CSE 421
Introduction to Algorithms
Winter 2000

The Network Flow Problem

How much stuff can flow from s to t?

Net Flow: Formal Definition

Given:
- A digraph G = (V,E)
- Two vertices s,t in V (source & sink)
- A capacity \( c(u,v) \geq 0 \) for each \((u,v) \in E\)
  (and \( c(u,v) = 0 \) for all non-edges \((u,v)\))

Find:
- A flow function \( f: V \times V \rightarrow \mathbb{R} \)
  s.t.,
  \[ f(u,v) \leq c(u,v) \] [Capacity Constraint]
  \[ f(u,v) = -f(v,u) \] [Skew Symmetry]
  \[ \forall u \neq s,t, f(u,V) = 0 \] [Flow Conservation]

Maximizing total flow \( |f| = f(s,V) \)

Notation:
\[ f(x,y) = \sum_{x \in X} \sum_{y \in Y} f(x,y) \]

Example: A Flow Function

\[ f(s,u) = f(u,t) = 2 \]
\[ f(u,b) = f(b,u) = -2 \]

\[ f(s,V) = \sum_{(u,v) \in V} f(u,v) = f(s,u) + f(u,s) + f(u,t) = -2 + 2 = 0 \]

Example: A Flow Function

- Not shown: \( f(u,v) \) if \( \leq 0 \)
- Note: max flow \( \geq 4 \) since \( f \) is a flow function, with \( |f| = 4 \)

Max Flow via a Greedy Alg?

While there is an s \( \rightarrow \) t path in G
- Pick such a path, p
- Find c, the min capacity of any edge in p
- Subtract c from all capacities on p
- Delete edges of capacity 0
- This does NOT always find a max flow:

If pick s \( \rightarrow b \) \( \rightarrow a \) \( \rightarrow t \) first, flow stuck at 2.
But flow 3 possible.
A Brief History of Flow

- $n =$ # of vertices
- $m =$ # of edges
- $U =$ Max capacity

Source: Goldberg & Rao, FOCS '97

Greed Revisited

Residual Capacity

- The residual capacity (w.r.t. $f$) of $(u,v)$ is $c_f(u,v) = c(u,v) - f(u,v)$

- e.g. $c_f(s,b)=7$; $c_f(a,x) = 1$; $c_f(x,a) = 3$

Residual Networks & Augmenting Paths

- The residual network (w.r.t. $f$) is the graph $G_f = (V,E_f)$, where $E_f = \{ (u,v) | c_f(u,v) > 0 \}$

- An augmenting path (w.r.t. $f$) is a simple $s \to t$ path in $G_f$.

A Residual Network

An Augmenting Path
Lemma 1

If $f$ admits an augmenting path $p$, then $f$ is not maximal.

Proof: “obvious” -- augment along $p$ by $c_p$, the min residual capacity of $p$'s edges.

Augmenting A Flow

Ford-Fulkerson Method

While $G_f$ has an augmenting path, augment

- Questions:
  - Does it halt?
  - Does it find a maximum flow?
  - How fast?

Cuts

- A partition $S,T$ of $V$ is a cut if $s \in S, t \in T$
- Capacity of cut $S,T$ is $c(S,T) = \sum_{u \in S, v \in T} c(u,v)$
**Lemma 2**

- For any flow \( f \) and any cut \( S,T \),
  - the net flow across the cut equals the total flow, i.e., \( |f| = f(S,T) \), and
  - the net flow across the cut cannot exceed the capacity of the cut, i.e. \( f(S,T) \leq c(S,T) \)

**Corollary:**

Max flow \( \leq \) Min cut

<table>
<thead>
<tr>
<th>Cut Cap</th>
<th>Net Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**Max Flow / Min Cut Theorem**

For any flow \( f \), the following are equivalent

1. \( |f| = c(S,T) \) for some cut \( S,T \) (a min cut)
2. \( f \) is a maximum flow
3. \( f \) admits no augmenting path

Proof:

1. \( \Rightarrow \) 2: corollary to lemma 2
2. \( \Rightarrow \) 3: lemma 1

**Max Flow / Min Cut Theorem**

(3) \( \Rightarrow \) (1)

- \( S = \{ u \mid \exists \) an augmenting path from \( s \) to \( u \} \)
- \( T = V - S; \ s \in S, t \in T \)
- For any \((u,v)\) in \( S \times T, \exists \) an augmenting path from \( s \) to \( u \), but not to \( v \).
  - \((u,v)\) has 0 residual capacity:
    - \((u,v)\) in \( E \) \( \Rightarrow \) saturated
    - \((v,u)\) in \( E \) \( \Rightarrow \) no flow
  - \( f(u,v) = f(v,u) = 0 \)

This is true for every edge crossing the cut, i.e.

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

**Edmonds-Karp Algorithm**

- Use a shortest augmenting path
  (via Breadth First Search in residual graph)
- Time: \( O(n m^2) \)

**BFS/Shortest Path Lemmas**

Distance from \( s \) is never reduced by:

- Deleting an edge
  - proof: no new (hence no shorter) path created
- Adding an edge \((u,v)\), provided \( v \) is nearer than \( u \)
  - proof: BFS is unchanged, since \( v \) visited before \((u,v)\) examined
Lemma 27.8 (Alternate Proof)

Let \( f \) be a flow, \( G_f \) the residual graph, and \( p \) a shortest augmenting path. Then no vertex is closer to \( s \) after augmentation along \( p \).

Proof: Augmentation only deletes edges, adds back edges.

Augmentation vs BFS

Theorem 27.9

The Edmonds-Karp Algorithm performs \( O(mn) \) flow augmentations.

Proof: \( \{u,v\} \) is critical on augmenting path \( p \) if it’s closest to \( s \) having min residual capacity won’t be critical again until farther from \( s \) so each edge critical at most \( n \) times.

Corollary

- Edmonds-Karp runs in \( O(nm^2) \)

Flow Integrality Theorem

If all capacities are integers
  - The max flow has an integer value
  - Ford-Fulkerson method finds a max flow in which \( f(u,v) \) is an integer for all edges \( (u,v) \)