COMP251: Network flows (2)

Jérôme Waldispühl
School of Computer Science
McGill University

Based on slides from M. Langer (McGill)
Recap Network Flows

\( G = (V, E) \) directed.

Each edge \((u, v)\) has a \textbf{capacity} \(c(u, v) \geq 0\).

If \((u,v) \not\in E\), then \(c(u,v) = 0\).

\textbf{Source} vertex \(s\), \textbf{sink} vertex \(t\), assume \(s \leadsto v \leadsto t\) for all \(v \in V\).

\[\begin{array}{c}
S & 1/3 & 0/1 & 2/2 \\
& 1/3 & 1/1 & 2/3 \\
& & 2/3 & 1/2 \\
& & & 1/2 \\
t
\end{array}\]

\textbf{Problem}: Given \(G, s, t,\) and \(c\), find a flow whose value is maximum.
Application

Maximize flow of supplies in eastern europe!
Recap (residual graphs)

Flow

Residual graph
Recap (Ford-Fulkerson algorithm)

\[ f \leftarrow 0 \]
\[ G_f \leftarrow G \]
while (there is a s-t path in \( G_f \)) do
\[ f \text{.augment}(P) \]
update \( G_f \) based on new \( f \)
Recap graph cuts

A graph cut is a partition of the graph vertices into two sets.

The crossing edges from $S$ to $V-S$ are $\{ (u,v) \mid u \in S, v \in V-S \}$, also called the cut set.
Cuts in flow networks

**Definition:** An s-t cut of a flow network is a cut $A, B$ such that $s \in A$ and $t \in B$.

**Notation:** We write the cut set as $\text{cut}(A, B)$. It is the set of edges from $A$ to $B$.

**Definition:** The capacity of an s-t cut is $\sum_{e \in \text{cut}(A, B)} c(e)$.
Examples

1+5=6

5+3+4=12

1+2=3

2+4=6
Objectives

For any flow network:

- Maximum value of a flow = the minimum capacity of any cut.

- Ford-Fulkerson gives the “max flow” and the “min cut”.
Application

How to cut supplies if cold war turns into real war!

The bottleneck
Flow through a cut

Claim: Given a flow network. Let f be a flow and A, B be a s-t cut. Then,

\[ |f| = \sum_{e \in \text{cut}(A,B)} f(e) - \sum_{e \in \text{cut}(B,A)} f(e) \]

Notation: \( |f| = f^{\text{out}}(A) - f^{\text{in}}(A) \)
Flow through a cut

Proof:
• for any \( u \in V - \{s,t\} \), we have \( f^{out}(u) = f^{in}(u) \).

• Summing over \( u \in A - \{s\} \):
  \[
  \sum_{u \in A - \{s\}} f^{out}(u) = \sum_{u \in A - \{s\}} f^{in}(u)
  \]

• \( |f| = f^{out}(s) = \sum_{u \in A} f^{out}(u) - \sum_{u \in A} f^{in}(u) \)

• Each edge \( e = (u,v) \) with \( u,v \in A \) contributes to both sums, and can be removed (Note: \( f^{in}(s) = 0 \)).

\[
|f| = \sum_{e \in \text{cut}(A,B)} f(e) - \sum_{e \in \text{cut}(B,A)} f(e) \\
\equiv f^{out}(A) - f^{in}(A)
\]
Upper bound on flow through cuts

**Claim:** For any network flow $f$, and any s-t cut$(A,B)$

$$|f| \leq \sum_{e \in \text{cut}(A,B)} c(e)$$

**Proof:**

$$|f| = f^{\text{out}}(A) - f^{\text{in}}(A)$$

$$\leq f^{\text{out}}(A)$$

$$\leq \sum_{e \in \text{cut}(A,B)} c(e)$$
Observations

• Some cuts have greater capacities than others.
• Some flows are greater than others.
• **But, every flow must be ≤ capacity of every s-t cut.**
• Thus, the value of the maximum flow is less than capacity of the minimum cut.
Value of flow in Ford-Fulkerson

- Ford-Fulkerson terminates when there is no augmenting path in the residual graph $G_f$.
- Let $A$ be the set of vertices reachable from $s$ in $G_f$, and $B=V-A$.
- $A,B$ is a s-t cut in $G_f$.
- $A,B$ is an s-t cut in $G$ ($G$ and $G_f$ have the same vertices).
- $|f|=f^{out}(A)-f^{in}(A)$
- We want to show: $|f| = \sum_{e \in \text{cut}(A,B)} c(e)$
- And in particular:

\[1\] $f^{out}(A) = \sum_{e \in \text{cut}(A,B)} c(e)$
\[2\] $f^{in}(A) = 0$
Value of flow in Ford-Fulkerson

(1) For any $e=(u,v) \in \text{cut}(A,B)$, $f(e)=c(e)$.
   
