Network Flow

A ***network*** is a directed graph **$G=(V, E)$** such that:

- **there are vertices $s,t \in V$ such that:**
  - **s has indegree 0 (s is the *source*)**
  - **t has outdegree 0 (t is the *sink*)**
- **each edge $(u,v) \in E$ has a non-negative capacity $c(u,v) \in R$ (the set of real numbers)**

(Assume nonexistent edges in G have capacity 0 and every vertex lies on some path from s to t)

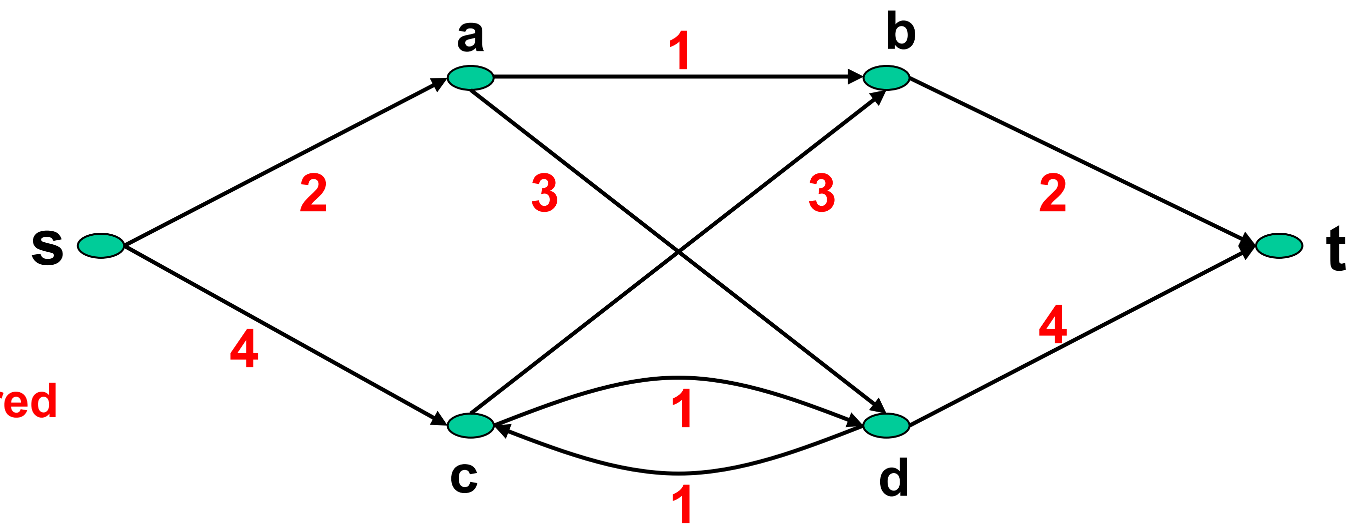
A *flow* in G is a function $f : E \rightarrow R$ such that:

- ***Capacity constraint*: for every edge, $0 \leq$ flow $\leq$ capacity**
- ***Flow conservation constraint*: for every vertex other than s and t, total incoming flow = total outgoing flow**

