Advanced Operations Research

Prof. G. Srinivasan

Dept of Management Studies

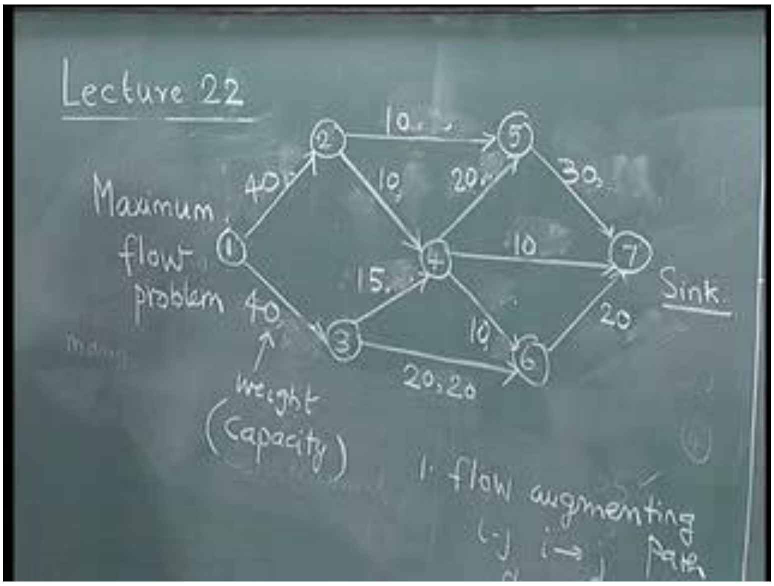
Indian Institute of Technology, Madras

Lecture - 22

Maximum Flow Problem

In this lecture, we continue our discussion on the maximum flow problem.

(Refer Slide Time: 00:20)



In the previous lecture, we introduced the maximum flow problem using this example. We also tried to evaluate the maximum flow possible by considering the paths in a certain order. When we did it with this particular order, we got a flow of 50 units. We first considered 1 to 2, 2 to 4, 4 to 5, 5 to 7 with flow equal to 10. Then, we consider the other paths in this order and we obtained a flow of 50 units. We also worked out the same problem by considering the paths in a different order. Now, with these 4 paths, we were able to get 55 units.

We also defined what is meant by a flow augmenting path. In fact, if we had considered the path 1 to 2, 2 to 4, 4 to 5, 5 to 7; which means we have a path from 1 to 2, 2 to 4, 4 to 5, 5 to 7, we realize that each of these arcs 1 to 2, 2 to 4, 4 to 5 and 5 to 7 have some spare capacity. 1 to 2 has a capacity of 40, 2 to 4 has 10, 4 to 5 has 20, and 5 to 7 has 30. Out of these capacities, the minimum being 10, that is, the maximum flow that can be augmented into the flow augmenting path. This flow augmenting path 1 to 2, 2 to 4, 4 to 5, 5 to 7 gave us flow equal to 10. We also defined a flow augmenting path this way, as a path that connects the source to sink through a set of forward arcs 1 to 2, 2 to 4, 4 to 5, 5 to 7. When I say a forward arc, I mean, when I say 1 to 2, there is an arc from 1 to 2, there is an arc from 2 to 4, and so on. If this path is made up of completely forward arcs and each forward arc has a residual capacity, which is strictly positive, then the minimum of those capacities will now become the flow. Now, such a path is a flow augmenting path.

We also observed that all these were flow augmenting paths, because they were able to augment the flow by a strictly positive quantity. In the second way, in which we did it, we considered flow augmenting paths which gave us 55 units of flow. We also observed from these 2 instances that this can further be improved to get the additional 5. Let us again work out this 50 on the network and then, show how we have to redefine the flow augmenting path, so that we are able to increase the flow from 50 to 55.

When we evaluated this 50, based on this definition of the flow augmenting path, we realise that there are no more flow augmenting paths. Therefore, we have to redefine the definition of the flow augmenting path, so that we are able to increase the flow from this to this. Let us go back to this example to see how these flows are captured on the network. 1 2 4 5 7 would be 1 to 2, 2 to 4, 4 to 5, 5 to 7, 40, 10, 20 and 30. Minimum being 10, we have indicated this as flow equal to 10. This is updated as (40, 10), (10,10), (20, 10) and (30, 10) where the second part gives the actual flow, while the first part gives the capacity of the arc.

We can consider the path 1 2 5 7, 1 to 2, 2 to 5, 5 to 7, here there is a capacity of 30 because the total capacity is 40. There is already a flow of 10 so remaining or residual capacity is 30. 2 to 5 has 10, 5 to 7 has 20, so the minimum among this 30, 10, and 20 is 10; the flow is now augmented by 10. This 10, becomes 20, this 10 is consumed and this 10 also becomes 20. We now consider the third path, 1 to 3, 3 to 4, 4 to 5, and 5 to 7. 1 to 3, 3 to 4, 4 to 5, 5 to 7, here, there is capacity of 40. There is no flow so all the 40 is available, all the 15 is available, here, 10 units are available, 20 minus 10 equal to 10, 30 minus 20 equal to 10. The flow can be augmented by a further 10 which is here.

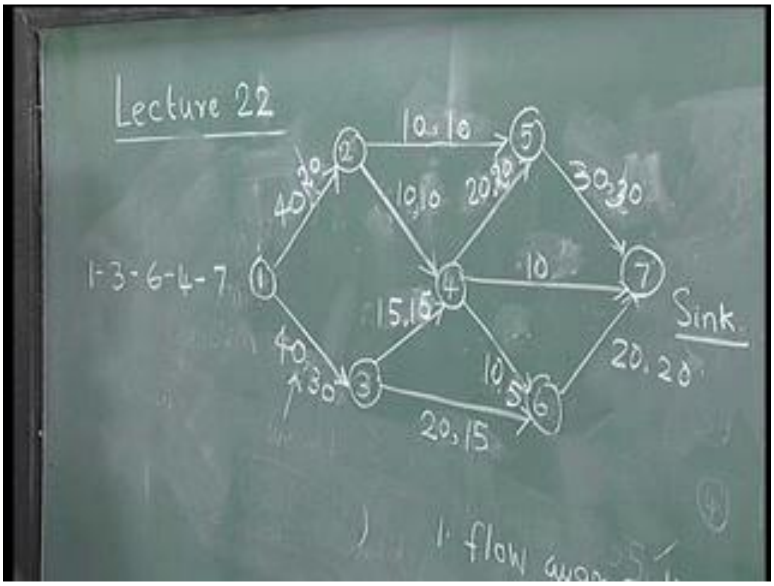
We update this (40, 10), (15, 10), (20, 20) and this becomes 30. Now, we consider the next path, 1 3 4 6 7. Here, there is a capacity of 30; here, there is capacity of 5, here, there is 10 and here there is 20. So, we got 30, 5, 10 and 20. The smallest number is 5, so the flow gets augmented by another 5. We update 1 to 3 is 15, 3 to 4 is also 15, 4 to 6 is 5 and 6 to 7 is 5. We then look at the path 1 3 6 7. Here, 25 is possible, 40 minus 15. Thus 1 3 6 7, 25 is possible here, 20 is possible. We do not have any flow on the arc 3, 6; so 20 is possible. Then 6, 7 15 is possible, so 25, 20 and 15. We take the minimum of them which is 15. Therefore, this gets augmented to 30, this becomes 15 and this becomes 20, this becomes 20.

