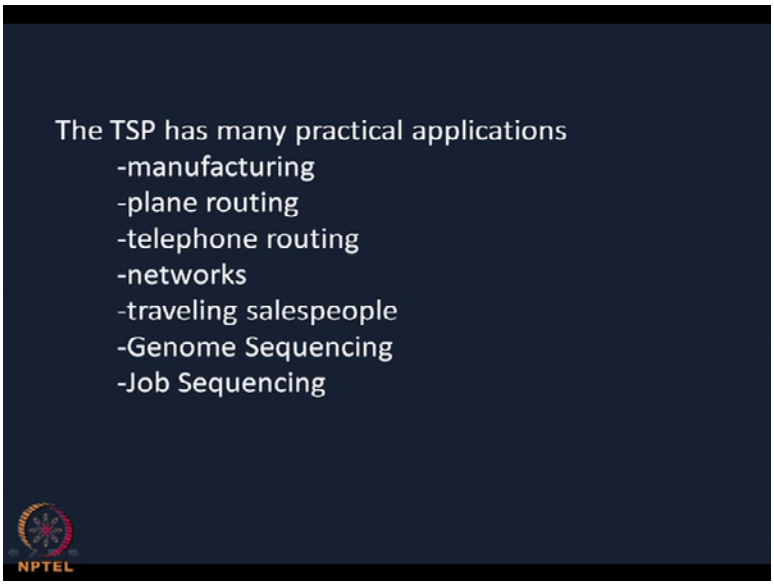


So, basically if you see there are n cities, if I draw something like this way. There are n cities which I am marking by these nodes such cities are c 1, c 2, c 3, c 4, c 5 something like this. So, from his home he starts. He has to visit all the cities one after another. The path may be so many that is I may visit c 1, c 2, c 3, c 4, c 5 and then, again I can come back to this, that is one path may be c 1, c 2, c 3, c 4, c 5 and then, again come back to c 1.

Similarly, I may go like this c 1. Then, I can visit like this way also that is c 1, c 5 and then, c 4, c 3, c 2 and then again I can come back to the original one. So, if you see like these different combinations can be made. In general, if we say that we have n cities, then there are total numbers of possible paths which will be factorial n minus 1 from here. So, you can understand if the number of cities is more, then what would be the complexity of finding all these paths because from these paths, we have to find out what is the optimum path. That is the path which will take the minimum time as I was telling you. This is basically the distance to cover from c 1 to c 5. It may be distance; it may be time whatever it may be. So, in factorial n minus time, factorial n minus 1, possibilities are there to cover n cities.

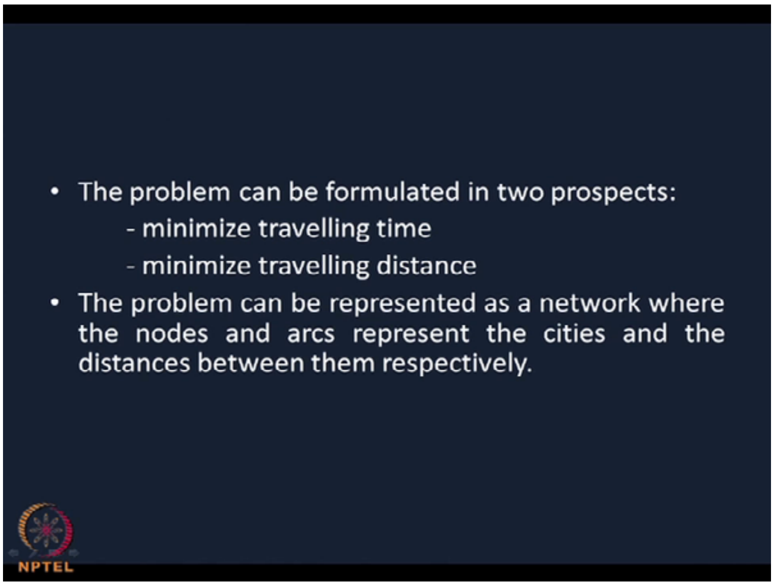
So, I was telling that there are salesmen as to visit n cities and then, he has n factorial n minus 1 possible choices round the trip. Round the trip means from where the salesman is starting, he has to come back on the same city. Now, since the salesman has to visit all the n cities, the optimal solution obviously will be independent of the selection of the starting problem that is whatever starting problem it may be, you will come back to the original one. You can start from any city.

(Refer Slide Time: 04:43)



Now, the TSP if you see what are the practical applications. In manufacturing system, we can apply this travelling salesman problem, in plane routing system, telephone routing system, in network, in travelling salesman problem, as genome sequencing, job sequencing. Like this way various applications are there of this travelling salesman problem.

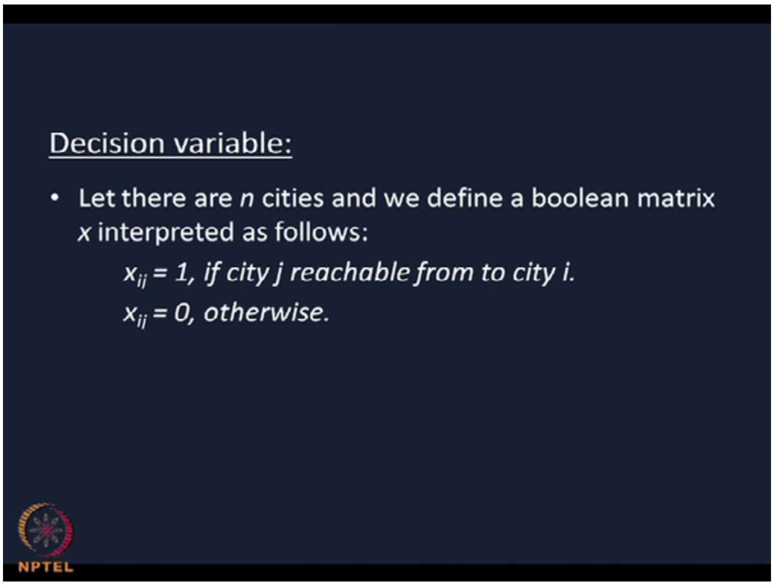
(Refer Slide Time: 05:15)



So, to summarize if you say we can tell that the problem can be formulated basically from the two prospective. One, I have to minimise the travelling time may be that is whenever you are travelling, the travelling time may be considered and I have to minimise the travelling time, so that I can visit all the cities only once, or you can pose the problem as minimize the travelling distance. So, the problem can be represented as a network where the nodes and arcs represent the cities and the distance between them respectively or in other sense, I want to say that I have withdrawn this one. I have drawn it something like a graph and from the graph, I have to find out basically the shortest path, so that each city has been visited exactly once, and there is a cycle means from where you have started, you have to come back from this.

So, basically I can solve it. By graphically I can represent the problem as a graph where I will have node, I will have arcs and obviously I can find out some mechanism over there.

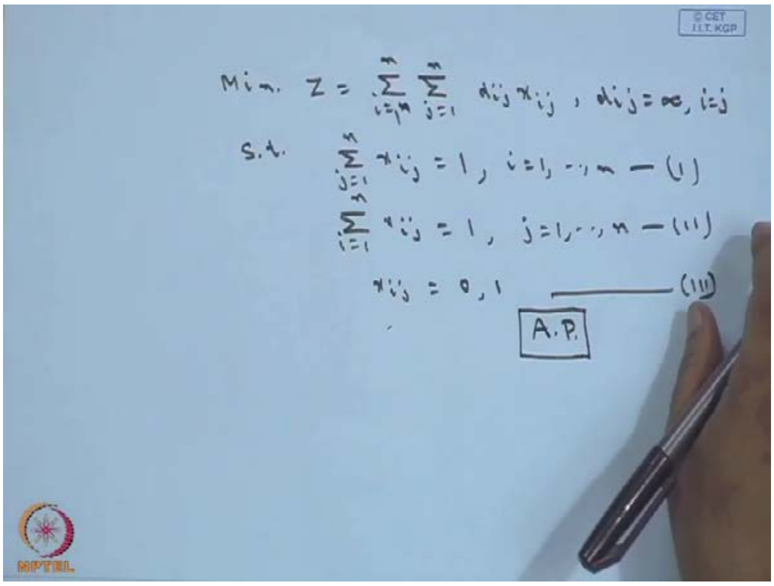
(Refer Slide Time: 06:28)



So, what are the decision variables? There are certain decision variables where if you see we are telling there are n cities and we have to define basically a Boolean matrix x, which is interpreted as follows as we have written small x i j equals 1, if city j is reachable from city i, that is if there is a connection between city i and city j. In that case, x i j equals 1, otherwise if there is no connection we say that it is not reachable or in other sense, I will form a matrix like this way where may be there will be c 1, c 2, c n cities are there. Like this way you will form the matrix.

