Admin:

→ PSet 2 Exit survey:
- some good feedback - keep it coming!
- actionable feedback best (but not required!)

↳ $\boxed{\text{PSet 3 Exit Survey out today}}$

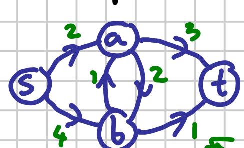
→ $\boxed{\text{QUIZ I IS COMING!}}$
- 3/14 (π-DAY!) 7:30-9:30 PM
- Covers Lectures 1-9
- Review sessions: 1) 3/10 - focus on prob. solving strategies 2) 3/11 - material review

---

Today: Maximum Flow Problem:
→ network flow
→ flow decomposition
→ s-t cuts
→ residual graph
→ augmenting paths
→ max flow / min cut theorem
→ Ford-Fulkerson alg.

Max Flow: One of the most central graph problems
→ used in logistics, routing, connectivity analysis
→ surprisingly versatile: Lots of other problems can be reduced to it (will see that in 6.046!)

Setup: Network = directed graph $G=(V,E)$



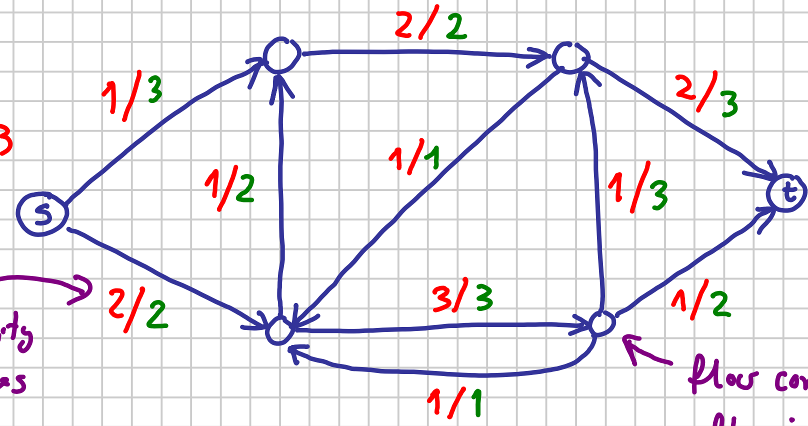
& source vertex $s \in V$
& sink vertex $t \in V$
& edge capacities $c: E \to \mathbb{R}^{\geq 0}$

define: $\boxed{c(u,v) = 0 \text{ if } (u,v) \notin E}$

Intuition: → Each edge = one-way road
→ capacity = # of lanes
→ Question: What $\underbrace{\text{rate}}_{\text{max}}$ of s→t traffic can we support?

Example: Flow

"Rate"/value = 3



feasibility: flow ≤ capacity for all edges

flow conservation: flow-in = flow-out for all $v \neq s,t$

Natural notation for flow: (used by CLRS)

"gross" flow: $g: E \to \mathbb{R}^{\geq 0}$

$g(u,v)$ = amount of flow on edge (u,v)

$\Rightarrow$ want: $\to$ $0 \leq g(u,v) \leq c(u,v) \quad \forall_{(u,v) \in E}$ (feasibility)

$\to$ $\sum_u g(u,v) - g(v,u) = 0 \quad \forall_{v \neq s,t}$ (flow conserv.)

But: This definition is a bit unwieldy to work with due to dealing with flow cycles of length 1 & 2

no "useful" flow

"useful" flow (net flow) = 1 unit from u to v

Definition we will use in class:

Net flow: $f : \underbrace{V \times V}_{} \to \mathbb{R}$ ← can be negative!

