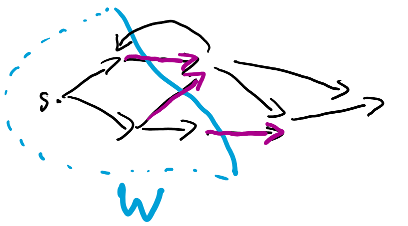
13 Max Flow, Min Cut Part II

Thursday, February 20, 2025 13:58

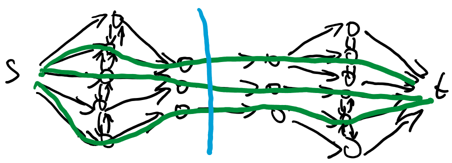
$G=(V,E)$ $s,t\in V$

st-cut $W\subseteq V$ $s\in W$ $t\notin W$



size of cut $\delta(W)=\sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} c_{uv}$

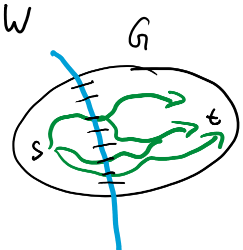
min st-cut is the smallest cut



max st flow
send as much flow as possible from s to t

min st cut
which edges form the bottleneck for sending more flow.

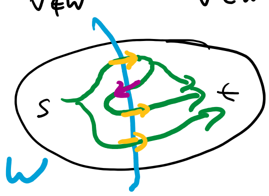
Lemma: For any st flow f and any st cut W
$\text{size}(f)\le\delta(W)$



Proof: $\text{size}(f)=\sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} f_{uv}-\sum_{\substack{(u,v)\in E\\ u\notin W\\ v\in W}} f_{uv}\le\sum_{\substack{(u,v)\\ u\in W\\ v\notin W}} f_{uv}$

how much is sent from s to t

$\delta(W)=\sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} c_{uv}$



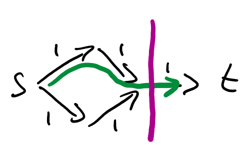
$\le\sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} c_{uv}=\delta(W)$

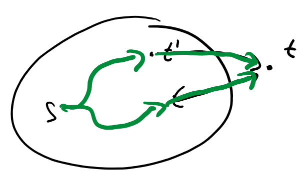
Let f st flow, W st cut, $\text{size}(f)=\delta(W)$

If there was a st flow f' with $\text{size}(f')>\text{size}(f)$
then $\text{size}(f')>\delta(W)$
$\Rightarrow$ contradiction to Lemma above

$\Rightarrow$ f is maximum

Similar argument tells us W is minimum





Algorithm from Tue. computed a flow
but is that output a max flow?

Idea: Find a cut that has same size as the flow returned by algorithm
then flow must be maximum.

Idea: Find a cut that has same size as the flow returned by algorithm
then flow must be maximum.

Claim: Given $G=(V,E)$, st flow f where G_f has no path from s to t

→ then f is maximum.

→ then $\exists W \subseteq V$ with $\text{size}(f) = \delta(W)$

Let $U \subseteq V$ be all vertices reachable from s in G_f

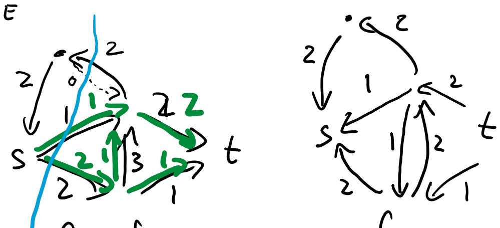
G_f

there are no outgoing edges in G_f from in W to out W

1) flow f never sends anything from outside W to inside W

2) for any edge in G that goes from inside to outside W, $f_{uv} = c_{uv}$

$$\text{size}(f) = \underbrace{\sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} f_{uv}}_{2)\,=\,c_{uv}} - \underbrace{\sum_{\substack{(u,v)\in E\\ u\notin W\\ v\in W}} f_{uv}}_{1)\,=\,0} = \sum_{\substack{(u,v)\in E\\ u\in W\\ v\notin W}} c_{uv} = \delta(W)$$



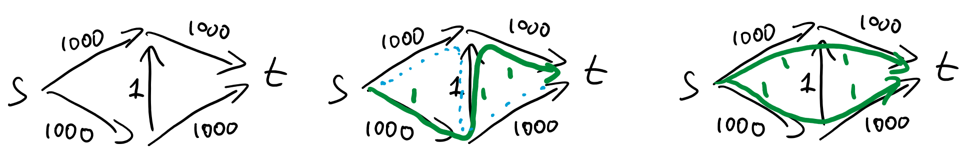
G, f $\qquad$ G_f

⇒ f is max flow
W is min cut

⇒ output of algo is optimal

2: 58

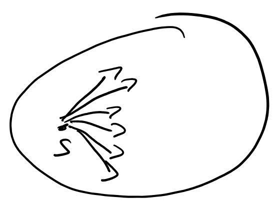
Tue algo $\quad O(F\cdot|E|) \quad F$ size of max flow



G $\qquad$ G $\qquad$ G

Pseudo polynomial because F scales wrt capacities

$$F \leq n \cdot \max_{(u,v)\in E} c_{uv}$$



Vant: $O\left(|E|^2 \cdot \log\left(\max_{(u,v)\in E} c_{uv}\right)\right)$

Input $(G=(V,E),\ s, t)$

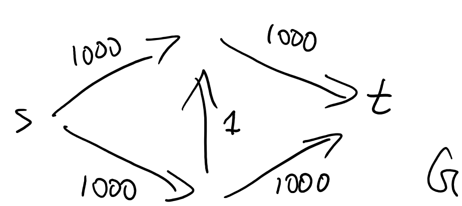
$f_e = 0 \quad \forall e\in E$

$\Delta := \max_{(u,v)\in E} c_{uv}$ rounded to next largest power of 2

While $\Delta \geq 1$

  While True

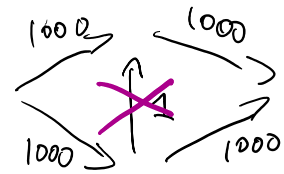
    construct G_f $\qquad$ capacity in G_f



G

```
while True
    construct G_f                                 capacity in G_f
    delete all edges from G_f with capacity < Δ
    if path s~>t exists in G_f
        augment f by path
    else
        break
Δ = Δ/2
```

G



G_f

- $O(|E|)$ iterations per Δ — to prove
- there are $\log(\max c_{uv})$ many Δs
- $O(|E|)$ time to find one path

$\Longrightarrow O(|E|^2 \log(\max c_{uv}))$