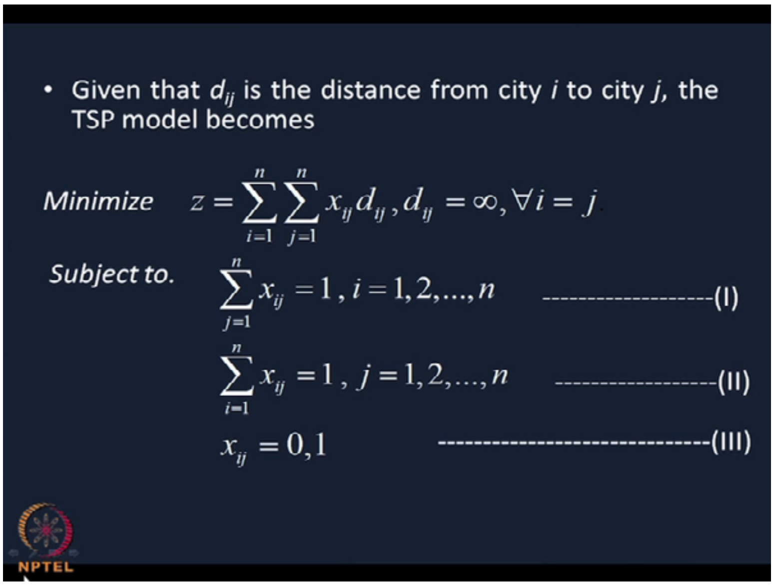
This I am from c 1 to c 1, c 2 to c 2, c n to c n. Nothing will come from c 1 to c 2. If there is space that is if we can move, then if we can reach, it will be 1, otherwise it will be 0. So, like this way some will be 1, some will be 0 and matrix 1 you can formulate Boolean matrix from here and suppose, here d i j is the distance from the city i to city j, if d i j is the distance from city i to your city j.

(Refer Slide Time: 08:03)



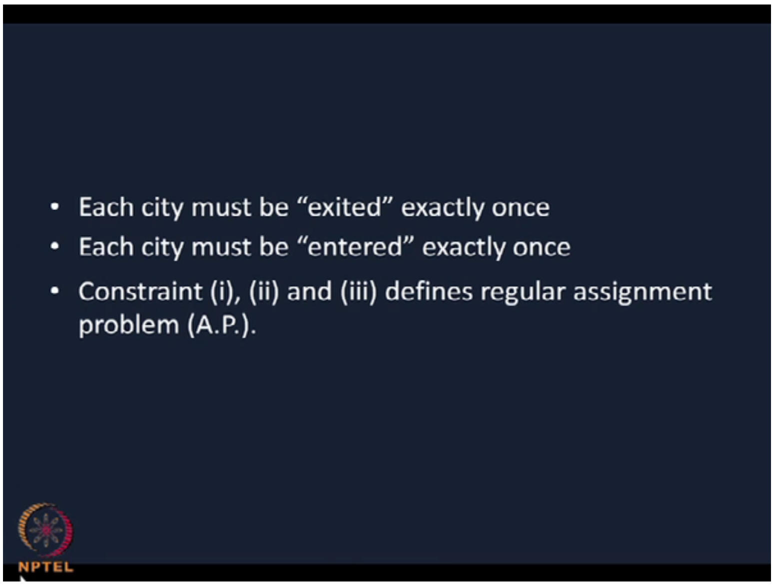
So, if d i j is the distance from city i to city j, we can formulate the problem as this minimise z equals summation over i equals 1 to n summation, j equals 1 to n d i j x i j and here subject to summation j equals 1 to n x i j equals 1 because they can take only one values. Here yours i is 1 to n. Similarly, summation over i equals 1 to n x i j. This equals 1. So, j equals 1 to n and obviously x i j, the addition variables can take only two values that is 0 or 1 which actually I have shown in the next slide.

(Refer Slide Time: 08:50)



If you see the minimise, this one summation over i equals 1 to n j equals 1 to n x i j d i j, where here d i j is the distance from city 1 to city n. If you note we have written d i j, this equals to d i j, this equals infinity. Whenever i equals to j, that is the distance, it is we are assuming for the same i and i that is 1. The same city, the distance will be remaining one. So, if you see the problem, the next one is summation over j equals 1 to n x i j equals 1 summation i equals 1 to n x i j equals to 1.

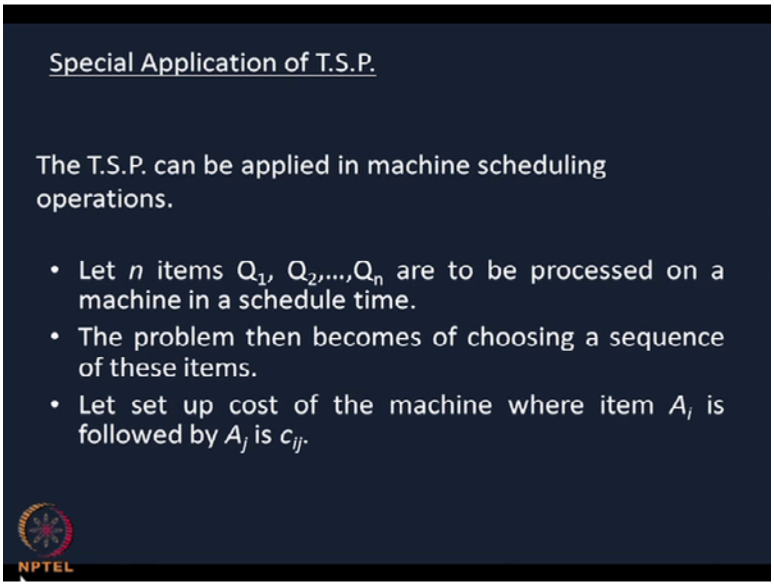
(Refer Slide Time: 09:44)



These conditions are coming due to the reason that here each city has to be exited only once, and each city must be entered exactly once. So, these are the two extra conditions. So, if you see the problem over here, now this problem if you see if you remember these constraints. If you see constraint 1, constraint 2 and this constraint 3 and subject to your minimization problem, this problem basically defines a regular assignment problem. If you see this is nothing, but the regular assignment problem or in other sense, I want to say that if we can formulate a travelling salesman problem like the problem minimise z equals summation over i equals 1 to n summation j equals 1 to n z i j c i j into d x i j into d i j subject to the constraints 1, 2, 3, we can solve it using modified assignment problems.

So, we can solve this problem using the modified assignment problem also, and in other way what I told you just now that is we can represent the problem as a graph and from the graph, we can find out the shortest path, so that with a condition it has to be exactly cyclic and each city has to be visited once, and we will come back to the city from where we have started.

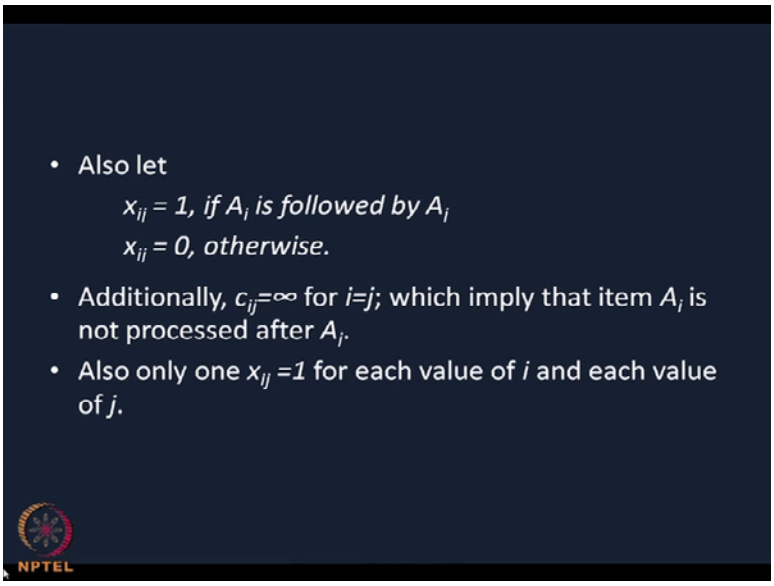
(Refer Slide Time: 11:12)



So, your special case is the transportation. Sorry, travelling salesman problem can be applied if you see in machine scheduling operations also. Why I am coming to the machine scheduling operation? Because whenever you are talking about the machine scheduling operation, it can be posed as an assignment problem and you can solve it using the assignment algorithm whatever we have done earlier.

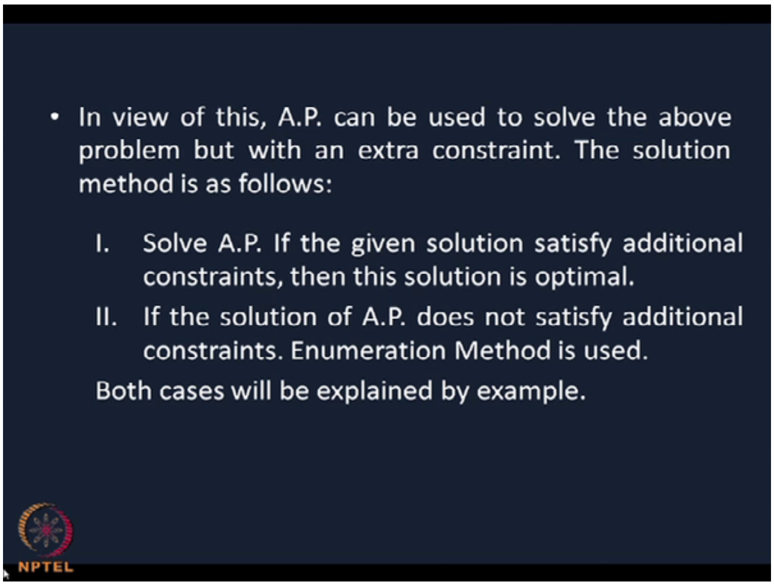
So, first we will discuss that part graphical also, but before solution I will start with the assignment. Then, we will see the graphical. Let us see how we can represent there are n items Q 1, Q 2, Q n are to be processed on a machine in a scheduled time period. So, n items are there we want to process on a machine. So, the problem for us is that we have to choose a sequence of these items that is which items should be processed first on the machine, which item should be processed second like this. So, we are assuming setup cost of the machine where item j is followed by item i is followed by item A j is c i j. So, basically now we are telling what is the cost? In terms of cost, we are telling that from item i A i to item A j the cost is c i j.

