Outline: Network Flow (aka "Max Flow") [Chapter 26]

- Definitions & notation
  - Flow network
  - Max flow problem
  - Residual network
  - Net flows
  - Cuts
- Max-flow min-cut theorem
  - Ford-Fulkerson alg
  - Edmonds-Karp alg

Def: A flow network is a directed graph \( G = (V, E) \)

with two distinguished vertices: a source \( S \) and a sink \( T \),

and a nonnegative capacity \( c(u, v) \) for each edge \( (u, v) \in E \).

[Assume \( c(u, v) = 0 \) if \( (u, v) \notin E \).] [Assume no "self-loops".]

Example:

```
S       A       B       C       T
\( 1 \) --\( 2 \) ----\( 4 \) --\( 2 \) ----\( 1 \)
\( 4 \) --\( \quad \) --\( 4 \)
        \quad     \quad     \quad
```

"Rate of flow"

\( c(u, v) \) = capacity of edge in units of stuff/unit of time

- cars/hour, amps, gallons/sec

[Want to maximize steady-state flow of stuff from \( S \) to \( T \)

that: (a) obeys capacity constraints, and

(b) has flow-in = flow-out at intermediate nodes (conservation).

[Like Kirchhoff's current law]

Def: A positive flow on \( G \) is a function \( p : V \times V \rightarrow \mathbb{R} \)

s.t.

- [Capacity constraint] \( 0 \leq p(u, v) \leq c(u, v) \) (\( \forall u, v \in V \))
- [Flow conservation] \( (\forall u \in V \setminus \{S, T\}) \)

\[ \sum_{v \in V} p(u, v) - \sum_{v \in V} p(v, u) = 0 \]

(flow out) (flow in)
**Def:** The value of a flow is net flow out of source:
\[ |p| = \sum_{v \in V} p(s,v) - \sum_{v \in V} p(v,s) \]

**Def:** Maximum-flow problem: Given a flow network, find a flow of maximum value.

**Example:** (cont)

```
\[ \text{Notation:} \]
\[ p(u,v)/c(u,v) \]
\[ \not\text{ division!} \]
\[ \text{1 is separator} \]
```

**Important problem!**

Many other problems are special cases, e.g.

- matching  
  (assignment)

```
\[ \text{find max \# of} \]
\[ \text{disjoint compatible pairs} \]
\[ BD, CE, AF \]
\[ \text{[hint at method: adding s \& t]} \]
```

**Def:** notations: net flow (instead of positive flow)

**Summation notation**

**Define:** net flow function \( F \) is net flow between vertices:

\[ f(u,v) = p(u,v) - p(v,u) \]

**Fact:** (Skew-symmetry) \( f(u,v) = -f(v,u) \)
Example:

\[
\begin{array}{c}
\begin{bmatrix}
\text{U} & \text{V}
\end{bmatrix}
\end{array}
\]

Normally either \( p(u,v) = 0 \) or \( p(v,u) = 0 \) (don't cycle stuff)

\[ f(u,v) = 0 \Rightarrow \begin{cases} p(u,v) = f(u,v) \\ p(v,u) = 0 \\ f(u,v) \leq c(u,v) \end{cases} \]

\[ f(u,v) \leq 0 \Rightarrow \begin{cases} p(v,u) = -f(u,v) = f(v,u) \\ p(v,u) = 0 \\ f(v,u) = -f(u,v) \leq c(v,u) \end{cases} \]

Enough to work with \( f \) only (not \( p \)), and may have constraints:

- \( f(u,v) = -f(v,u) \) \( \quad (\forall u,v) \) \[ \text{skew-symmetric} \]
- \( f(u,v) \leq c(u,v) \) \( \quad (\forall u,v) \) \[ \text{capacity} \]

For conservation use summation notation:

\[
f(u,X) = \sum_{v \in X} f(u,v)
\]

\[ f(X,v) = \sum_{u \in X} f(u,v) \quad X \subseteq V \]

\[ f(X,Y) = \sum_{u \in X} \sum_{v \in Y} f(u,v) \quad X, Y \subseteq V \]

