Applications of Max Flow Min Cut

Rich Schwartz

April 4, 2022

The purpose of these notes is to give several applications of the Max Flow Min Cut Theorem. Nothing in these notes is original to me. All the applications are classic ones, and actually I found the proofs just by looking around online.

0.1 The Main Theorem

We first recall the statement of this theorem. We have a directed graph $G = (E, V)$ with a source vertex s and a sink vertex t. The edges are labeled with non-negative capacities. Here are the two main definitions:

- A *feasible flow* on G is an assignment of non-negative numbers to each edge such that each edge is assigned a number no larger than its capacity. Additionally, the amount of flow into each vertex v equals the amount of flow out of v provided that $v \neq s, t$. The *value* of the flow is the amount flowing out of s.

- A *cut* of G is a partition of the vertices of G into two disjoint sets S and T such that $s \in S$ and $t \in T$. The capacity of the cut is the sum of all the capacities of edges pointing from S to T.

Here is the theorem.

Theorem 0.1 (Max Flow Min Cut) *The maximum value of a feasible flow on G equals the minimum capacity cut of G. Moreover, if the capacities of G are integers, then there is a maximal flow with integer values.*

Proof: (sketch) Start with a maximal feasible flow. (These exist even in the irrational case, by "compactness".) Starting with s let S be the set of all vertices of G reachable by augmenting flow paths. Let T be the remaining vertices. If $t \in S$ then we can augment the flow and increase its value. Any edge pointing from S to T must be used to capacity because otherwise we'd be able to reach a vertex of T by an augmenting flowline. Likewise and edge pointing from T to S is not used at all because otherwise we could again reach the vertex in T by an augmenting path. Hence the capacity of the $S - T$ cut equals the value of the flow.

In the integer case, we produce the maximal flow by starting with the 0-flow and making a series of integral modifications coming from integral augmenting paths. Hence, the final flow, which is maximal, has integer values. ♠

0.2 Sums of Unit Flows

Consider an integer feasible flow on a directed graph, as above. Say that a *unit flow* is a flow which has flow 1 on some edges and 0 on other edges, and such that the union of edges having flow 1 is either a cycle or a path. In the path case, we insist that the path have p and q as endpoints.

Theorem 0.2 *Every integer feasible flow can be written as a sum of unit flows.*

Proof: The proof goes by induction on the sum of all the numbers that the flow assigns to the edges. For simplicity we assume that s and t are not adjacent. If they are, we can just add extra vertices on the edges joining s and t. Call two edges e_1 and e_2 compatible if these edges share a vertex and one of the edges points into the vertex and the other one points out.

Suppose first that the flow has positive value. Start with an edge e_1 incident to s that has positive flow. Then there is some edge e_2 compatible with e_1 and having positive flow. We continue like this, producing edges $e_1, e_2, ...$ We keep going until either we reach t or we reach some previously encountered vertex. In the first case, we can subtract off a unit flow path from our flow. In the second case we can subtract off a unit flow loop.

Suppose now that the flow has value 0. Then we do the same thing, except that we start at edges not incident to s or t. The process described above produces a unit flow loop, which we subtract off.

In either case, we can subtract off a unit flow path or a unit flow loop. Subtracting this off leaves a feasible integer flow having smaller total. ♠

0.3 Hall's Matching Theorem

We use Max Flow Min Cut to prove the Hall Matching Theorem. Suppose that $H = (A, B)$ is a bipartite graph satisfying Hall's criterion. This is to say that the set of neighbors in B of any subset $S \subset A$ is at least as large as S.

Constructing the New Graph: Let G be the following directed graph. We make all the edges of H point from A to B. We add a vertex s and join it to every vertex in A. We make these new edges point into A. We add a vertex t and join it to every vertex B. We make these new edges point to t. To get a flow problem, we assign capacity 1 to all the edges involving s and t. We assign capacity ∞ (or, if you prefer, some finite number larger than the total number of edges of G.)