The *value* $val(f)$ of a flow f is the total flow out from s

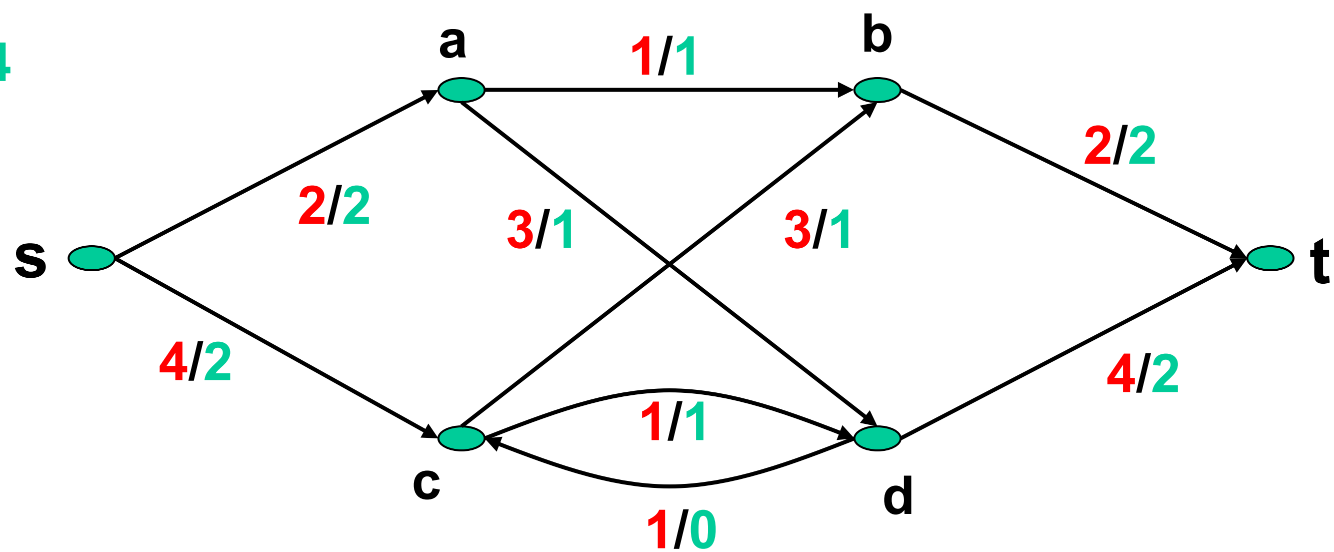
- **or equivalently, the total flow into t**