Now, we have a total flow of 50. Let us also check whether we can have any more flow augmenting path. For the path, 1 2 5 7 no more flow is possible, because this is exhausted. 1 2 4 5 7, this is exhausted as well as this. Then 1 2 4 7 this is exhausted, 1 2 4 6 7 this is exhausted. With 1, 3, 1 3 4 5 7 this is exhausted, 1 3 4 7 this is exhausted, 1 3 4 6 7 this is exhausted and 1 3 6 7 this is exhausted. By this definition of flow augmenting path, we do not have any more flow augmenting paths.

Remember once again, the definition of this flow augmenting path. Consider a path from i to j, which is made up entirely of forward arcs and evaluate the excess flow that the arc, each of these arcs can handle. The minimum of them is the flow that we can augment. If we consider all these flows, we are unable to get a flow augmenting path and therefore we stop at 50. How do we increase this 50 to 55. As I said, we have to now redefine the flow augmenting path idea, so that we are able to increase the flow further in this network. How do we do that?

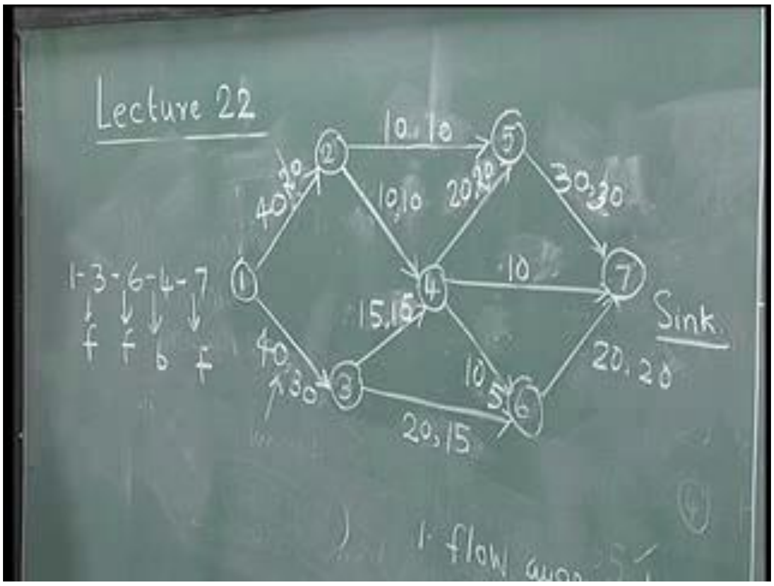
Let us go back and consider a path like this. With these kinds of arrangements, with this capacity and with this flow, let us consider a path like this. Let us consider 1 3 6 4 7. We are looking at 1 to 3, 3 to 6, 6 to 4 and 4 to 7.

(Refer Slide Time: 09:40)



1 to 3 is a forward arc, there is an arc from 1 to 3; 3 to 6 is a forward arc, there is an arc from 3 to 6. 6 to 4, there is no arc in this network 6 to 4, but we are still considering it because there is an arc from 4 to 6. Then, 4 to 7 is a forward arc with an arc going from 4 to 7. Let us consider this once again. Now, 1 3 is a forward arc, because there is an arc from 1 to 3, 3 6 is a forward arc, 6 to 4 is a backward arc and 4 to 7 is a forward arc.

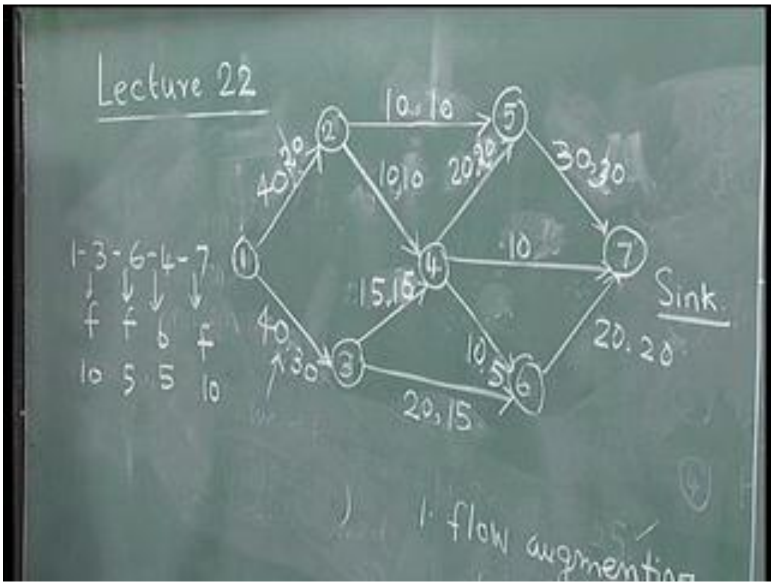
(Refer Slide Time: 10:25)



Here 6 to 4 becomes a backward arc because there is a forward arc from 4 to 6. Wherever there is a forward arc from i to j, we may consider j to i as a backward arc under an important condition. In this, if we observe, if we look at 1 to 3, 1 to 3 is a forward arc with capacity 40, flow 30. So, there is an excess 10 that it can absorb. Now, what is the increase that is possible here? It is 40 minus 30 which is 10. 3 to 6 is again a forward arc; capacity is 20, flow is 15. So, this can absorb 5 which is the difference between 20 and 15 which means the flow can be increased by 5 units here.

