Max Flow

- In optimal paths, every arc has a length, cost, or value.
- Now, every arc has a width.

\[ b_{i,j} \] is the amount of flow that can go from \( v_i \) to \( v_j \).
Max Flow
\[ \text{max } \nu = ? \]

\[ \sum_{i} x_{i,j} - \sum_{k} x_{j,k} = \begin{cases} 
+\nu & j = s \\
0 & j \neq s, t \\
-\nu & j = t 
\end{cases} \]

\[ 0 \leq x_{i,k} \leq b_{i,k} \]
Max flow algorithm

Capacity of a cut:

\[ C \left[ X, \overline{X} \right] = \sum_{i \in X, j \in \overline{X}} b_{i,j} \]
Labeling Process

Step 0. \( v_s \in X \)
Step 1. If \( v_i \in X \) and \( x_{i,j} < b_{i,j} \) then \( v_j \in X \)
Step 2. If \( v_i \in X \) and \( x_{j,i} > 0 \) then \( v_j \in X \)
Max Flow

Graph 1:
- Source: S
- Sink: T
- Edges: (S, a) = 3, (a, c) = 2, (c, d) = 1, (d, T) = 3, (b, a) = 1, (a, b) = 2, (b, d) = 2

Graph 2:
- Source: S
- Sink: T
- Edges: (S, T)
Multi-Terminal Max Flow

- Undirected edges only. i.e. \( b_{i,j} = b_{j,i} \)
- Given edge capacities

\[\begin{array}{ccc}
  & j & \\
 i & 3 & 5 \\
  & 4 & k \\
  & 7 & j \\
  & 7 & \\
i & 8 & k
\end{array}\]

- Result of Max Flows
For any three nodes, $F_{i,j}, F_{j,k}, F_{i,k}$ satisfy

$$F_{i,k} \geq \min(F_{i,j}, F_{j,k})$$

$$F_{i,k} = C[ X, \overline{X} ]$$
Max Flow

- $F_{i,j} \geq \min(F_{i,k}, F_{k,j})$
- $F_{j,k} \geq \min(F_{j,i}, F_{i,k})$
- Among any three values, two values must be the same, and the third one is bigger or also the same value.
Max Flow

Representation of

Given $n = 100$ - node network

$$\binom{n}{2} \text{ Max Flows} = (n - 1) \text{ Max Flows}$$
Max Flow

Edge Capacities

Result of Max Flows
For any undirected network, there is a tree-shape network which has the same max flows and the same $n - 1$ cut value.

Gomory-Hu cut tree

G - H tree

Not only flow - equivalent, but also cut equivalent.