Example network with flow



capacities in red

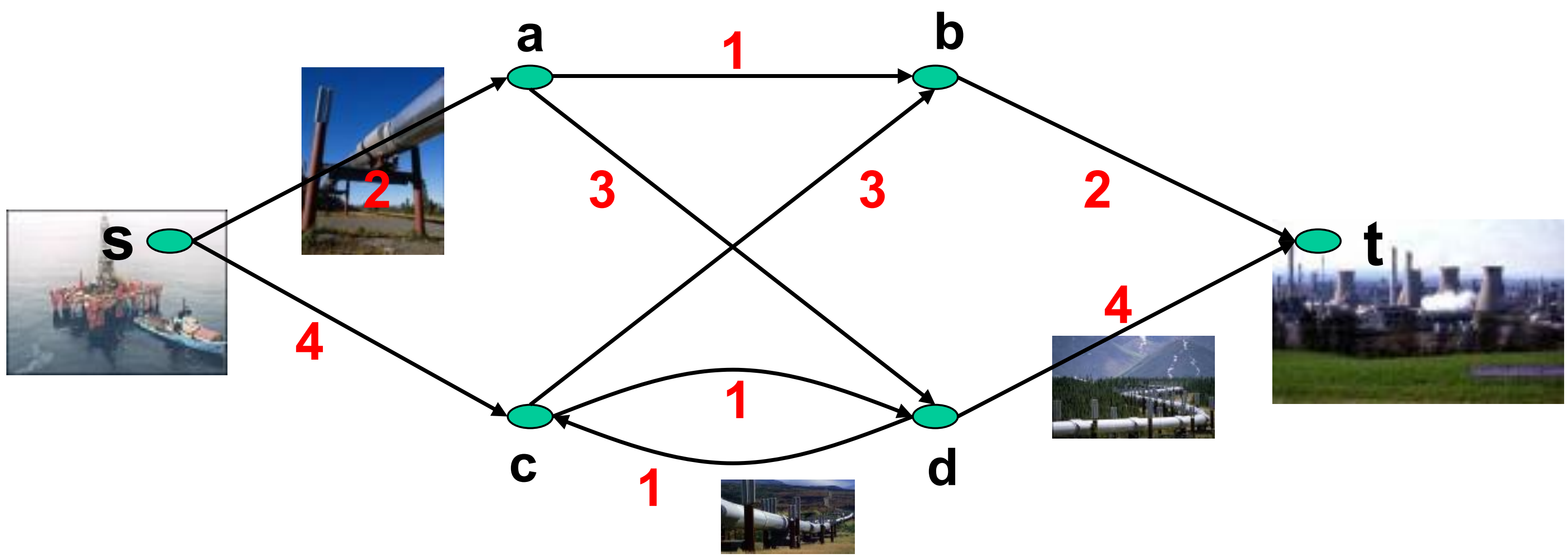
flow f, val(f)=4



flows in green

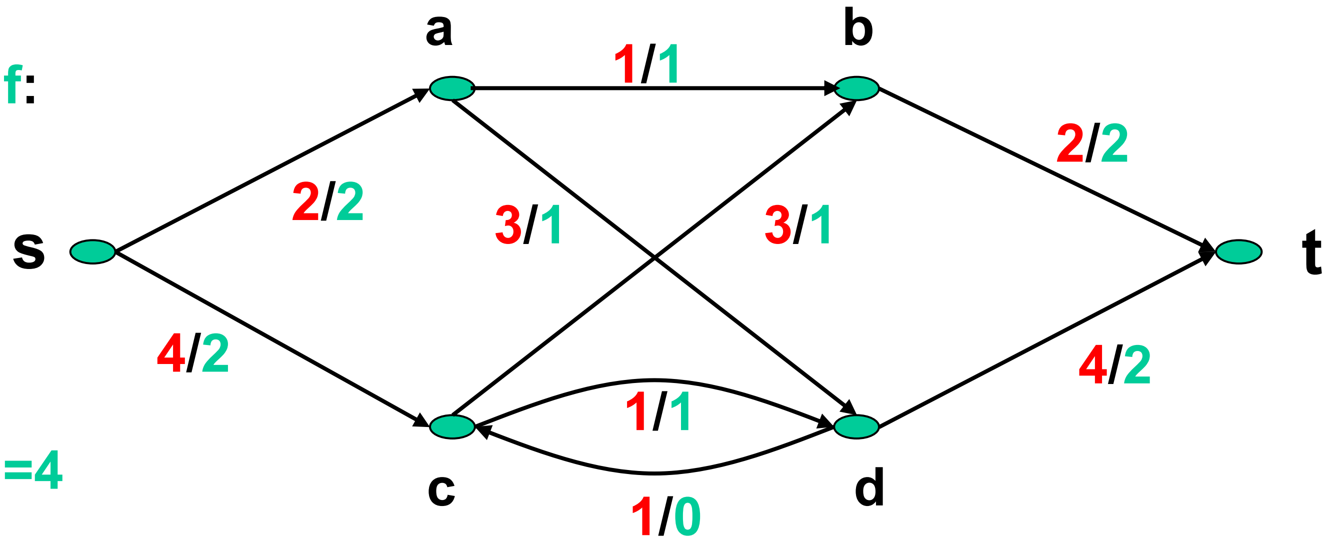
Applications of network flow

- Transportation
- Resource allocation
- Telecommunications
- Data mining
- Project selection
- Airline scheduling
- Baseball elimination
- Network connectivity
- Network reliability
- Distributed computing



