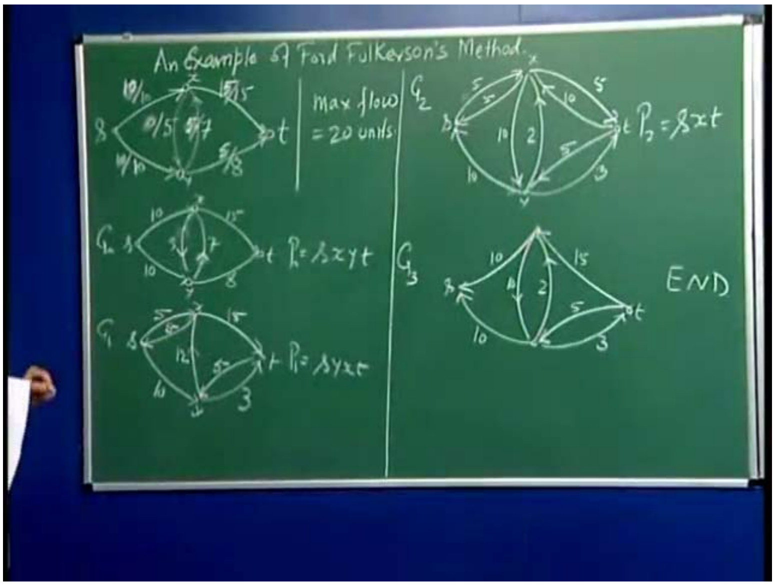
Computer Algorithms – 2
Prof. Dr. Shashank. K. Mehta
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

Lecture - 11
Edmond Karp Algo

Today, we will begin with an example of Ford Fulkerson's method on a very small network.

(Refer Slide Time: 00:26)



Suppose, the given network has 4 nodes the source and the sink node, and the 2 additional nodes, what you see here are the capacities of the various edges. So, the step one, initializes the flow to zero value so, we have values zero out of given capacity on each of the edges. Hence, the residual graph is precisely the initial network so, let us say the residual graph G naught is, we have this.

Now, in this graph, my mistake we have decided to put the direction as this so in this, we have to choose a directed path from source to the sink and suppose, we choose the path to be this. So, that would be s say, this is x and y, x and y and t so, I choose my path to be s x y t, path P naught. The minimum capacity on this path is 5 hence, we can augment 5 units of flow on this so, in the next graph, the flow is may be I can show the flow here, will be 5 units here, this will be 5 and this will be 5.

So, the edges of the residual graph, I will still have an edge with 5 units this way but, I will have an reverse edge also 5. We will have a edge with 10 capacity this way now, this is saturated so, this edge will go away and we will have edge with 5 capacity but, we also have additional 7 capacity. So, this will be a 12 capacity edge, this is 15 and this has 3 and the reverse edge has 5 capacity so, we have the new graph. Now, suppose we choose a path to be s y x so, let us choose path P 1 to be s y x t, on this path we have 10, 12 and 15 hence, 10 is the minimum capacity.

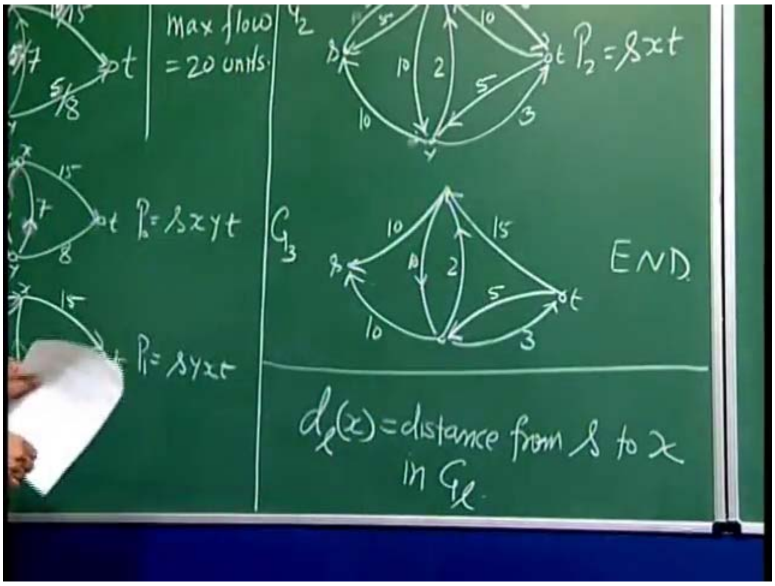
So, I can augment 10 units so, this flow will be 10, we will augment 10 so, this will be 0 and this will be 5 here and this will be 10 units. The new residual graph will have still a 5 capacity edge this way and a 5 capacity this way, this time there is a 10 capacity because, this is saturated this way. Now, we have a total capacity this way is 10 units and 2 units this way so, this is 10 and this is 2, capacity of 5 this way and 10 this way and we have a capacity of 3 and a capacity of 5 in the reverse edge.

