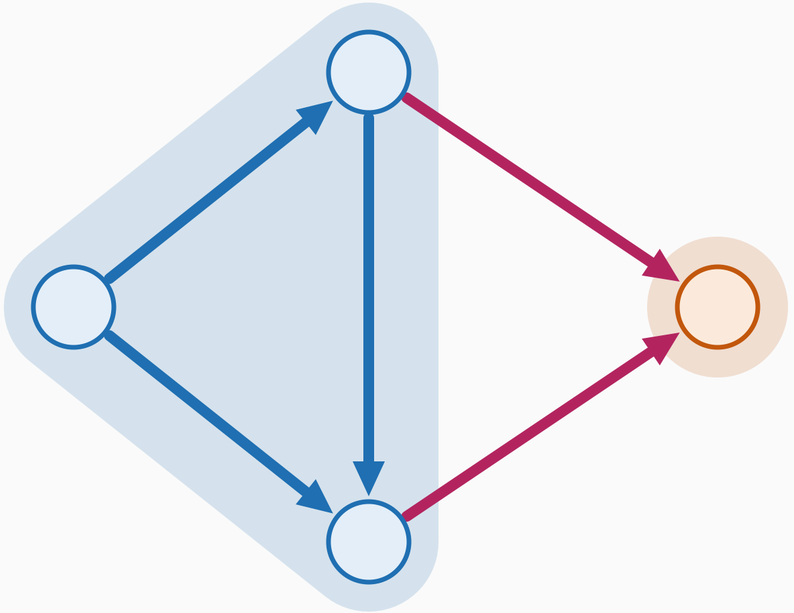
ALGORITHMS AND DATA STRUCTURES · LECTURE 11

Minimum cuts

The correctness of Ford–Fulkerson, and max-flow min-cut



Heng Guo

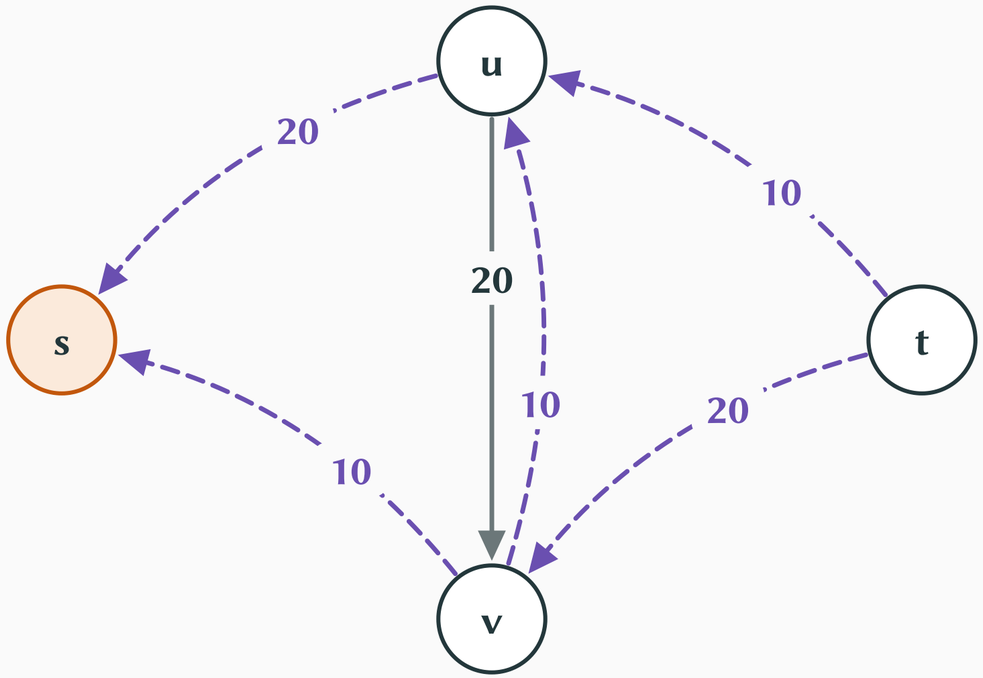
School of Informatics, University of Edinburgh

Monday 26 October 2026 · 16:10–17:00

Ford–Fulkerson: what is left?

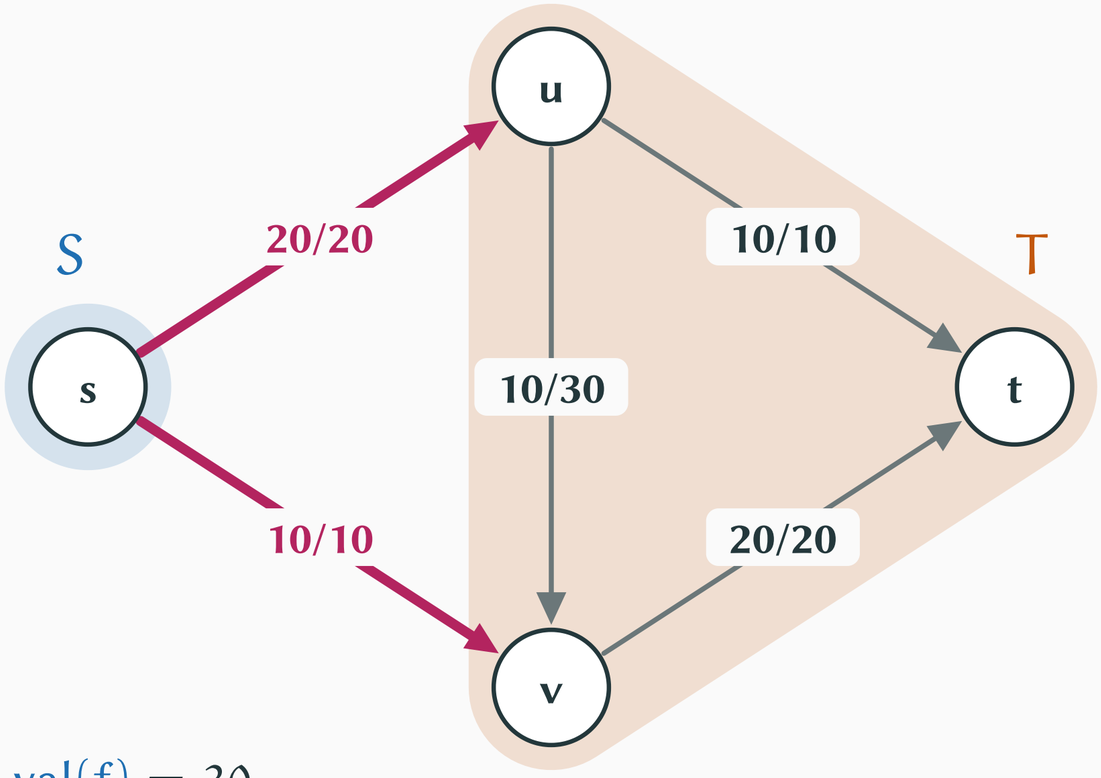
- ✓ **Feasibility.** Is every result a flow? LECTURE 9
- ✓ **Termination.** Does it stop? LECTURE 9
- ✓ **Running time.** How fast? LECTURES 9, 10
- ? **Optimality.** Is the flow maximum? TODAY

the last residual graph G_f



? *G_f has no path from s to t. Why can no other flow be larger?*

Why 30 cannot be beaten



$\mathrm{val}(f) = 30$

Our flow has value 30.

? *Could another flow do better?*

All flow starts at s:
it has to leave through (s, u) and (s, v).