If you look at this backward arc 6 to 4, there is a forward arc 4 to 6 and more importantly there is a flow in the direction 4 to 6. This flow 5 can actually be reduced or diverted. Therefore, we consider the possibility of considering only this flow, if it is a backward arc so that this flow can be diverted and the total flow can be increased. We go back to 4 7, 4 7 is a forward arc; there is capacity of 10 flow of 0. So, there is a 10 that is possible. Look at the minimum of these numbers.

(Refer Slide Time: 12:02)

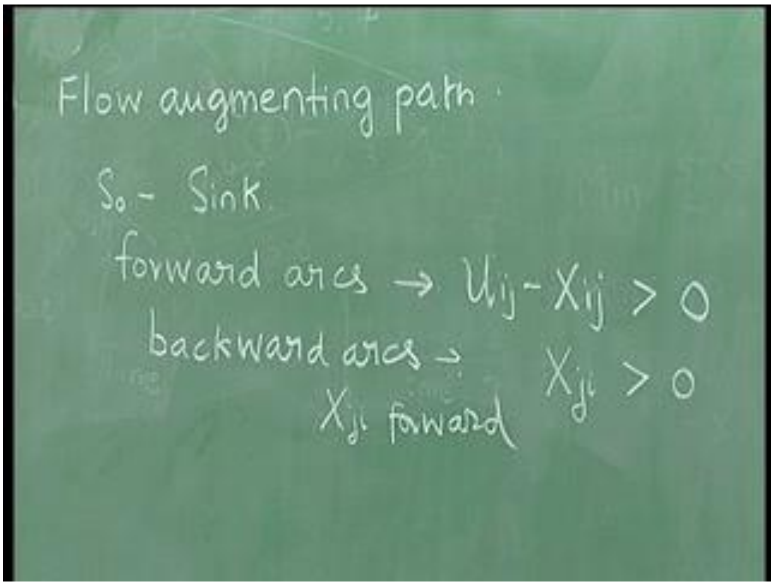


Minimum is 5, this 5 or this 5 is either the minimum of the additional possible flows in the forward arc and the actual flows in the backward arc. Whenever we consider a backward arc into a flow augmenting path, we have to ensure two things. One is there is a forward arc in that direction. When we consider 6 4, we know that there is an arc 4 6 in the network and more importantly this has a strictly positive flow, which can be diverted. By considering this backward arc, this flow augmenting path, which comprises of forward arcs and backward arcs, we can augment the flow by 5. So, we can go back and add 1 3 6 4 7, which will give us 5 and we get that 55.

The flows have to be updated. This is a forward arc, so 30 will become 35. 3 to 6 is a forward arc, 15 will become 20. 6 to 4 is a backward arc. So, 10 will become 0. Now, 4 to 7 is a forward arc so it gets increased by 5. Wherever there is a forward arc, there is a flow, there is an increase and wherever there is a backward arc, there is a decrease.

It is absolutely necessary that when backward arcs are considered, there should be positive flow. By diverting that positive flow, we will be able to increase the total flow in the network. We can go back and show that this kind of a flow representation actually corresponds to this solution and the maximum flow that we got is 55. At the end, what we have to do is, we have to now start defining the flow augmenting path in two ways.

(Refer Slide Time: 14:05)



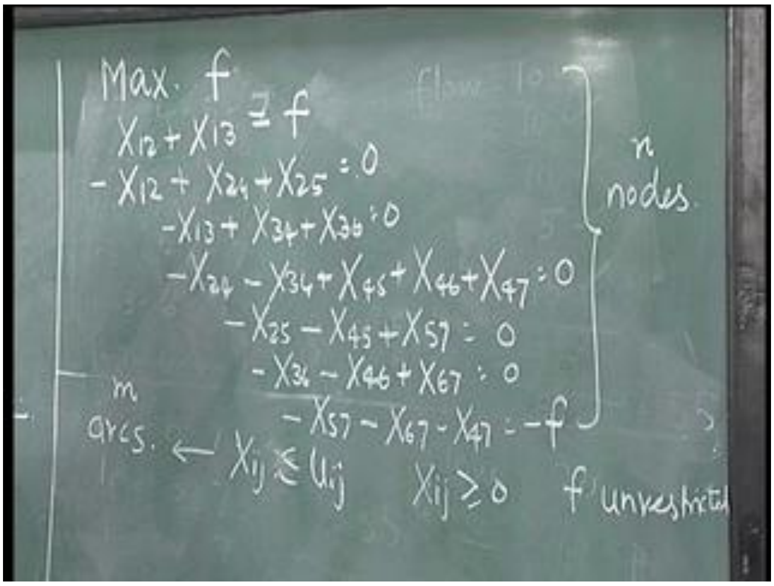
Flow augmenting path is a path from source to sink, made up of forward arcs and backward arcs, such that for every forward arc in the path, the present X_{ij}. C_{ij} minus X_{ij}, let us call the capacity as C_{ij}, or we may call the capacity as U_{ij}, U_{ij} minus X_{ij} is strictly greater than 0. If it is a backward arc, then there is an X_{ji} in the forward direction. Only then, it becomes a backward arc and the flow X_{ji} is strictly greater than zero. When we do this, then we take the minimum value of the U_{ij} minus X_{ij}, if it is a forward arc and the flow X_{ji}. If it is a backward arc, then take the minimum of these and all of these being strictly positive. The minimum will be a strictly positive quantity and the flow gets increased by that quantity.

Correspondingly, if it is a forward arc, then this increases. If it is a backward arc, then this decreases. In this case, we earlier had 5 and now it has decreased to zero. This essentially means we are actually taking away this flow and redistributing it here, so that this flow goes into this. Then, there is an additional flow that can come into this network because of the redistribution that we have actually done. One has to understand that we are not deleting any flow. We are only redirecting or redistributing this flow, whenever a backward arc comes into the flow augmenting path.

The flow augmenting path now has been redefined and therefore the algorithm that we have seen is called the 'flow augmenting path algorithm'. The flow augmenting path algorithm is simple to identify, update the flow corresponding to the flow augmenting path. If there is no flow augmenting paths possible the algorithm terminates. The definition of the flow augmenting path is any path from the source to the sink, which is made up of forward arcs, such that the forward arc has excess U_{ij} minus X_{ij}. In the backward arcs, if there is an X_{ji} in the forward direction, then there is a strictly positive flow in this. This is how the flow augmenting path algorithm actually works.