Now, we have one possible directed edges s x t so suppose, we choose the next path to the s x t the two edges, both have capacity 5 so, the minimum capacity is 5, we can augment 5 units of flow in this. So, that gives me 10 units here and 15 units this way and the residual graph for this flow, we have a saturated edge from here to here so, there is a reverse edge of 10 capacity and this is also saturated. So, we have a reverse edge 15 the other edges are untouched, we will have 10 this way and 2 this way, I will have 10 capacity edge this way, 5 this way and a 3 this way.

In this case, since right here we see there is no way to get out of s, there cannot be path from s to t so, this is the end of the algorithm. We have reached a stage where, there is no path hence, this flow has to be the maximum flow and notice that, the total fluid exiting from s is 10 plus 10 units so, the mod so, maximum flow is 20 units. Now, in this example, what we notice is that, there was an edge x y which was once saturated and hence, we had no edge going from x to y.

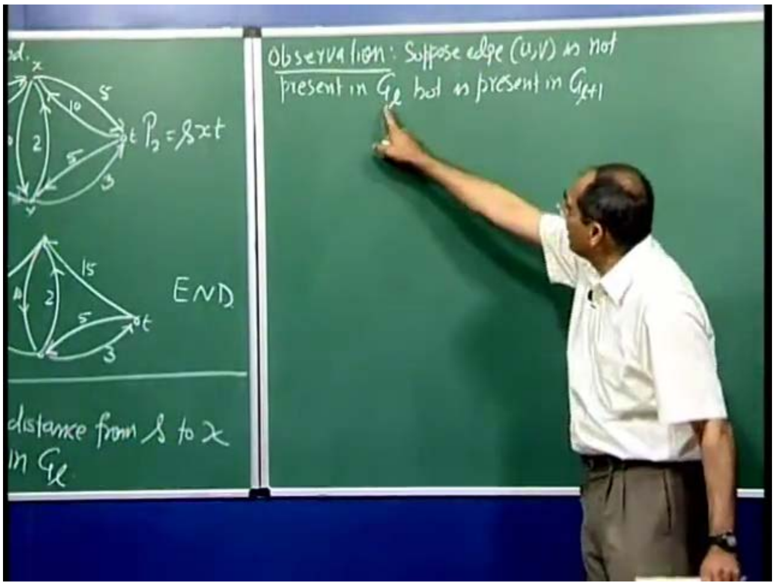
At a later stage what we notice is that, x to y edge reemerges at some stage hence, it is possible that a certain edge, which vanishes at some stage in the residual graph may reappear. So, in general, there is no way to tell, whether these iterations will terminate and this is the problem with this general method, unless we give a way to guarantee the termination of these iteration, this cannot be considered as an algorithm in classical sense now, we want to make one observation in this context.

(Refer Slide Time: 10:05)



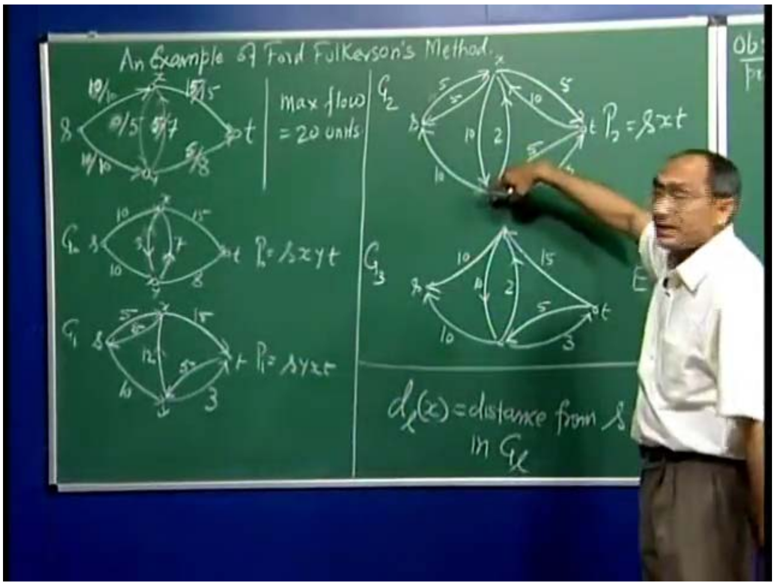
I would notation first of all, we will denote the successive residual graphs by G 0, G 1, G 2, G 3. The path computed, the respective paths computed are P 0, P 1, P 2, etcetera and one more notation I will use D l x to denote distance from s to vertex x in graph G l, these are the three notations I need from now onwards.