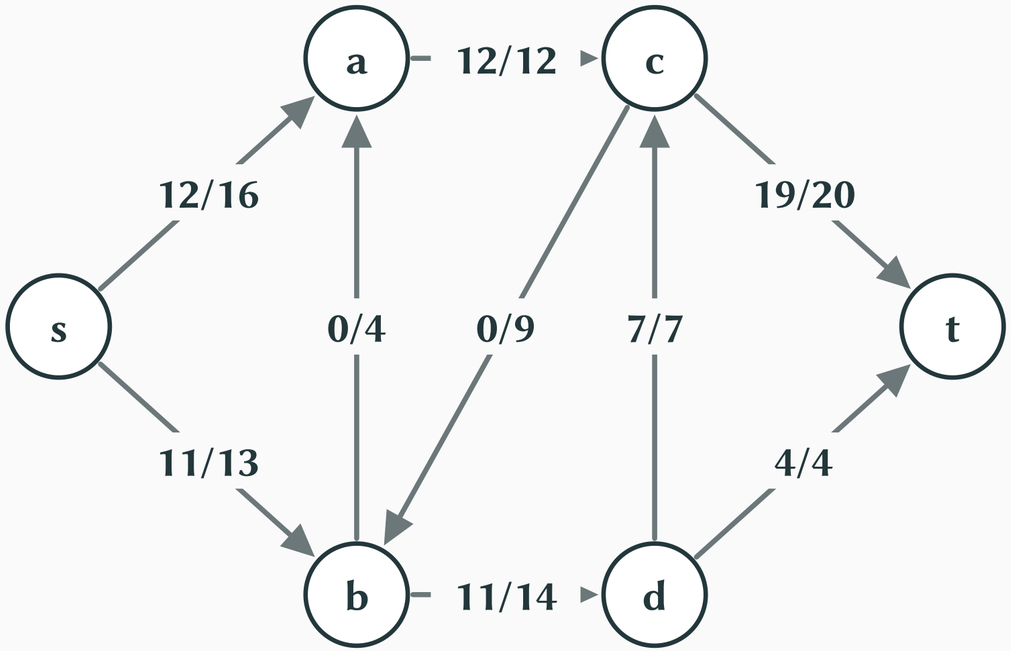
These two edges carry at most
$20 + 10 = 30$.

> No flow has value more than 30.
> Ours is **maximum**.

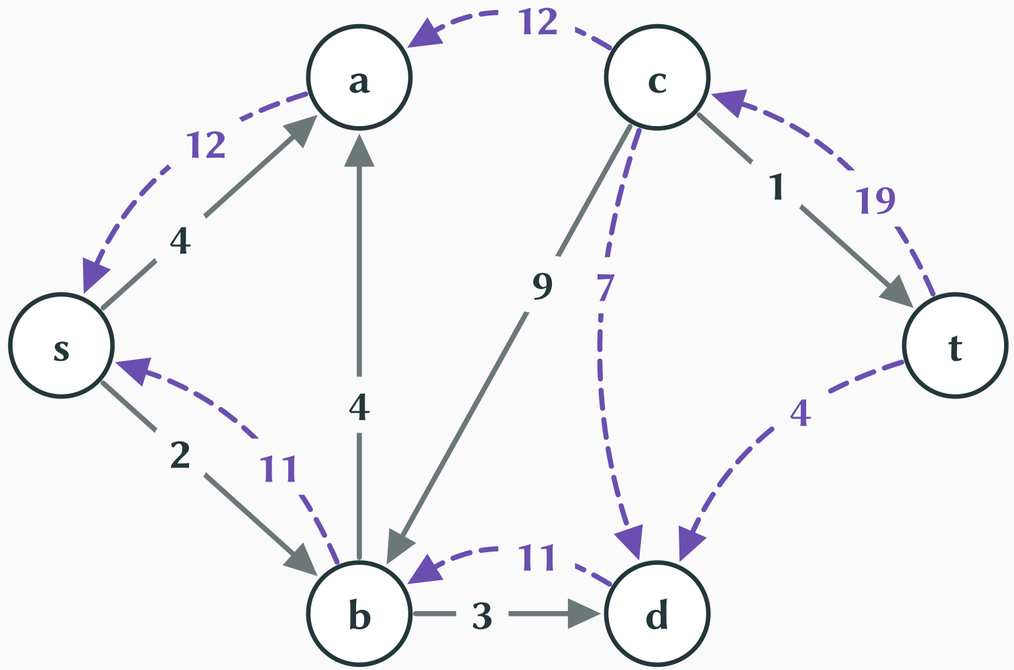
? *Is there always such a bottleneck?*

In a larger network

G with a flow f of value 23



its residual graph G_f: no s–t path



? *How do we know this flow is optimal?*

Cuts and their capacity

Definition.
An (s, t)-**cut** is a partition of the nodes into two sets S and T with $s \in S$ and $t \in T$.

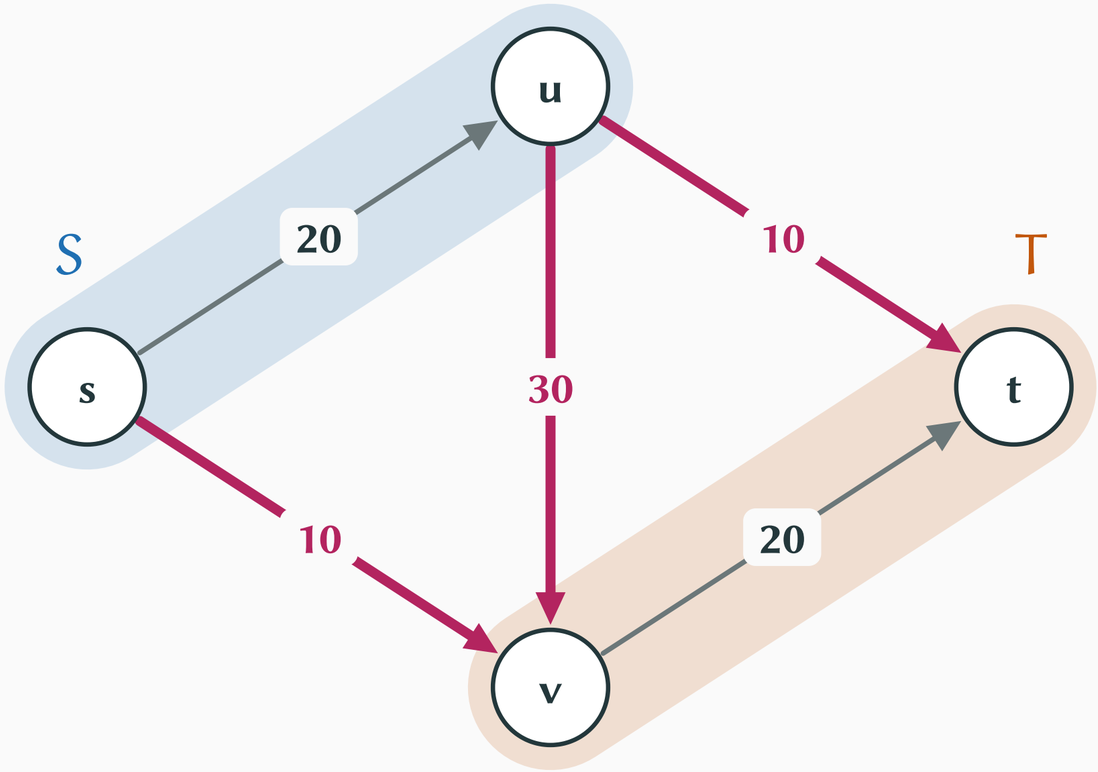
Example: $S = \{s, u\}$ and $T = \{v, t\}$.

