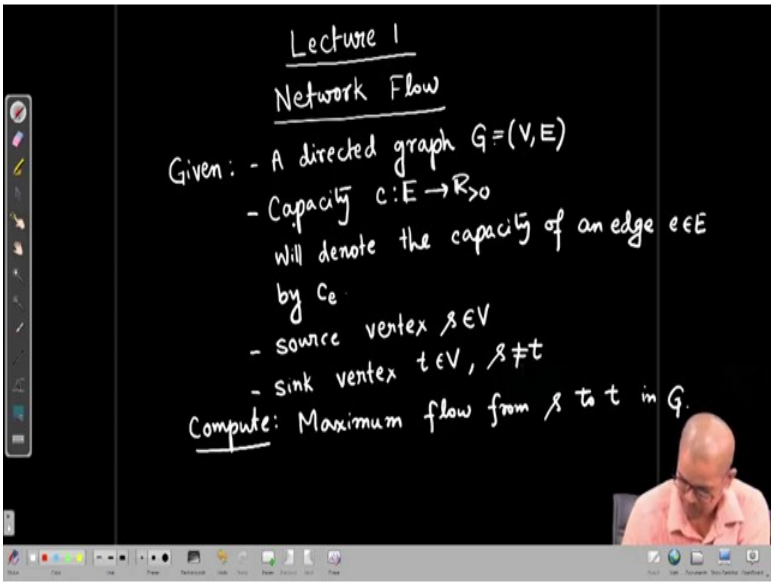
Selected Topics in Algorithm
Prof. Palash Dey
Department of Computer Science and Engineering
Indian Institute of Technology – Kharagpur

Lecture – 01
Introduction to Maximum Flow

Welcome to Selected Topics in Algorithms and this is the first lecture and we will be starting network flows. This is a fundamental problem in graph algorithms.

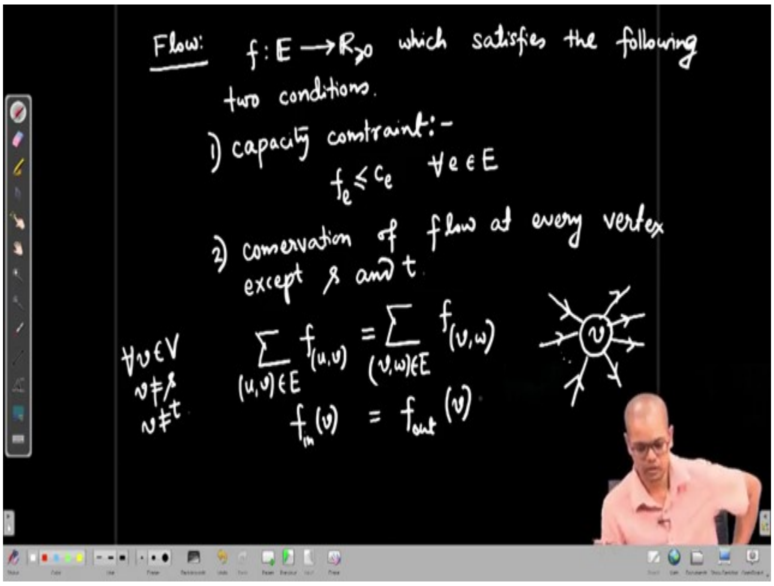
(Refer Slide Time: 00:40)



In the network flow problem, we are given a directed graph G and each edge has certain capacity. So, capacity C from edge set to positive real numbers, so, this graph G is with vertex at V and edge set E. So, we will denote the capacity of an edge E by C_e and there are two special vertices one is source vertex s and sink vertex t and we will assume that source is not equal to sink.

And what we need to compute is the maximum flow from s to t? Compute maximum flow from s to t in the flow graph G.