(Refer Slide Time: 10:58)



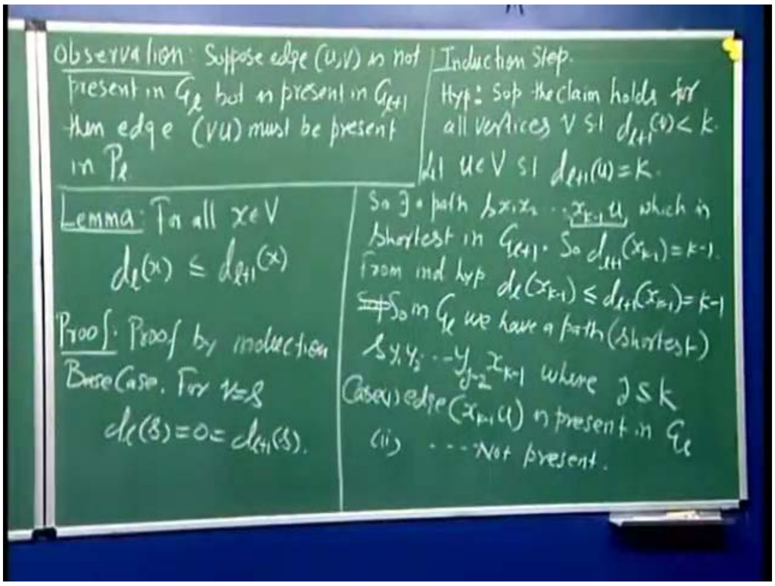
So, we want to make an observation that suppose, edge U V is not present in G l but, is present in G l plus 1. Suppose, it so happens that, there was an edge which is not available in G l but, is present in G l plus 1.

(Refer Slide Time: 11:41)



For example, edge x y is not present in this graph but, it is present in the next one and what can we say about this particular edge. Why is this not present here, the reason must be that, that is a saturated edge, there is no capacity there. Something must have happened in our augmentation says that, the capacity from x to y reemerges. And that can happen only if, we augment flow in such a way, that there is new flow from y to x and you notice that, there is edge y x present in the path P 1.

(Refer Slide Time: 12:46)



So, we are saying that, if U V is not present in G l but, present in G l plus 1 then, edge V U must be present in P l, the path computed from G l. Now, we would like to prove an interesting and significant result, which leads to a very interesting way to modify this method into an effective algorithm. So, we have a lemma which says, for all vertices x d l of x is less than equal to d l plus 1 of x, the distance from s to x in G l cannot be more than distance from s to x in the next graph.

Let us try to prove this, we will prove this result by induction, what we will do is that, we will take this measure for induction, the distances of vertices from s in G l plus 1. So now, base case notice that, the only vertex which has distance 0 from s is s and that is true in every single graph. Hence, what we do know is that, for v equal to s, d l of s is 0 which is also in the next graph. So, this inequality holds for the vertex, which has d l plus 1 value 0, there is no other vertex where, the distance 0 hence, we have proven this claim for the distance 0 case.

Now, we take the induction step and from induction hypothesis, suppose the claim is true, the claim holds for all vertices v says that, d l plus 1 of v is less than K. So, if we notice that, there is a vertex with a d l plus 1 v less than or equal to K minus 1, the claim holds and we want to prove that, the same holds when this distance is equal to K for all those vertices so, now suppose, let u be a vertex says that, d l plus 1 of u is equal to K.