(Refer Slide Time: 12:35)



So, here you are using the cost and we are assuming that, obviously I told you little earlier that x i j, this equals 1 if A i j is followed. A i is followed by A j x i j equals 0. Otherwise, that means there is a connection between the items which is followed by one for that case x i j will be 1, otherwise x i j equals infinity. Additionally c i j will be infinity whenever i equals j, that is for x 1 1 1 2 3 3 like this or for the diagonals of the matrix always, it will be the cost will be taken as infinity, and the third point if you see only one x i j this is equals one for each value of i and each value of j. So, basically for one case only x i j will be one and for other case it will be equals to 0.

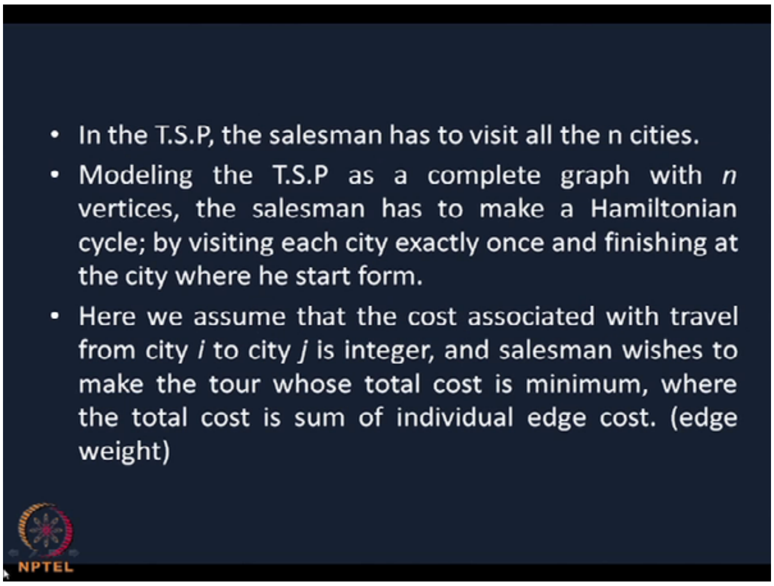
(Refer Slide Time: 13:38)



So, the machine scheduling problem can be used like this. So, in view of this, the assignment problem can be used to solve the above problem with an extra constraint. That we will see in the solution method. If you see the solution method, it will be something like this. I have to solve the assignment problem. We know how to solve an assignment problem. If the given solution satisfy the additional constraint of the transportation of the travelling salesman problem, then this solution is optimum. Now, additional constraint means I want to say from where I have started, I should come back to the original starting position or in other sense, it should be cyclic. So, this is the additional condition.

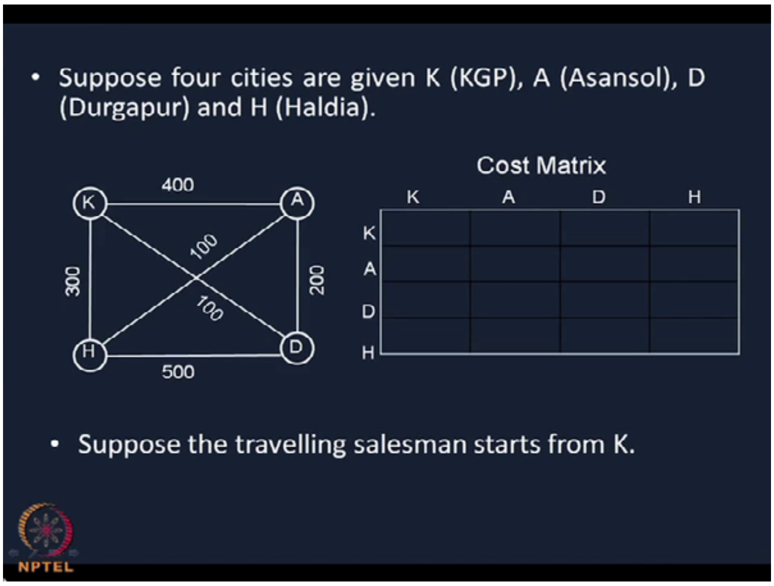
Now, if the solution of AP does not satisfy the additional constraint, then some enumeration method we will use. Obviously, we will explain these both cases with the examples.

(Refer Slide Time: 14:36)



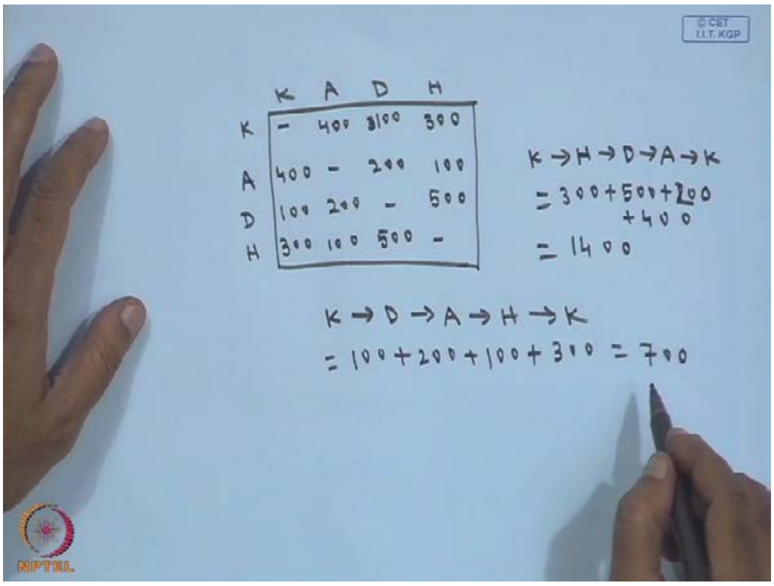
So, in the travelling salesman problem, the salesman obviously has to visit all the n cities. Now, as I told you modelling of the transport travelling salesman problem as a complete graph with n vertices, the salesman has to basically make one Hamiltonian cycle by visiting each city exactly once, and finishing at the city where he starts from or in other sense, as I was telling you if you see this earlier figure. In this figure, if you start from this c 1, you can visit in any way, but ultimately you have to come back to this c 1 only, and we have to form with this we are calling as the Hamiltonian cycle and each city you are visiting only once, and you are coming back to the earlier position and of course, we are assuming that the cost associated with travelling from city i to city j is an integer, and the salesman wishes to make the tour whose total cost is minimum. Obviously, if the cost associated your assuming cost moving cost from c 1 to c 2, it may be 10, it may be 7, it may be 15, something like this way. So, the cost is there. Obvious the problem is we have to minimise the cost.

(Refer Slide Time: 16:12)



If you see the total cost is minimum, where the total cost is the sum of individual edge costs. That is what you can say it as the edge weight. You will see this particular diagram. You have the 4 cities. We are telling k mean KGP, A-Asansol, D-Durgapur and H is Haldia. In the figure if you see there are routes from K to A and the associated cost is 400. Similarly, from K to H, you can move the associated cost is 300 like this.

(Refer Slide Time: 16:41)



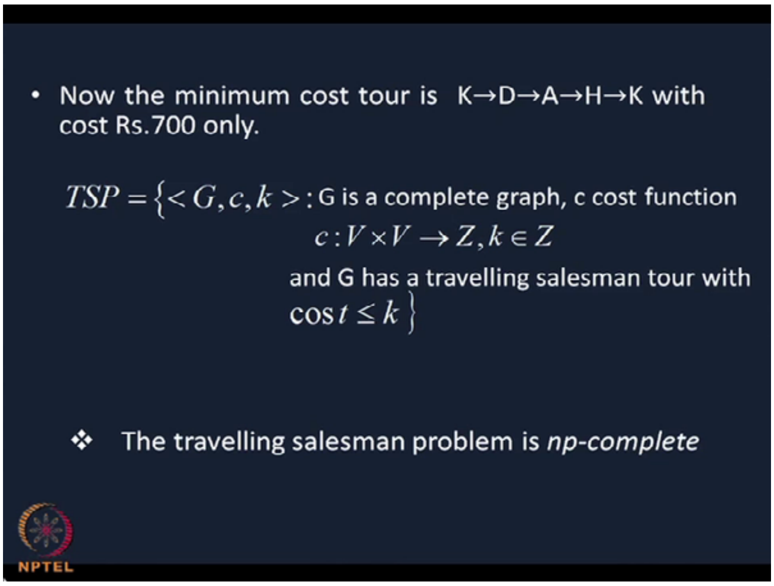
So, we can formulate one cost matrix. Immediately the cost matrix can be calculated like this. You are having K A D and H and here, it is also row wise K A D H. Now, from K to K, there is no associated cost. So, we are giving one blank space. If you see the figure, again from K to A, it is your 400, your K to D is 100 and K to H is 300. So, this you can write down now on this matrix. So, K to A is 400, K to D is 100 and K to H is 300. Similarly, we can fill up the other parts. I am filling up this one. This is 200, this is 100 and for D to K, it is 100. 200, it is blank and it is 500 for H, it is 300, it is 100-500 and this one is blank.

Suppose, the salesman wants to start from K. If you see the salesman wants to start from K, whenever they are starting wants to start from K and if you see the figure over there, just let us see the figure. We can move in any direction. I can say that I can move from K to H, then H to D, D to A, A to again K like this way. So, many paths I can avail and for each path, I have to find out the associated cost. If you see for K to H, it is 300, H to D it is 500, that is 500 plus 300 is 800. D to A is 200, that is 1000 and A to K is 400, that is 1400.

