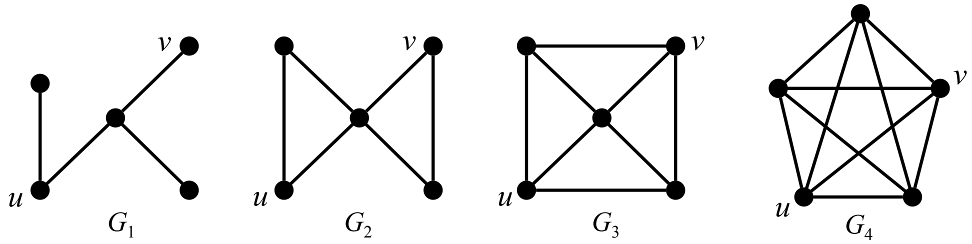
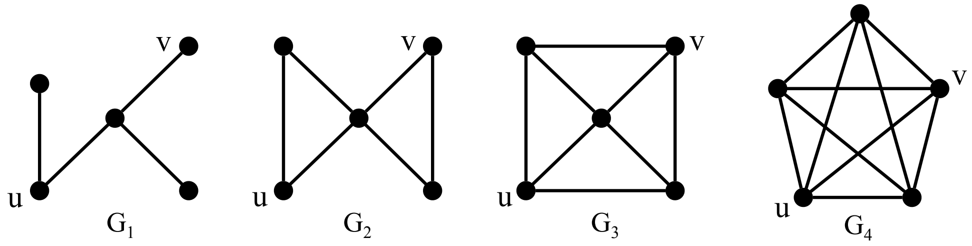
Connectivity and Menger's theorems

We have seen a measure of connectivity that is based on invulnerability to deletions (be it vtcs or edges). There is another reasonable measure of connectivity based on the multiplicity of alternative paths. We will see (in Menger's thms) that these two notions are, in fact, the same.



- G_1: 1 unique path between u and v
- G_2: 4 paths between u and v
- G_3: 10 paths between u and v
- G_4: many paths between u and v

Definition Two paths from u to v are *internally disjoint* if they have no common internal vertex. The paths are *edge disjoint* if they have no common edge.



$\kappa(G_1) = 1$	$\kappa(G_2) = 1$	$\kappa(G_3) = 3$	$\kappa(G_4) = 4$
$\kappa'(G_1) = 1$	$\kappa'(G_2) = 2$	$\kappa'(G_3) = 3$	$\kappa'(G_4) = 4$

Between u and v, we have....

- G_1: 1 internally disjoint path; 1 edge disjoint path

- G_2: 1 internally disjoint path; 2 edge disjoint paths
- G_3: 3 internally disjoint paths; 3 edge disjoint paths
- G_4: 4 internally disjoint path; 4 edge disjoint paths

Theorem (Menger - vertex, undirected, global version)**.** *A graph G is k-connected if and only if every pair of vertices are joined by k pairwise internally disjoint paths.*

Theorem (Menger - edge, undirected, global version)**.** *A graph G is k-edge-connected if and only if every pair of vertices are joined by k pairwise edge disjoint paths.*

There are several versions of Menger's theorem, all of which can be derived from the max-flow, min-cut theorem.

Directed graphs and network flows

Definitions In a *directed graph*, or *digraph*, each edge has a direction: (u, v) is different than (v, u). So a digraph is a graph whose edges are ordered pairs of vertices.

Directed edges are often called *arcs*.

For edge (u, v), u is called the *tail* and v is called the *head*.

We define the *in-degree* and *out-degree* of a vertex v as

$$\begin{aligned}\text{in-degree of } v = d^-(v) &= \#\text{ of edges with head } v = \{(u,v) : u \in V(G)\}\\ \text{out-degree of } v = d^+(v) &= \#\text{ of edges with tail } v = \{(v,u) : u \in V(G)\}\end{aligned}$$

Definitions A *network* is a digraph G together with a *capacity function* $c : E(G) \to \mathbb{R}_{\geq 0}$ and two distinguished vtcs $s, t \in V(G)$ known as the *source vertex* and the *sink vertex*.

Definition A *feasible flow* in a network is a function $f : E(G) \to \mathbb{R}_{\geq 0}$ such that

(a) $0 \leq f(e) \leq c(e)$ for all directed edges e. [capacity constraints]

(b) For all vtcs $v \in V(G) - \{s, t\}$,

$$\underbrace{f^-(v) = \sum_u f(u,v)}_{\text{flow in}} = \underbrace{\sum_w f(v,w) = f^+(v)}_{\text{flow out}}.$$

[conservation constraints]

The *flow value* is $f^+(s) - f^-(s)$.