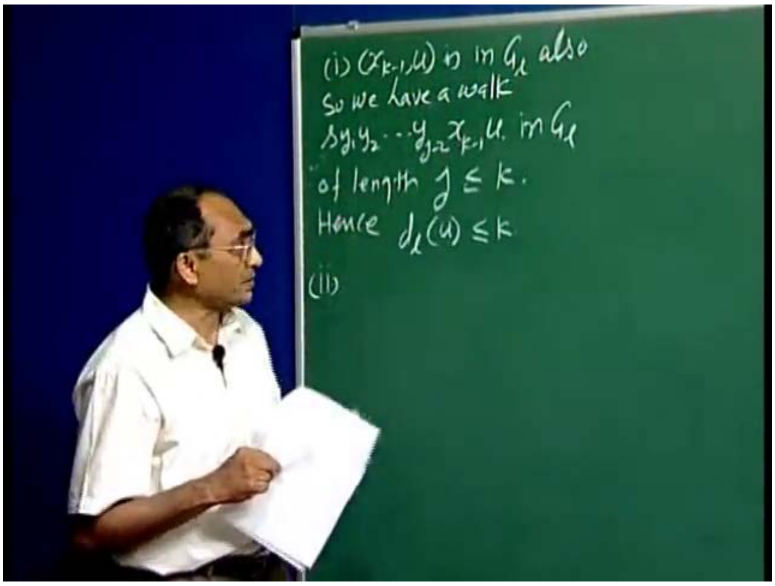
Distance is the length of the shortest path so, there exists a path s x 1 x 2 x k minus 1 u, which is shortest in G l plus 1 that is because, we are given so…

Now, as a result what we notice is that, the vertex x k minus 1 has a distance k minus 1 so, d l plus 1 of x k minus 1 is k minus 1 but, this qualifies under this condition. Hence from induction hypothesis, d l of x k minus 1 is less than equal to d l plus 1 of x k minus 1, which is equal to k minus 1. So, in graph G l also, the distance of x k minus 1 from s is less than equal to k minus 1. Now, let us say, we have a path suppose, in G l we have a path s y 1 y 2 y j minus 2 x k minus 1, we have a shortest path, path as shown as it is shortest and this as suppose, we do have.

So, in G l, we have a path where, j has to be less than equal to k because, the length of this path is j minus 1 and that should be less than equal to k minus 1. Now, let us take a look at this edge, k minus 1 u now, this edge is present in G l plus 1 so, there are two possibilities, either it is present in G l or it is not. So, we will consider 2 cases, case 1, the edge x k minus 1 u is present in G l and case 2 is not present.

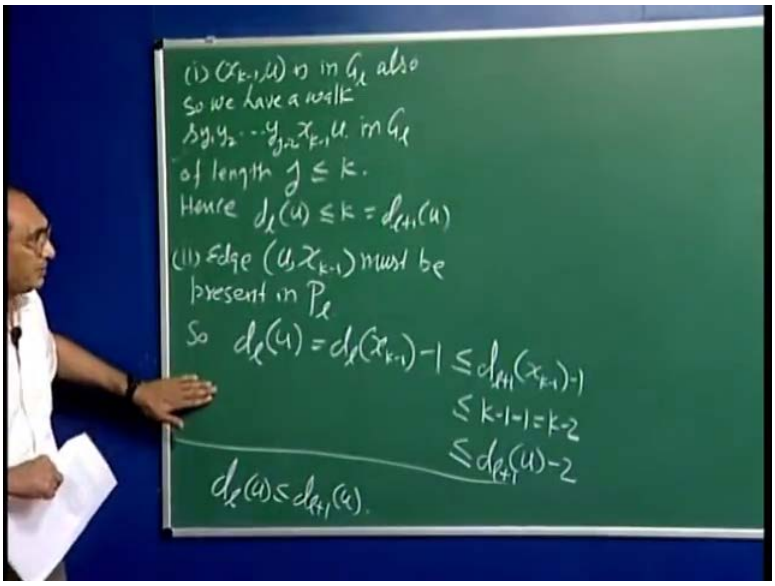
Now, consider the first case, if indeed this edge is available to us then, I can take this path and attend this edge to it, that will give me a walk of length K. So, there has to be a path of length less than or equal to K from s to u and that is what, we want to show that, if d l plus 1 u is equal to K. Then, there is a path of length no more than k from s to u in d l as well.

(Refer Slide Time: 21:54)