Definition.
The **capacity** of the cut is

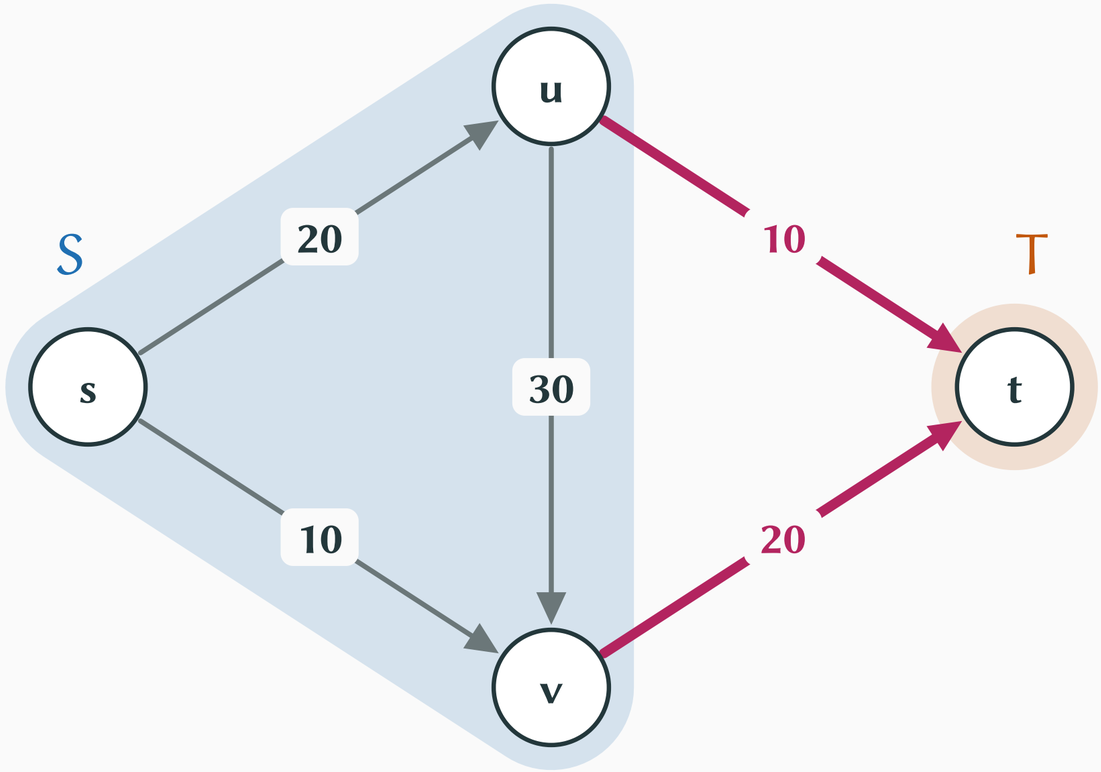
$$c(S, T) = \sum_{e \text{ out of } S} c_e,$$

the edges from S to T
(edges from T to S do not count).

$c(S, T) = 10 + 30 + 10 = 50$



Four cuts



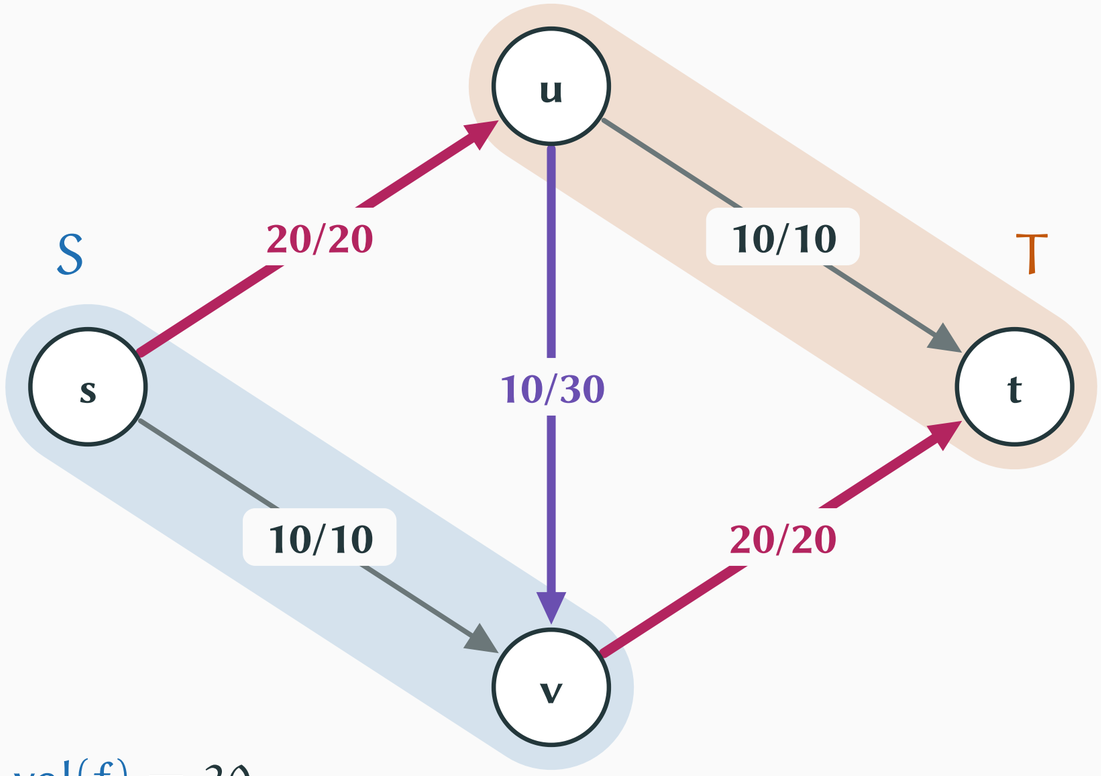
S	capacities out of S	$c(S, T)$
$\{s\}$	$20 + 10$	30
$\{s, u\}$	$10 + 30 + 10$	50
$\{s, v\}$	$20 + 20$	40
$\{s, u, v\}$	$10 + 20$	30

An edge from T to S does not count:
(u, v) for $S = \{s, v\}$.

Smallest: 30, reached twice.
With n nodes there are 2^{n-2} cuts.

How much flow crosses a cut?

$f^{out}(S)$: the flow on the edges **out of** S
$f^{in}(S)$: the flow on the edges **into** S



$val(f) = 30$

The cut $S = \{s, v\}$, $T = \{u, t\}$:

$f^{out}(S) = 20 + 20 = 40$

$f^{in}(S) = 10$

$$f^{out}(S) - f^{in}(S) = 40 - 10 = 30 = val(f)$$

S	$f^{out}(S)$	$f^{in}(S)$	$f^{out}(S) - f^{in}(S)$
$\{s\}$	30	0	30
$\{s, u\}$	30	0	30
$\{s, v\}$	40	10	30
$\{s, u, v\}$	30	0	30

? *A coincidence?*

Fact 1: the net flow across a cut

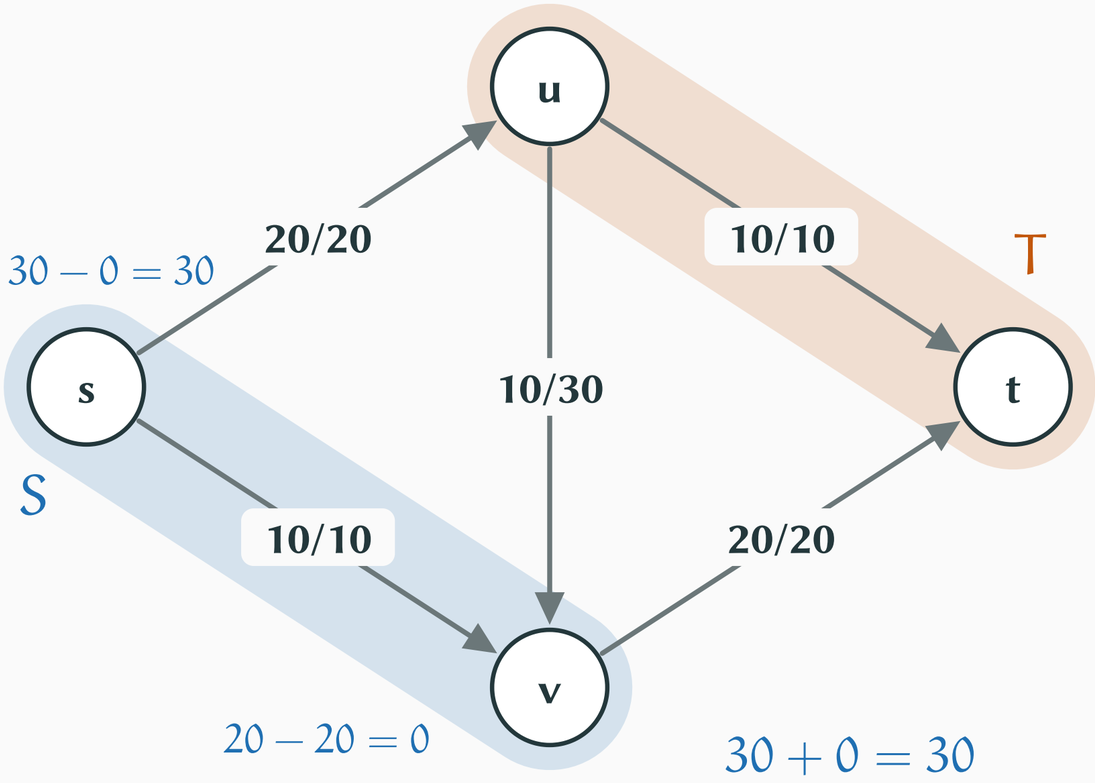
Fact 1 (flow value lemma).
For every flow f and every cut (S, T),