The next thing that we have to do is to show that the flow augmenting path algorithm is indeed optimal. To do that, we first write the primal and the dual of the maximum flow problem. Then, show some kind of primal dual relationships that will explain how and why the flow augmenting path algorithm is optimal and works effectively.

(Refer Slide Time: 18:00)



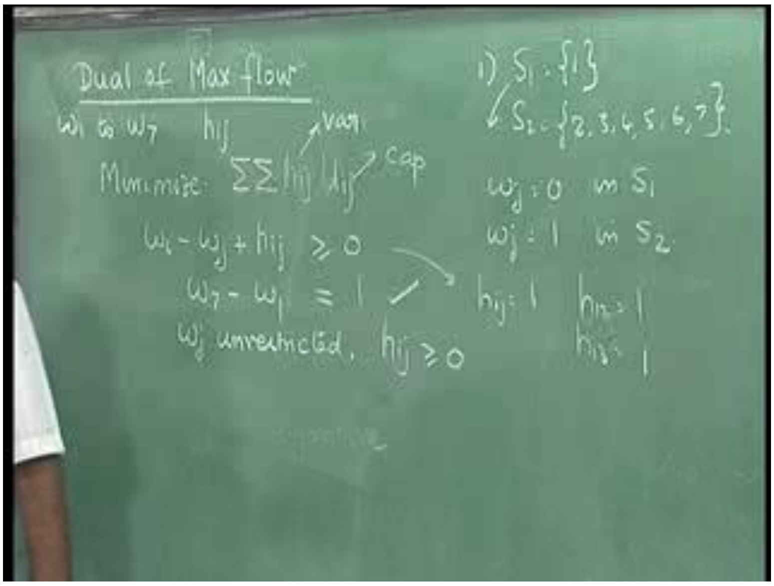
Let us write the primal and the dual; so primal will be written like this. Let us call 'f' as the maximum flow in this network, so maximize f. Now, there are 7 nodes in this. As far as this node is concerned, X_{12} plus X_{13} equal to f, because whatever goes out the total flow, that goes out of the source through X_{12} and X_{13} so, X_{12} plus X_{13} equal to f. Whenever we have an intermediate node like this we have flow balance equation. For this one, we will have minus X_{12} plus X_{24} plus X_{25} equal to 0. For this, we will have minus X_{13} plus X_{34} plus X_{36} equal to 0. For this, we will have minus X_{24} minus X_{34} plus X_{45} plus X_{46} plus X_{47} equal to 0. For this, we will have minus X_{25} minus X_{45} plus X_{57} equal to 0. For node six we will have minus X_{36} minus X_{46} plus X_{67} equal to 0. For node seven, we have minus X_{57} minus X_{67} minus X_{47} equal to minus f. Then, we have X_{ij} less than or equal to U_{ij}, X_{ij} greater than or equal to 0. This is the primal formulation for the maximum flow problem.

We have deliberately introduced a variable called f, which is the maximum flow. We may actually avoid this f by simply saying maximize X_{12} plus X_{13} or maximize X_{57} plus X_{67} plus X_{47}, we do not do that. For the sake of some convenience, we introduced a variable called f here. It becomes maximize f and then, we do this. Then, we also need to define whether this f is greater than or equal to 0 or whether it is unrestricted. If all X_{ij}s are greater than or equal to 0, then it is fairly obvious that f should be greater than or equal to 0. It makes sense to put f greater than or equal to 0, but we now define this f as an unrestricted variable. This is acceptable, because X_{ij} is greater than or equal to 0, in spite of the fact that f is unrestricted; X_{ij} greater than or equal to 0 will force f to be greater than or equal to 0.

There is a slight advantage in defining f as an unrestricted variable, which we will see when we actually write the dual of this particular primal. Before that, we also need to look at this constraint. This is an important constraint because whatever quantity that flows in an arc should be less than or equal to the arc capacity. So, X_{ij} should be less than or equal to U_{ij}. This is important because there are as many constraints as the number of arcs in the network. There are as many constraints as the number of nodes in the network. For every node, you have a constraint here; for every arc, you have a constraint here. If the network has n nodes, m arcs, then there are the X_{ij}s; there are m X_{ij}s plus 1 f. So there are (m plus 1) variables and there are m plus n constraints, so this is the formulation of this.

We also observe that this portion is unimodular. You can simply add vertically, you will get 0 equal to 0. This is unimodular and the max flow problem by itself has the unimodularity property. It is also not so much necessary to define X_{ij} as integers because X_{ij}s represent the flows on these arcs. X_{ij}s can be defined as continuous variables. So, the question of defining X_{ij}s to be integers and exploiting the unimodularity property does not exist. We can simply define X_{ij}s to be greater than or equal to 0 and the problem will become a linear programming problem by itself. Let us write the dual of this problem. The dual of the max flow problem will look like this.

(Refer Slide Time: 22:59)



To write the dual of this problem, we now start defining dual variables. There are seven nodes; there are seven constraints here. So, we can start defining dual variables w_1 to w_7. We start defining dual variables w_1 to w_7 for these seven nodes and here, we have as many dual variables as the number of arcs. We are going to define h_{ij} as the dual variables associated with this. With that in mind, let us write the dual.

We can always bring this f into this side of the equation, by putting a minus f here. We can bring this to the other side. All the right hand sides are 0 here; so this is the only place where we have a right hand side. The problem will be to minimize, the primal being a maximization problem, the dual will be a minimization problem. Problem will be to minimize double sigma h_{ij} into U_{ij}. Remember that h_{ij} is the variable and U_{ij} is the capacity. The objective function is to minimize h_{ij} into U_{ij}.

Now, what are the constraints? For every variable, there will be a constraint. So, if we look at a variable, a typical variable X_{ij} or for that matter, X_{12}. X_{12} will appear in the first constraint, second constraint and 1 constraint here. The dual variables will be w_1 here, w_2 here and h_{12} here. In general, if you consider a variable X_{ij}, it appears with a plus 1 coefficient in the i'th constraint, minus 1 coefficient in the j'th constraint and again, there is a constraint which is of the form X_{ij} less than or equal to U_{ij}. The corresponding dual constraint will become for a typical $X_{i,}$ it will become w_i minus w_j plus h_{ij}. The objective function value is 0; every variable X_{ij} has an objective function value 0. It is a minimization problem, so you will have greater than or equal to 0. For any variable $X_{ij,}$ you will have w_i minus w_j plus h_{ij} will be the left hand side. The right hand side will be zero, this will be a greater than or equal to because you have a maximization problem, where all X_{ij}s are correctly defined. So, you have w_i minus w_j plus h_{ij} greater than or equal to 0.