So, if you write down K to H, then H to D, D to A and A to K, the associated cost from here you can find out K to H is 300, H to D is 500. From this cost matrix, you can get D to A is 100, D to A is 200 and the last one is A to K is 400. So, the cost is 1400, but from here if you see, if I make the path as like this K to D, D to A, A to H, H to K. If I make, I can make another path like this K to D. It is moving from D to K to D. K to D is 100, D to A D is 200 and then, A to H A to H is 100 and H to K H to K H to K is 300.

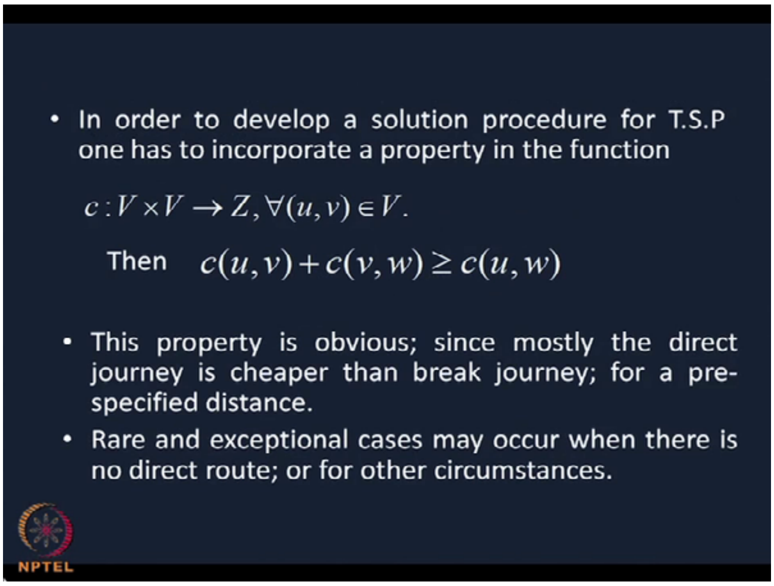
So, the total cost becomes 700 only. So, like this way I have to find out basically all possible paths and then, I have to check which path gives me the lowest cost and that will be the optimum solution over here.

(Refer Slide Time: 20:05)



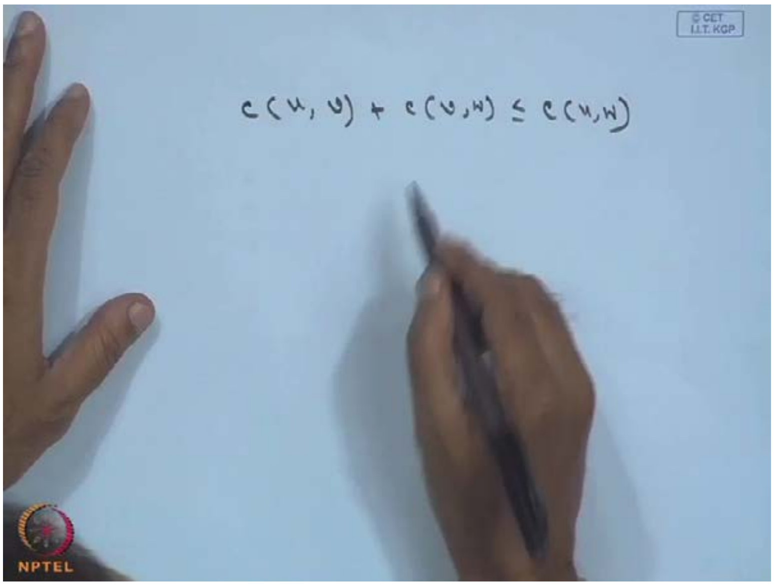
So, like this way you are having so many paths and from here, you will find out what is the minimum path. So, whenever n is very large, it is very difficult to find out all the cases. So, the travelling salesman problem if you see can be described as a triple set, set of three elements G, c and k where G is the complete graph. I am talking now in terms of graph only and here, a cost function is there which I am writing c, such that V to V and G has a travelling salesman tour whose cost is with cost less than equals t with cost is less than equals t and the travelling salesman problem of course as I told you, it is the np-complete problem. It can be solved in the polynomial time. So, we say that it is np-complete problem.

(Refer Slide Time: 21:00)



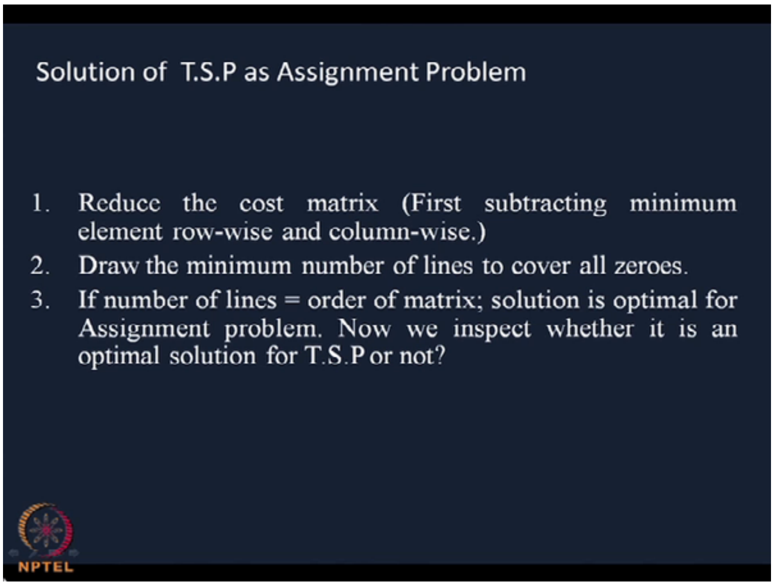
Now, in order to develop the solution procedure for the travelling salesman problem, one more condition has to be used that if you see we have written c of u, v plus c of v, w should be greater than equals c of u, w.

(Refer Slide Time: 21:18)



If I am writing here c of u, v plus c of v, w that is cost of this should be less than equals c of u, w or in other sense, whenever I want to move from u to w, directly you can move from u to w or for some other reason, you can use some other path that is from u to I will move to city V from city V. I will move to city w, but in that case, the cost of both should be less than equals original cost that is if you move directly, the cost always should be less than equals. It is this property of course we say that it is obvious since mostly the direct journey of course is cheaper than your big journeys. It is well known to us.

(Refer Slide Time: 22:11)



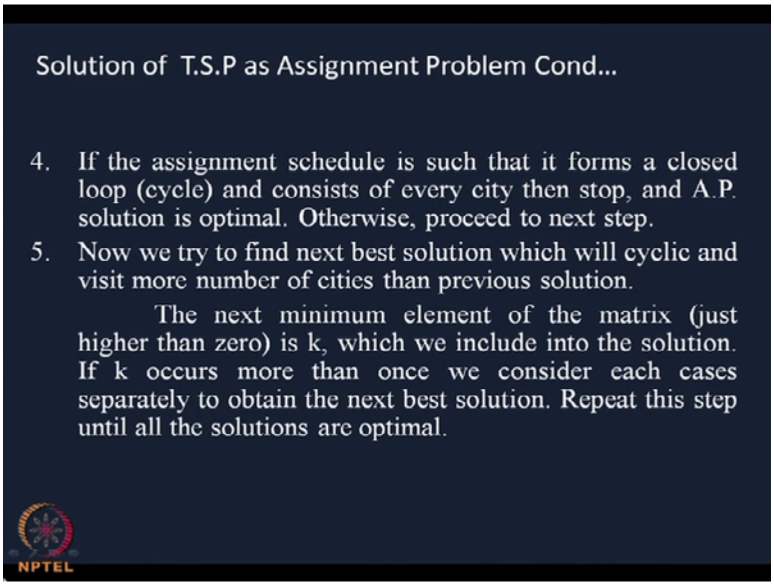
Now, the solution of the transportation problem if you see, what we have to do is first I am going through the solution as the transportation problem. Then, I will see the next one that is as I told you is not transportation problem, but the travelling salesman problem can be treated as assignment problem or the travelling salesman problem can be treated as a graph also.

So, first we will try to find out the solution of the travelling salesman problem as the assignment problem. So, what are the steps to be followed that first we are telling and then, we will discuss it with examples. First one if you see, I have to reduce the cost matrix. It is known to us for finding the solution of assignment problem. We use it that is first subtract the minimum elements row wise and column wise, that is find out the row minimum and subtract it from each row, then find the column minimum, subtract it from each column and then, I have to draw a minimum number of lines which will cover all the zeros of that matrix. So, we have formulated the cost matrix.

Now, if I draw the minimum number of lines to cover all zeros or not. Now, if number of lines is equals to order of the matrix, then the solution is optimal for the assignment problem. So, in that case, I have to first find out what is the optimal solution of the assignment problem. So, you have to note one thing. You will treat the problem. You have the cost matrix; treat it as an assignment problem. We solve the assignment problem on the same way you solve the assignment problem. So, you are finding out the number of lines which will cover all the zeros. If it happens that the number of lines is less than equals the order of the matrix since the matrix is a square matrix. So, if it is less than equals order of the matrix, then you know you have to find out the lowest element and from the lowest element, you have to subtract the uncovered lines, uncovered elements and wherever there is a intersection, you have to add that part. We have discussed earlier.

So, first find out the optimal solution of the assignment problem. This is the first step. The second step is once I have obtained the optimum solution of the assignment problem, now we have to find out or inspect whether it is the optimal solution of the travelling salesman problem. That is travelling salesman problem is having another criteria. The criteria is, it has to be cyclic that is from where you started, you have to visit two problems. One is you have to visit each city only once and from where you have stared, you should come back to the original position. So, that you have to check.