$$\mathrm{val}(f) = f^{\mathrm{out}}(S) - f^{\mathrm{in}}(S).$$

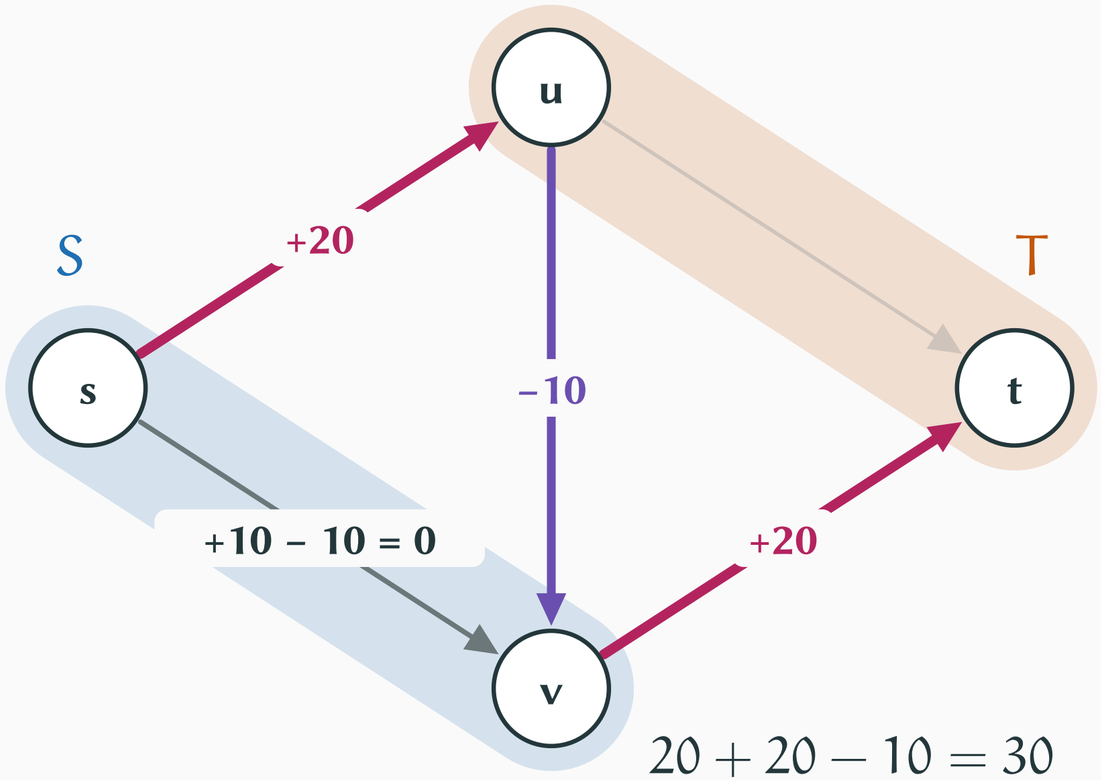
1. $\mathrm{val}(f) = f^{\mathrm{out}}(s) - f^{\mathrm{in}}(s)$, as no edge enters s.
2. Every other $w \in S$ is an inner node, as $t \in T$: $f^{\mathrm{out}}(w) - f^{\mathrm{in}}(w) = 0$, by conservation.
3. Add this up over all of S:

$$\mathrm{val}(f) = \sum_{w \in S} \left(f^{\mathrm{out}}(w) - f^{\mathrm{in}}(w) \right).$$

? *Why is this sum the net flow across the cut?*



Fact 1: counting edge by edge



Each edge $e = (x, y)$ appears in the terms of its ends:

$+f(e)$ in the term of x, if $x \in S$

$-f(e)$ in the term of y, if $y \in S$

both ends in S	$+f(e) - f(e) = 0$
from S to T	$+f(e)$
from T to S	$-f(e)$
both ends in T	not in the sum

$$\begin{aligned} \mathrm{val}(f) &= \sum_{e \text{ out of } S} f(e) - \sum_{e \text{ into } S} f(e) \\ &= f^{\mathrm{out}}(S) - f^{\mathrm{in}}(S). \qquad \square \end{aligned}$$

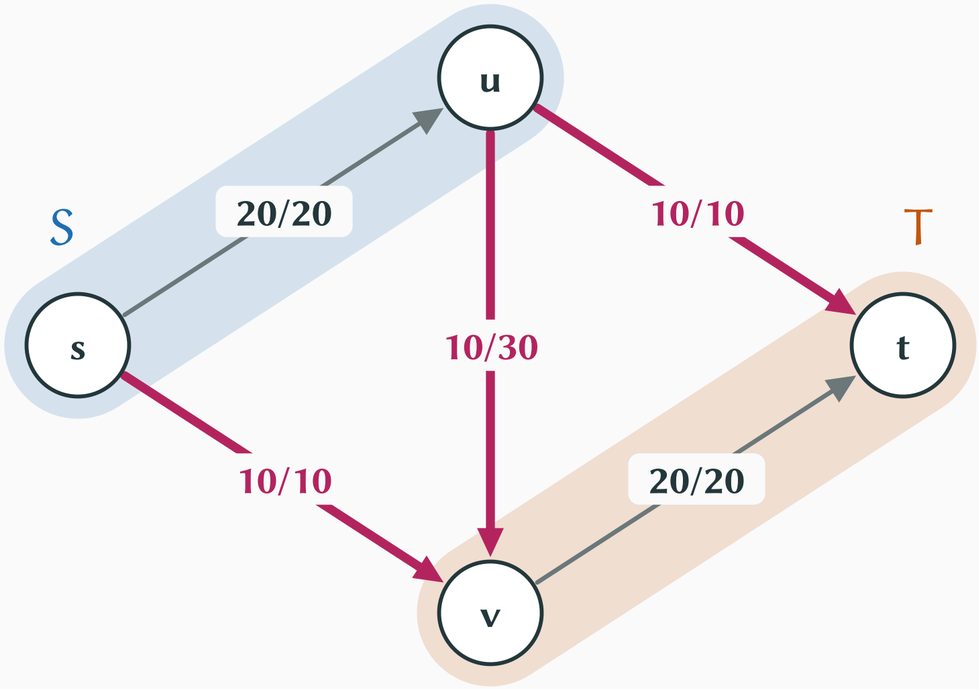
Fact 2. From the T side: $\mathrm{val}(f) = f^{\mathrm{in}}(T) - f^{\mathrm{out}}(T)$.

The edges out of S are the edges into T, and vice versa.

Fact 3: no flow beats a cut

Fact 3.
For every flow f and every cut (S, T):
$\mathrm{val}(f) \leqslant c(S, T)$.

$$
\begin{aligned}
\mathrm{val}(f) &= f^{\mathrm{out}}(S) - f^{\mathrm{in}}(S) && \text{Fact 1} \\
&\leqslant f^{\mathrm{out}}(S) && f^{\mathrm{in}}(S) \geqslant 0 \\
&= \sum_{e \text{ out of } S} f(e) && \text{definition} \\
&\leqslant \sum_{e \text{ out of } S} c_e && \text{capacity: } f(e) \leqslant c_e \\
&= c(S, T) && \text{definition} \quad \square
\end{aligned}
$$



? *Can equality always hold, for some flow and cut?*

When is the bound tight?