An alternative flow

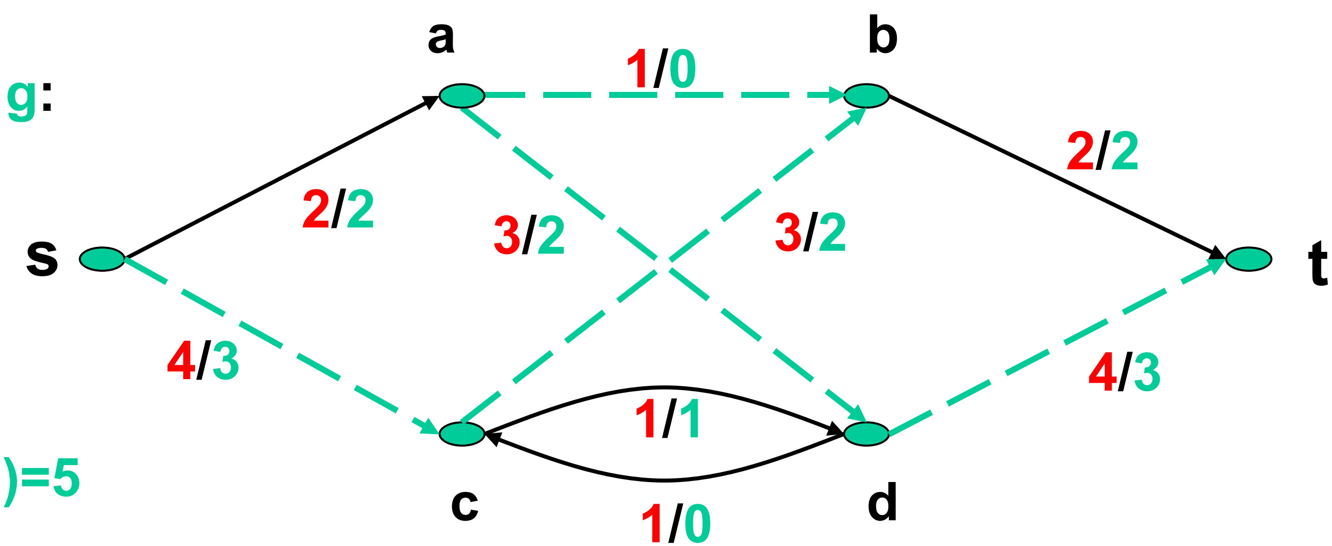
Flow f:



val(f)=4

There is another flow g with larger value

Flow g:

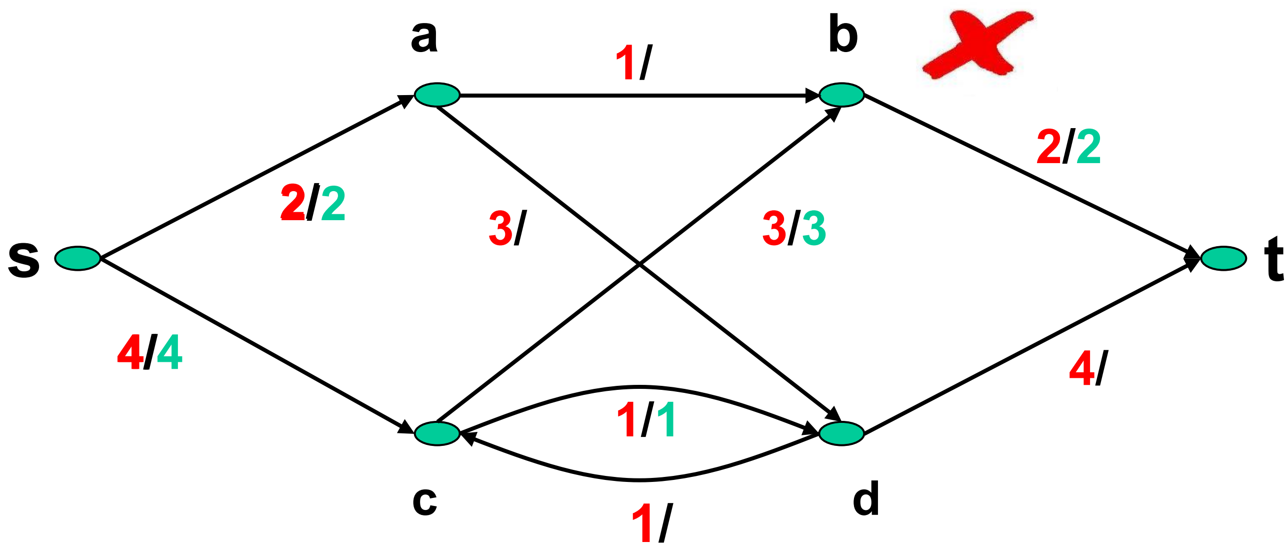


val(g)=5

Can we find a flow with even larger value?

