Math 38 — Graph Theory Nadia Lafrenière

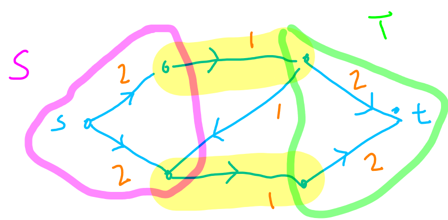
Max-flow min-cut theorem 05/15/2020

Recall from last lecture the following theorem (not yet proved):

Theorem (Max-flow Min-cut, Ford-Fulkerson, 1956)
The value of a maximum flow in a network is the minimum capacity of a source/sink cut.

Our goal for today is to prove it.

Proposition
Let f be a feasible flow. Any source/sink cut has capacity at least value(f).



Capacity 2

Proof
Consider a source/sink cut [S,T].
Recall that the capacity of the cut is the total capacity of the edges from S to T. For each edge e in [S,T], $f(e) \leq c(e)$.
Also, every path from the source to the sink uses edges from S to T; otherwise, it contradicts the fact that [S,T] is a cut.
Finally, the value of a flow on a path cannot be higher than the capacity of any edge of the path. Since every path passes through [S,T], every path has maximum capacity that of the edge(s) it uses from the cut.
Hence, the total capacity of the cut is at least the value of the flow.

We have that, for any flow f and any source-sink cut [S,T]

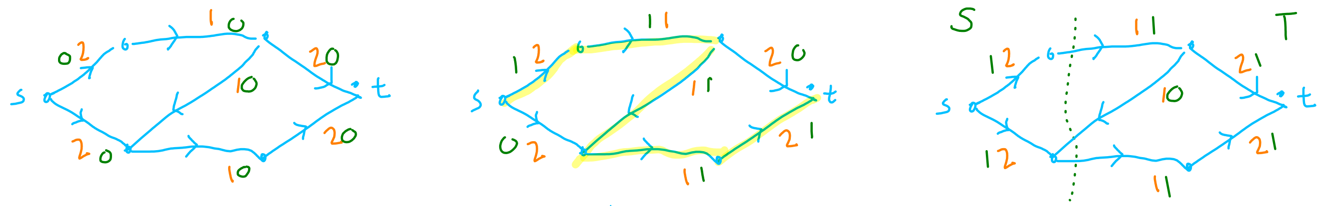
$$\text{value}(f) \leq \text{capacity}([S,T]).$$

That means that if we find a flow with value k and a cut with capacity k in the same network, the flow is necessarily maximum and the cut is necessarily minimal.

Max-flow: Ford-Fulkerson algorithm

Just as in Dijkstra's algorithm, Ford-Fulkerson's algorithm relies on the principle of visiting and reaching vertices. The goal is to find a maximum flow and a minimum cut.

Algorithm to increase the value of a flow, if possible



Value 0 Value 1 Value 2
Maximum flow

Input: A feasible flow f (can be the flow that has 0 on every edge)
Output: An f-augmenting path or a cut with capacity value(f)

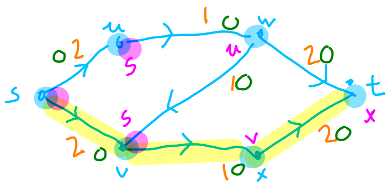
2 sets of vertices that we update: S (source set) and R (reached).
First, $R=\{s\}$ and S is empty.

Iteration: As long as $R \neq S$, and t is not in R.
Choose v in $R-S$.
- For each edge vw with $f(vw)<c(vw)$ and $w \notin R$, add w to R.
  For w, record that you reached it from v.
- For each edge uv with nonzero $f(uv)$ and $u \notin R$, add u to R.
  For u, record that you reached it from v.

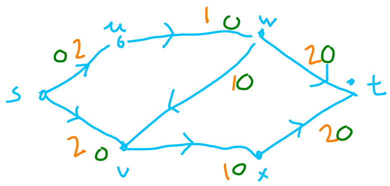
After visiting all the edges incident to v, add v to S.

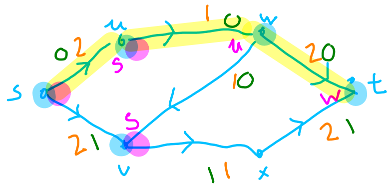
If t is reached, we found an augmenting path. Return it (using the recorded vertices). If $R=S$, there is no possibility of f-augmenting path, so $[S, V-S]$ is a source-sink cut.

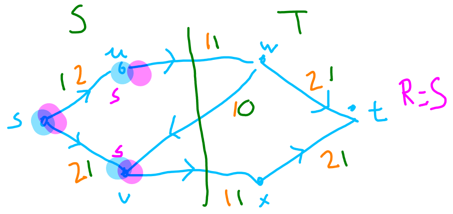
To get a maximum flow:
Start with the zero flow, and use the former algorithm to find an augmenting-path. When you get a cut, the flow is maximal.



- Flow
- Capacity
- Visited vertices (S set)
- Reached vertices (R set)







Max-Flow Min-Cut Theorem

Theorem (Max-flow Min-cut, Ford-Fulkerson, 1956)
The value of a maximum flow in a network is the minimum capacity of a source/sink cut.

Sketch of proof
The key is to use the algorithm, starting with any feasible flow.

- If it terminates with t, we found an augmenting path: all the edges going forward are not used at full capacity, and all the edges going backward have nonzero flow. This means we increase the flow by the tolerance of the augmenting path.
- If it terminates with a cut, the flow is maximal because the edges leaving S are at maximal capacity.

Reference: Douglas B. West. Introduction to graph theory, 2nd edition, 2001. Section 4.3