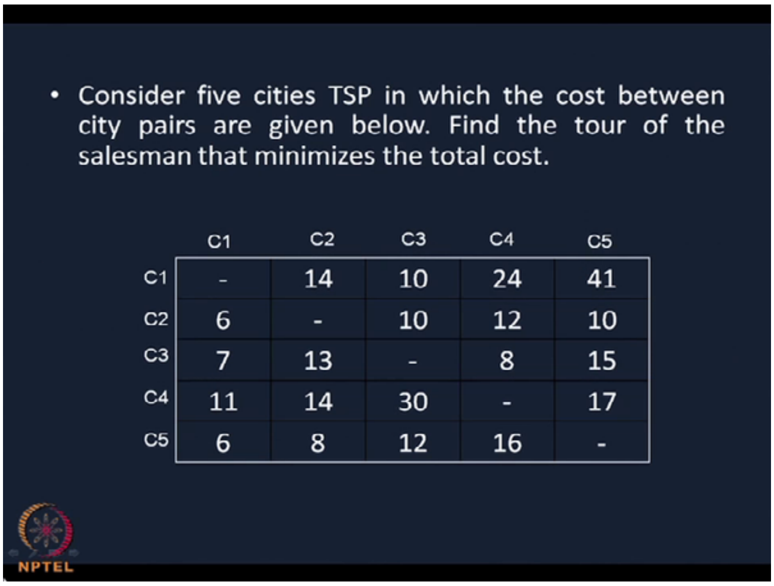
(Refer Slide Time: 24:58)



Now, if the assignment schedule is such that it forms a closed loop or cycle as I was telling you, and it consists of every city, then stop. That is it has covered every city and it is forming a closed loop, then you stop and this solution of the optimal problem is the solution of the travelling salesman problem also. Otherwise, you go to the next step. In the next step, what actually we have to do is, we will try to find out the next best solution which will be cyclic and this one. So, for this, what I have to do is we have to find out the next minimum element of the matrix, that is you have the cost matrix. I have to find out the minimum element which is greater than 0, say it is k which we will include in the solution. That element k will be one optimum solution and then, find out the assignment of the other elements also.

Now, if k occurs more than once, then for each case you have to repeat the process which I will show you with one example.

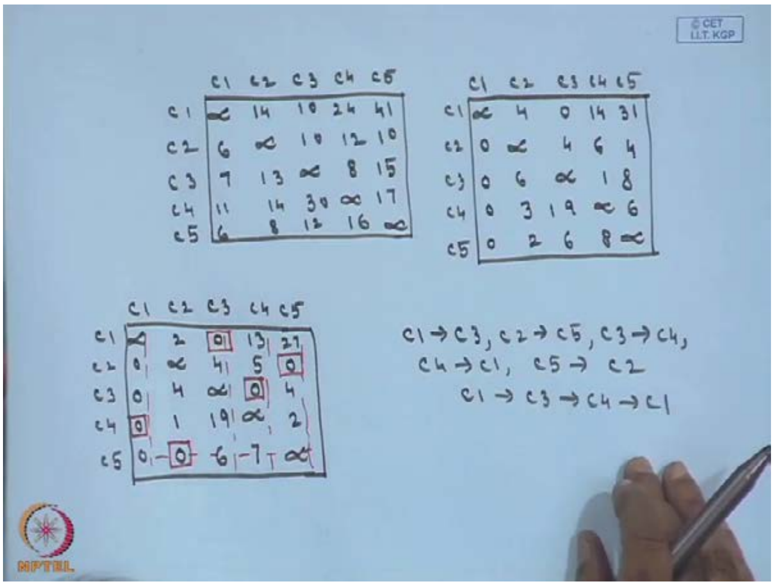
(Refer Slide Time: 26:11)



	C1	C2	C3	C4	C5
C1	-	14	10	24	41
C2	6	-	10	12	10
C3	7	13	-	8	15
C4	11	14	30	-	17
C5	6	8	12	16	-

So, let us go to the example. You consider there are 5 cities and the cost between them is pair wise. City pairs are given just like C1. C1 is blank and C1, C2-14, C3-10 like this way. So, at first I will rewrite the transportation. Sorry, the travelling salesman problem as an assignment problem with the cost for C1 to C1 as infinity as I told you c i j is equals to infinity whenever i equals j.

(Refer Slide Time: 26:48)



So, at first what you are getting? You are having the matrix. Over here, you are having C1 C2 C3 C4 and C5. Similarly, here also you are having C1 C2 C3 C4 and C5. I am just doing it initially. So, this C1 to C1 infinity C2 to C2 infinity C3 to C3 infinity C4 to C4 infinity C5 to C5 infinity and all other elements, whatever was there in the cost matrix, that is first row will be 14 10 24 41 like this. So, it is 14 10 24 41. The next one will come as I am copying it 6 10 12 and 10. The next one will be 7 13 infinity, then 8 15 11 14 30 infinity and 17. The last one will be 6 8 12 16 infinity.

So, this is the assignment problem which we got by making c i j as infinity as C1 C2. Like this way first thing is, subtract the minimum row element from each row. For the first row, minimum row element is 10. So, what I am telling you again is I am not explaining that part properly because already we have done this thing. C1 C2 C3 C4 and you are having C5. So, if minimum is 10 over here, so it becomes 4 0 14 and 31. So, this is infinity subtracting 10. In the next row if you see the minimum element is 6. So, subtract it. So, just like that way I am now writing it will be 4 6 and 4. For the next one, the minimum element is 7. So, it will be 0, 6, infinity, 1 and 8. For the next one, again 11 is the minimum element. So, it will become 0 13 19 infinity and 6. For the last one, 6 is the minimum element, so 0 2 6 8 and infinity.

So, by subtracting each row, you are getting it. Now, what I have to do? I have to subtract the minimum from each column. So, again you are having C1 C2 C3 C4 and C5.

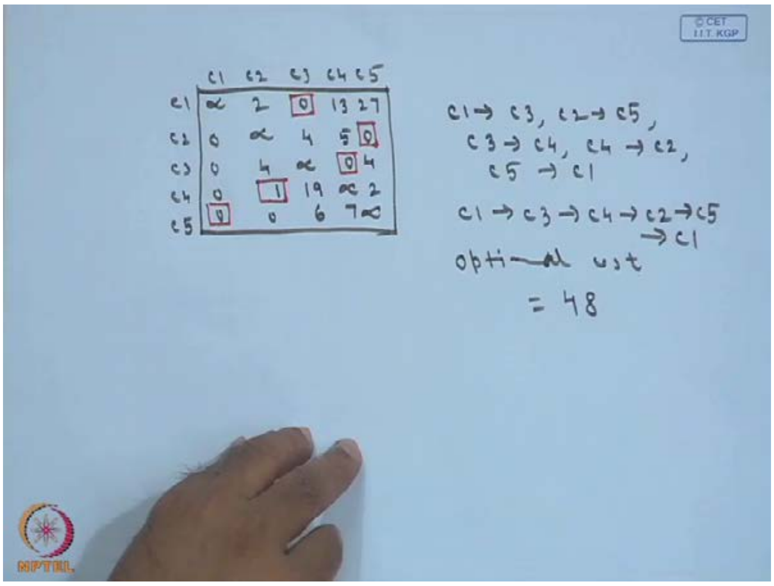
On these you are having C1 C2 C3 C4 and C5 and in the first column already 0 is there. So, there will be no change. In the second column, you see the minimum element is 2. So, it will be subtracted, so that you will get 2, infinity. You have 4 1 and 0. In the next one already you are having 0. So, there will be no change that is 0 4, infinity 19 and 6. In the next one, minimum element is 1. So, it will be 13 5 0 infinity and 7. In the last row, minimum element is four. So, it will become 27 0 4 2 and infinity.

So, if you see here now in this case, what we are doing? You are finding the minimum row and column. Now, try to make the number of rows, number of columns, sorry number of lines you draw, so that you can delete all the rows and all the columns. So, I can draw a line like this one. If I want I can do it 2, I can do 3, 4. This is not covered. So, I can make 5. So, if you see now you are making the number of lines which are covering all the elements. So, it has covered all the elements. Number of lines are 1 2 3 4 and 5 which equals the number of rows of the matrix.

So, now we can find out the assignment of this one. If you see I am not just writing here, you can assign like this because you know how to do this thing. One will be this, then you will get C1 to C5, C3 this one will come, C4 will be 0 and C5 is this. I am not discussing how I am allotting because already we have discussed this whenever we were doing the assignment problem.

So, basically what is the optimal assignment? Now, C1 to C3, then C2 to C5, after that C3 to C4, and then C4 to C1 and C5 to C2, so this is the optimal assignment, but if you see basically you are having C1 C3 C4, from C4 you are coming to C1 again, but you have not visited all the cities here. So, you have not visited C2 and C5. So, by this way whatever optimal assignment is there, here it is not satisfying your criteria of the travelling salesman problem because it is not visiting all the cities, and from where I am starting, I am not coming back to this. Therefore, this travelling salesman problem will not have the optimum solution. So, now what I have to do? We have to first allot what is the lowest element between the unticked elements. If you see the lowest element is one over here.