Defined for any u & v

$\Rightarrow$ Want: $\to$ $f(u,v) \leq c(u,v) \quad \forall_{u,v \in V}$ (feasibility)

$\to$ $\sum_u f(u,v) = 0 \quad \forall_{v \neq s,t}$ (flow conservation)

$\to$ $f(u,v) = -f(v,u) \quad \forall_{u,v}$ (skew symmetry)

$\Rightarrow$ Skew symmetry implies that:

Eliminates flow cycles of length 1

$\to$ $f(u,u) = 0 \quad \forall_u$

$\to$ $f(u,v)$ = net flow from u to v

Eliminates flow cycles of length 2

$\boxed{\Rightarrow \text{ if } f(u,v) < 0, \text{ flow of } |f(u,v)| \text{ from } v \text{ to } u}$

Note: $f(u,v) = 0$ if $(u,v) \notin E$ & $(v,u) \notin E$

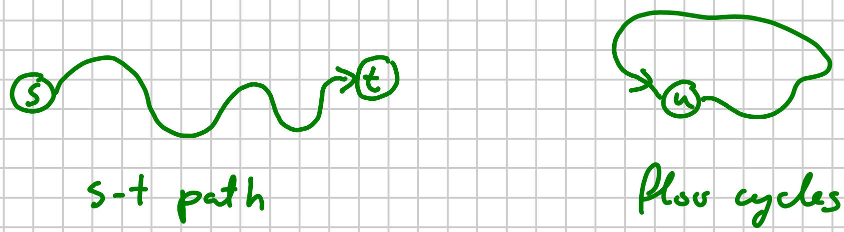
Value of flow f: $\quad |f| = \sum_v f(s,v)$

Maximum flow problem:

Given a network $G = (V, E, s, t, c)$, find a flow in G of maximum value

---

What are the "simplest" examples of non-zero flows?



Observation: Any flow can be decomposed into a collection of s-t flow paths and flow cycles

$\Rightarrow$ s-t flow paths & flow cycles = elementary "building blocks" of flows

Formally: Let $supp_f(G)$ = subgraph of G of edges (u,v) with $f(u,v) > 0$

$\Rightarrow$ $supp_f(G)$ = graph of "gross" flows of f

Lemma (Flow decomposition): For any flow f, $|f| \geq 0$ $supp_f(G)$ can be decomposed into a collection of s-t flow paths & flow cycles

Proof: By induction on # of edges in $supp_f(G)$

1) $supp_f(G)$ has no edges $\Rightarrow$ claim trivially holds

2) $supp_f(G)$ has some edges

(*) **Crucial obs.:** By flow conservation, if $(u,v) \in supp_f(G)$ & $v \neq t$ $\Rightarrow$ $(v,u') \in supp_f(G)$ for some u'

4

Case (a): $(s,v) \in \mathrm{supp}_f(G)$ for some v

→ use (*), possibly many times, to find an s-t path OR an s-s cycle P in $\mathrm{supp}_f(G)$

→ s-t path / cycle P found!

→ Remove P from $\mathrm{supp}_f(G)$ by reducing f on each edge $e \in P$ by bottleneck capacity

$$\boxed{C_f(P) = \min_{e \in P} f(e)}$$

→ this removes at least one edge from $\mathrm{supp}_f(G)$ (the edge that is $\arg\min_{e \in P} f(e)$) & keeps $|f| \geq 0$

⇒ Can use inductive claim to what remains!

Case (b): $(v,t) \in \mathrm{supp}_f(G)$ for some v

→ follow this edge backwards, analogously to Case (a)

⇒ again, identifies an s-t path or cycle that we can remove

(This case is not really needed, as if no edge leaves s in $\mathrm{supp}_f(G)$ then no edge enters t in $\mathrm{supp}_f(G)$. But we do not need to prove that. Exer. Prove it!)

Case (c): all edges (u,v) in $\mathrm{supp}_f(G)$ have $u \neq s$ & $v \neq t$

(this is the last case to cover)

⇒ $|f| = 0$ (since $|f| \geq 0$)

⇒ no edges adjacent to s or t in $\mathrm{supp}_f(G)$

→ Use (*), possibly many times, on u to find a u-u cycle

→ Proceed as before to remove it & use inductive claim

$\boxed{4}$

Exercise: Wlog flow decomposition has ≤ # of edges s-t flow paths & flow cycles

Exercise 2: If $|f| > 0$ then there has to be ≥ 1 s-t flow path (can't have only flow cycles)

We understand now the structure of flow

How about <u>finding</u> the maximum flow?