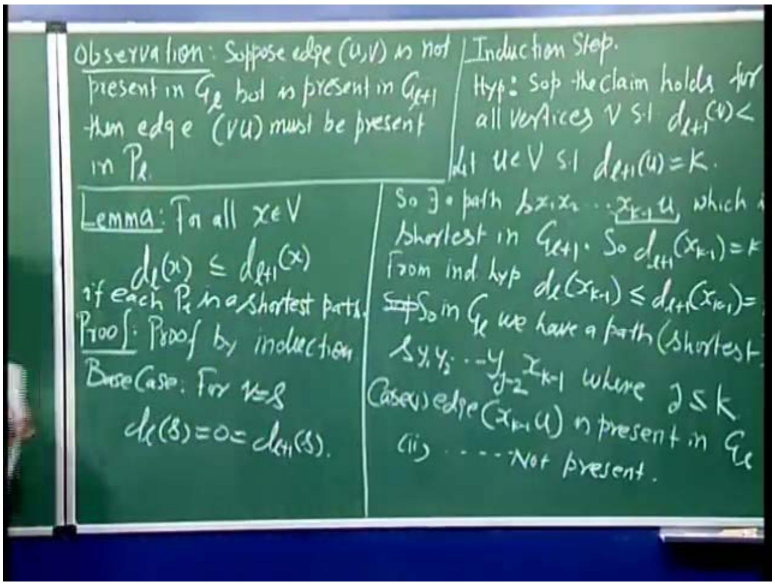
So, we have first case where, x k minus 1 is u is in G l also so, we have a walk s y 1 y 2 y j minus 2 x k minus 1 u in G l, this is a walk from s to u hence, there is the length of this walk is j, the length of this is j of length j, which is less than equal to K, as we have seen earlier. Now, the walk is of length limited by K then, there has to be a path inside this, which also is not greater than K. Hence, the shortest path length from s to u in G l d l u has to be less than equal to K. Now, let us take a look at the case 2, in this case, the edge x k minus 1 u was not present in G l but, certainly was available in G l plus 1.

(Refer Slide Time: 23:46)



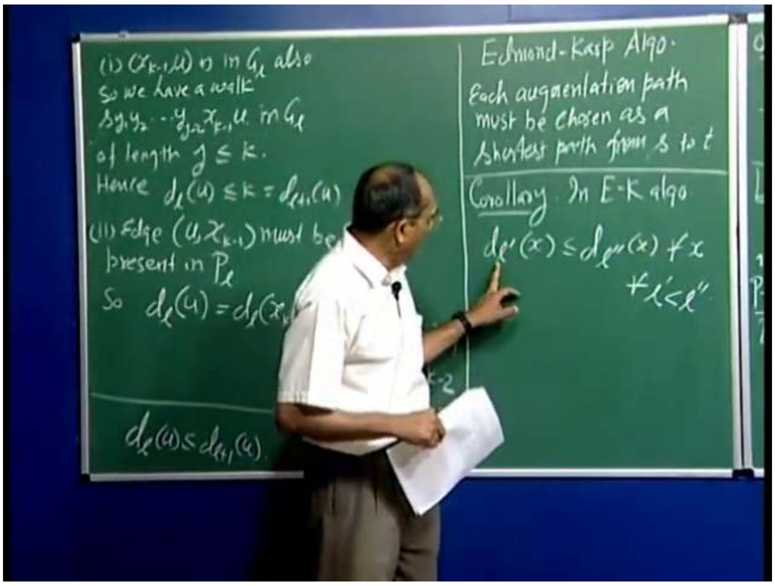
From our observation, we know that, edge u comma x k minus 1 must be present in P l in the augmentation path computed from G l must have had this edge. Now, so and notice that, the augmentation path, if we take this to be the shortest path then, d l of u is d l x k minus 1 minus 1. So now, I am assuming that, my augmentation path is a shortest path from s to t under that assumption, we have this available. And from induction hypothesis, this is less than equal to d l plus 1 x k minus 1 minus 1 and that in turn, is less than equal to k minus 1 minus 1 is k minus 2. So, if I assume that, my P l is a shortest path then, once again this of course, less than equal to d l plus 1 u minus 2. So, in both the cases, we have this is actually, equal to d l plus 1 u so either way, we have shown that, d l of u is less than equal to d l plus 1 u, if the augmentation path is a shortest path.

(Refer Slide Time: 26:29)