Fact 3 loses something in two places. By the flow value lemma (Fact 1),

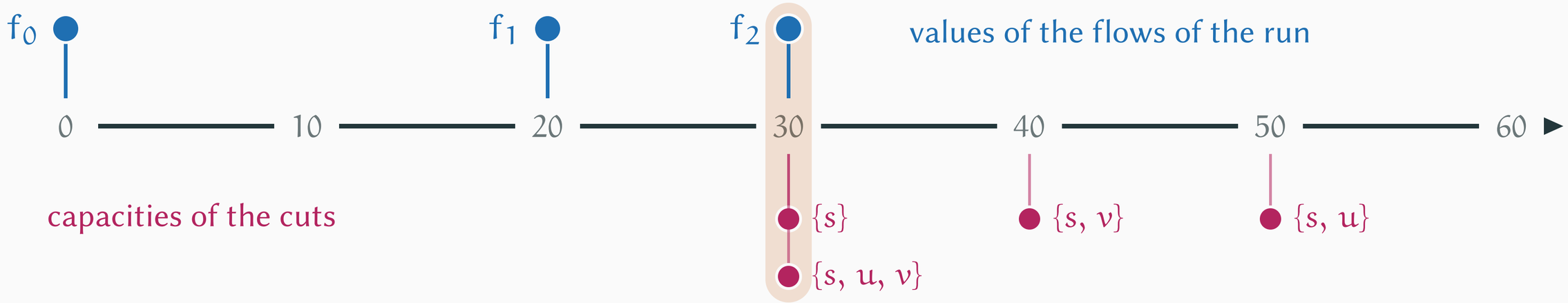
$$c(S,T) - \mathrm{val}(f) = \sum_{e \text{ out of } S} \big(c_e - f(e)\big) + f^{\mathrm{in}}(S).$$

S	$c(S,T)$	**unused capacity on the way out**	**flow coming back into S**	$c(S,T) - \mathrm{val}(f)$
$\{s\}$	30	0	0	0
$\{s, u\}$	50	20 (u, v)	0	20
$\{s, v\}$	40	0	10 (u, v)	10
$\{s, u, v\}$	30	0	0	0

Tight ($c(S,T) = \mathrm{val}(f)$) exactly when every edge out of S is **full**, $f(e) = c_e$, and every edge into S is **empty**, $f(e) = 0$.

Flows below, cuts above

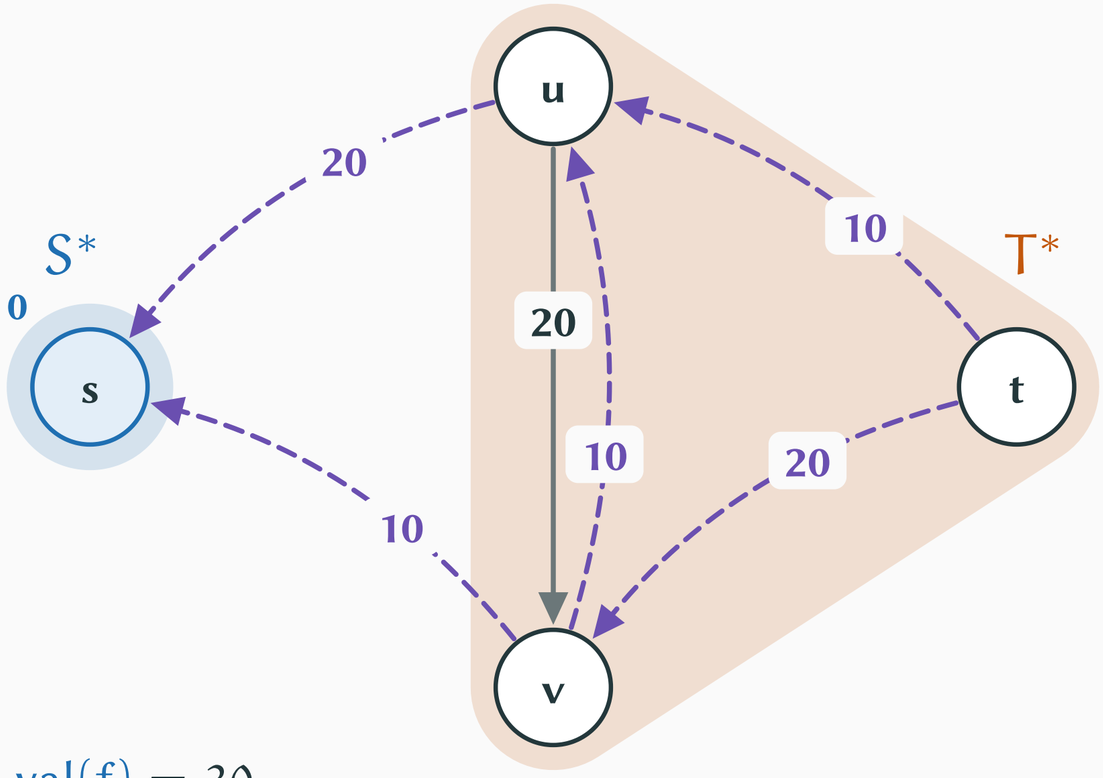
$\mathrm{val}(f) \leqslant c(S, T)$ for every flow f and every cut (S, T)



> If $\mathrm{val}(f) = c(S, T)$, then f is a **maximum flow** and (S, T) is a **minimum cut**.

? *Can we always find such a pair?*

One search, two outcomes



$\mathrm{val}(f) = 30$

Search from s in G_f (BFS or DFS):
which nodes can we reach?

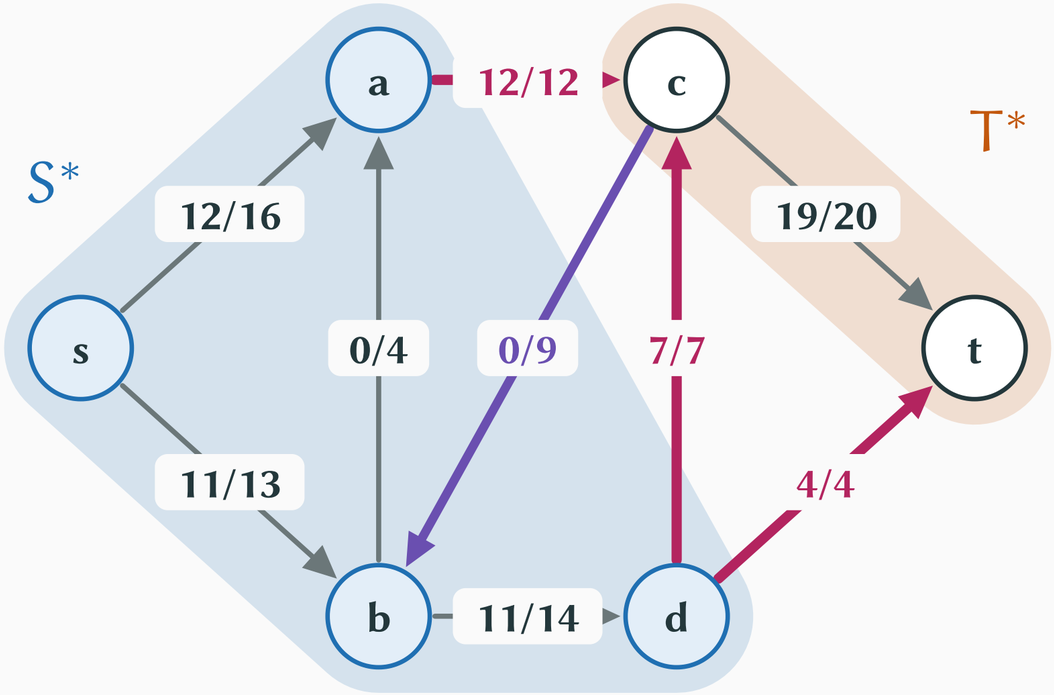
t is reached: an augmenting path
$(s, v, u, t$, when $\mathrm{val}(f) = 20)$.
f is not maximum; augment.

t is not reached: let S^* be
the set of nodes reached, T^* the rest.
(S^*, T^*) is a cut.

