Second Level Algorithms

Prof. Palash Dey

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

Week – 07

Lecture 33

Welcome to the thirty-third lecture of the second-level algorithms course. In this lecture, we will see an important result concerning graph theory, which is called Hall's marriage theorem, and we will see how this theorem can be easily proved using again maximum flow, OK? So, let us begin. So, this is regarding perfect matchings in bipartite graphs.

So, what is a perfect matching? A matching in a graph is called a perfect matching Let's give a name to the matching, say M. So, a matching M is called a perfect matching if every vertex of the graph belongs to one edge of M. In this case, every vertex is matched.

We want to ask: What are the conditions to guarantee that there exists a perfect matching in a bipartite graph? So, let us consider a bipartite graph with bipartition L and R. They are independent sets; edges go only across. What are the necessary and sufficient conditions for the existence of a perfect matching in a bipartite Obviously, not every bipartite graph has a perfect matching.

For example, if the bipartite graph is ageless, there is no perfect matching. On the other hand, there exist graphs on which there exists a perfect matching. An extreme example is a complete bipartite graph. What is the complete bipartite graph? Between every pair of vertices, one vertex from L and the other vertex from R, there exists an edge. So, those complete bipartite graphs always have a perfect matching.

So, we are interested in studying what are the necessary and sufficient conditions for the existence of a perfect matching in a bipartite graph. So, let us first ask: what are some necessary conditions? A necessary condition for a perfect matching to exist in a bipartite graph is that every subset of vertices, say, call it X, a subset of L should have enough neighbors in R. So, we introduce the concept of a neighborhood. What is a neighborhood? So, this is a bipartite graph L R. Let us consider a subset X of L. The neighborhood of X is the endpoint of all the edges whose one endpoint is in L and the other endpoint is in R. The R-side endpoint of all the edges whose one endpoint is in L. So, you look at the set of edges that are incident on any vertex in X, and the set of vertices in R on which these edges are incident—that is the neighborhood of X. We denote it by N(X). So, formally, N(X), which we call the neighborhood of X.

Is all the vertices in R such that there exists a vertex in X with an edge between x and y. So, this is how we define the neighborhood of a subset of vertices in L. A clear necessary condition for a perfect matching to exist is that for all subsets of vertices in L, the size of its neighborhood should be at least as much as the size of the subset itself. This is because the vertices in X can be matched only with the vertices in N(X).

The celebrated Hall's marriage theorem states that this necessary condition is actually sufficient. A bipartite graph G with bipartition L and R has a perfect matching if and only if for all subsets X of L, the size of the neighborhood of X is at least as much as the size of X. Proof. We have already argued that the conditions in 1—let us call this inequality 1—are necessary for the existence of a perfect matching in a bipartite graph. Now we will show sufficiency. That means, let G be a graph that satisfies all the inequalities in 1; we need to show that it has a perfect matching. So, it is a proof by contradiction.

So, suppose not; then the size of the maximum matching in G is less than the size of L, which is also the size of R. One more condition, of course: we need the size of L and the size of R to be the same. So, now consider the reduced instance of s-t flow, which we used for computing a maximum matching in a bipartite graph. Consider the reduced instance G' of the maximum s-t flow problem. Let us recall how we constructed the equivalent instance of the maximum s-t flow problem. We are given a bipartite graph G with bipartition L and R. L and R are independent sets; edges only go across. For this graph, we have to find a maximum bipartite matching.

We added a new vertex S and a new vertex T. We add an edge from S to every vertex in L, an edge from every vertex in R to T, and we orient the edges between L and R to go from L to R. So, this was our reduced instance of the s-t flow problem; we call it G'. We observe that the value of any maximum s-t flow in G' is the same as the size of any maximum cardinality matching in G which is strictly less than the cardinality of L or the cardinality of R—they are the same thing. But now, recall the max-flow min-cut theorem from max flow. Min-cut theorem. We know that the capacity of any minimum s-t cut in $G^{'}$ is the same as the value of any maximum s-t flow, which we have argued is strictly less than the cardinality of L. So, let us consider any minimum s-t cut in $G^{'}$. Here is a schematic diagram of $G^{'}$. So, let us consider any minimum s-t cut of $G^{'}$. Let us call it A comma V minus A, a minimum s-t cut in $G^{'}$. So, A should include S and should not include T. In general, A can include some vertices of L, some vertices of R, and S—this is how A looks like—and the remaining set of vertices is the other part.

So, let us highlight the set of edges that contribute to the capacity of the cut. Let us recall the capacity of every edge in $G^{'}$ was 1. So, the capacity of the cut, which is the sum of the capacities of all the edges going from the S side to the T side, is in this case the number of edges going from the S side to the T side. So, let us highlight the set of edges that go from the S side, which is A, to the other side.

So, these are the edges that go from vertex S to the other side; these are the edges. which go to vertex t from the A side, and these are the edges which go from the A side to the other side, which goes from L to R. Let us assume that the number of the edges of the third is k. So, let us call these type 1 edges, these type 2 edges, and these type 3 edges. So, the capacity of the S-T cut is the number of type 1 edges. What is the number of type 1 edges?

These are the number of edges in L minus, let us call it n_A, and let us call it n_B. So, it is, and let us call the cardinality of L, which is the same as the cardinality of R, to be n. So, the number of edges of type 1 is $n-n_A$. The number of edges of type 2 is n_B, and the number of edges of type 3 we have assumed to be k. So, the capacity of this S-T cut is $n-n_A+n_B+k$. From the max-flow min-cut theorem, we know that the capacity of the min S-T cut is the same as the value of any maximum S-T flow, which is less than the cardinality of L, which is n in this case. So, the capacity of this cut should be strictly less than n. Cancelling n from both sides, we get that n_A should be at least or strictly more than n_B+k. Now, consider x to be the set of vertices in A which belong to L also, that is A intersection L, whose cardinality is n_A. Let us find out what is the cardinality of n_X. The number of neighbors of x within A can be at most N_B, and outside A can be at most k. So, the cardinality of n_X is at most n_B+k. This is the maximum possible number of neighbors of x in A and k is the maximum possible number of neighbors of $x \in B$. But then what we have seen, let us call it equation 2.

From equation 2, what do I get? That the cardinality of x is more than the cardinality of its neighborhood, which contradicts our assumption that the input graph satisfies all conditions in one. Hence, the size of the maximum matching in G must be equal to n, that is, G must have a perfect matching, which concludes the proof of Hall's marriage theorem, OK. So, let us stop here. Thank you.