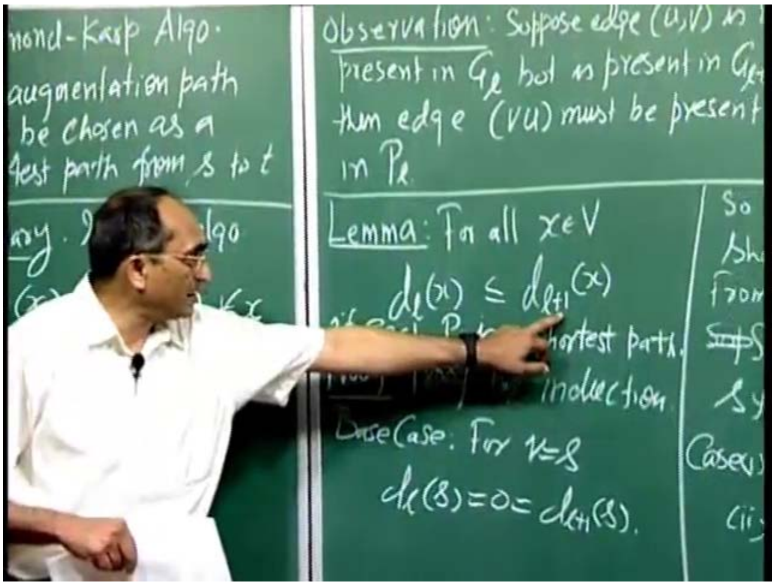
So, the lemma to be précised is, this is true if each P i is a shortest path now, we will propose a modification or actually, an addendum in the Ford Fulkerson algorithm.

(Refer Slide Time: 27:04)



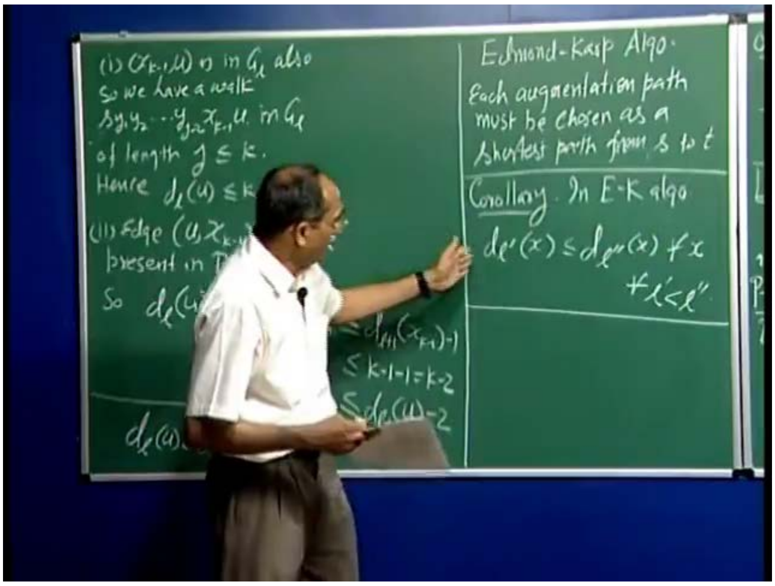
And which is proposed by originally Edmond and Karp so, it is known as an Edmond Karp algorithm, this algorithm is nothing but, Ford Fulkerson algorithm. But, they say in addition to this, each augmentation path, path must be chosen as a shortest path from s to t, this is the only modification in the method. And now, we are ready to show that, the number of iterations are going to be bounded so, let us try to prove actually, let us try to determine the number of iterations will be at most how much and from that, we will be able to determine the complexity of the algorithm. So now, the next thing I would like to prove, is a simple corollary that is, in Edmond Karp algorithm, d l prime of x is less than equal to d l double prime of x, for all x and for all l prime less than l double prime. That is, in any subsequent iteration, the distance of a vertex can never decrease.

(Refer Slide Time: 29:20)



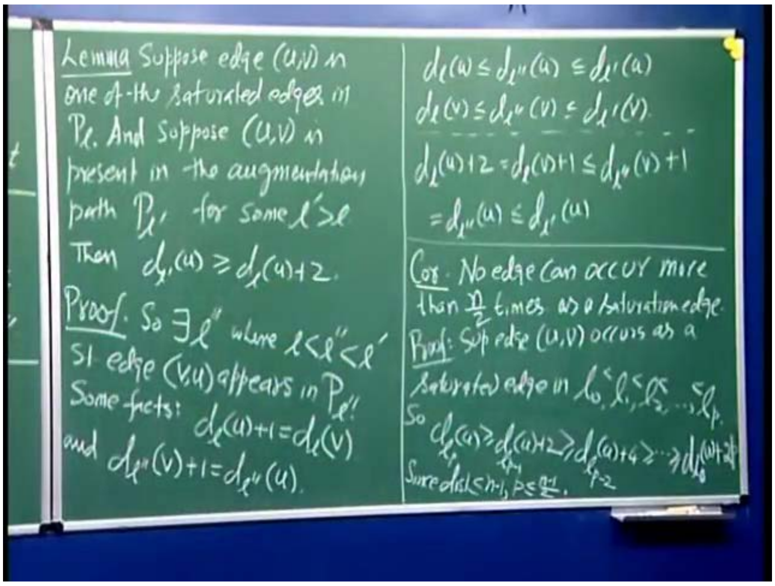
Now, that is a trivial consequence of this lemma because, if the distance in l plus 1 is greater than equal to distance in l then iteratively, the distance in l plus 2 will be greater than equal to distance in l plus 1 and so on, so this is trivially true.