There is also the variable f. Now, this variable f has already moved in here. It has moved in here, so this will have w_7 minus w_1. This will have w_7 minus w_1 equal to 1, because the objective function coefficient is 1. So, this comes here with the minus sign, this comes here with a plus sign. So, plus w_7 minus w_1 equal to plus 1 the equation comes because f is unrestricted here. Defining f as an unrestricted variable helps us in getting this equation here. We will see how we use this equation effectively.

Now the dual variables, w_j are unrestricted, because these are the seven equations that correspond to the w's. They are all equations, so the corresponding w's are unrestricted. Then, you have h_{ij} greater than or equal to zero, because you have a maximization problem here, you have less than or equal to constraint. You will have a minimization problem here with a greater than or equal to variable. From the primal dual relationships, we can write this as the dual of the max flow problem.

What do we gain through this dual and what is the advantage of putting this equation here, the equal to sign here while defining f as unrestricted?

This is the next thing we will have to look at. Before we do that, let us go back to this network again and try to understand how we define the maximum flow on this network. Let us come back here to this network. Now, let us try and see what is the maximum flow possible in this.

If we look at the source one or suppose if we look at the sink, all the flow that leaves the source will have to now come to the sink. The sink has three edges that get into the sink. The capacities of these three are 30, 10 and 20. The maximum that can enter the sink is 60 here. Similarly, if we go back to the source, everything has to leave the source and you leave the source only in two ways; so through 1 2 and 1 3. The maximum is 80, so the maximum that can leave this source is 80.

This 40 plus 40 equal to 80 represent some kind of an upper limit or an upper bound on what can be the maximum flow. From this, we can say very confidently that the maximum flow on this cannot be more than 80. Applying the same thing to the sink, we now can say that the max flow cannot be more than 60. Now, let us look at these two things again in a slightly different context. If we say that considering this source 1, the maximum flow that can happen is 80. It is like saying, if I send 80, I have this source and these are the two edges that go out of the source, which have capacity of 80. If I remove both these edges here, I have a disconnected network.

I have disconnected this network with source on one side and sink plus rest of them on the other side. I know that the capacity that I have actually pulled out in order to disconnect this is 40 plus 40 equal to 80. If I am able to pull out some edges, such that I disconnect this network, there is no path possible. Then, the capacity of those arcs actually gives me an upper estimate of what the flow can be. I just put it back here to get 40, 20 and I get 40 and 35. If I remove these two, I know that I disconnect this network or alternately, if I can identify a set of edges such that by disconnecting those edges, I am able to divide this network into two parts.

Then, I realise that the capacity of such edges now gives me an upper limit on what my flow can be. The same logic can be used here. Now, by removing these three edges, I can simply disconnect it by all of them here and seven being there. Now, each of these gives us a capacity of 60. We can go back and say that the maximum flow now can only be less than or equal to 60. Now, that brings us to another question. When we do this, what have we effectively done?

We have by removing these three edges, we have now created two sets of nodes, one set of node which will be 1 to 6 and the other set of node which is 7 which means effectively the source should always be in s_1, the sink should always be in s_2. So this has to be in s_1, this has to be in s_2, the rest of them can be anywhere.

What I mean to say is instead of saying I have removed 5-7, 4-7, 6-7 with the capacity of 60, thereby I have disconnected this network, thereby I have got this s_1 and s_2 defined accordingly, that is one way of saying it. The other way of saying it is, if I take the nodes 1 to 7 and if I simply divide these nodes into two sets, s_1 and s_2, such that node 1, which is the source is always in s_1 and node 7 or sink is always in s_2. The rest of the nodes can be anywhere and for such an s_1 and s_2 defined, it is always possible to remove the corresponding edges such that the graph becomes disconnected with s_1 and s_2 defined accordingly.

For example, if I define s_1 as 1 2 3 and if I define s_2 as 4 5 6 7, which means I am effectively removing this many portion of the graph, whereby the capacity that I am pulling out is 10 plus 10 equal to 20 plus 15 equal to, 35, 35 plus 20 becomes 55. Now, 55 becomes an upper limit for the flow that I can have. It is fairly obvious to understand that we now have a feasible flow with 55, 55 being an upper limit. Therefore, 55 is optimal. But leave that alone, we can now say that for every definition of s_1 and s_2, such that the nodes are now divided into two sets.

The source is always in s_1, the sink is always in s_2, the rest of the nodes are distributed. If we are able to get for every possible definition of s_1 and s_2, it is always possible to go back to this network and find out the removal of which of these arcs will create this and disconnect this network. The capacities of such arcs will give an upper limit on the maximum flow.

For example, if we consider this, then it means that we are effectively removing this, this, this and this. We are removing arcs with capacities 55 and therefore, the maximum flow can only be 55. 55 is an upper bound to the maximum flow. Now, what does this tell us. This can be done in multiple ways. It is all about the number of nodes being here, making sure that the source is always in s_1, the sink is always in s_2 and the rest of the nodes can be put anywhere else.

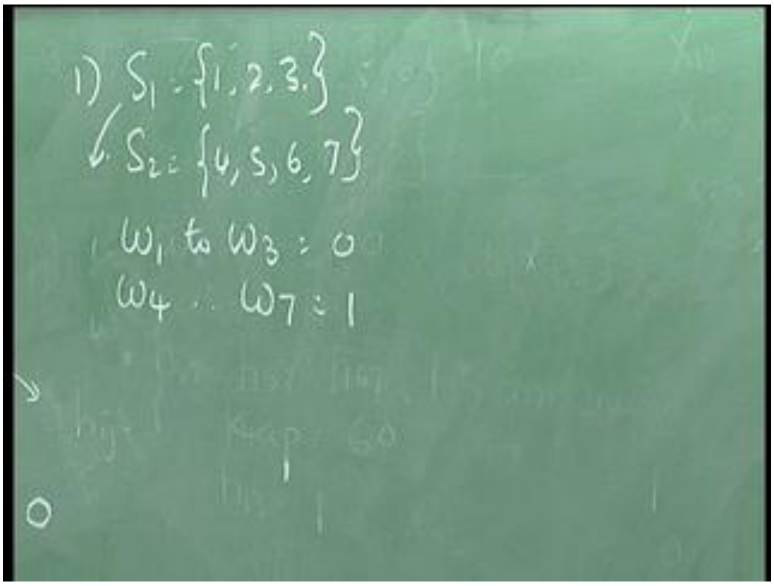
Effectively, we have created two sets of nodes, where the source is in s_1 and the sink is in s_2. We have divided the network or cut the network, such that the nodes are now into two sets. This kind of a thing is called a cut or it is called a cutset. For every cut that we can create or every cutset that we can create, we can go back and find out what is the capacity of the arcs that have to be removed, in order to create this cut and each one of them is an upper bound to the maximum flow. This is the relationship between the maximum flow and the cut. It is also obvious that we have seen three different cuts.