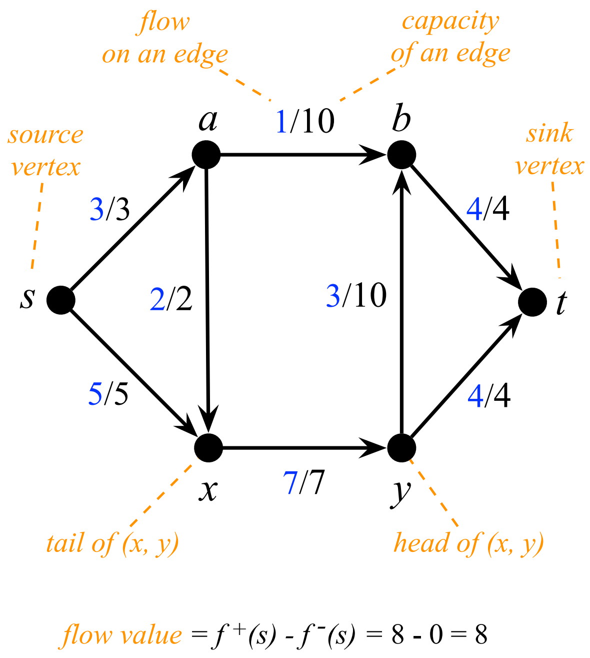


Figure 1: A *network* is shown, along with a feasible flow on the network.

Max flow problem The max flow problem is to find a feasible flow f with the maximum flow value.

Definition A *network cut*, or an *$s-t$ cut*, is an edge cut $[S,\overline{S}]$ where $s \in S, t \in \overline{S}$. The *capacity* of a network cut $[S,\overline{S}]$ is the sum of the capacities of the edges in the cut,

$$\mathrm{cap}[S,\overline{S}] = \sum_{e \in [S,\overline{S}]} c(e).$$

Min cut problem The min cut problem is to find an $s-t$ cut with the minimum possible capacity.

Lemma. *For any feasible flow f and $s-t$ cut $[S,\overline{S}]$ in a network G, we have*

$$\textit{flow value} = \sum_{e \in [S,\overline{S}]} f(e) - \sum_{e \in [\overline{S},S]} f(e).$$

Proof.

$$\begin{aligned} \text{flow value} &= f^+(s) - f^-(s) \\ &= f^+(s) - f^-(s) + \sum_{v \in S - \{s\}} \underbrace{(f^+(v) - f^-(v))}_{=0} \\ &= \sum_{v \in S} f^+(v) - \sum_{v \in S} f^-(v). \qquad (\star) \end{aligned}$$

Let $e = (u,v)$ be any edge in G. Then

- if $u, v \notin S$, then $f(e)$ is not counted at all;
- if $u \in S, v \notin S$, then $f(e)$ is counted once in the first sum but not in the second sum;
- if $u \notin S, v \in S$, then $f(e)$ is counted once in the second sum but not in the first sum;
- and if $u, v \in S$, then $f(e)$ is counted once in the first sum and once in the second sum, so these two terms cancel.

Thus, we can express $(\star)$ as

$$\text{flow value} = \sum_{v\in S} f^+(v) - \sum_{v\in S} f^-(v) = \sum_{e\in[S,\overline{S}]} f(e) - \sum_{e\in[\overline{S},S]} f(e).$$

■

Max flow and min cut problems are dual optimization problems. (Recall that min vertex cover and max matching problems were duals to one another).

Theorem (Weak duality)**.** *If f is a feasible flow and $[S,\overline{S}]$ is an $s-t$-cut, then*

$$\textit{flow value of } f \le \operatorname{cap}[S,\overline{S}].$$

Proof.

$$\begin{aligned}
\text{flow value of } f &= \sum_{e\in[S,\overline{S}]} f(e) - \sum_{e\in[\overline{S},S]} f(e) && \text{by previous lemma}\\
&\le \sum_{e\in[S,\overline{S}]} f(e) && \text{since } f \text{ nonnegative}\\
&\le \sum_{e\in[S,\overline{S}]} c(e) && \text{by capacity constraints of } f\\
&= \operatorname{cap}[S,\overline{S}].
\end{aligned}$$

■

Remark In particular,

$$\text{max flow value} \le \text{min cut capacity}.$$

Certificate of optimality: If f is a flow and $[S,\overline{S}]$ is an $s-t$ cut such that

$$\text{flow value of } f = \operatorname{cap}[S,\overline{S}],$$

then both the flow and the cut must be optimal.

Theorem (max flow, min cut – strong duality)**.** *Let G be a network. The maximum value of a flow equals the minimum capacity of a cut.*

We prove this strong duality result via the Ford-Fulkerson algorithm.

Basics of Ford-Fulkerson algorithm

(1) Start with a zero flow everywhere.

(2) Form residual network.

> residual capacities: $r(e) = c(e) - f(e) + f(e^{-1})$
> measures how much we can increase flow along an arc or virtually do so (by decreasing flow along reverse arc)

(3) Augment flow along s–t path P in residual network by $\min_{e \in E(P)} r(e)$.

(4) Go to step 2 and repeat. Continue repeating until there is no $s - t$ path in residual network.

Ford-Fulkerson algorithm returns a feasible flow f and an $s - t$ cut $[S, \overline{S}]$ such that the flow value of f is equal to the capacity of the $s - t$ cut, thereby proving the optimality of both.

Lemma. *If a flow f has a residual network with an augmenting path P, then f is not maximum. Furthermore, there is another flow f' whose value is* $r(P)$ *greater than the value of f.*

Outline of proof. Form a new flow by augmenting along path P.

- Check that the capacity constraint holds, namely the flow on every arc is between 0 and capacity.
- Check that flow conservation constraint holds for new flow.

■