(Refer Slide Time: 29:50)



Now, we are going to show the change in distance in a special situation, is little bit greater than a simple inequality.

(Refer Slide Time: 30:03)



So, we have a lemma then suppose, some edge U V is one of the saturated edge, edges in P l what I mean is that, in the graph G l, the path that we had selected was P l and the flow that was computed was looking at the minimum capacity and the minimum among the edges, which had minimum capacity was this one. So now, on this edge, we have fullest possible flow hence, this edge was a saturated in this path. Now, it is clear therefore, that this edge will be absent in the subsequent graph namely G l plus 1.

But suppose U V is present in the augmentation path P l prime for some l prime greater than l, we know of course, this cannot happen for l prime equal to l plus 1. But, may be some later stage, as we had seen in our example, it had reappeared so happens that, the same edge is present in the augmentation path of a later graph. Then in this case, we can say that, d l prime of u is greater than equal to d l of u plus 2 so, we already know of course, this claim we know from our previous result now, this is a stronger claim for such special cases.

So, how do we prove this, now notice that, this edge was saturated, there was no further capacity left for a appending any flow in this direction. If it again shows up in l prime that means, somewhere in between, there must be some graph G l double prime where, we must have sent the flow through edge V U. If U append a flow in the direction V U then, this flow will create some residual capacity for edge U V, it will reappear in the subsequent stage.

So, there exists l double prime where, l double prime is between l and l prime such that, edge V U appears in P l double prime, I hope that is clear. Because here, we had made a flow from V to U where by, we have created capacity from U to V and hence, in l double prime plus 1 stage, the edge u v reappears. So now, let us see, what are the facts we know, we know that U V was present in P l and V was present in P l double prime. So, some facts here namely, d l of u plus 1 is d l of v.

So, we had d l v because, in P l edge U V occurs, U occurs first V occurs later, this is shortest path. So, the distance from s to u has to be 1 less than the distance from s to v, the reverse occurs in P l double prime. So, we have d l double prime v plus 1 is d l double prime of u, we have this as well. From our corollary, we have a few additional facts we know that, d l of u is less than equal to d l double prime of u, which is less than equal to d l prime of u.

And finally, we also have d l of v is d l double prime of v, which is less than equal to d l prime of v, we have these four facts with us. What do we want to prove, we want to show this inequality so, let us say, we look at the value of d l u plus 2, d l u plus 1 is d l v so, this is equal to d l v plus 1, this is less than equal to d l double prime of v plus 1.

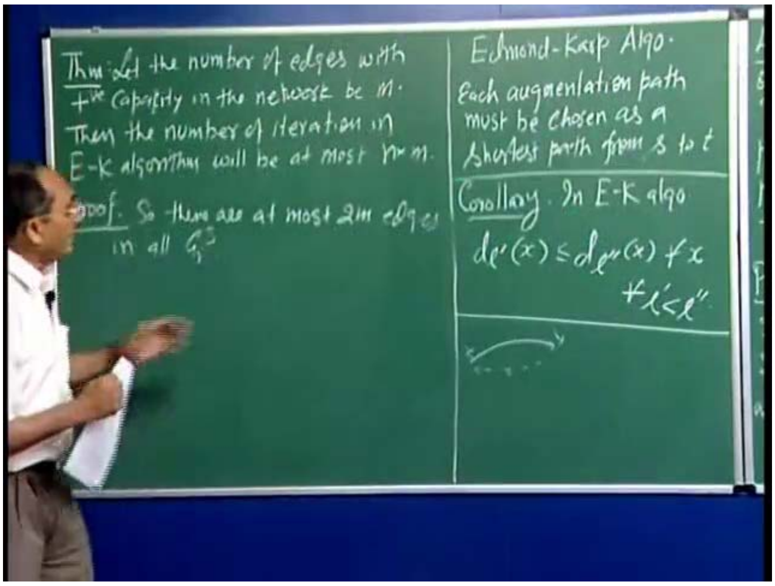
Because, this is greater than or equal to this, d l double prime v plus 1 is equal to d l double prime u, this is equal to d l double prime u, d l double prime u is less than equal to d l prime.