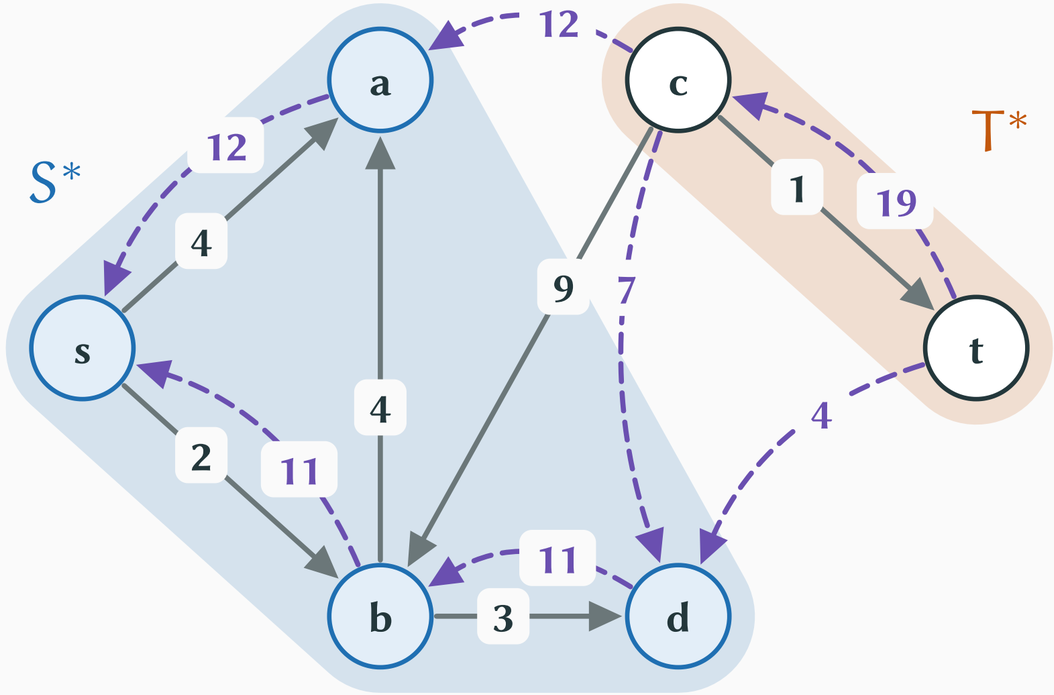
> $c(S^*, T^*) = 20 + 10 = 30 = \mathrm{val}(f)$

The cut (S^*, T^*) on the larger network

G with the flow f of value 23



the residual graph G_f

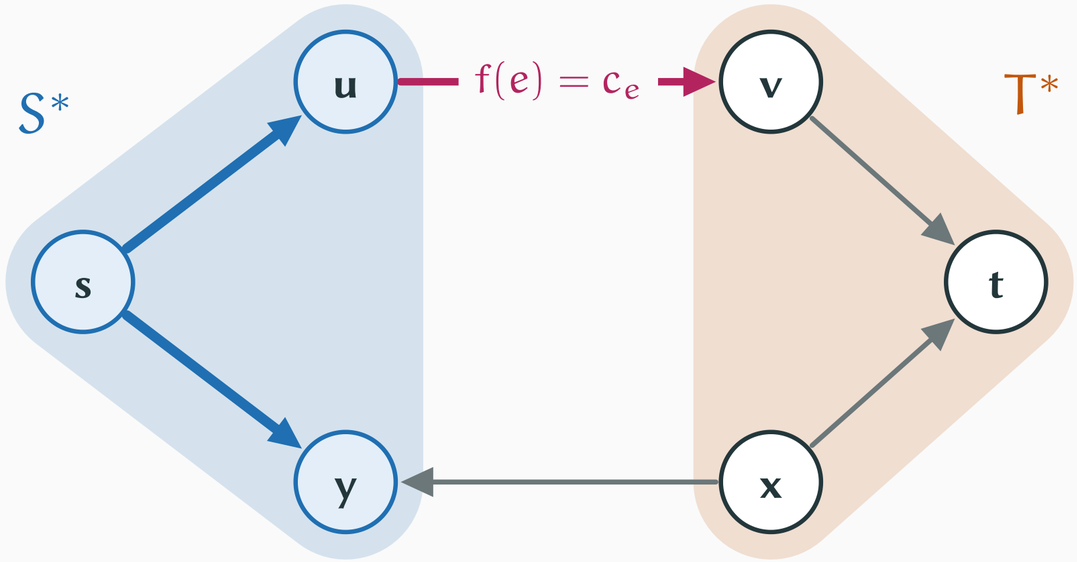


Reachable from s in G_f: $S^* = \{s, a, b, d\}$; the rest: $T^* = \{c, t\}$

Out of S^*: all **full** ($f = c$). Into S^*: all **empty** ($f = 0$).

This is sufficient for Fact 3 to be tight (slide 11): $c(S^*, T^*) = \mathrm{val}(f) = 23$.

Fact 4: edges out of S^* are full



S^*: the nodes reached from s in G_f. $t \notin S^*$.

Take an edge $e = (u, v)$ with $u \in S^*$, $v \in T^*$.

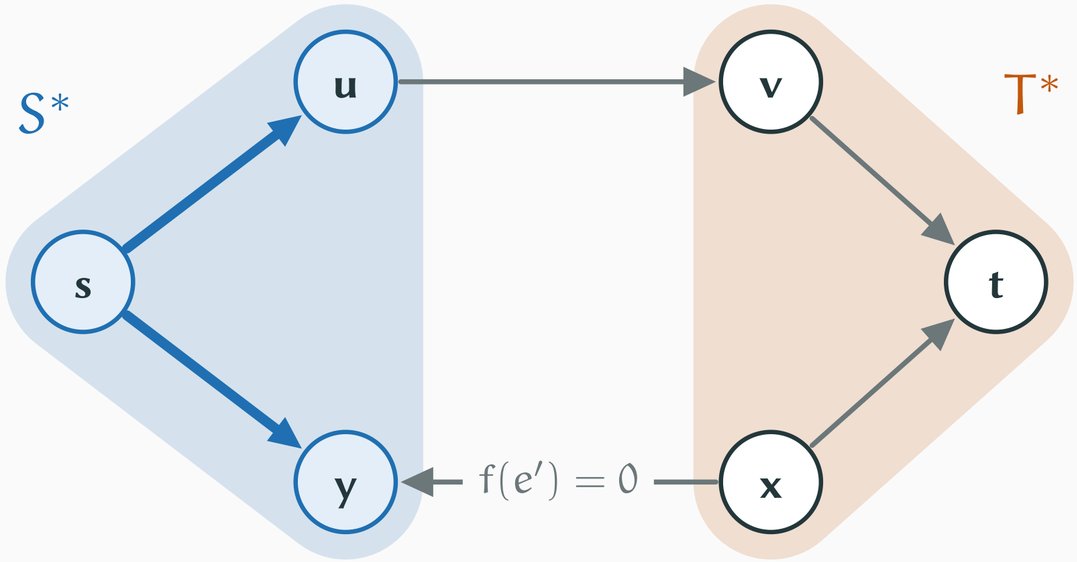
Suppose $f(e) < c_e$. Then G_f has the **forward arc** (u, v).

u is reached, so v is reached: $v \in S^*$.

Contradiction.

> $f(e) = c_e$: every edge **out of** S^* is **full**.

Fact 4: edges into S^* are empty



S^*: the nodes reached from s in G_f. $t \notin S^*$.

