Max Flow Min Cut, Bipartite Matching

Yin Tat Lee
Residual Graph

Original edge: $e = (u, v) \in E$.
- Flow $f(e)$, capacity $c(e)$.

Residual edge.
- "Undo" flow sent.
- $e = (u, v)$ and $e^R = (v, u)$.

Residual capacity:

$$c_f(e) = \begin{cases} 
  c(e) - f(e) & \text{if } e \in E \\
  f(e) & \text{if } e^R \in E
\end{cases}$$

Residual graph: $G_f = (V, E_f)$.
- Residual edges with positive residual capacity.
- $E_f = \{ e : f(e) < c(e) \} \cup \{ e^R : f(e) > 0 \}$. 
Augmenting Path Algorithm

Augment\((f, c, P)\) {
    \(b \leftarrow \text{bottleneck}(P)\)
    \(\text{foreach } e \in P \{\)
        \(\text{if } (e \in E) f(e) \leftarrow f(e) + b\)
        \(\text{else } f(e^R) \leftarrow f(e) - b\)
    \}\n    \(\text{return } f\)
}

Ford-Fulkerson\((G, s, t, c)\) {
    \(\text{foreach } e \in E \ f(e) \leftarrow 0\)
    \(G_\pi \leftarrow \text{residual graph}\)
    \(\text{while } (\text{there exists augmenting path } P) \{\)
        \(f \leftarrow \text{Augment}(f, c, P)\)
        \(\text{update } G_\pi\)
    \}\n    \(\text{return } f\)
}
Max Flow Min Cut Theorem

**Augmenting path theorem.** Flow $f$ is a max flow iff there are no augmenting paths.

**Max-flow min-cut theorem.** [Ford-Fulkerson 1956] The value of the max s-t flow is equal to the value of the min s-t cut.

**Proof strategy.** We prove both simultaneously by showing the TFAE:

(i) There exists a cut $(A, B)$ such that $v(f) = \text{cap}(A, B)$.
(ii) Flow $f$ is a max flow.
(iii) There is no augmenting path relative to $f$.

(i) $\Rightarrow$ (ii) This was the corollary to weak duality lemma.

(ii) $\Rightarrow$ (iii) We show contrapositive.

Let $f$ be a flow. If there exists an augmenting path, then we can improve $f$ by sending flow along that path.
(iii) => (i)
No augmenting path for f => there is a cut (A,B): \( v(f) = \operatorname{cap}(A,B) \)

- Let f be a flow with no augmenting paths.
- Let A be set of vertices reachable from s in residual graph.
- By definition of A, s \( \in \) A.
- By definition of f, t \( \notin \) A.

\[
v(f) = \sum_{e \text{ out of } A} f(e) - \sum_{e \text{ in to } A} f(e)
\]

\[
= \sum_{e \text{ out of } A} c(e)
\]

\[
= \operatorname{cap}(A,B)
\]
Running Time

Assumption. All capacities are integers between 1 and C.

Invariant. Every flow value $f(e)$ and every residual capacities $c_f(e)$ remains an integer throughout the algorithm.

Theorem. The algorithm terminates in at most $v(f^*) \leq nC$ iterations, if $f^*$ is optimal flow.

Pf. Each augmentation increase value by at least 1.

Corollary. If $C = 1$, Ford-Fulkerson runs in $O(mn)$ time.

Integrality theorem. If all capacities are integers, then there exists a max flow $f$ for which every flow value $f(e)$ is an integer.

Pf. Since algorithm terminates, theorem follows from invariant.
Applications of Max Flow: Bipartite Matching
Maximum Matching Problem

Given an undirected graph $G = (V, E)$. A set $M \subseteq E$ is a matching if each node appears in at most edge in $M$.

Goal: find a matching with largest cardinality.
Bipartite Matching Problem

Given an undirected bipartite graph $G = (X \cup Y, E)$ A set $M \subseteq E$ is a matching if each node appears in at most edge in $M$.
Goal: find a matching with largest cardinality.
Create digraph $H$ as follows:

- Orient all edges from $X$ to $Y$, and assign infinite (or unit) capacity.
- Add source $s$, and unit capacity edges from $s$ to each node in $L$.
- Add sink $t$, and unit capacity edges from each node in $R$ to $t$. 

In some proof

In algorithm
Bipartite Matching: Proof of Correctness

**Thm.** Max cardinality matching in $G = \text{value of max flow in } H$.

**Pf. $\leq$**

Given max matching $M$ of cardinality $k$.

Consider flow $f$ that sends 1 unit along each of $k$ edges of $M$. $f$ is a flow, and has cardinality $k$. □
Bipartite Matching: Proof of Correctness

**Thm.** Max cardinality matching in $G = \text{value of max flow in } H$.

**Pf. (of $\geq$)** Let $f$ be a max flow in $H$ of value $k$. 
Integrality theorem $\Rightarrow k$ is integral and we can assume $f$ is 0-1.
Consider $M = \text{set of edges from } X \text{ to } Y \text{ with } f(e) = 1$.
- each node in $X$ and $Y$ participates in at most one edge in $M$
- $|M| = k$
Perfect Bipartite Matching
Perfect Bipartite Matching

Def. A matching $M \subseteq E$ is **perfect** if each node appears in exactly one edge in $M$.

Q. When does a bipartite graph have a perfect matching?

Structure of bipartite graphs with perfect matchings:

- Clearly we must have $|X| = |Y|$.
- What other conditions are necessary?
- What conditions are sufficient?
Def. Let $S$ be a subset of nodes, and let $N(S)$ be the set of nodes adjacent to nodes in $S$.

Observation. If a bipartite graph $G$ has a perfect matching, then $|N(S)| \geq |S|$ for all subsets $S \subseteq X$.

Pf. Each $v \in S$ has to be matched to a unique node in $N(S)$. 
Marriage Theorem

Thm: [Frobenius 1917, Hall 1935] Let $G = (X \cup Y, E)$ be a bipartite graph with $|X| = |Y|$.
Then, $G$ has a perfect matching iff $|N(S)| \geq |S|$ for all subsets $S \subseteq X$.

Pf. $\Rightarrow$
This was the previous observation.
If $|N(S)| < |S|$ for some $S$, then there is no perfect matching.
Marriage Theorem

Pf. \( \exists S \subseteq X \text{ s.t., } |N(S)| < |S| \iff G \text{ does not a perfect matching} \)

Formulate as a max-flow and let \((A, B)\) be the min s-t cut

G has no perfect matching \( \Rightarrow \) \( v(f^*) < |X| \). So, \( cap(A, B) < |X| \)

Define \( X_A = X \cap A, X_B = X \cap B, Y_A = Y \cap A \)

Then, \( cap(A, B) \geq |X_B| + |Y_A| \)

Since min-cut does not use \( \infty \) edges, \( N(X_A) \subseteq Y_A \)

\( |N(X_A)| \leq |Y_A| \leq cap(A, B) - |X_B| = cap(A, B) - |X| + |X_A| < |X_A| \)