Properties of the Min Cut: The only edges in a minimal cut can be those involving s or t. Otherwise the capacity of the cut is enormous. Suppose that A has n vertices. Notice that the cut with $S = \{s\}$ has n edges. So, the min cut has capacity at most n. On the other hand, consider an arbitrary cut of capacity at most n. Let A' denote the set of vertices of A involved in the cut and let B' denote the set of vertices of B involved in the cut. There are no edges from $A - A'$ to $B - B'$ because we have a cut. So, all the neighbors of $A - A'$ lie in B'. Hence $|B'| \geq |A - A'|$, by Hall's Criterion. But then the number of edges in the cut is

$$|A'| + |B'| \geq |A'| + |A - A'| = |A| = n.$$

This proves that the min cut has capacity n.

Properties of the Max Flow: By Max Flow Min Cut, there is an integer maximal flow from s to t having value n. Since there are n edges leaving s, this flow must use each edge leaving s with to the max. In other words, one unit of flow comes out of each edge from s into A. Hence 1 unit of flow comes into each vertex of A. Since the flow is integral, exactly one edge leaving a vertex of A has positive flow, and this flow must be 1. The point is that there is one unit of ingoing flow and hence one unit of outgoing flow. Let M be the set of edges of H which have flow. Two such edges have different endpoints in A for the reason just mentioned. For the same reason, they have different endpoints in B. Hence M is a matching between A and some subset of B. This completes the proof.

0.4 Menger's Theorem: Edge Version

Let H be connected (undirected) graph. let $s, t \in H$ be two vertices that are not adjacent. An $s-t$-*path* is an embedded path joining s to t. Two such paths are *edge disjoint* if they have no edges in connon. An *edge cut* is a collection of C of edges in H such that every st-path uses at least one edge in the set. Notice that the minumum size of an edge cut must be at least as large as the maximum number of pairwise edge disjoint st-paths. This situation is a lot like Max Flow Min Cut, except that H is not directed.

Theorem 0.3 (Menger) *The maximum number of pairwise edge disjoint st-edges equals the minumum size of an edge cut.*

To prove this result, we introduce a modified (multi-)graph G that is directed. We give G the same vertices as H, but we replace each edge ab of G by the two directed edges $a \to b$ and $b \to a$. We then give all edges of G capacity 1.

Properties of the Min Cut: Consider a minimum capacity cut of G. This is a partition of the vertices of G into two sets S and T such that $s \in S$ and $t \in T$. Since all edges of G have capacity 1, the capacity of this cut is just the number of edges pointing from S to T. Such a cut corresponds to an st-cut in the sense of Menger. Hence, the min cut of G has at least as many edges as the min cut of H. Let us remember this as:

$$N = |\text{min cut}(G)| \geq |\text{min edge cut}(H)|. \tag{1}$$

Properties of a Max Flow: By Max Flow Min Cut there is a maximum integer flow having value N. Each edge of G gets value 0 or 1. We can write our flow as the sum of unit flow paths and unit flow cycles. Since the value of the flow is N, and all edges have capacity 0 or 1, there must be N unit flow paths joining s to t. The conservation property implies that two of these paths cannot merge together and have a common edge. This would give us an incoming flow of 2 and an outgoing flow of 2 on a single edge. This is above capacity, so it cannot happen. Hence our unit flow paths give us N pairwise edge-disjoint st-paths. Let us record this as

$$\text{number of edge disjoint st} - \text{paths} \geq N. \tag{2}$$

But we have produced at least as many edge disjoint st-paths in H as the minimum cut of H. This shows that the maximum number of pairwise edge-disjoint st-paths is at least as large as the minimum edge-cut of H. Since we automatically get the other inequality, we in fact must have equality. This completes the proof of Menger's Theorem.

0.5 Menger's Theorem: Vertex Version

Let H be a (undirected) graph and let s, t be non-adjacent vertices in H. We say that two st-paths in H are *vertex disjoint* if they have no vertices in common except s and t. (Sometimes this is called being *internally disjoint*).

Say that a *vertex cut* of H is a collection of vertices such that every st-path in H contains one of these vertices. Here is the vertex version of Menger's Theorem.

Theorem 0.4 (Menger) *The maximum number of pairwise vertex disjoint st-edges equals the minumum size of a vertex cut.*

To prove this result, we again convert H into a directed graph G. This time the conversion is really cool. We first make a white vertex for s and a black vertex for t. We then make a black-white pair of vertices for each other vertex of H. We join black to white vertices in G if and only if the corresponding vertices are joined in H. We use blue edges for this. We also join each black-white pair by a red edge. This gives us a bipartite graph. For the sake of notation, we denote a black vertex with a $(-)$ and a white vertex with a $(+)$. We direct the blue edges from white to black, and the red edges from black to white. We give capacity 1 to all edges of G. Figure 1 gives an example.