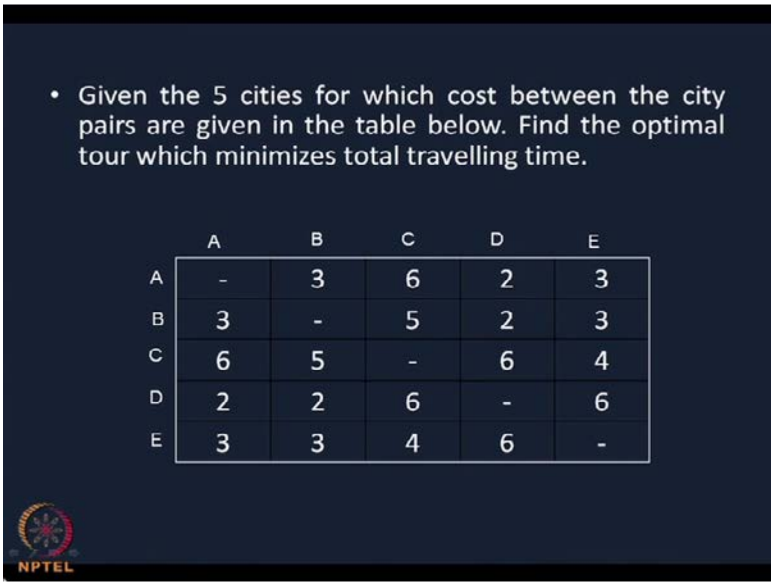
(Refer Slide Time: 33:10)



So, I am rewriting this particular table first. It will be clear to you. In that case, at first I am writing, you are having these elements C1 C2 C3 C4 and C5 and values are infinity 2 0 13 27 0 infinity 4 5 0 0 4 infinity 0 4 0 1 19 infinity 2 0 0 6 7 and infinity. In this if you see the lowest element between the unticked, the elements which were not covered by the lines, the lowest element is 1. So, what I will do according to the theory is, first allot this one, then you start allotting the others. So, that part I can do it. Obviously, like this way one will be this. In this case, it will come to here. For C3, it will come to here and for C5, it will come to this. So, once you are allotting these, others you follow the normal assignment problem whatever we have done earlier. So, now, if you see the allocation C1 to C3, C2 to C5, then C3 to C4, C4 to C2 and C5 to C1, so I can make the cycle something like this C1 to C3, C3 to C4, C4 to C2, C2 to C5 and C5 to C1.

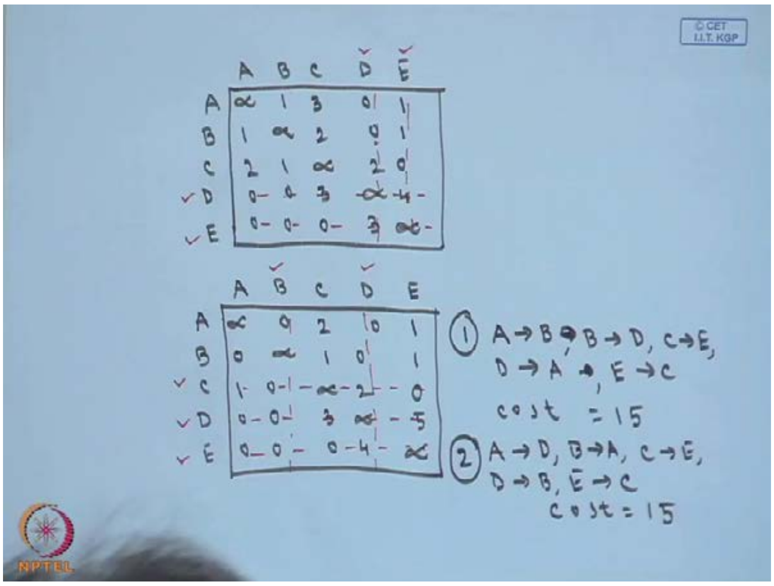
So, if you see this, now I have visited all the cities C1 C2 C3 C4 C5 and it is forming one cycle that is I started from C1 and I am stopping at C1 also. So, this is the optimal path then. What is the optimal cost? Your optimal cost in that case will be equal to the sum of moving from C1 to C3. On the original table if you see whatever is the cost we have written that is 14 10 like this, all this from here you can find out the cost and your cost will become the 48.

(Refer Slide Time: 35:34)



So, this is one way to do it. You see the next problem. In the next problem, the problem is similar, but I will just do it directly.

(Refer Slide Time: 35:46)



After doing this, after finding out the row minimum and the column minimum by subtracting, I am writing this one. You are having A B C D and E. You are having 5 cities A B C D and E. The calculation is similar, but I am doing it because of one reason. After subtracting the row minimum column, we will get this table, that is infinity 1 3 0 1 and then, 1 infinity 2 0 1 2 1 infinity 2 0 and then 0 0 3 infinity and 4 0 0 0 3 and infinity.

So, if you see after finding the row minimum, that is subtract minimum element from each row, subtract minimum element from each column, we will obtain this table from the given problem. Now, what are the minimum number of lines which will be required to cover all these? If you see I can draw this. If you wish I can draw this. So, I have covered all the zeros by how many lines? By total 1 2 3 and 4 lines, but what is the number of rows, that is number of rows equals to 5. So, therefore, this solution is not optimal solution for the assignment problem.

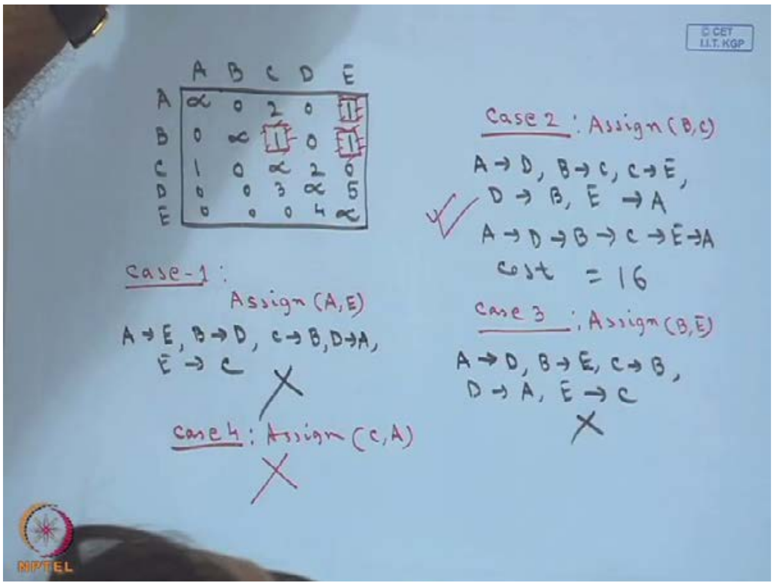
So, since this solution is not the assignment solution of the assignment problem from the remaining uncovered elements, I have to find out what is the minimum element. So, I will subtract it from all other elements and whenever I am having the intersection, that is this position, this position I will add the minimum element one. So, doing that job I will obtain this table A B C D and E. The columns will be infinity 0 2 0 and 1 and then, 0 infinity 1 0 1 1 0 infinity 2 and 0 0 0 3 infinity and 5 0 0 0 4 and infinity.

If you see over here, after subtracting from here using the assignment problem whatever we do there are three lines. So, I can draw C D and E that is something like this. I am just giving one faded on reading C D and E and on this B and D that is if you see, I have covered all the zeros. So, number of lines are 5 which is equal to number of rows. So, now I can assign, I am leaving it to you. Basically if you see this assignment problem is having two solutions. This assignment problem I can assign in two different ways. In both cases, optimal cost will remain same.

The first case, you will get A to B. Sorry A to B, B to D, C to E, D to A and E to C. You can just do it of your own. If you calculate the cost, the cost will become 15. If you see here, this solution assignment problem is not the solution of the THP because this is not forming a loop. You are starting from A or any point, but you are not going back to the other one. So, this is your solution 1, the solution 2. You can obtain like this A to D, B to A, C to E, then D to B, E to C, 1 2 3 4 5. Here also if you calculate the cost, your cost will become 15 only, but again this solution is also not satisfying the criteria of the TSP. So, this is solution 1, this is solution 2. So, for the solution 2, if you see now what happens for the solution 2?

Let me draw first and then, I will have to now check since both solution 1 and solution 2 are the solutions of the assignment problem, but not the solution of the TSP.

(Refer Slide Time: 41:05)



So, now I have to check both the cases separately. I will purposefully take the second case first. That will be easier. Let me draw the table first. A B C D E, this one if you draw it and here, I am having A B C D and E. The values are infinite 0 2 0 1 0 infinity 1 0 1 1 0 infinity 2 0 0 0 3 infinity 5 0 0 0 4 and infinity. So, if you see from this problem from here, what is the minimum element after deduction of the rows and elements or from the uncovered minimum elements? If you see 1 1 1 and 1, how many 1’s are there in this from here? It will be clear minimum element which is greater than 0. Here four 1’s are there. So, basically I have to check all the 4 cases. How you will check? You will check something like this. In case 1, what I will do is you assign first A E that is A to E. That means, you will assign first these A to E and then, assign all others. So, if you assign like this, you will obtain the assignment as these A to E, B to D. You can check it of your own C to B and D to A and E to C.

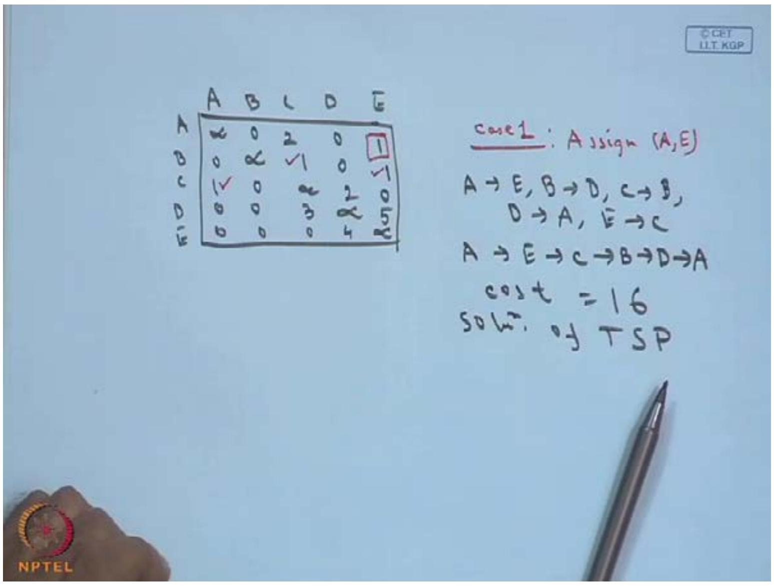
So, for the case 1, first you assign A to E and then, assign the others following the same rule whatever we have done earlier. So, your assignment will be this one. If you see, again this assignment is not satisfying the criteria of the transportation. Sorry of the travelling salesman problem. So, therefore this case, this is not the optimum solution. Now, case 2, this one we considered. Now, consider the case 2 over here. In case 2, what you do you is assign the other one that is say this one. Now, this is not there. So, now, you assign. So, you assign B C that is you assign first this one and then, assign other rows and elements.