Flow conservation \( \equiv (\forall u \in V - \{s,t\}) f(u,V) = 0 \)

\( |f| = \text{value of flow } f \)

\[ = \text{net flow out of } s \]

\[ = f(s,V) \]
Facts:
1. \( f(x, x) = 0 \) for all \( x \in V \)  
2. \( f(x, y) = -f(y, x) \) for all \( x, y \in V \) 
3. \( f(x \cup y, z) = f(x, z) + f(y, z) \) if \( X \cap Y = \emptyset \)

[by s.s. \( f(u, v) = -f(v, u) \)]

Theorem: \( |f| = f(V, t) \)

"Net flow out at s" = "net flow into t"

Proof:
\[
|f| = f(\{s\}, V) = f(V, V) - f(V - s, V) = f(V, V - s) = f(V, t) + f(V, V - s - t) = f(V, t)
\]

[Note: omit braces]

[Check on example flow network]

- We now introduce notion of a cut, so we can state max-flow/min-cut theorem, which allows us to know when we have a max flow.

- **Def.** A cut is a partition \((S, T)\) of \( V \)
  
  \[ \text{i.e. } S \cup T = V \text{ and } S \cap T = \emptyset \]

  s.t. \( s \in S \) and \( t \in T \).

- **Def.** If \( f \) is a flow, then flow across cut \((S, T)\) is \( f(S, T) \).

Example:

\[
\begin{align*}
S &= \{s, b\} \\
T &= \{a, c, t\} \\
f(S, T) &= 3 + 3 - 2 = 4
\end{align*}
\]
Theorem:  \(|f| = f(S,T)\) for any cut \((S,T)\).

Proof: (really rather intuitive: has to cross cut to get from \(s\) to \(t\).)

\[f(S,T) = f(S,V) - f(S,S)\]
\[= f(S,V)\]
\[= f(s,V) + f(S-s,V)\]
\[= f(s,V)\quad \text{[flow conservation } \forall u \in V - \{s,t\} \text{]}\]
\[= |f|\]

Def: The capacity of a cut is \(c(S,T)\).

Example: cut has value \(y + 2 + y = 10\).

Theorem: For any cut \((S,T)\), \(f(S,T) \leq c(S,T)\).

Proof: \(|f| = f(S,T)\)
\[= \sum_{u \in S} \sum_{v \in T} f(u,v)\]
\[\leq \sum_{u \in S} \sum_{v \in T} c(u,v)\]
\[= c(S,T)\]

[Look at other cuts in example.]

[Note: \(c\) is not skew-symmetric!]
So, \( f \) must satisfy various constraints

\[ |f| \leq c(S, T) \]

one for each cut. Tightest such constraint is achievable:

**Theorem:** [Max-flow/min-cut]

\[
\max |f| = \min \text{ cuts } (S, T) c(S, T)
\]

**Idea [details next time & proof]:**

Given \( f \), can explore edges with excess capacity

using BFS from \( s \).

- If we reach \( t \), \( |f| \) is not maximum
- If we can't reach \( t \), then let

\[
S = \text{reachable vertices} \\
T = V - S
\]

Assert \( |f| = c(S, T) \). (every \( S-T \) edge at capacity)

**Example:** show \( S = \{ s, A, C \} \quad T = \{ B, t \} \)

\[ |f| = f(S, T) = c(S, T) = 4. \]
Outline:
- Review
  - Residual graph, residual capacities, augmenting path
    - Max-flow/min-cut theorem
    - Ford-Fulkerson algorithm $\Theta(|E| \cdot |f^*|)$ [integral capacities]
    - Edmonds-Karp $O(VE^2)$

Review:
- Flow network: $G = (V, E)$, source $s$, sink $t$, capacities $c(u, v)$
- Net flow $f(u,v)$: $f(u,v) = c(u,v) - |f| = f(s,v)$
- Cut $(S,T)$: $S \cup T = V$, $S \cap T = \emptyset$, $s \in S$, $t \in T$
- Theorem: $|f| = f(S,T)$ for any cut $(S,T)$
- Theorem: $|f| \leq \Phi(S,T)$ for any cut $(S,T)$

