Proof of “Flow Theorem”

Vassos Hadzilacos

Let \( \mathcal{F} = (G, s, t, c) \) be a flow network, where \( G = (V, E) \). The theorem below relates the value of an arbitrary flow \( f \) in \( F \) to the traffic on the edges connecting the two components of an arbitrary \((s, t)\)-cut \((S, T)\) of \( F \). This result is key in the proof of correctness of the Ford-Fulkerson algorithm, as we saw in class.

**Notation.** If \( u \in V \), \( \text{out}(u) \) is the set of edges out of \( u \), i.e., \( \text{out}(u) = \{(u, v) : \exists v \in V \text{ such that } (u, v) \in E\} \); similarly, \( \text{in}(u) \) is the set of edges into \( u \), i.e., \( \text{in}(u) = \{(v, u) : \exists v \in V \text{ such that } (v, u) \in E\} \). We generalize this for \( X \subseteq V \) in the obvious way: \( \text{out}(X) \) is the set of edges out of nodes in \( X \), i.e., \( \text{out}(X) = \cup_{u \in X} \text{out}(u) \); similarly, \( \text{in}(X) \) is the set of edges into nodes in \( X \), i.e., \( \text{in}(X) = \cup_{u \in X} \text{in}(u) \). Note that if both endpoints of an edge are in \( X \), then the edge is in \( \text{out}(X) \) as well as in \( \text{in}(X) \). The theorem below states that the value of any flow can be determined by looking at the traffic on the edges that cross any cut \((S, T)\): Add up the traffic on the edges entering \( T \) from \( S \) and subtract the traffic on the edges entering \( S \) from \( T \).

**Flow Theorem.** For any flow \( f \) and any \((s, t)\)-cut \((S, T)\) of the flow network \((G, s, t, c)\),

\[
\mathcal{V}(f) = \sum_{e \in \text{out}(S) \cap \text{in}(T)} f(e) - \sum_{e \in \text{out}(T) \cap \text{in}(S)} f(e).
\]

**Proof.** Let \( f \) be an arbitrary flow and \((S, T)\) be an arbitrary \((s, t)\)-cut of \((G, s, t, c)\). By definition of \( \mathcal{V}(f) \) and the fact that there are no edges into \( s \) we have that \( \mathcal{V}(f) = \sum_{e \in \text{out}(s)} f(e) - \sum_{e \in \text{in}(s)} f(e) \); and for all nodes \( v \in S - \{s\} \) by the conservation property we have that \( \sum_{e \in \text{out}(v)} f(e) - \sum_{e \in \text{in}(v)} f(e) = 0 \). Therefore,

\[
\begin{align*}
\mathcal{V}(f) &= \sum_{v \in S} \left( \sum_{e \in \text{out}(v)} f(e) - \sum_{e \in \text{in}(v)} f(e) \right) \\
&= \sum_{v \in S} \sum_{e \in \text{out}(v)} f(e) - \sum_{v \in S} \sum_{e \in \text{in}(v)} f(e) \quad \text{[rearrange terms]} \\
&= \sum_{e \in \text{out}(S)} f(e) - \sum_{e \in \text{in}(S)} f(e) \quad \text{[def. of out, in]} \\
&= \left( \sum_{e \in \text{out}(S) \cap \text{in}(T)} f(e) + \sum_{e \in \text{out}(S) \cap \text{out}(T)} f(e) \right) - \left( \sum_{e \in \text{in}(S) \cap \text{out}(T)} f(e) + \sum_{e \in \text{in}(S) \cap \text{in}(T)} f(e) \right) \\
&= \sum_{e \in \text{out}(S) \cap \text{in}(T)} f(e) - \sum_{e \in \text{out}(T) \cap \text{in}(S)} f(e).
\end{align*}
\]

In going from the pre-penultimate to the penultimate line in the above derivation, we use the fact that an edge out of a node in \( S \) goes either into a node in \( S \) or to a node in \( T \) but not both (because \( S \) and \( T \) are disjoint); and, similarly, an edge into a node in \( S \) goes out of either a node in \( S \) or a node in \( T \) but not both. Thus, \( \text{out}(S) = (\text{out}(S) \cap \text{in}(S)) \cup (\text{out}(S) \cap \text{in}(T)) \); and, similarly, \( \text{in}(S) = (\text{in}(S) \cap \text{out}(S)) \cup (\text{in}(S) \cap \text{out}(T)) \), where both unions are disjoint. \( \square \)