(Refer Slide Time: 03:59)

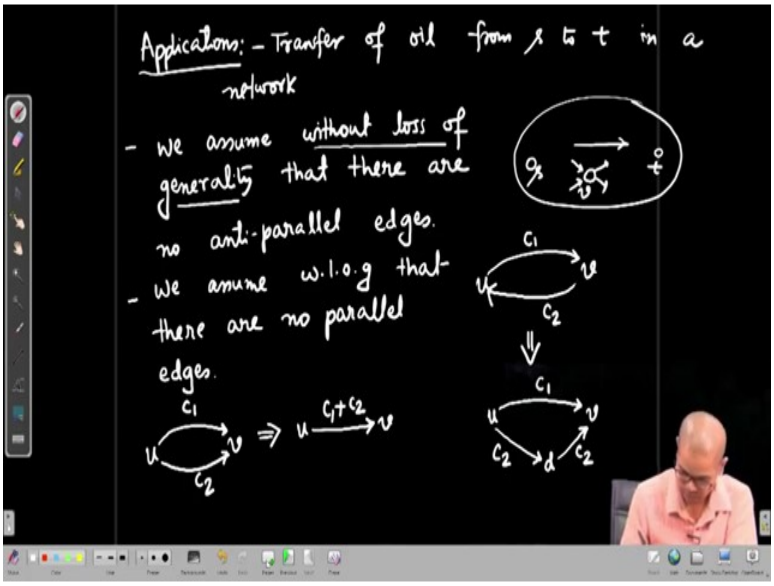


So now, we need to define what is a flow? So, flow is a function from set of edges to real numbers, non-negative or greater than equal to 0 which satisfies the following two conditions. The first condition is capacity constraint that means that the amount of flow in any net any edge e, f_e should be less than equal to the capacity of the edge. This should hold for all edge $e \in E$.

And the second condition is conservation of flow, at every vertex except is the source s and the sink t. What is conservation of flow? Let us see V be a vertex and what is the sum of flows across all it is incoming edges? That sum should be same as the sum of flows leaving the vertex. So, for all $u, v \in E$ $f_{(u,v)}$ this is the sum of flow entering V. This should be same as $v, w \in E$ $f_{(v,w)}$. This should hold for all vertex $v \in V$ except s and t.

So, the first term is called $f_{\text{in}}(v)$ and this should be same as $f_{\text{out}}(v)$. Now, this problem appears in many real world applications.

(Refer Slide Time: 08:09)



For example, so, applications, for example, transfer of say oil from s to t in a network. Suppose there is a network and each edge has certain capacity each edge is a oil pipeline and there is a source is where the oil is produced. And there is a sink t in a city from where the oil will be distributed to a city, to all oil stations in the city. And how fast we can send oil from s to t and where each edge has certain capacity.

So, each edge cannot transfer oil beyond it is capacity and there should not be any excess flow or any lack of flow in any internal node. So, the flow in and flow out should be same at every internal node, V. So and also there are so, you can think of instead of this oil network, this could be internet network with routers and again each link has a capacity and in terms of bandwidth.

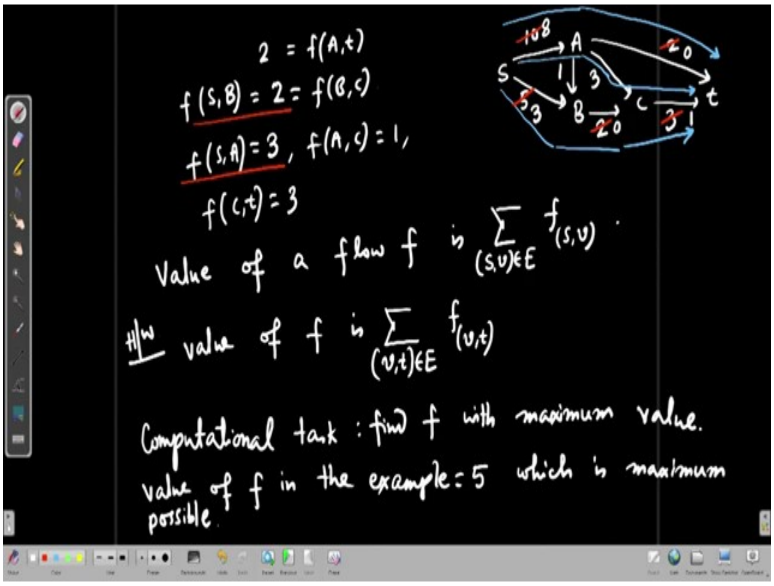
And again, the question is from source to destination how fast data can be transferred? So now, we will make a couple of an important observation, so, we assume, without loss of generality that there are no anti-parallel edges. So, anti-parallel edges look like suppose there are two vertices u and v. There is an edge from u to v and there is an edge from v to u. So, we can assume this without loss of generality, why we can assume without loss of generality?

