COL 351: Analysis and Design of Algorithms

Lecture 31

Mathematical Formulation

Given: A directed network $G = (V, E, c)$ with

- source node s, and sink node t.
- Capacity function: Edge e has a capacity $c(e) \geq 0$.

Define: $f_{out}(x) = \sum_{(x,\, y)\, \in E} f(x, y)$, and similarly $f_{in}(x) = \sum_{(y,\, x)\, \in E} f(y, x)$

Maximize: $f_{out}(s)$ or $f_{in}(t)$

Subject to:

1. $0 \leqslant f(e) \leqslant c(e)$, for $e \in E$

2. $f_{out}(x) = f_{in}(x)$, for $x \neq s, t$

Construction of Residual Graph

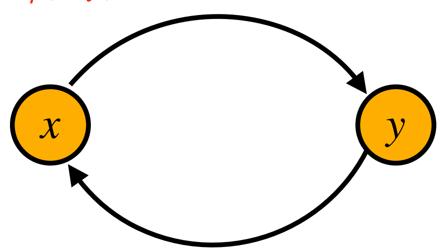
For each $(x, y) \in E(G)$:

If $c(x,y) - f(x,y) > 0$	Include (x,y) in G_f and set $c_r(x,y) = c(x,y) - f(x,y)$	Forward edge
If $f(x,y) > 0$	Include (y,x) in G_f and set $c_r(y,x) = f(x,y)$	Backward edge

$c(x,y) = 50$

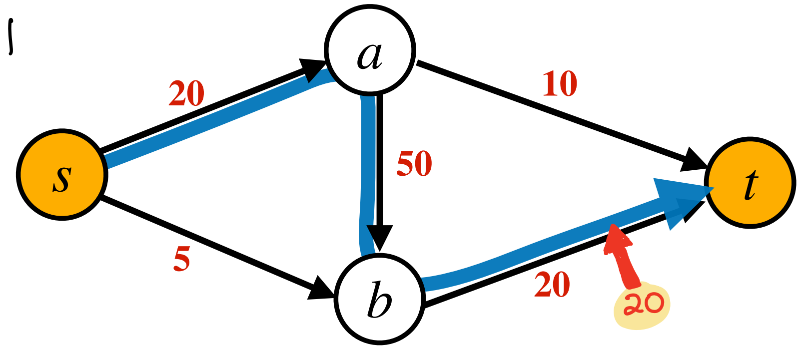
$f(x,y) = 30$

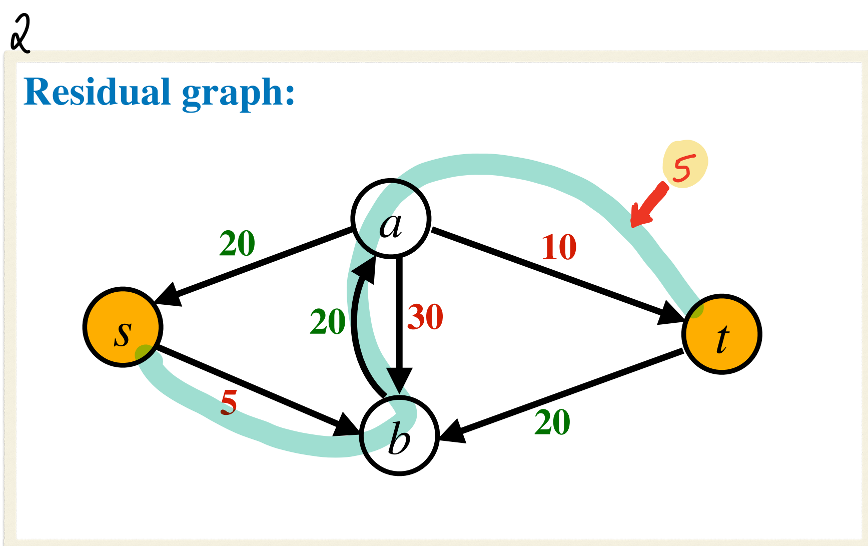
$c_r(x,y) = 50 - \overset{30}{\cancel{20}} = \cancel{30}\ 20$