A flow is *saturating* if f(s,v) = c(s,v) for all vertices v

- In our example, a saturating flow would have value 6



- Need a flow of at least 3 out of vertex b, so no saturating flow exists

A *maximum flow* is a flow whose value is maximum

Maximum flow problem
Input: Network G=(V,E) with capacity function c
Output: Maximum flow f in G

Augmenting paths

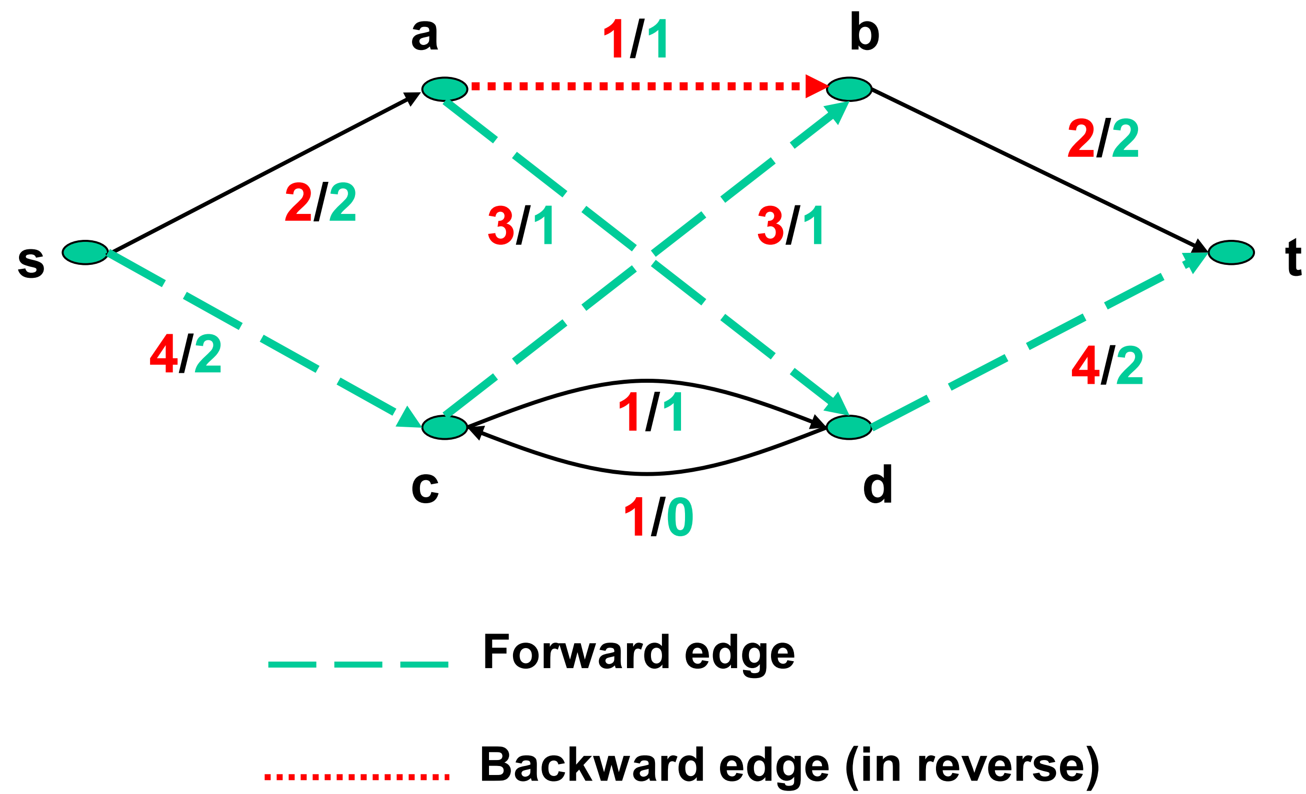
An *augmenting path* with respect to a flow f is a path from s to t comprising edges of G but not necessarily directed as in G

Each edge (u,v) in the path must satisfy one of the following two conditions:

1. $(u,v) \in E$ (i.e. (u,v) is in the same direction as in G) and $f(u,v) < c(u,v)$
   - (u,v) is called a *forward edge*
   - The difference $c(u,v) - f(u,v)$ is the *slack* of (u,v)

2. $(v,u) \in E$ (i.e. (u,v) is opposite in direction to an edge in G) and $f(v,u) > 0$
   - (u,v) is called a *backward edge*