The cut here gave us a capacity of 80, the cut here gave us a capacity of 60, while a cut here gave us a capacity of 55. Each of the capacities of the cut now gives an upper bound or an upper limit to the max flow. Since each of these gives an upper limit, the smaller the capacity of the cut, the more value and information it gives us, because from this we can say 55 is an upper bound on the maximum flow that is possible on this network.

The cutset gives us that kind of an information, so the smaller the cutset is the closer we are to the optimum of the max flow problem. Obviously, since every cutset being a feasible solution, gives an upper bound to the maximum flow, there has to be a strong relationship between the cutset and the dual of the max flow problem.

Let us move now to the dual of the max flow problem and try to map these cutsets with the dual of this problem.

(Refer Slide Time: 36:33)



First, let us consider the first cutset, which is made up with s_1 equal to 1, the source and s_2 equal to 2 3 4 5 6 7, that is, we are going to consider this cutset. If we do this here, for every cutset, we can actually find a feasible solution to the dual. How do we do that? For every element which is in s_1, we put w_j equal to zero, if it is in $s_{1.}$ w_j is equal to 1, if it is in s_2.

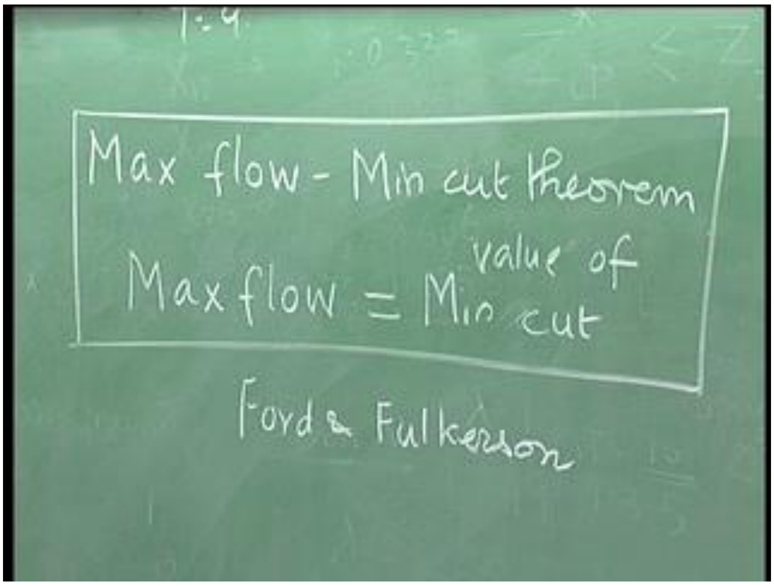
to this cutset, s_1 will be 0, s_2 3 4 5 6 7 etc., will be 1. This is always satisfied, because w_7 is always in s_2, the sink is always in s_2, the source in s_1. So, the difference is always equal to 1. If we go back to this 1 2, these are the arcs, which we are removing: X_{12} and X_{13}. Now, go back here wherever w_i is 0 and w_j is 1 so we get a minus 1 here, the corresponding h_{ij} will be equal to 1, that is, h_{12} will be equal to 1 and h_{13} will be equal to 1, so that this is satisfied.

In all other cases, if there is an arc that connects nodes in this, the corresponding h_{ij} will be equal to 0. For every cut or cutest, you can get a feasible solution to the dual and the capacity of such a cutset, if you see, will be U_{ij} h_{ij}, so that will be 40 into 1 plus 40 into 1 which is 80 which was the capacity of these two. Now, the same kind of interpretation can be done for the other cutset, where we have s_1 equal to 1 2 3 4 5 6 and s_2 is equal to 7, which means we are looking at this cutset. For this cutset, we will have w_1 to w_6 will be equal to 0, w_7 will be equal to 1 and we will have h_{47} equal to h_{57} equal to h_{67} equal to 1. Capacity of the cutset will be equal to 60.

For the third case that we looked at, we said 1 2 3 is s_1, 4 5 6 7 is s_2, we will have w_1 to w_3 equal to 0, w_4 to w_7 equal to 1. These four things h_{24}, h_{25}, h_{34} and h_{36} equal to 1. So, this is always satisfied this is also satisfied, you get feasible solution to the dual.

The objective function value will be sum of these capacities will be 55. It is easy to show the relationship between the cutset and the max flow. Every cutest is a feasible solution to the dual and therefore, every feasible solution to the dual is an upper bound to that of the primal. By weak duality theorem, every feasible solution to the dual, dual being a minimization problem is an upper bound to Z-star where the original primal is the maximization problem. Every cutset solution is a feasible solution to the dual of the max flow problem and by weak duality theorem gives an upper bound to the flow and by the optimality criterion theorem or by the main dualitive theorem at the optimum, the maximum flow and the minimum of these cuts that we have will be equal.

(Refer Slide Time: 00:41:17)



We get a very strong relationship which is called the max flow - min cut theorem which says at the optimum the maximum flow in the network is equal to the value of the minimum cut. This is a very important relationship. It is called the max flow - min cut theorem and it is attributed to Fords and Fulkerson who actually proved this max flow and min cut theorem.

What we have effectively done is, we have taken this example and we have shown here, that there is a strong relationship between the definition of the cut and the max flow. From here, it is known that every cut represents an upper bound. Later, we wrote the dual of the max flow problem and showed that every cut actually provides a feasible solution to the dual. Therefore, it is an upper bound to the Z-star by weak duality theorem. By the strong duality or the main duality theorem, we can show that the maximum flow is equal to the minimum cut.