Given: Flow network with flow $f$, how to augment (increase) flow?
Answer: Find $s \to t$ path in residual graph $G_f$

$\begin{array}{c}
\text{Residual graph } G_f = (V, E_f) \text{ with edges having} \\
\text{strictly positive residual capacities} \\
c_f(u,v) = c(u,v) - f(u,v) > 0 \\
\end{array}$

$\begin{array}{c}
G_f \\
\begin{array}{c}
\text{Lemma: } |E_f| \leq 2|E| \\
\end{array}
\end{array}$
**Def:** Any path from $s$ to $t$ in $G_f$ is an **augmenting path** in $G$ wrt $f$.

Flow value can be increased along augmenting path $\rho$ by

$$C_f(\rho) = \min_{(u,v) \in \rho} \{ C_f(u,v) \}$$

**Example:**

$G = \begin{array}{c}
  \textcircled{5} \rightarrow \textcircled{u} \rightarrow \textcircled{v} \rightarrow \textcircled{w} \rightarrow \textcircled{x} \rightarrow \textcircled{t} \\
\end{array}$

$G_f = \begin{array}{c}
  \textcircled{5} \rightarrow \textcircled{4} \rightarrow \textcircled{2} \rightarrow \textcircled{1} \rightarrow \textcircled{3} \rightarrow \textcircled{t} \\
\end{array}$

$\rho = s \xrightarrow{3/5} u \xrightarrow{2/4} v \xrightarrow{5/5} w \xrightarrow{2/3} x \xrightarrow{2/4} t$

$C_f(\rho) = 2$

$G' = \begin{array}{c}
  \textcircled{5/5} \xrightarrow{4/6} \textcircled{4/6} \rightarrow \textcircled{0/2/3} \rightarrow \textcircled{4/4} \rightarrow \textcircled{t} \\
\end{array}$

(after augmenting $b = 2 \left( \frac{1}{6} \right)$)
Max-flow/min-cut theorem:

The following are equivalent:

1. \(|f| = c(S, T)| \) for some cut \((S, T)\)
2. \(f\) is a max flow
3. \(G_f\) has no augmenting paths.

Proof:

1. \(\Rightarrow 2\): Since \(|f| \leq c(S, T)| \) by theorem, \(f\) is a max flow.

2. \(\Rightarrow 3\): (\(\equiv -3 \Rightarrow -2\))
   
   If \(G_f\) has an augmenting path, \(f\) is not a max flow.

3. \(\Rightarrow 1\): Suppose \(f\) has no augmenting paths.

   Let \(S = \{v \in V: \exists \text{ path in } G_f \text{ from } s \text{ to } v\}\)
   
   \(T = V - S\)

   Note: \(s \in S, t \in T\) so \((S, T)\) is a cut

   Consider \(u \in S, v \in T:\)

   \(\Rightarrow\)

   Must have \(c_f(u, v) = 0\) (otherwise \(v \in S\))

   \(\therefore f(u, v) = c(u, v) \quad [\text{since } c_f(u, v) = c(u, v) - f(u, v)]\)

   Sum over all \(u \in S, v \in T\)

   \(\Rightarrow f(S, T) = c(S, T)\)

   \(|f| \quad \square\)
**Ford-Fulkerson max-flow alg.**

\[
\begin{align*}
\text{while } & \exists \text{ augmenting path } p : \\
& \text{ augment } f \text{ along } p \text{ by } \Delta f(p)
\end{align*}
\]

Can be slow:

- FF algorithm with integral capacities runs in time \( O(E \cdot |f^*|) \)
  where \( |f^*| \) is optimal flow.

\[\text{Theorem: If capacities are integral, then max flow } \uparrow \text{ with integral flow on each edge.}\]

\[\text{PG: FF alg.}\]

**Application:** e.g., matching

- Can we fix dependence on \(|f^*|\)??