Suppose the capacity of the top edge is C_1 and the capacity of bottom edge is C_2. So, if we have, if our network has such a structure then what we can do? Is that we instead consider another network u, v we introduce a dummy node, a new node d in the top edge. Let it remain same with capacity C_1 but the bottom edge passes through d and the capacity of both the bottom edges are C_2.

So, it is easy to check that the by this transformation we can get rid of all anti-parallel edges. Moreover, the value of the maximum flow remains same. Second, is we assume again without loss of generality that there are no parallel edges. Again, if there are parallel edges, say between u and v. There are two edges of capacities C_1 and C_2 . Then we can transform this, we can add a new edge between u and v and delete the old two edges.

And make the capacity of the new edge to be $C_1 + C_2$ and again it is easy to check that it does not affect the maximum flow value in the network.

(Refer Slide Time: 13:52)



So now, to get a feel, let us look at a concrete example of a max flow problem and to see how the max flow values may look like? So, suppose this is S and this is suppose A B C maybe t is here. Suppose these are the various edges and I am putting some numbers as capacities 10 5 1 2 3 3. Now, how does the capacity? How does the flow look like? A flow so, let us try a natural greedy approach of taking a path and try to send a flow.

So, let us maybe take this path, S to A to t and we see that you know we can push two amount of flow along this path S to A to t because the bottleneck here is the A to t edge of capacity 2. So, our current flow value is f of S, A is 2 and this is also f of A, t and rest all are 0. And so, what is the remaining capacities in the network to the after sending flow along of value 2? Along S A t path, the remaining capacities look like now, this 10 this 2 becomes 0.

There is no capacity left and this becomes 8. Next, maybe let us take another path. Maybe let us take the bottom path and then the flow values will be $f(S,B)$ is 2 and this is also the flow $f(B,C)$ which is $f(C,t)$. So, new flow values will become then this in S B there are three capacities left here. It is 0 and it is 1 and there is still a path S to A to C to t and that way the flow value of $f(S,A)$ this edge, was carrying 2 units of flow before now it is one more unit of flow is added.

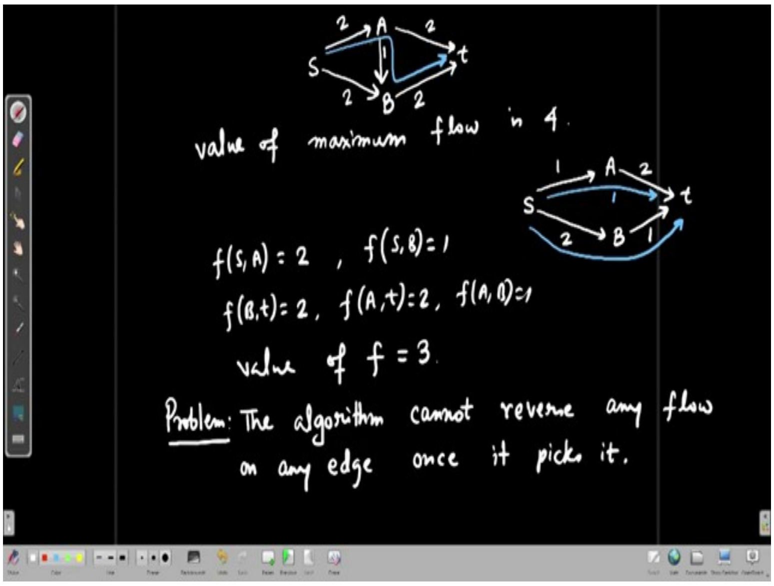
So, the new flow value becomes three in this edge. So, these, let me erase and $f(A,C)$ is 1 and $f(C,t)$ before it was carrying 2 unit of flow. Now, one more unit of flow is added, so, it has become 3 so, I erase this. So, these are the flow values of various edges but what is the value of the flow? So, the value of the flow if the total amount of flow leaving is so, these we define the value of the flow.