Augmenting path – example

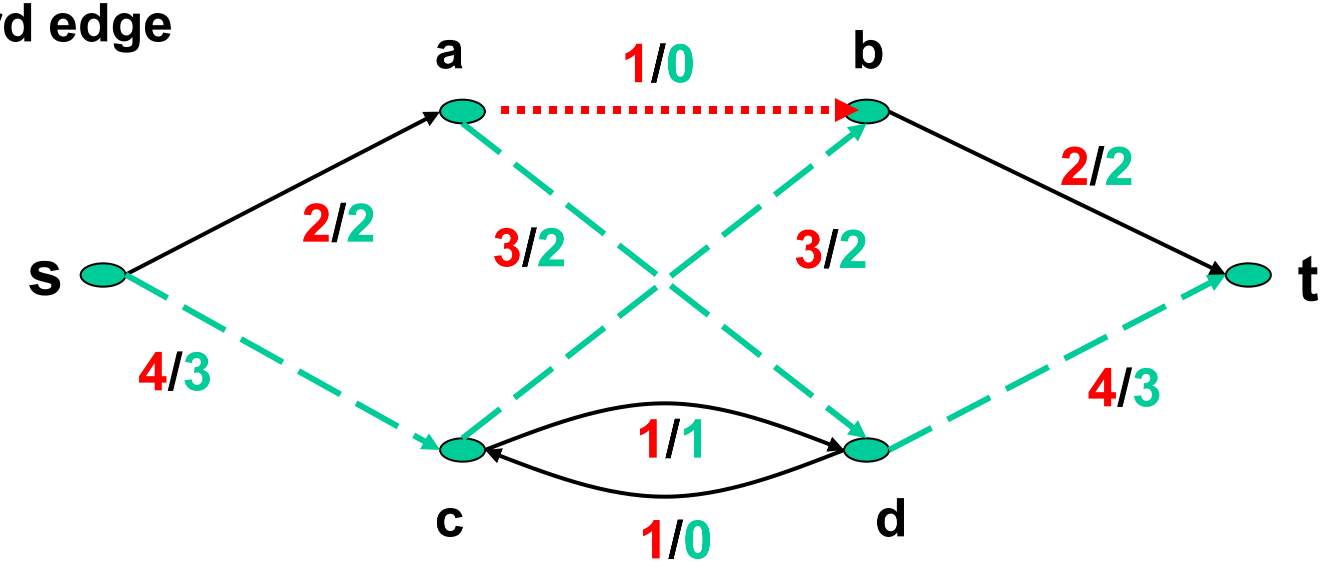


Augmenting path: (s,c) (c,b) (b,a) (a,d) (d,t)

Augmenting a flow along an augmenting path

Start with flow f (value 4) from earlier slide (#2)

- **f admits an augmenting path (shown on previous slide)**
- **Push an extra unit of flow along each forward edge**
- **Borrow a unit of flow from an edge in the opposite direction to a backward edge**



- **End up with flow g (value 5) from earlier slide (#4)**
- **Edges leaving the source must be forward edges of an augmenting path**
- **So an augmenting path allows us to increase the value of a flow**

Augmenting path theorem

A flow is maximum if and only if it admits no augmenting path

Proof (only if): Suppose f admits an augmenting path $P = \langle e_1, e_2, \ldots, e_r \rangle$

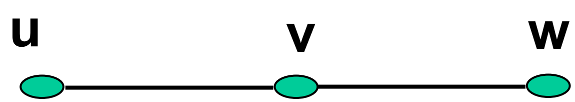
- Can push additional flow of m_i along each **forward** edge $e_i = (u,v)$ with slack $m_i = c(u,v) - f(u,v)$ (note $m_i > 0$)
- Can borrow a flow of m_i from each edge (v,u) in the opposite direction to a **backward** edge $e_i = (u,v)$ where $m_i = f(v,u)$ (note $m_i > 0$)
- Let $m = \min\{m_i \ : 1 \le i \le r\}$ (then $m > 0$)
- **Aim:** increase value of flow f by m
- Define a flow g as follows: initially set $g \equiv f$
- Now let i ($1 \le i \le r$) be given and consider edge $e_i = (u,v)$ on the augmenting path P