   • $f(e)<c(e) \Rightarrow e=(u,v)$ would be a forward edge in the residual graph $G_f$ with capacity $c_f(e)=c(e)-f(e)>0$.
   
   • $v$ reachable from $s$ in $G_f$ $\Rightarrow$ contradiction. $

(2) f^{\text{in}}(A)=0$: $\forall$ $e=(v,u) \in E$ such that $v \in B$, $u \in A$, we have $f(e)=0$.
   
   • $f(e)>0 \Rightarrow \exists$ backward edge $(u,v)$ in $G_f$ such that $c_f(e)=f(e)$
   
   • $v$ is reachable from $s$ in $G_f$ $\Rightarrow$ contradiction. $

Max flow = Min cut

- Ford-Fulkerson terminates when there is no path s-t in the residual graph $G_f$
- This defines a cut in $A,B$ in $G$ ($A =$ nodes reachable from $s$)
- $|f| = f^{out}(A) - f^{in}(A)$
  $$= \sum_{e \in cut(A,B)} f(e) - \sum_{e \in cut(B,A)} f(e)$$
- Ford-Fulkerson flow = $\sum_{e \in cut(A,B)} c(e) - 0$
  $$= \text{capacity of cut}(A,B)$$

**Note:** We did not proved uniqueness.
Computing the min cut

Q: Given a flow network, how can we compute a minimum cut?

Answer:

• Run Ford-Fulkerson to compute a maximum flow (it gives us $G_f$)
• Run BFS or DFS of s.
• The reachable vertices define the set $A$ for the cut
Bipartite matching

Suppose we have an undirected graph bipartite graph \( G = (V, E) \).

Q: How can we find the maximal matching? (Recall Lecture 11)
Bipartite matching with network flows

Define a flow network $G'=(V',E')$ such that:

- $V' = V \cup \{s,t\}$
- $E' = \{ (u,v) \mid u \in A, v \in B, (u,v) \in E \} \cup \{ (s,u) \mid u \in A \} \cup \{ (v,t) \mid v \in B \}$
- Capacities of every edge = 1.

Motivation: Max flow $\Rightarrow$ max matching.
Max flow in bipartite graphs

Exercise: The maximal flow found by Ford-Fulkerson defines a maximal matching in the original graph G (the maximal set of edges \((u,v)\) \(u \in A \& v \in B\) such that \(f(u,v)=1\).
Max matching with Ford-Fulkerson

Ford-Fulkerson will find an augmenting path with $\beta=1$ at each iteration. They are of the form:

Or have more than one zig-zag.

Note:
- No edge from B to A in $E'$. The back edges are in the residual graph.
- Edges $e$ such that $c(e)=0$ are not shown.
Q: How long will it take to find a maximal matching with Ford-Fulkerson?

- The general complexity of Ford-Fulkerson is \(O(C \cdot |E|)\), where
  \[
  C = \sum_{u} c(s,u)
  \]
- Suppose \(|A|=|B|=n\)
- Then, \(C=|A|=n\) and \(|E'|=|E|+2n=m+2n\) (Assume \(m>n\))
- Thus, \(C \cdot |E'| = n \cdot (m+2n)\)
- Running time is \(O(nm)\)
Example

What is max flow? What is min cut?
Example

What is max flow? What is min cut?

Max flow $|f|=2$.
There are other flows having $|f|=2$.
What is the minimum cut?
Example

Find any min cut with capacity 2.

Not a cut!

Not a min cut!

Not a cut!

min cut!
Example

To find a min cut compute a max flow.

Flow

Residual graph

backward forward
To find the cut run BFS (or DFS) from \( s \) on the residual graph. The reachable vertices define the (min) cut.