Let $\boxed{f^* = \text{a max flow in } G}$ (f^* does not need to be unique)

& $F^* = |f^*|$ (max flow value)

<u>Warmup question:</u> How to check if $F^* > 0$?

Let G^+ = subgraph of edges with positive capacities

<u>Note:</u> If there exist an s-t path P in G^+

$\Rightarrow F^* > 0$ since P can support positive flow

<u>But:</u> By flow decomposition Lemma

this is not only sufficient but also necessary!

(By Exercise 2, $F^* = |f^*| > 0 \Rightarrow \exists$ s-t path in $\mathrm{supp}_{f^*}(G) \subseteq G^+$)

<u>So:</u> Existence of an s-t path in G^+ <u>certifies</u> $F^* > 0$

<u>But:</u> How to <u>certify</u> that $\nexists$ s-t path in G^+, i.e., that $F^* = 0$?

$\Rightarrow$ Use a cut!

Let $\hat{S} = \{ v \in V \mid \exists \text{ s-v path in } G^+ \}$

$\Rightarrow s \in \hat{S}$

$\Rightarrow$ if $F^* = 0$, i.e., $\nexists$ s-t path in G^+,

then $t \notin \hat{S}$, i.e., $t \in V \setminus \hat{S}$

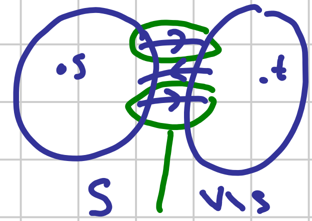
$\Rightarrow \hat{S}$ is an $\boxed{\text{s-t cut}}$ that separates s from t

6

Def. s-t cut = a cut $(S, V\setminus S)$ s.t. $s \in S$ & $t \in V\setminus S$

→ capacity of a cut:

$$c(S) = \underbrace{c(S, V\setminus S)}_{\text{set notation - implicit sum}} = \sum_{u\in S} \sum_{v \in V\setminus S} c(u,v)$$



contribute to $c(S)$

⇒ $c(S)$ = total capacity of edges that leave S

(edges entering S don't count)

⇒ $c(S) = 0 \Leftrightarrow$ cut separates s from t

(so, $\hat{S}$ above had $c(\hat{S}) = 0$)

To summarize:

$F^* = 0 \Leftrightarrow \nexists$ s-t path in $G^+ \Leftrightarrow \exists$ s-t cut S w/ $c(S) = 0$

So, existence of a (separating) s-t cut implies non-existence of an s-t path

⇒ s-t paths and s-t cuts are $\boxed{\text{dual}}$ to each other

---

There is more !

minimum s-t cut problem: Given $G = (V, E, s, t, c)$

find an s-t cut of minimum capacity

Let S^* be the min capacity s-t cut

i.e., $\boxed{S^* = \underset{S \text{ s-t cut}}{\arg\min} \; c(S)}$ (min s-t cut might not be unique)

(I)

<u>So far, we concluded</u>: $F^* = 0 \iff c(S^*) = 0$

(Recall: Some might be < 0)

For an s-t cut S & flow f,

let $$f(S) = f(S, V \setminus S) = \sum_{u \in S} \sum_{v \in V \setminus S} f(u,v)$$

$\Rightarrow$ $f(S)$ = <u>net</u> flow across the cut

(II)

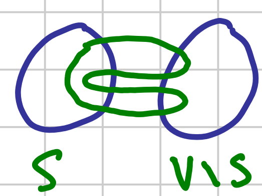
By feasibility, $\boxed{f(S) \leq c(S)}$ for any s-t cut S

<u>Claim I</u>: $f(S) = f(S')$, for <u>any</u> two s-t cuts S & S'