So, if you do like this, your assignment will be A to D, B to C, C to E, D to B, E to A. Here, you see you have covered all the elements, all the cities A to D, then D to B. You can write B to C, C to E and again, you are coming back to A and cost is 16. So, therefore in one case, the solution of your travelling salesman problem is follow this path A to D, B C E and the cost is 6. So, if you see the cost is cost of the optimum TSP is not equals to cost of optimum assignment problem. If you remember in the earlier case, the cost was 15 and here it is 16. So, this is case 2.

Now, you consider again the case 3 that is this one you have taken. Now, forget about this. You take case 3. In case 3, what will happen is you assign now the other one, that is assign your B E that is you assign. Now, if you assign B E, you will get the optimum assignment as A to D, B to E, C to B, D to A and E to C. Again, this path is not cyclic, although it is visiting all the cities. Therefore, the solution is not optimum. So, now the other minimum one is there and that is this one. So, your case 4, you have to take now in case 4, you assign your C A.

In assigning C A, what you will obtain? In this case also, if you see the assignment for C A in the same way that is C to A and all others, you will find that you are not getting the optimum solution. So, for one solution of AP, you have got 4 different cases because minimum value one was occurring in 4 cases. You are considering case 1, case 2, case 3 and case 4 separately from here. Only for case 2, I got the optimum solution for the transportation problem. So, this was related to case 2. Now, we will see related to case 1 what happens? On the same way related to case 1, what happens is just once I will draw the table, that is you are having A B C D and E. Here also, you are having A B C D and E.

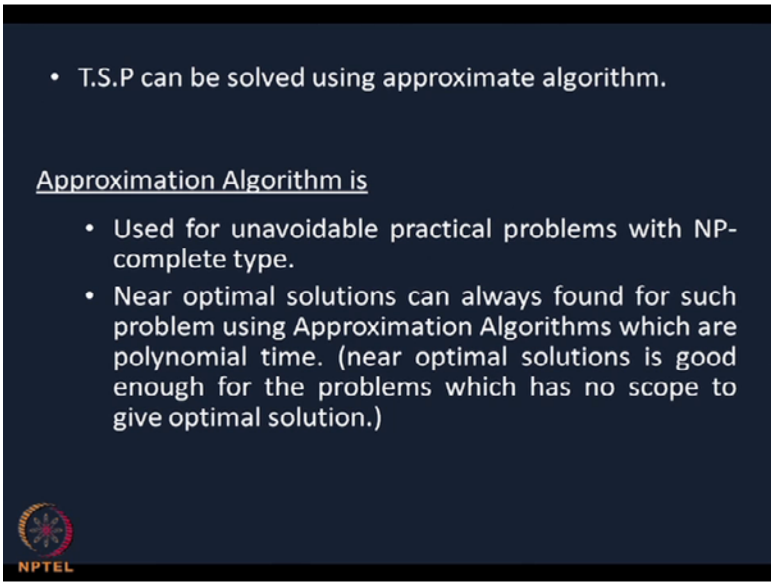
(Refer Slide Time: 46:17)



So, you are writing all the elements infinite 0 2 0 and 1 0 infinite 1 0 and 1 1 0 infinite 2 and 0 0 0 3 infinite 5 and then, 0 0 0 4 and infinite. So, in this case, again four different cases will be occurring because minimum element is 1 and 1 is occurring in four different places. So, just I have done if for the other case, I will just do case 1. In case 1, what you are doing is assign A E. If you assign A E, that is if you assign this one in that case, your solution will be A to E, then B D, then C to B, D to A and E to C. So, from here if you see A to E, you are going to C, C to you are going to B, B to D, D to you are going again coming back to A.

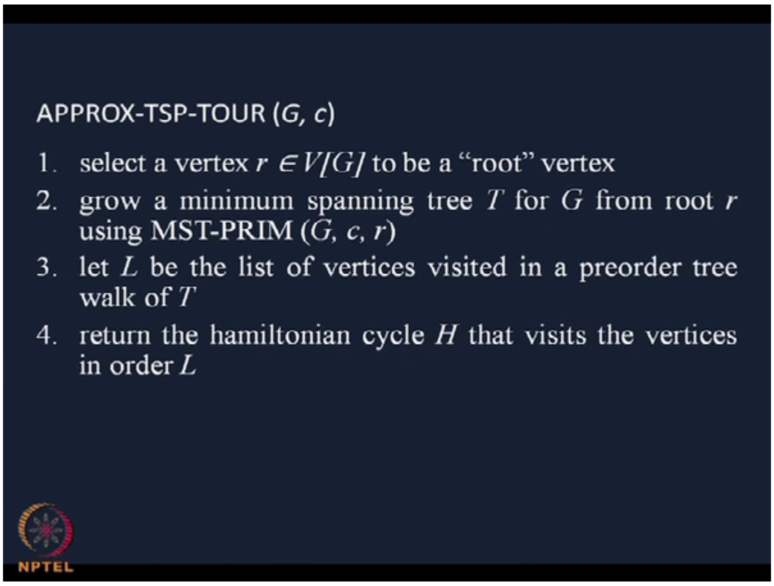
So, therefore, I have visited all the cities and it is cyclic. I am coming back to A only if you calculate the cost. The cost is 16. So, therefore, this is another solution of the travelling salesman problem. So, just like four cases, I derived for the earlier case here also I did it for A E. Then, you will take this one. You have to take, you will take this one and then, this one, this one and you may see the solution of this one. One thing may be noted as I told you from here that one solution of the TSP I obtained as this path, other solution I am obtaining as this path. So, solution of travelling salesman problem is not unique that we have to understand. So, this we did it using the assignment problem or converting the travelling salesman problem as assignment problem and then, you find out the cities or the shortest route where the cost will be minimum, that is other method.

(Refer Slide Time: 48:57)



As I was telling, there is some approximate method also which we call approximation algorithm. By this approximation algorithm, you will obtain the shortest path, the cyclic part, but the solution will be approximate, that is the cost or the time may not be optimal. We can obtain the near optimum solution.

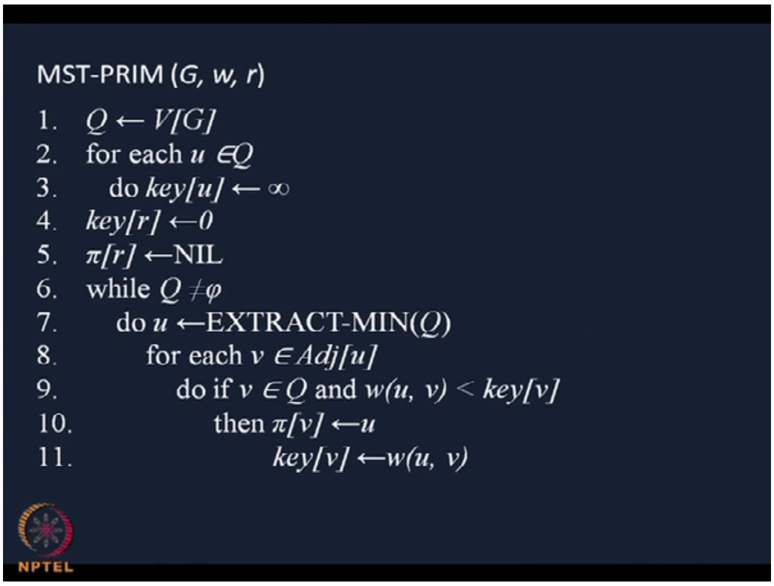
(Refer Slide Time: 49:17)



In the approximate TSP tour, we are finding out or we are finding, we are selecting a vertex r which will be a root vertex as we have told for the travelling salesman problem. It is not essential from where I will start and from where I will stop. That is not essential.

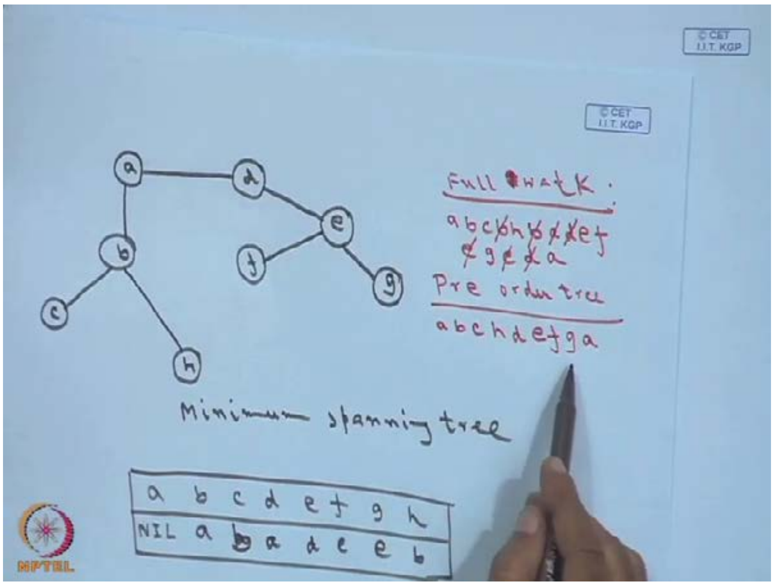
I can start from anywhere and I can go to anywhere. So, for the travelling salesman problem, at first you are selecting a vertex and then, I have to find out a minimum spanning tree. Actually for shortage of time, I may not be able to complete this minimum spanning tree, but what is minimum spanning tree. At least by one example I will show you and then, this is the step number 2, minimum spanning 3 and then, you find out one is the least of vertices visited in preorder tree walk of T. What is preorder tree walk? Then, again I will tell and then, we will return the Hamiltonian cycle that is the cycle which we got.