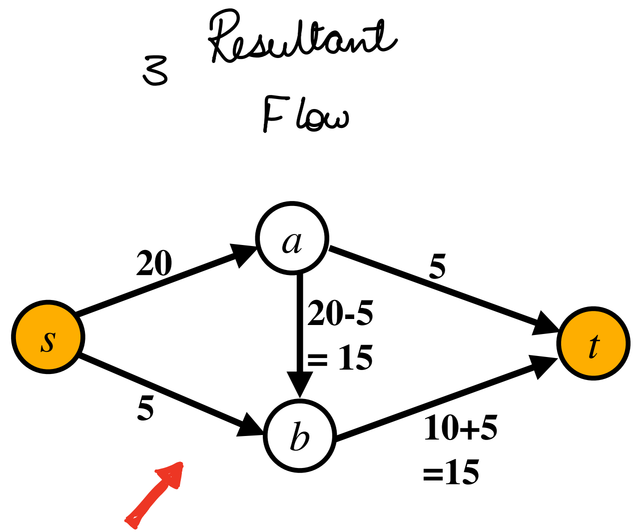
$c_r(y,x) = 30$

Increasing Flow using Residual graph





Introduce reverse edges
that can cancel flows.



Residual graph G_f here will have no (s,t) path as both out-edges of s are fully saturated

Ford Fulkerson Algorithm

Ford-Fulkerson-algo(G, s, t):

1. Initialise $f = 0$
2. **While**($\exists\ s \to t$ path in G_f):

    2.1 Let P be an $s \to t$ path in G_f

    2.2 Let $c_{min} = \min\{c(e) \mid e \in P\}$

    2.3 **For each** $(x, y) \in P$:

    If (x, y) is forward edge : $f(x, y) = f(x, y) + c_{min}$

    If (x, y) is backward edge : $f(x, y) = f(x, y) - c_{min}$

3. Return f.

* To compute G_f we look at current (s,t)-flow and original capacities in G

Is this a valid flow?

Ford Fulkerson Algorithm

<u>Ford-Fulkerson-algo(G, s, t):</u>

1. Initialise $f = 0$
2. **While**($\exists\ s \to t$ path in G_f):

   2.1 Let P be an $s \to t$ path in G_f

   2.2 Let $c_{min} = \min\{c(e) \mid e \in P\}$

   2.3 **For each** $(x, y) \in P$:

   If (x, y) is forward edge : $f(x, y) = f(x, y) + c_{min}$

   If (x, y) is backward edge : $f(x, y) = f(x, y) - c_{min}$
3. Return f.

⊛ For any (x,y), $f_{new}(x,y) \leq C_{original}(x,y)$ if flow passed in forward direction

Is capacity constraint satisfied?

$$x \xrightarrow[f_{old}(x,y)=10]{c(x,y)=50} y$$

$\Rightarrow$ In $G_{f_{old}}$ $c_r(x,y) = 50-10 = 40$

$\Rightarrow$ $C_{min} \leq 40$

$f_{new} = f_{old} + C_{min} \leq 10+40 = 50$

Ford Fulkerson Algorithm

Ford-Fulkerson-algo(G, s, t):

1. Initialise $f = 0$
2. **While**($\exists\ s \to t$ path in G_f):

   2.1 Let P be an $s \to t$ path in G_f

   2.2 Let $c_{min} = \min\{c(e) \mid e \in P\}$

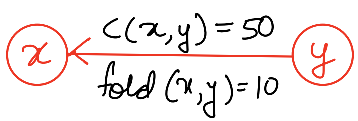
   2.3 **For each** $(x, y) \in P$:

   If (x, y) is forward edge : $f(x, y) = f(x, y) + c_{min}$

   If (x, y) is backward edge : $f(x, y) = f(x, y) - c_{min}$
3. Return f.

(*) For any (x,y), $f_{new}(x,y) \geq 0$ if flow passed in Backward direction

Is capacity constraint satisfied?



$\Rightarrow$ In $G_{f_{old}}$ $c_r(y,x) = 10$

$\Rightarrow$ $c_{min} \leq 10$

$f_{new} = f_{old} - c_{min} = 10 - c_{min} \geq 0$

Ford Fulkerson Algorithm

Ford-Fulkerson-algo(G, s, t):

1. Initialise $f = 0$
2. **While**($\exists\, s \to t$ path in G_f):

    2.1 Let P be an $s \to t$ path in G_f

    2.2 Let $c_{min} = \min\{c(e) \mid e \in P\}$

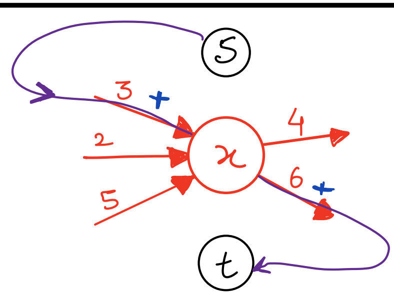
    2.3 **For each** $(x, y) \in P$:

    If (x, y) is forward edge : $f(x, y) = f(x, y) + c_{min}$

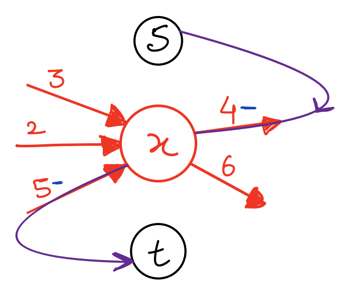
    If (x, y) is backward edge : $f(x, y) = f(x, y) - c_{min}$

3. Return f.

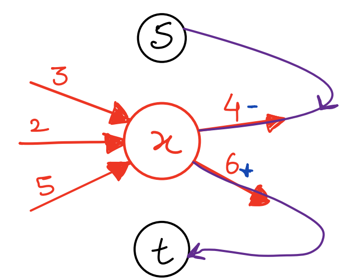
Is flow at each node conserved?



Case 1: fin, fout incremented by Cmin

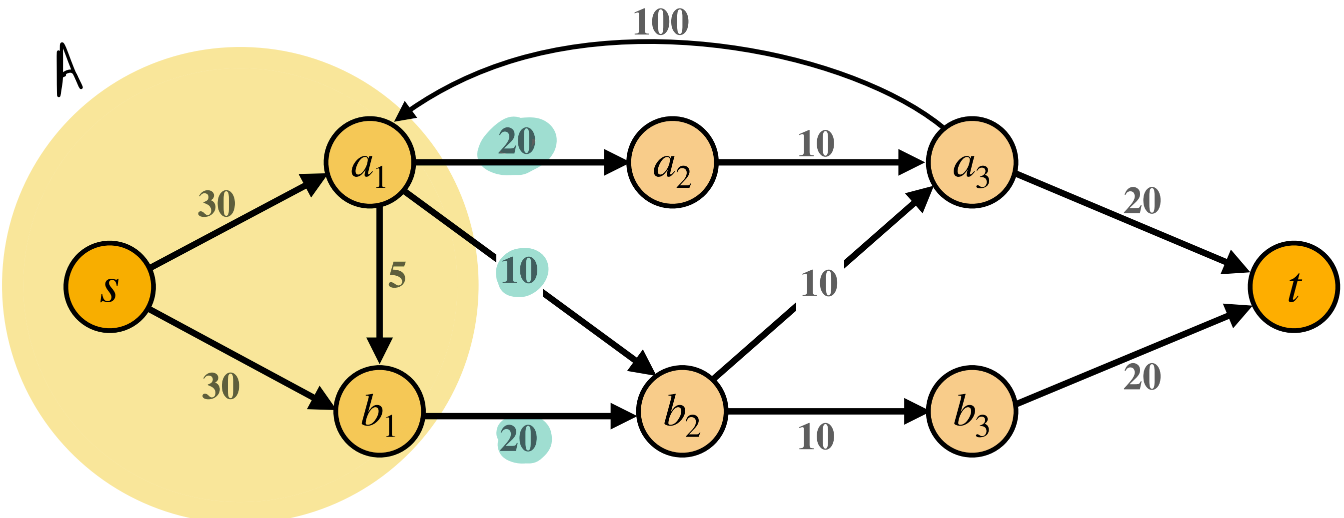


Case 2: fin, fout decremented by Cmin



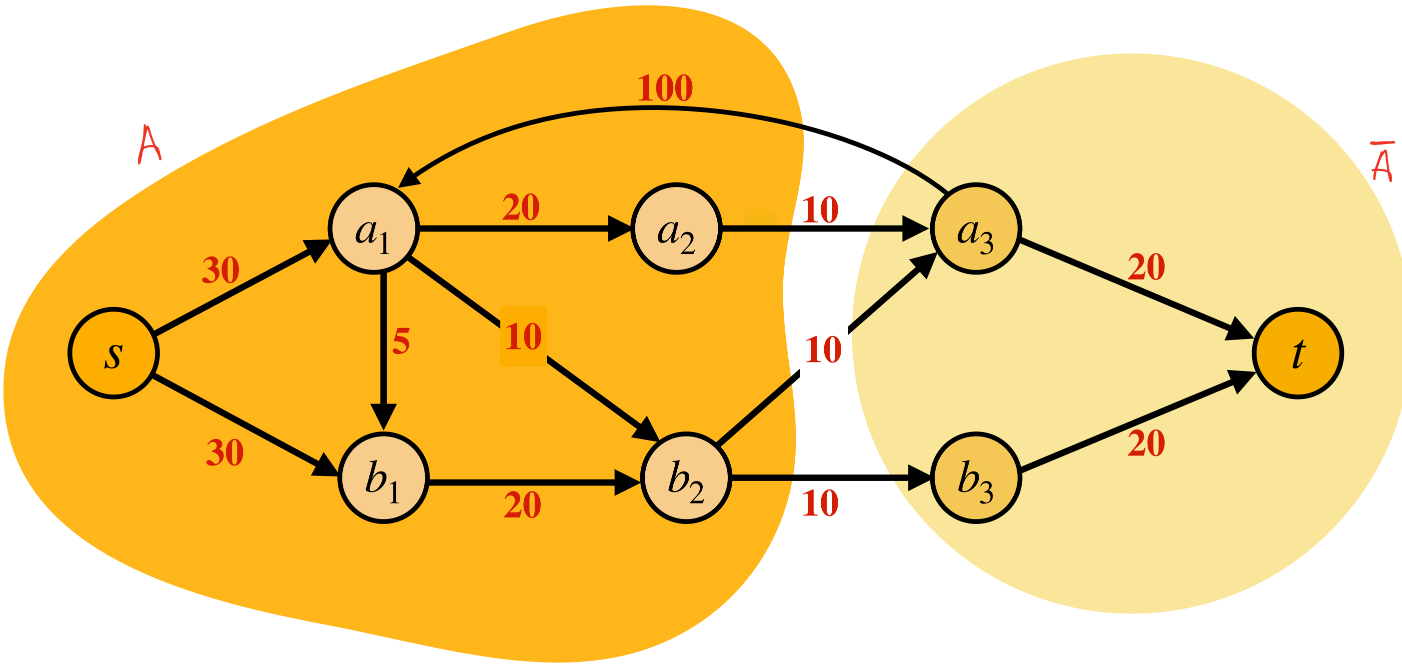
Case 3: fin, fout remains same

Natural Upper bound on (s,t)-max-flow



Value of max-flow $\leq 20 + 10 + 20 = 50$

Natural Upper bound on (s,t)-max-flow



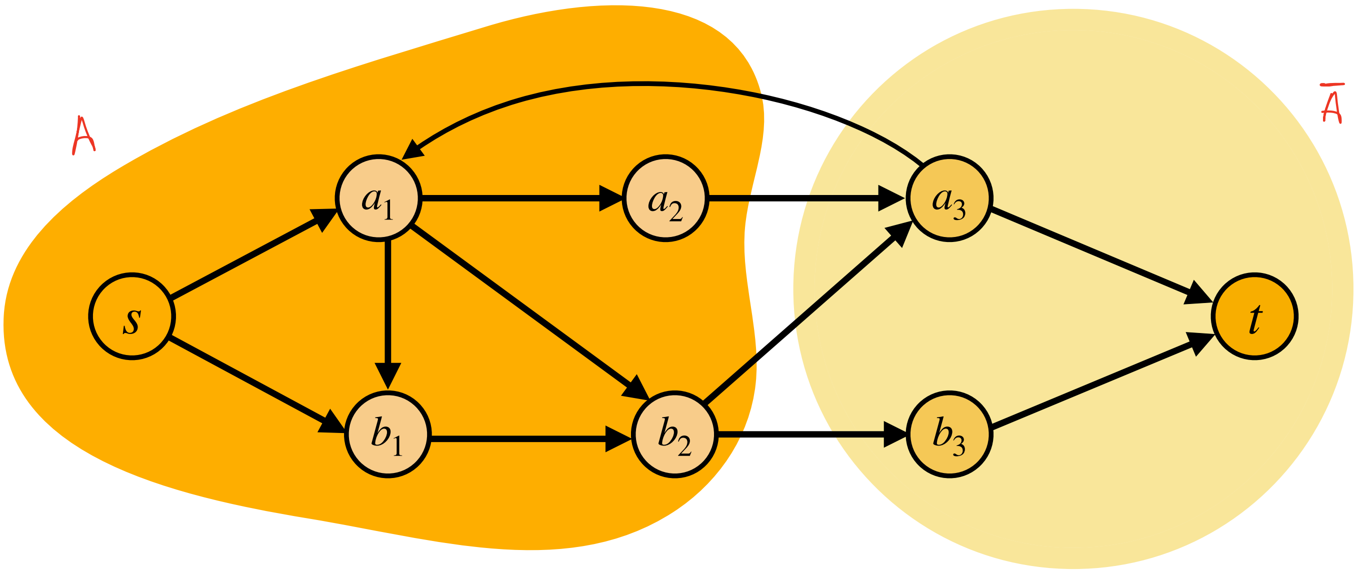
Lemma 1:

For any partition $(A, \bar{A})$ of vertices with $s \in A,\ t \in \bar{A}$,

$$(s,t)\text{-max-flow-value} \ \leq \sum_{(x,y)\in(A\times\bar{A})\cap E} c(x,y)$$

Proof will follow from Property on last slide

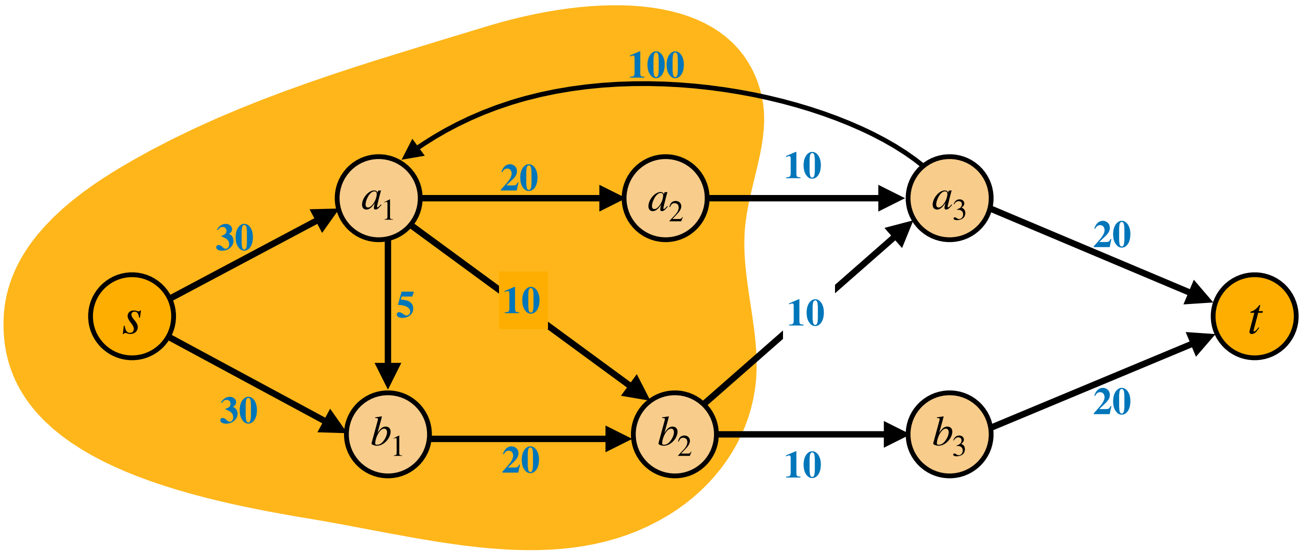
(s,t)-Cuts



Definition:

Any partition $(A, \bar{A})$ of vertices with $s \in A,\ t \in \bar{A}$

Definitions



Definition: For any cut $(A, \bar{A})$,

$$c(A, \bar{A}) = \sum_{\substack{(x,y) \in \\ (A \times \bar{A}) \cap E}} c(x, y)$$

Capacity of cut

$$f_{out}(A) = \sum_{\substack{(x,y) \in \\ (A \times \bar{A}) \cap E}} f(x, y)$$

out-flow

$$f_{in}(A) = \sum_{\substack{(x,y) \in \\ (\bar{A} \times A) \cap E}} f(x, y)$$

in-flow

Property of Flows & Cuts

Property: For any $(s, t)-$cut $(A, \bar{A})$ and any flow f,

$$\text{value}(f) \;=\; f_{out}(A) - f_{in}(A)$$

Homework : Provide mathematical proof