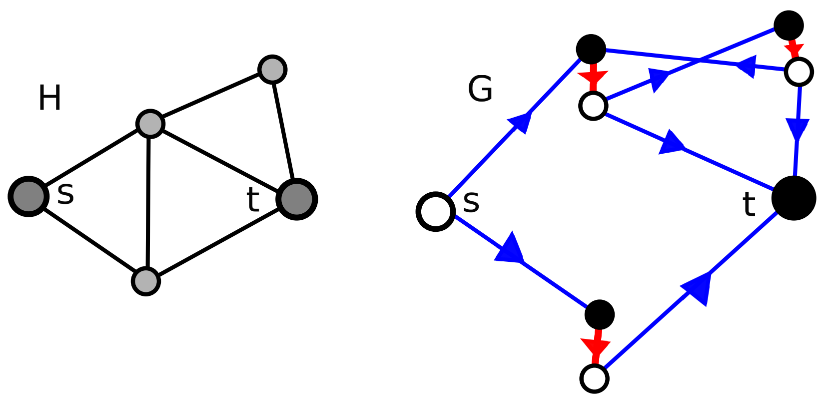

Figure 1: Graph Replacement

Properties of a Min Cut: Let N be the capacity of a minimal (S,T)-cut of G. By definition, N is the number of blue edges pointing from S to T plus the number of red edges pointing from S to T. We are now going to show how to produce another minimum cut which only uses red edges. The strategy is to show that we can increase the number of red edges in case some blue edge exists.

There are 3 kinds of blue edges we need to consider. Suppose that S contains a blue edge pointing from a white vertex $a_+ \in S - \{s\}$ to some $b_- \in T$. Let $S' = S - a_+$ and $T' = T \cup a_+$. We are just letting S "donate" a_+ to T. Since $a_+ \neq s$, the pair (S', T') is still a valid cut. Notice that the blue edge $a_+ \to b_-$ is no longer counted when we compute the size of the (S', T') cut. We may lose some other blue edges as well. Since there is only one edge pointing to a_+, the only edge we gain, potentially, is the red edge $a_- \to a_+$. If (S,T) is really a minimum cut, then so is (S', T'), and this cut has one more red edge.

Suppose that S contains a blue edge connecting s to some $a_- \in T$. Then $a_- \neq t$ because s and t are not adjacent. This time we let $S' = S \cup a_-$ and $T' = T - a_-$. In the (S', T') cut, we lose the blue edge connecting s to a_-, and the only edge we gain, potentially, is the red edge pointing from a_- to a_+. So, (S', T') is a min cut with one more red edge than (S,T).

Finally, suppose that S contais a blue edge connecting $a_+ \in S$ to t. Then $a_+ \neq s$. We let $S' = S - \{a_+\}$ and $T' = T \cup \{a_+\}$. The argument now proceeds in the previous case.

Thus we can assume, without loss of generality, that the only (S,T) edges are red. These red edges canonically define a set of vertices in the original graph H. These vertices in H must be a vertex-cut. Otherwise, we could take an st-path in their complement and look at the corresponding directed path in G. This directed path would go from s_+ to t_- avoiding the red edge associated to the (S, T) cut. But then it would be contained entirely in S, which is a contradiction.

We have proved the following:

$$N = \text{min G cut} \geq \text{min vertex H cut} \tag{3}$$

Properties of a Max Flow: By Max Flow Min Cut there is a maximal integer flow of value N associated to G. As in the edge case, we can write this flow as the sum of unit flow paths and unit flow cycles. There are at least N unit flow paths because the flow has capacity N. But each such path alternates red and blue as it makes its way from s_+ to t_-. Thus path thereby defines an edge path in H which uses the corresponding vertices. The capacity 1 condition precludes the possibility that two such flow paths in G use the same red edge. Hence the corresponding st-paths in H are vertex disjoint. The rest of the proof is just like in the edge case.