And I will let you check that value of the flow value of f is also the total amount of flow entering t. These two numbers are same. The total amount of flow leaving is same as the total amount of flow entering t. And this value I want to maximize so, the computational task is maximize or let me write this way find f with maximum value. So, for example, the value of the flow in this example is 3.

So, let me write the value of f in the example, let us see what it is? It is not 3. So, in S 2 A 3 amount of flow is there and in S to B 2 a unit of flow is there. So, total flow the value of the flow is 5. Is this the value of the maximum flow possible? And this is yes, it turns out that this is yes because for this, graph for this flow network, we see that you know the total amount of the total sum of capacities of all the edges entering t is 5.

And because the value of the flow is also the sum of the flows entering t and that can be at most the sum of the capacities of the edges entering t. So, in this particular example, the flow the value of the maximum flow is indeed 5. So, the value of f in this example is 5 which is maximum possible. Now, it gives an impression that you know this our greedy approach of finding a path and pushing as much flow as possible may give a maximum flow.

(Refer Slide Time: 22:49)



And looks like a viable approach so, whenever we have an algorithm, we have two options either we prove that it is correct. It is correct that means that for all instances it gives the right answer or the second option is that you show that it is not correct and to show that it is not correct one way or the most straightforward approach is to give a counter example, where the algorithm fails.

So, it turns out that this greedy approach is wrong and here is a counter example. So, what is the value of the maximum flow? We can send 2 units of flow along the path S to A to t and 2 units of flow along the path S to B to t. So, the value of maximum flow is 4 which is maximum possible because the sum of the capacities of all the edges leaving S is 4 which is same as the sum of the capacities of all the edges entering t.

Any flow must has to respect the flow the capacity constraint, property and hence the value of maximum flow, is at most four. Now, let us try to apply the greedy approach, it finds any path. So, if it finds the first path that it computes to send the flow, if it is S to A to t or S to B to t then it is fine. Then it indeed finds the maximum flow but the algorithm fails if the first path where it pushes the flow is S to B to t.

Now, if it is S to B to t then let us see what is the flow values? $f(S,A)$ is $f(A,B)$ is $f(B,t)$ which is 1 and now let us see how does the network look like? S to t A to B the capacity of A to B the edge A to B is exhausted and it does not have any capacity left. Now, it has only two disjoint S to t paths, S to A to t and S to B to t. And along each path maximum one unit of flow can be pushed.

So, the algorithm pushes 1 unit of flow on top path then S to A path becomes saturated. And then if it pushes 1 unit of flow in the bottom path then B to t edge becomes saturated and now in the resulting graph. There is no path left from S to t. So, with $f(S,A)$ will be 2 $f(S,B)$ will be 1 and $f(B,t)$ will be 2 $f(A,t)$ will be 2 and $f(A,B)$ is 1. This is the final flow value but it is value of f is 3.

So, what has gone wrong? The problem is that if the algorithm has made some wrong choice at any point of the iteration. The algorithm has no scope of reversing the decision, so, it is sort of committing whatever intermittent choices it is making. So, the problem, the algorithm cannot rectify or not reverse any flow on any edge once it fix it so that is the problem it.

The optimal solution in this example does not send any flow on the path A B and by and if any flow which sends any flow along the edge A B. Then that flow cannot be the optimal flow and that is the problem. And we will solve this problem in the next lecture and we will get our first algorithm for maximum flow. Thank you.