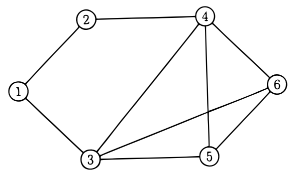
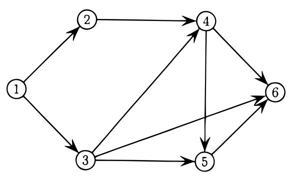
The Ford-Fulkerson Theorem

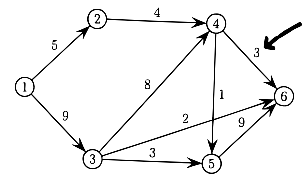
So far, we have been studying "simple graphs" like



By orienting the edges of a simple graph, we get a **directed graph** (**digraph** for short).



We are going to think of these edges as one-way pipes of varying sizes. This gives us a **network**.



the **capacity** of the edge $4 \to 6$.

For conveience, we label the vertices by $[n]$.

Goal: Figure out how much can be pumped from vertex 1 (the "source") to vertex n (the "sink").

Define

$c_{i,j}$ = capacity of the edge from i to j (0 if there is no such edge).

A **flow** is a set of numbers $f_{i,j}$ which tell how much is pumped from i to j. Flows must satisfy three constraints:

1. $f_{i,j} \geq 0$ for all i, j.
2. $f_{i,j} \leq c_{i,j}$ for all i, j.
3. Conservation of flow. For all $i \neq 1, n$,

   Flow in = Flow out

   $$\sum_{k \in [n]} f_{k,i} = \sum_{k \in [n]} f_{i,k}.$$

We want to find the maximum flow, i.e., the flow that maximizes

$$\sum_{k \in [n]} f_{1,k}.$$

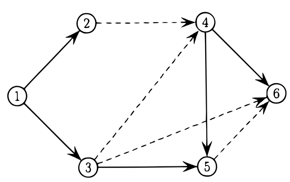
(Note that this is the same as $\sum_{k \in [n]} f_{k,n}$.)

First, what about an upper bound?

Def: A path from u to v is a way to get from u to v following the edges of the graph, without visiting the same vertex twice.

Def: A cut in a network is a set of edges whose removal leaves no path from source (1) to sink (n).

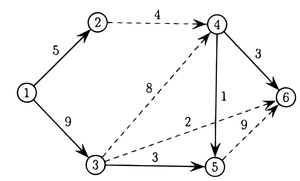
For example:



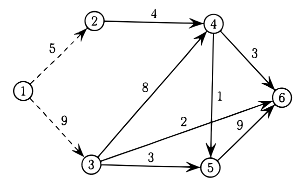
(The dashed edges are a cut.)

Def: The capacity of a cut is the sum of the capacities of every edge in the cut.

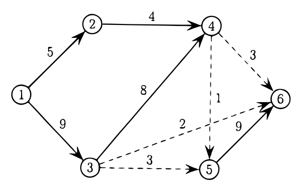
Examples:



Capacity:
4+8+2+9
= 23.



Capacity:
5+9
= 14.



Capacity:
2+1+3+3
= 9.

Clearly no flow can pump more from the source to the sink than the capacity of the minimum cut, so this is our upper bound.

In 1956, Ford and Fulkerson and (independently) Feinstein and Shannon proved that this can be acheived.

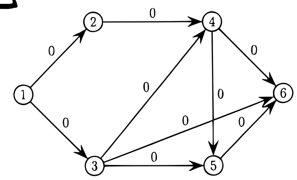
Max-Flow Min-Cut Theorem: In every network, the maximum capacity of a flow equals the minimum capacity of a cut.

The Ford-Fulkerson Algorithm

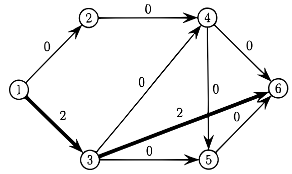
Start with 0 flow.
If possible, find an "augmenting path", a path from the source to the sink with excess capacity, and add this.
When no augmenting paths exist, we are done.

An example of the Ford-Fulkerson Algorithm

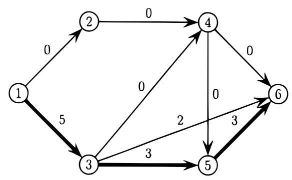
Begin:



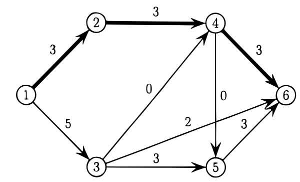
First path:



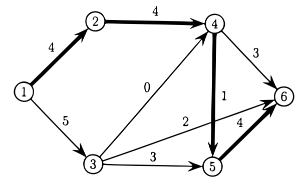
Second path:



Third path:



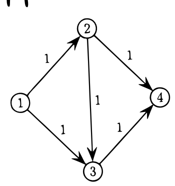
Last path:



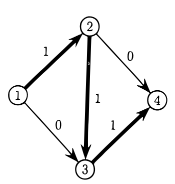
The capacity of this flow is $4+5=9$. Since this equals the capacity of a cut, we know it is best possible.

One more example.

Suppose our network is



And we start out with



Then what should we do next?

We now prove that this algorithm always works.

First, we need to formally define augmenting paths.

Def: The sequence of vertices $1 = v_0, v_1, \ldots, v_k = n$ is an *augmenting path* if, for each $1 \le i \le k$, either

1. The edge is not saturated, $c_{v_{i-1}, v_i} - f_{v_{i-1}, v_i} > 0$, or
2. There is flow in the opposite direction, $f_{v_i, v_{i-1}} > 0$.

What if we can't find an augmenting path?

Let $X = \{ v : v$ can be reached from 1 by an augmenting path $\}$,

Y = all other vertices.

Now if $x \in X$ and $y \in Y$, then

1. If there is an edge from x to y, it must be saturated, $f_{x,y} = c_{x,y}$.
2. If there is an edge from y to x, its flow must be 0, $f_{y,x} = 0$.

Therefore, if we cannot improve our flow, its capacity must equal that of a cut. ■