(Refer Slide Time: 50:16)



This is basically your tree. This is the algorithm for finding out the minimum spanning tree. You just go through; I am not explaining this one.

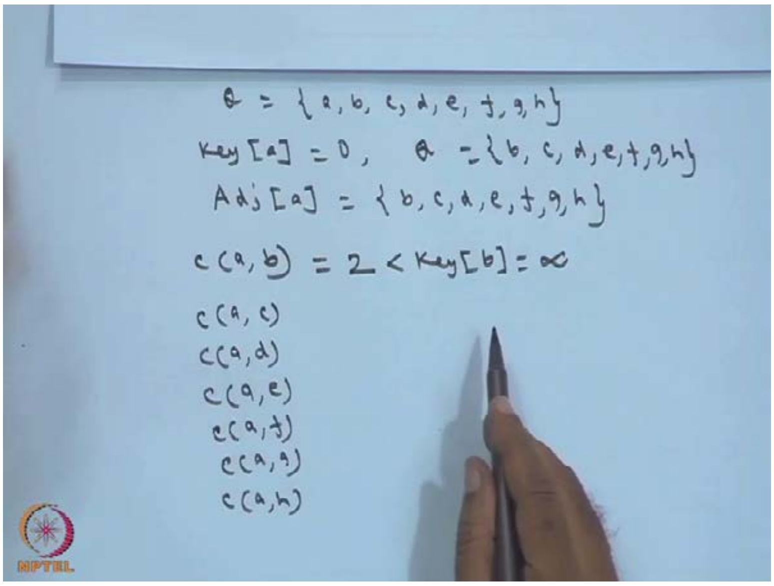
(Refer Slide Time: 50:43)



So, what I will do now? I will just let us take this example. First, let us take the example. Suppose you are having city something like this a, here you are having say city d, you are having b, you are having f, somewhere here it is c, it is h, it is say somewhere e, and it is g. Some cities are located something like this. Now, we want to find out what would be the shortest route of this one. Let us just find out the minimum spanning tree. First for just one minute, this is unloading. If you see, it is of course not fully, it is if I just increase it to 16 say this one is the different values. Whatever we have got for this one, different value we are getting. From a to a, it is 0, a to b 2 and like this way I am drawing what are the cost matrix that is for moving from a to b, the cost is 2. Moving from b to c, it is 3, something like this way. So, I am not explaining this one. So, first you will make the cost matrix over here and once I am making the cost matrix, after that the cost is calculated like this, the Euclidian distance between a and d. If you see cost of a and e is root over 3 square plus 1 square, that is equals 6.10 here.

What I have to do? First, I have to find out a key. What is key? If you see I am making this is little bit larger, so that you can see it. Initially you are having how many nodes? A b c d e f g h. Initial cost for all of them will be infinity initially. Now, I will start my route is say a. Here once my route is a, in that case I will make the corresponding key value corresponding to this as 0. From here it is corresponding to this. I will make it as 0 and all others will be infinity. If you see here I have to initialize the values also of phi and for the values will be the nil. Initially each and everything will be nil. You have this figure. I am not going to this figure. At first this one, I am just writing this. Initially you are having the Q is a b c d e f g and h.

(Refer Slide Time: 53:48)



Now, what is the minimum of this a b c d e f g h from the mean value? Obviously, the value of a is 0 and all others are infinity. Your key of a, this is equal to 0 and since, Q of a, is 0, all others are infinity. Now, you remove this a from here, so that your Q becomes b c d e f g and h. So, your Q is this one. Once I got this, what is the adjacent of a? Now, adjacent of a will be all this b c d e f g and h. Now, you calculate the cost between a b. Cost of a b if you see, it is equal to from the figure if it is say 2, in that case, it is less than the cost of key of b because initially the b is infinity.

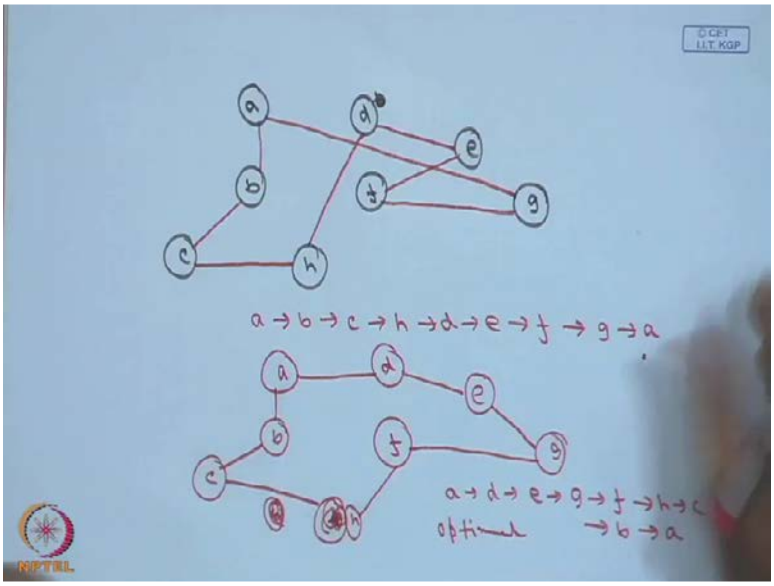
So, there would be one value between, there will be a path which comes from a to b. So, basically a path will come from a to b like this way. Now, I have to calculate paths from, now I have to calculate after this a c cost of a d cost of a e and then, cost of a f and cost of a g and cost of a h. All this I have to calculate just like we have calculated from the given matrix and from here, you can see that once I am calculating on this, you will see that from c to a, from a where I can go if you see from a.

So, you are moving like this way. You are calculating the cost and ultimately, you will obtain a, the minimum spanning tree something like this. Due to shortage of time, I could not explain the entire thing, but this is ultimately your, this will be your minimum spanning tree. Why we are saying minimum spanning tree? Because you will find that there will be a minimum spanning key. What we are telling is you will get a table something like this. You are having here a b c d e f g and h, for a predecessor is nil, for b it is a. You will find for d, it is a. For b it is a, for c it will be b, for d it is a, for e it is d, for f it is e, for g it is e and for h it is b. So, from here I can obtain the minimum spanning tree.

So, once I have obtained the minimum spanning tree, then what I have to do is I have to make a full walk of this. I have to make a full walk of the tree. What do I mean by full walk of the tree? The full walk of the tree means it will be a, then b and from b, it is moving to c, c to you are coming back to b, b to it is h, h to you are coming back to b, b to you are going back to a, a you are going to d, d to you are going to e, e to you are going to f, f to you are coming back to e, e to you are going to g, g to you are going back to e, e to you are going back to d, d to you are going back to a.

So, this we are calling as full walk. From this full walk we are finding out the preorder tree. What is the preorder tree? Now, your preorder tree will be from here. You go in this order. Whatever has appeared, you write only those once, that is a b c b has appeared here. So, you will cut it after that it is h b. Again a has appeared, after that h d will come, after d, this d will be not come and then, e will come. After e, f will come and after f, it will be this is not coming. G will come and all these will be cut and this.

(Refer Slide Time: 59:07)



So, this is the basically a b c h d e f g a or in other sense, if you draw it like this using this preorder, this one you will obtain. You are having here a this is your say, d, this is the b, this is your f, somewhere here, your having c, here you are having h, you are having e and you are having f. F is here somewhere you are having g. So, once I have drawn this, now you follow this path a b c that is a from b, b to c from c to h, h to d. So, from c to h, you are moving from d to e from e to f, e to f to you are moving to g and from this preorder g to a.

So, this is the optimum path is therefore you are getting a to b, b to c, c to h, h to d, d to e, e to f, f to g and g to a. So, this is the approximate tool. If you see I will tell this is the approximate tool. So, first you are finding out the minimum spanning tree and after finding out the minimum spanning tree, you have to find out the preorder tree, and from the preorder tree, you can decide what would be the cycle path. Of course, if you see this one, just a b d f here you are having say c, somewhere here you are having h, here you are having e, you are having g. If I move something like this a to b, b to e, e to f, f to g, g to h, h to c, c to b and b to a, that is say a d e g f h c b and a. If you find out the cost of this one, you will find this is the optimal one. This I have drawn just by observation, but by our algorithm, you will obtain this.

So, therefore, we are saying that you will obtain the shortest path which will visit each city only once, and I will come back to the route that is original city, but the cost will be near optimum, but not optimum. So, please note this thing that the travelling salesman problem can be solved by two ways. One is I can form it as an assignment problem and find the solution. Other one is I can form a graph. From the graph, I can find out what is the minimum spanning tree. From the minimum spanning tree, I can find out what is the preorder and from the preorder, I can fix what would be the shortest path which will be the nearest optimal path. Please note that near optimal, it is not the optimal path. That is the reason we call it as the approximate algorithm.

Thanking you very much.