Second Level Algorithms

Prof. Palash Dey

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

Week – 04

Lecture 17

To the 17th lecture of the second-level algorithms course. In the last class, we started studying the Ford-Fulkerson method for computing a maximum s-t flow in a given network. In the last class, we have seen that when the algorithm terminates, the output flow is indeed a maximum s-t flow. In this lecture, we will begin by arguing why the Ford-Fulkerson method will always terminate if all the input capacities are rational numbers. So, let us begin. So, here is the lemma. The Ford-Fulkerson method always terminates if all the input capacity values are rational numbers. Proof. This lemma is much easier to see if the capacity values are all integers. If the capacity values are all integers, then the maximum value of the maximum s-t flow could be at most the sum of the capacity values. For any max-flow instance, the value of a maximum s-t flow is at most the sum of all the capacities of the edges. If all capacity values are integers, that means the value of a maximum s-t flow is an integer. If all the capacity values are integers, then this upper bound on the value of a maximum s-t flow is integral.

Now, observe that in this case whenever we augment some flow along an augmenting path in a residual graph the minimum residual capacity or the residual capacities of all the edges in the residual graph is also integer and hence the minimum residual capacity on the augmenting path which we choose to augment the current flow is also integer. In this case, the capacity values of the initial residual graph are all integral. Hence, the minimum capacity of any age in the st augmenting path is also integral. So, here is your residual graph G_f and here is S. We find the augmenting path from S to T and the question is how much flow we can send along this path. It depends on what is the capacities of individual edge. and the amount of flow that we can send along this ST augmenting path is the minimum capacity of any edge in this ST augmenting path that is the maximum amount of flow that we can send along this path. we hence the flow value increases by some integral amount from the initial zero value. So, at the beginning the flow value was 0 after first iteration flow value increases with some integral value. So, after first iteration the flow value remains integral and hence the residual graph after first iteration the capacities of every edge in the residual graph remain integral. the residual graph after every flow augmentation. contains only integral kappa city values okay in particular

The value of the flow increases by at least one. In every iteration. Hence, the maximum number of iterations that the Ford-Fulkerson method makes is at most the sum of the capacity values of all the edges. Hence, the Ford-Fulkerson algorithm always terminates if all the capacity values are integral. Next, we will see how if all the capacity values are rational numbers, then this also holds. If the capacity value of every edge is a rational number, then we can assume without loss of generality that all these rational numbers have the same denominator, say λ. How can we assume this without loss of generality? There are rational numbers, and we can make the denominator of all those rational numbers the LCM of those denominators.

So, λ can be chosen as the LCM of all the denominators, and then by multiplying both numerator and denominator with a suitable natural number, we can ensure that the denominator of all the rational numbers is the same, which we will call λ. Then, by the same argument as with integer capacity values, the value of the flow increases by at least $\frac{1}{\lambda}$, in every iteration. Hence, the number of iterations that the Ford-Fulkerson method takes is at most λ times the sum of the capacity values of all the edges. In particular, the Ford-Fulkerson method always terminates if all the capacity values of the edges are rational numbers. So, this concludes the proof of the correctness of the Ford-Fulkerson method because, in the last lecture, we saw that when the Ford-Fulkerson method terminates—that is, when there is no s-t path in the residual graph—the flow value is a maximum s-t flow. The Ford-Fulkerson method is correct if all the capacity values of the edges are rational numbers. Our proof of the termination of the Ford-Fulkerson method, when the capacity values are integers, also gives a very important corollary.

We can say that there exists a maximum s-t flow with integral flow values in every edge if all the capacity values of all the edges are integers. So that is a very important corollary which we will use in lots of applications of maximum flow. So here is the corollary. If the capacity values of the edges are integers, then there exists an integral maximum s-t flow that is, f(e) is some natural number including zero for every edge, okay. So now let's turn our attention to the time complexity of the Ford-Fulkerson method. So let's assume that all the flow values are integers, then let's compute what the time complexity is. So, every iteration takes—let us see how much time. In every iteration, we construct the residual graph, which takes $O(|V|+|E|)$ for constructing the residual graph.

Then we need to find the s-t path in the residual graph, which can be done using breadth-first search, taking time $O(|V|+|E|)$ for finding a path in the residual graph. And once we have found the path, we need flow augmentation, which is changing the flow value along that path. The length of the path could be at most $|V|-1$. So, we need to change the flow value of at most $|V|-1$ edges.

So, flow augmentation takes V go of $|V|$. for flow augmentation. So, total time taken in every iteration is $O(|V|+|E|)$ and we have to multiply this with the number of iterations. So, if the capacity values are all integers, Then the number of iterations is at most some of the capacity values and this holds even for rational capacity values also.

So total time complexity is the sum of the capacity values of all the edges. This is the number of iterations times $|V|+|E|$. So this runtime is not polynomial. Such a runtime is called pseudo polynomial because it depends on the value of the input numbers.

Its runtime is polynomial in the value of the input numbers and other parameters. So such a runtime is not polynomial. So this is not polynomial this sort of runtime which is polynomial in the input numbers are called pseudo polynomial. So this time complexity for rational age capacity.

So, in the next lecture, we will see another algorithm based on the Ford-Fulkerson method, which achieves polynomial time complexity for computing a maximum s-t flow. So, let us stop here. Thank you.