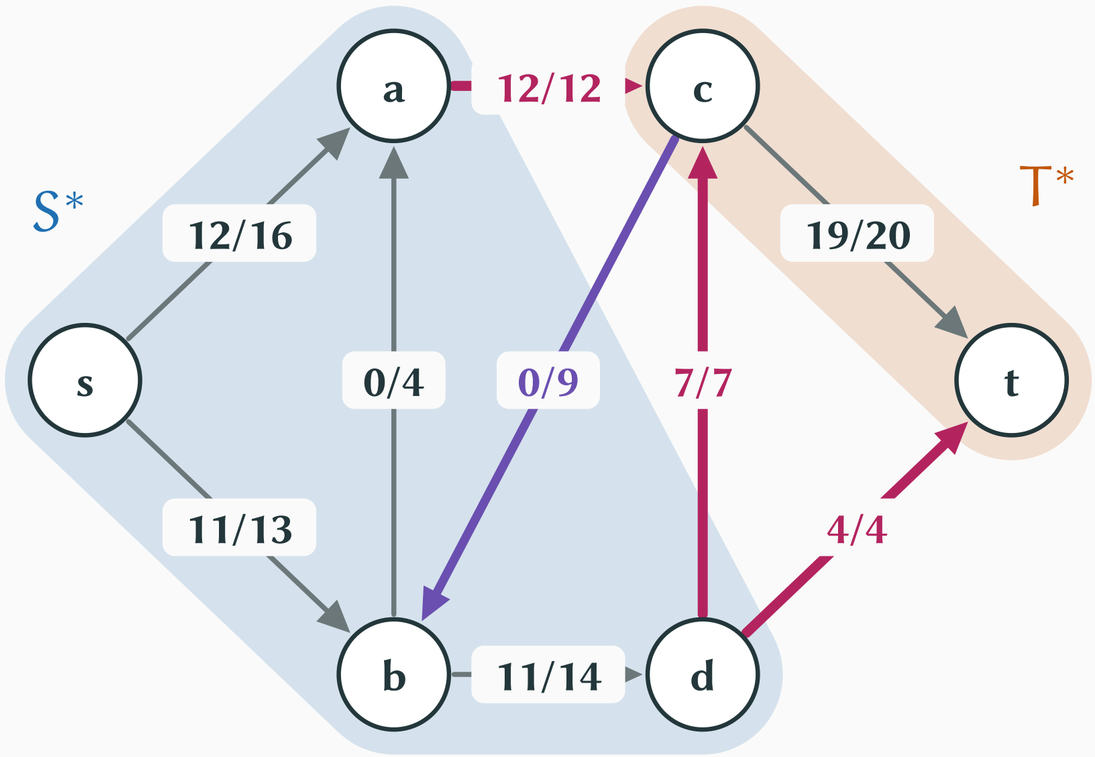
Take an edge $e' = (x, y)$ with $x \in T^*, y \in S^*$.

Suppose $f(e') > 0$. Then G_f has the **backward arc** (y, x).

y is reached, so x is reached: $x \in S^*$.
Contradiction.

> $f(e') = 0$: every edge **into** S^* is **empty**.

Fact 4: the capacity of the cut



Fact 4.
If G_f has no s–t path,
then (S^*, T^*) is a cut with $c(S^*, T^*) = \mathrm{val}(f)$.

$$\mathrm{val}(f) = f^{\mathrm{out}}(S^*) - f^{\mathrm{in}}(S^*)$$
Fact 1

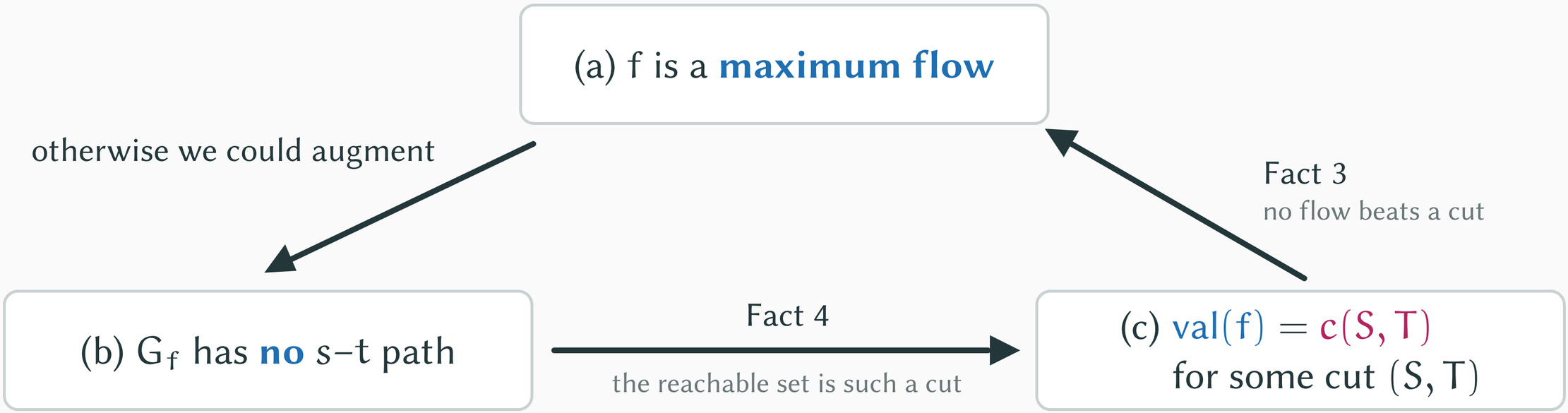
$$f^{\mathrm{out}}(S^*) = \sum_{e \text{ out of } S^*} c_e$$
every edge out of S^* is full

$$f^{\mathrm{in}}(S^*) = 0$$
every edge into S^* is empty

So $\mathrm{val}(f) = c(S^*, T^*)$. □

Ford–Fulkerson is correct

For a flow f, the following three statements are **equivalent**:



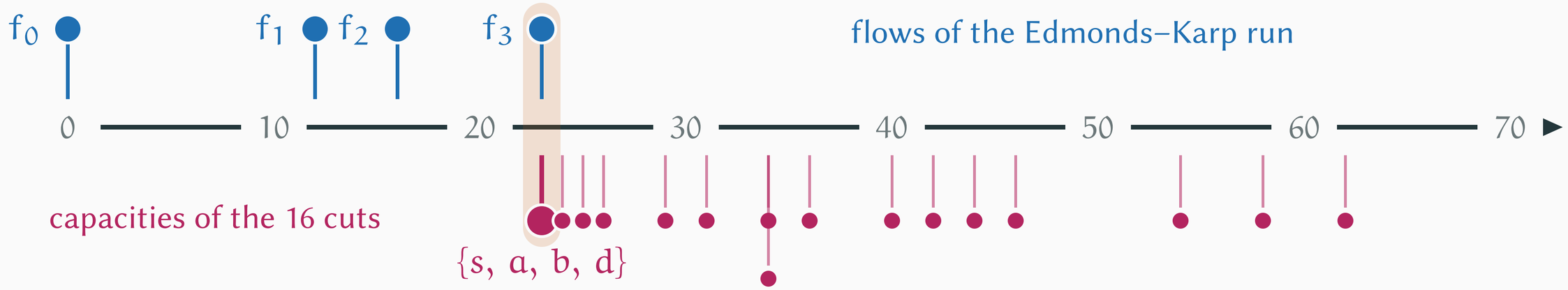
Ford–Fulkerson stops exactly when (b) holds:
its flow is a **maximum flow**.

It stops for integer capacities: Lecture 9.
Edmonds–Karp always stops: Lecture 10.

The max-flow min-cut theorem

Theorem.
In every flow network, the value of a maximum flow equals the capacity of a minimum cut:

$$\max_{f} \operatorname{val}(f) = \min_{(S,T)} c(S,T).$$



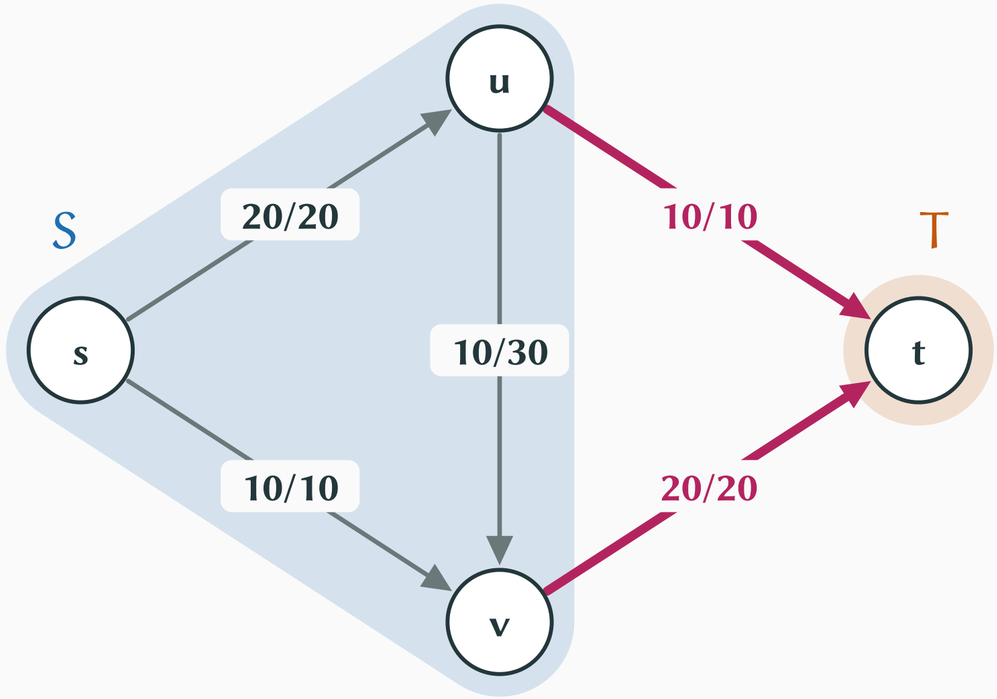
Proof. $\leqslant$: Fact 3 (no flow beats a cut).