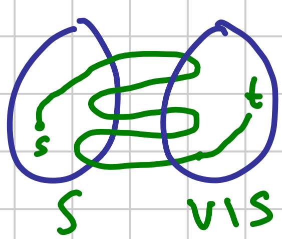
<u>Proof</u>: Flow decomposition $\Rightarrow$ f collection of flow cycles & s-t flow paths

$\Rightarrow$ Every flow cycle contributes <u>zero</u> to both $f(S)$ & $f(S')$

(zig-zags cancel out)



$\Rightarrow$ Every s-t flow path contributes the same amount of flow to both $f(S)$ & $f(S')$



(odd # of zigs & even # of zags

$\Rightarrow$ zig-zags cancel out, except the last one!)

4

Claim I

$\Rightarrow$ Since $\boxed{|f| = f(\{s\})}$ $\Rightarrow$ $\boxed{|f| = f(S) \text{ for any s-t cut } S}$ (III)

$\Rightarrow$ **Max flow $\leq$ min s-t cut:**

$$F^* = |f^*| \overset{(III)}{=} f^*(S^*) \overset{(II)}{\leq} c(S^*)$$ (IV)

(Weak duality of flows and s-t cuts)

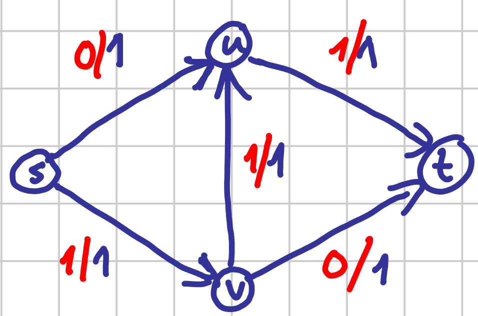
↳ More in Lecture 9

Back to finding max flow:

Qs: Given a flow f, how to increase its value (or conclude that $F^* = |f|$)?

Idea: Just try to find an s-t path to push more flow along it.

Problematic case:



→ Value $|f| = 1$

→ Can't push more flow through (s,v) or (u,t) ("saturated")
$\Rightarrow$ All s-t paths "clogged"

Idea: Undo some of the existing flows!

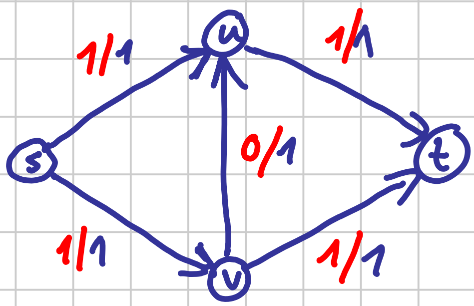
→ Is there no way to increase the flow?

→ If we <u>decrease</u> the flow on (v,u) the <u>net</u> u-v flow will <u>increase</u>

(Note: even though there is no edge (u,v) in G working with net flow enables us to still talk about pushing a flow from u to v)

⇒ Increasing flow on (s,u) & (v,t) by 1 and decreasing the flow on (u,v) by 1 increases the value of the flow by 1

⇒ <u>New flow</u>:



→ Value $|f'| = 2$

→ Can we improve it further?

→ No! By weak duality of flows & s-t cuts (IV)

$$2 = |f'| \leq F^* \leq c(S^*) \leq c(\underbrace{\{s\}}) = 2$$

example cut with capacity = 2

(Could have also used $\{s,u\}$ or $\{s,u,v\}$ here)

$$\Rightarrow \boxed{F^* = 2}$$

$\Rightarrow$ $\boxed{f' \text{ is a max flow}}$

How to look for such "non-obvious" s-t path to increase the flow algorithmically?