- Yes: find augmenting paths with BFS, \(\text{(i.e., use a shortest aug. path)}\)
  \(\text{with fewest edges)}\)

\[\Rightarrow \text{Edmonds-Karp alg. (EK)} \quad \text{[actually, due to Dinic: their analysis]}\]
Def: Let $\delta_f(s,v) = \text{distance (\# edges) from } s \text{ to } v \text{ in } G_f$.

(Monotonicity)

Lemma: Let $\delta(v) = \delta_f(s,v)$ during EK algorithm. (Changes as we augment flow, of course.) Then $\delta(v)$ never decreases.

Proof:

Case 1: $f$ is flow, and augmented flow is $f'$.
Let $\delta'(v) = \delta_f(s,v)$, $\delta(v) = \delta_f(s,v)$.
Prove $\delta'(v) \geq \delta(v)$ by induction on $\delta'(v)$.

Base case: $\delta'(v) = 0 \Rightarrow v = s \Rightarrow \delta(s) = 0 \Rightarrow \delta'(v) \geq \delta(v)$.

Inductive case:
Consider BFS path $s \rightarrow u \rightarrow v$ in $G_{f'}$.
Must have $\delta'(v) = \delta'(u) + 1$ and so $\delta'(u) = \delta(u)$ by induction.

Certainly $(u,v) \in E_{f'}$.

Case 1: $(u,v) \in E_f$:
$\delta(v) \leq \delta(u) + 1$ (A-ineq)
$\leq \delta'(u) + 1$ (induction)
$= \delta'(v)$ (BFS)

Case 2: $(u,v) \notin E_f$:

$G_f$ (after):

$G_f$ (before):

How? Any path must have $(v,u) \in E_f$ where $p$ is BF path in $G_f$.

$G_f: p = \text{add arrow} \text{ (.stem)}$
\[
\delta(v) = \delta(u) - 1 \quad \text{(BFS)}
\]
\[
\leq \delta'(u) - 1 \quad \text{(induction on } \delta', \text{ recall.)}
\]
\[
= \delta'(v) - 2 \quad \text{(BF path)}
\]
\[
< \delta'(v) \quad \blacksquare
\]

**Theorem:** In EK, # flow augmentations is \( \Theta(VE) \).

**Proof:** Suppose \( p \) is aug path, & \( \zeta_{G_f}(u,v) = \zeta_{G_f}(p) \) for some \((u,v)\)\(\in\)p.

Then \((u,v)\) is critical and disappears from res. graph after aug.

**Ex:**

![Graph](image)

First time \((u,v)\) is critical, \( \delta(u) = \delta(u) + 1 \) since \( p \) is BF.

Must wait until \((v,u)\) is on new aug path before \((u,v)\) can be critical again.

Let \( \delta' \) be dist. func. when \((v,u)\) on aug. path. Then

\[
\delta'(u) = \delta'(v) + 1 \quad \text{(BF)} \quad \text{in } G_f,
\]
\[
\geq \delta(v) + 1 \quad \text{(monotonic)}
\]
\[
= \delta(u) + 2 \quad \text{(BF)} \quad \text{in } G_f
\]

\( \Rightarrow \) Each time \((u,v)\) is critical, \( \delta(u) \) has increased by \( \geq 2 \).

\( \Rightarrow \) \((u,v)\) is critical \( \Theta(V) \) times, since \( 0 \leq \delta(u) \leq |V| - 1 \).

Since residual graph has \( \Theta(E) \) edges,

\[
\# \text{augs} \leq \Theta(VE) \quad \blacksquare
\]

**Thm:** EK alg runs in time \( \Theta(VE^2) \)

**PF:** BFS & other ops take time \( \Theta(E) \) per augmentation.

**Other alg:** \( \Theta(V^3) \) Goldberg push/relabel (in book)

\( \Theta(VE \log \frac{E}{V}) \) King-Rao-Tarjan.