- If $e_i = (u,v)$ is a forward edge of P then set $g(u,v) = f(u,v)+m$
- Otherwise, $e_i = (u,v)$ is a backward edge of P; set $g(v,u) = f(v,u)-m$

Check that g satisfies
(1) capacity constraints
(2) flow conservation constraints

For (2), consider each vertex $v \in V \setminus \{s,t\}$
Either (a) v is incident to 0 edges of P
or (b) v is incident to 2 edges of P: (u,v), (v,w)

For (b), consider 4 cases according to whether each of (u,v) and (v,w) is a forward / backward edge of P

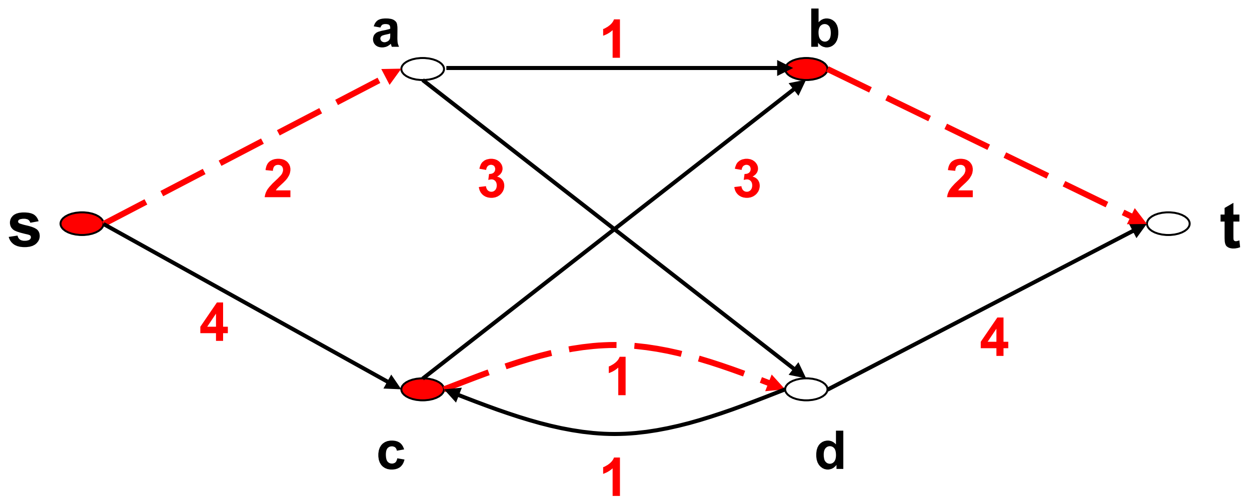


Edge e_1 is a forward edge of P, so $val(g) = val(f) + m > val(f)$
Thus f is not a maximum flow □

(if): suppose f admits no augmenting path.
Need to prove that f is maximum...

Cuts

- A set of edges separating the source from the sink
- Formally: let V be partitioned into A and B, where $s \in A$ and $t \in B$, (i.e., $A \subseteq V$, $B = V \setminus A$). Then the set of edges $C = \{(u,v) \in E : u \in A \text{ and } v \in B\}$ is a *cut*
- Example:



- $A = \{s,b,c\}$, $B = \{a,d,t\}$ and $C = \{(s,a),(c,d),(b,t)\}$
- The *capacity* of a cut C, $cap(C)$, is the sum of the capacities of the edges in C
- In the example, $cap(C) = 5$

Equivalence of the definitions

- If C is defined with respect to the formal definition, removing the edges of C leaves no path from s to t
- Proof: suppose after removing C, there is a path $\langle e_1, e_2, \ldots, e_r \rangle$ from s to t
    - $e_1=(s,v)$ for some v, and $s \in A$, so $v \in A$
    - $e_2=(v,w)$ for some w, and $v \in A$, so $w \in A$
    - …
    - $e_r=(z,t)$ for some z, and $z \in A$, so $t \in A$, contradiction
- Converse can also be shown