Residual network $G_f = (V, E_f, s, t, c_f)$ of flow f in network G

→ residual capacities

$$c_f(u,v) = c(u,v) - f(u,v)$$

(= how much extra net $u \to v$ flow can send)

Note: By feasibility of f, $\boxed{0 \le c_f(u,v) \le c(u,v) + c(v,u)}$

→ Edge $(u,v) \in E_f$ whenever $c_f(u,v) > 0$

(discards saturated edges)

Observe: If f a flow in G & f' a flow in G_f

then $f + f'$ is a flow in G

⇒ Reduces improving a flow f to finding a non-zero flow in G_f

But: What if it is impossible to find a non-zero flow in G_f?

⇒ By our previous discussion, ∃ s-t cut S with $\boxed{c_f(\hat{S}) = 0}$

↳ residual capacity of S

$$c_f(S) = \sum_{u \in S} \sum_{v \in V \setminus S} c_f(u,v)$$

$$\boxed{\begin{aligned}&\textbf{Observe:}\ \text{For any } s\text{-}t \text{ cut } S,\\ &c_f(S) = \sum_{u\in S}\sum_{v\in V\setminus S} c_f(u,v) = \sum_{u\in S}\sum_{v\in V\setminus S} c(u,v) - f(u,v) =\\ &= c(S) - f(S)\end{aligned}}$$

So, if $c_f(\hat{S}) = 0$ then

$$c(\hat{S}) = f(\hat{S}) \overset{\text{Claim I}}{=} |f|$$

$\Rightarrow$ By (IV), $c(\hat{S}) = |f| \leq F^* \leq c(S^*) \leq c(\hat{S})$

$\Rightarrow$ $|f| = \boxed{F^* = c(S^*)} = c(\hat{S})$

$\Rightarrow$ f is a max flow & $\hat{S}$ is the min s-t cut !

Summary:

1. f is a max flow, i.e., $|f| = F^*$
   iff max flow value of G_f is 0
   (iff $\nexists$ s-t path in G_f)

2. Max Flow - Min Cut Thm:

$$\boxed{F^* = c(S^*)}$$

Strong duality of flows & s-t cuts

↳ More in Lecture 9

Ok, but how about a max flow alg.?

Augmenting path = directed s-t path in G_f

→ Can push additional flow along such path P up to (residual) bottleneck capacity

$$\boxed{c_f(P) = \min_{(u,v)\in P} c_f(u,v)}$$

(increases $|f|$ by $c_f(P)$)

(V) **Note:** f is max flow iff $\nexists$ augmenting path

Ford-Fulkerson: [1956]

→ Start with a zero flow

→ Find an augmenting path P (e.g. via DFS)

→ Augment the flow by pushing $c_f(P)$ along P

→ Repeat until no more augmenting paths

$O(E)$ time each

Correctness? Follows from (V)

Running time?

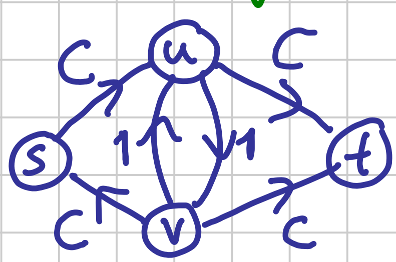
① if capacities are integers $\in [0, C]$

then $\leq |f| \leq c(\{s\}) \leq \underbrace{|V|}\cdot C$ augmentations

(since $c_f(P) \geq 1$ always)

($\leq |V|$ edges leaving s)

→ Can happen!



Alternate paths:

$s \to u \to v \to t$

& $s \to v \to u \to t$

⇒ Total Runtime: $\boxed{O(E \cdot V \cdot C)}$

→ pseudopolynomial algorithm = polynomial in sum of input numbers
i.e., polynomial in unary encoding of input
(NOT binary)

→ Note: Since all capacities integral
⇒ F-F alg. finds a max flow with integral flows on all edges
(since $c_f(p)$ is always integral)

⇒ Flow integrality thm:
If G has all capacities integral then $\exists$ max flow that is integral too

(2) If capacities rational
⇒ running time still finite (& pseudopoly.)

(3) If capacities are real
⇒ potentially infinite runtime!