$\geqslant$: Edmonds–Karp stops (Lecture 10) with a flow f^* whose G_{f^*} has no s–t path.

Fact 4 (the reachable set is a tight cut): $c(S^*, T^*) = \operatorname{val}(f^*)$.

So $\max_f \operatorname{val}(f) \geqslant \operatorname{val}(f^*) = c(S^*, T^*) \geqslant \min_{(S,T)} c(S,T)$. □

Finding a minimum cut



$c(S, T) = 10 + 20 = 30$

? *How do we find the **value** of a minimum cut?*

Run Ford–Fulkerson (or Edmonds–Karp):
it is the value of the flow it returns.

? *How do we find a **minimum cut**?*

Search the last G_f once more:
S^* = the nodes reached from s (here $\{s\}$).

Time: that of Ford–Fulkerson (or Edmonds–Karp),
plus one search, $O(m)$.

Not unique: $\{s, u, v\}$ is a minimum cut too.

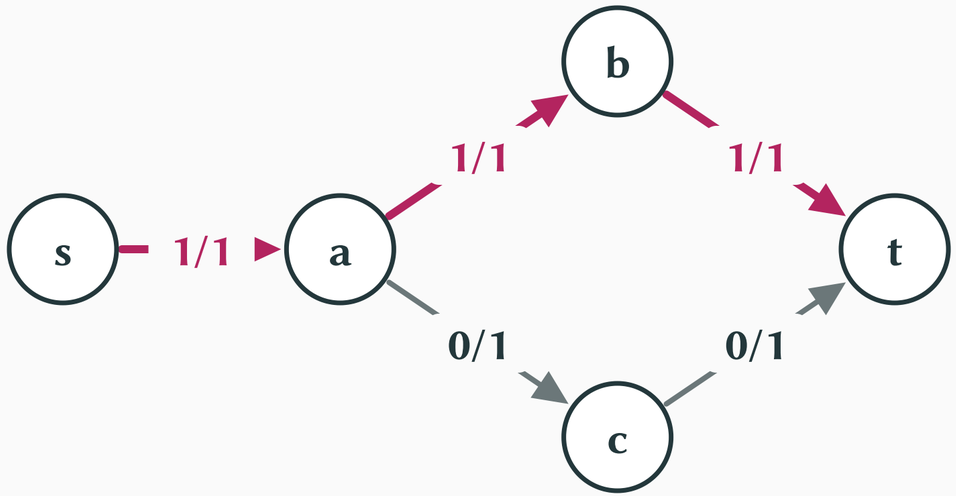
Integrality

Theorem.
If all capacities are integers, some maximum flow has an **integer** value $f(e)$ on every edge.

1. Ford–Fulkerson starts with $f = 0$: integral.
2. If f is integral, so are the residual capacities and the bottleneck b, hence so is $f' = f \pm b$ on P.
3. It stops with a maximum flow (today): an integral one. □

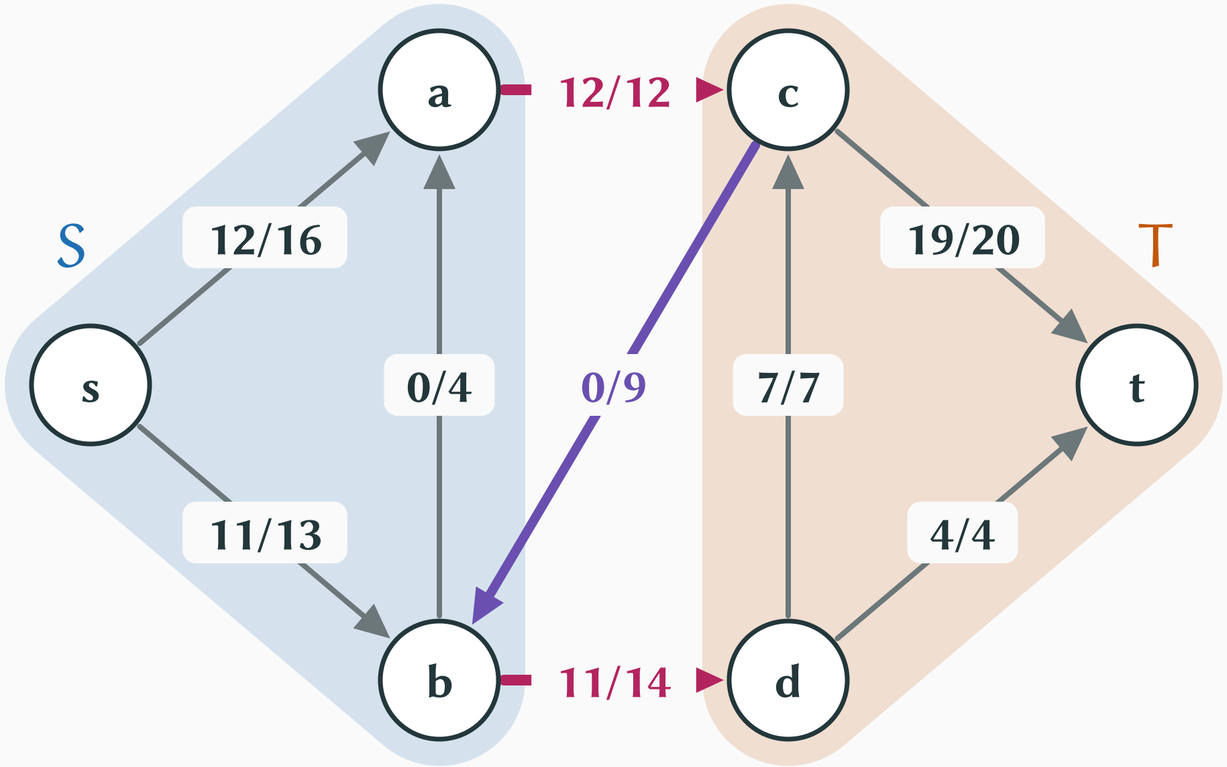
Why it matters: with capacities 1,
flows are yes/no choices (Lecture 12: matchings).

a maximum flow that Ford–Fulkerson finds



value 1 in both cases

Try it: cuts of a network



$S = \{s,\ a,\ b\}$

$T = \{c,\ d,\ t\}$

capacity of the cut

$c(S, T) = 12 + 14 = 26$

net flow across it

$f^{out}(S) - f^{in}(S) = 23 - 0 = 23$

gap = unused capacity + flow coming back

$26 - 23 = 3 + 0$

Summary

1. **Cuts.** $c(S, T)$ counts the edges **from** S **to** T.
   The net flow across every cut is $\mathrm{val}(f)$, so $\mathrm{val}(f) \leqslant c(S, T)$.
2. **The cut from** G_f**.** If G_f has no s–t path,
   the nodes reachable from s form a cut with $c(S^*, T^*) = \mathrm{val}(f)$.
3. **Consequences.** Ford–Fulkerson returns a maximum flow
   and a minimum cut; integer capacities give an integral flow.

> Max-flow min-cut: the value of a maximum flow
> equals the capacity of a minimum cut.

Reference: KT, Chapter 7.2; CLRS, Chapter 24.2