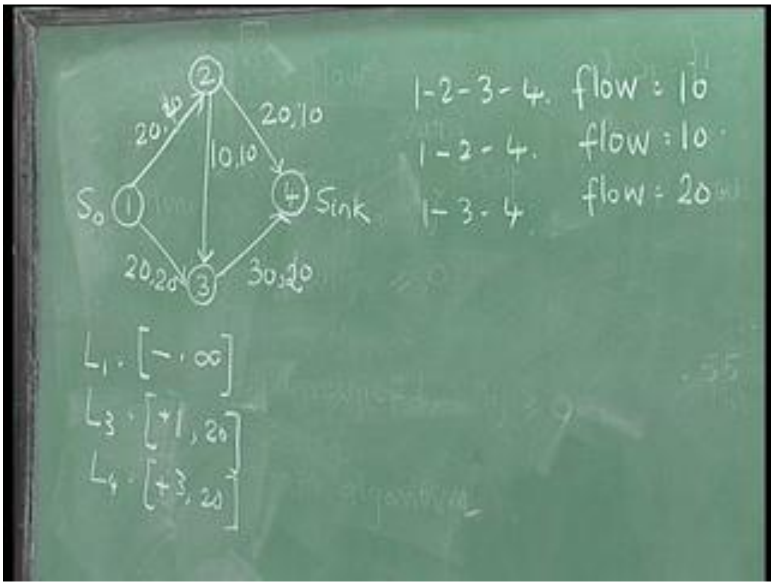
If the flow augmenting path algorithm is able to get the flows in such a manner that no more flow augmenting path is possible then we have actually exhausted all the possible flows here. This means, we have solved the max flow problem and we have also got the value of the min cut that is associated with the max flow.

Coming back to the flow augmenting path algorithm, we define the flow augmenting path as a path that comprises of arcs from source to sink. If it is a forward arc then there is a residual capacity which means there is a difference between the U_{ij} and the

X_{ij}. If it is a backward arc ij then there must be a forward arc ji with strictly positive flow.

Having seen the flow augmenting path algorithm and the way we evaluated all the flow augmenting paths, the next question is can we have some kind of a labeling algorithm which will help us program or efficiently get the maximum flow given a network. We now look at a labeling algorithm, which we can use to solve the max flow problems. Let us take a simple network here so that we can illustrate the labeling algorithm better.

(Refer Slide Time: 44:00)



Let us look at a network and try to define or do a labeling algorithm which will help us get the flow augmenting paths. The first step what we will do, this is the source, this is the sink, what we do first is, we first define L_1 is equal to minus comma infinity. This is a simple initialization; we do not need to worry about it. Now, from 1 go to an arc, such that there is a forward arc; so from 1, we go to 2. Start defining label L_2 will be from 1 to 2, there is a capacity of 20, there is a flow of 0. It is a forward arc from 1 to 2, so we can increase the capacity by 20. We put plus 1 and we put 20.

The plus 1 comes because we have looked at arc 1 2, arc 1 2 has a capacity of 20. Now, flow is 0 so plus 1 and an increase of 20 is possible. Go back from 2 and see if there is a forward arc. There is a forward arc 2 3, so now look at L_3. Now, 2 3 has capacity of 10, flow equal to zero, so 2 3 will be plus 2, because it comes from node two.

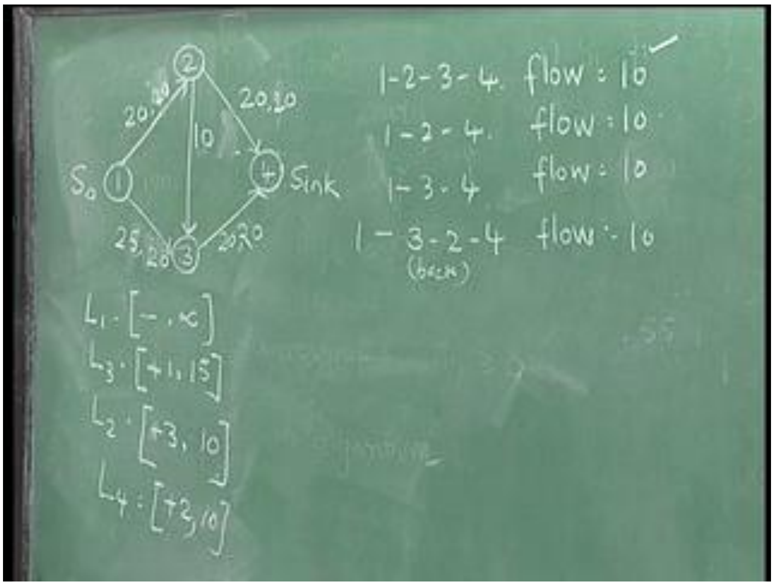
The U_{ij} minus X_{ij} is 10 and the earlier value is 20. Take the smaller of them, so you get plus 2 and 10. We are in three, see if there is a forward arc, so 3 to 4. You will do L_4 will be equal to plus 3, because it comes from 3. The capacity is 30, the flow is zero0, so 30 is possible. Take the minimum of this 30 and this 10, which is 10. Now, the moment the sink is labeled, go back and augment or update the flows. This is the last value that we have here associated with the sink is the quantity of flow that is going to be increased.

Now, we have flow equal to 10, the path will be 4 and plus 3, 3 and plus 2, 2 and plus 1. So, the path will be 1 to 2, 2 to 3, 3 to 4 and flow equal to 10. Update the flows, so 1 to 2, 2 to 3, 3 to 4, flow is equal to 10. Now, go back here and again start with $L_1$is equal to minus infinity. Look at the forward arc. 1 to 2 is a forward arc with 10 capacity, 20 minus 10 that is, 10. You will do plus 1 and 10, this 10 comes from 20 to 10. See if there is a forward arc 2 to 3. There is, but the capacity is 0, so ignore it. See if there is a backward arc 2 to 3, you do not have here. Then, you go back to 2 to 4, there is a forward arc 2 to 4. So you will do L_4 equal to plus 2.

There is 20 and 0, 20 capacity but you have 10 here, so minimum of 10 and 20 is 10. You will see that the sink is labeled. Therefore, the flow can be increased by 10. Then, you have 2 to 4, 1 to 2. You have 1 to 2, 2 to 4 with flow equal to 10, now update it. 1 to 2 becomes 20, 2 to 4 becomes 10, now go back and do this again. You have 1 to 2, you cannot do. So, you move from 1 to 3, at 20 and 0 so you put a label L_3 as plus (2, 20).