So, this is d l prime of u, this is all we needed to prove, this is the claim so, what we notice is, that if the same edge reappears on a subsequent augmentation path and first occurrence, it was a saturation edge. Then, the distance of vertex U increases in the second occurrence by at least 2 so now, I am claiming a corollary to this claim is that, no edge can occur more than n by 2 times. That is say, in n by 2 iterations, distinct iterations, no edge can occur as a saturation edge so suppose, let us try to prove this.

Suppose, edge U V occurs as a saturated edge in l, in the iteration number l naught, l 1, l 2, l p so, these are the different successive. And I am assuming that, it first occurred in iteration l naught then, in l 1 then, in l 2 and so on so, we know that, d l p of u has to be greater than equal to d of l p minus 1 of u plus 2 and in d l p minus 2 u plus 4. So finally, in d l 1, the inequality is that, this is at least 2 p smaller than this d l 0. So, what we notice is, the shortest distance in G l p has to be at least 2 p may be, at least 2 p plus 1 if it is not s but, at least 2 p.

But, the distance can never be more than n minus 1 in a graph of n vertices so, since distances are always less than equal to n minus 1, we conclude that p has to be less than or equal to n minus 1 over 2. What this result does is that, it puts a bound on number of times a particular edge can occur as a saturated edge in different iterations in different augmentation path. Now, this is all we need, we have only so many edges, we have n square edges, in every path in every residual graph, the augmentation that we compute must have at least one saturated edge and each edge is allowed to appear at most n by 2 times, that allows me to a bound on the total number of iterations.

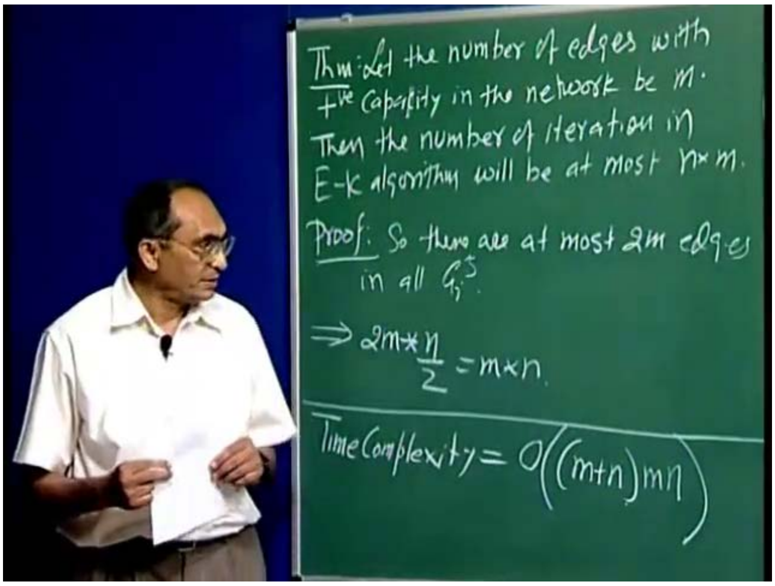
(Refer Slide Time: 43:32)



So, let us do the last step so, we have a simple result now, theorem let the number of edges with positive capacity in the network be m then, the number of iterations in Edmond Karp algorithm will be atmost n times m. If there are n edges in a network then, at most 2 m edges can occur because, if you have an edge x y to begin with but, you have 0 capacity here.

But, in some subsequent graph, you may very well have an edge y x in the residual graph. So, the total number of distinct edges that can show up in various residual graph can be at most 2 m so, there are atmost 2 m edges in all G I's, these are the only edges that can show up. This edge can be a saturation edge in atmost n by 2 iteration, every iteration must have at least one saturation edge.

(Refer Slide Time: 46:07)



So, we can directly say that, number of iteration can never exceed this number, if you have more than this many iterations than at least one edge, must have occurred n by 2 plus 1 times but, that is not allowed. So, once we have put a bound on this, the total time complexity can be determined now, in each iteration, we compute a shortest path from s to t and then, only on those edges, we need to modify the capacities.

Hence, only those edges will get modified in the next residual graph and that modification takes only order n times because, on a path, there are only at most n edges. So, the time to compute a shortest path n time to update the graph, is only order m plus n and there are atmost m n iterations. So, the time complexity of Edmond Karp's algorithm is m plus n times m n so, that completes our discussion of the flow networks. So, from next lecture, we will begin the discussion of matrix operations and in particular way, we will discuss matrix multiplication, inversion and decomposition.