Look at 3 to 4, 20 is possible. So, L_4 will become plus 3. The minimum of this 20 and this 20 is 20. Once again, the sink is labeled. Go back and update the flow, so flow is equal to 20. You have 20 and 20 updated 20 and this will also be updated to 20. This will give us 1, so 3 will be plus 1, 20. This will give 1 to 3, 3 to 4 and flow is equal to 20.

(Refer Slide Time: 50:00)



See whether you have any other possibility, so 1 to 2, you cannot move 1 to 3, no more flow augmenting path exists and therefore, you get flow equal to 20. The one thing that we have not actually shown in this example is how you get a backward arc into the picture. If you want to do that, then if for example, if at some point, we have 20, 20 here and say you have 25, 20 here. If we did the labeling algorithm, even so you have 10 1 2 3 4. Let us redo this. We had 1 2 3 4 with flow of 10, which is correct.

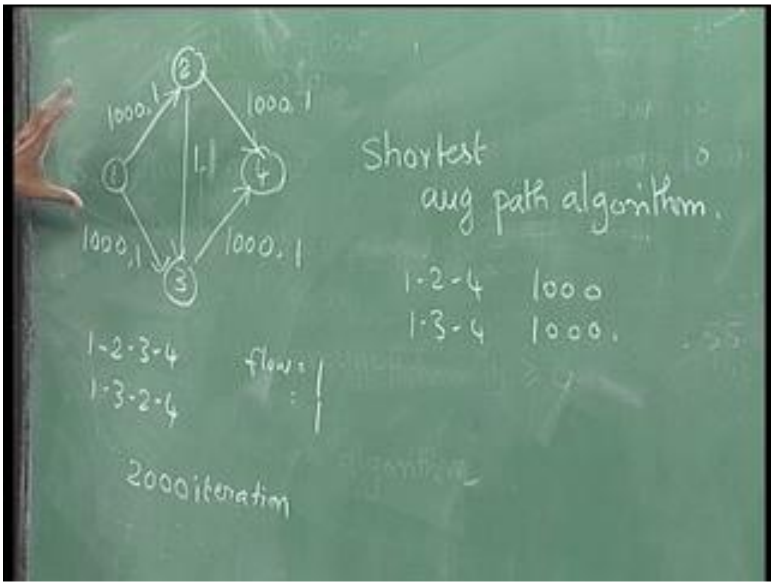
Let us have the capacity here as 20. Then, we had 1 2 4 with flow 10, now you have 20 here and then we have 1 3 4; 1 3 4 with flow 10 so that this becomes 20 and this becomes 10. If we have this network with capacities 20, 10, 20, 20, and 25 and three iterations of the flow augmenting path would have given us something like this. Go back and look at a flow augmenting path like this. L_1 is labeled as minus comma infinity. We realize 1 to 3 is a forward arc, so I can label L_3 as plus 1, 15, that I have here, I can do plus 1 and 15. Now, 3 to 4, I cannot have a forward arc because this is full, but I can do 3 to 2 as a backward arc, because 2 to 3 is a forward arc with flow equal to 10.

I can go back and do L_2 as plus 3 and minus 3 and 10 here. This minus indicates that it is a backward arc and 3 indicates that it is coming from 3 to 2 backward arc, which means 2 to 3 forward arc. Now, I have reached 2, now I go back here i can label L_4 as plus 2 with a gap of 10. 10 is smaller, the minimum of this 10 and this 10 which happens to be 10 here. I can increase the flow further by considering 4 2 2 and then 3 and then 1. So, 1 3 2 4, now 3 to 2 is a backward arc, flow will be equal to 10.

We can update this, as this becomes 20, this becomes 0 and this becomes 20. We can define this labeling algorithm to handle both the flow augmenting paths that are made up entirely of forward arcs as well as flow augmenting paths which have a combination of forward arcs and backward arcs. We saw the other example right now, how you model a backward arc.

So you do 1 to 3, 3 to 2, 2 to 4 and then, increase the flow here by 10. Reduce the flow or divert the flow and bring it back to zero and then increase it to 20. This is how the labeling algorithm for the flow augmenting path works. So far, the labeling algorithm as well as the flow augmenting path algorithm are very good ways to solve the maximum flow problem. But are there limitations to that?

(Refer Slide Time: 54:40)



We can see that through a very simple example. If we consider a very simple example, where let us say this has a capacity of thousand, this has a capacity of thousand, this has a capacity of thousand, this has a capacity of 1000 and this has a capacity of 1. If we apply the flow augmenting path algorithm, we realize that we could start with 1 2 3 4 with flow equal to 1. You could get a 1 here, a 1 here, and a 1 here. Then, we may consider 1 3 2 4 with a backward arc. This becomes 1, this becomes 0, this becomes 1, flow equal to 1.

Again we can consider 1 2 3 4, again we can consider 1 3 2 4, and so on. Then, we realise that we actually require 2000 iterations, with each iteration augmenting the flow by 1 unit to get a maximum flow of 2000 which is possible here. In some sense, the flow augmenting path algorithm is not a very efficient algorithm, when it comes to running time, running time as indicated or shown by this particular counter example. So, the flow augmenting path algorithm, the way we define would actually the extent of computation, would actually be dependent on the maximum of these capacities.

Later, people came up with refinements, where we look at what is called s'hortest augmenting path algorithm', where we always find an augmenting path which has as few edges as possible. We realise that sometimes a wrong choice would lead to solutions which require backward arc flows. If the paths are chosen very carefully, then there can be no need for a backward arc path. The shortest augmenting path algorithm will try to first exhaust all the forward paths with increasing number of arcs inside and only when absolutely required, we will evaluate arcs that require backward arcs or will evaluate paths that would require backward arcs.

If we apply the shortest augmenting path algorithm, we straight away do 1 2 4 with flow equal to thousand and we do 1 3 4, with flow equal to 1000. This would ensure that we first exhaust all the forward arc paths and we go to backward arc only when it is absolutely necessary. In fact, this would not even evaluate a backward arc path. The shortest path augmenting path algorithm is actually an improvement over the normal flow augmenting path algorithm. But, the flow augmenting path algorithm helps us to understand the maximum flow problem and its implications from an OR point of view.

In the next lecture, we will see the minimum cost flow problem and continue our discussion in OR models.