Applications of flows and cuts

(and, time permitting)

Circulations

Kent

(and, time permitting)

min-cost flow

Thursday, March 15

Team	Wins	Games left
New York	92	2
Baltimore	91	3
Toronto	91	3
Boston	89 +	2 = 91

Can Boston catchup?

Team	Wins		Games left	
New York	92		2	
Baltimore	91		3	
Toronto	91		3	
Boston	90	+	2	92

Can Boston catch up?

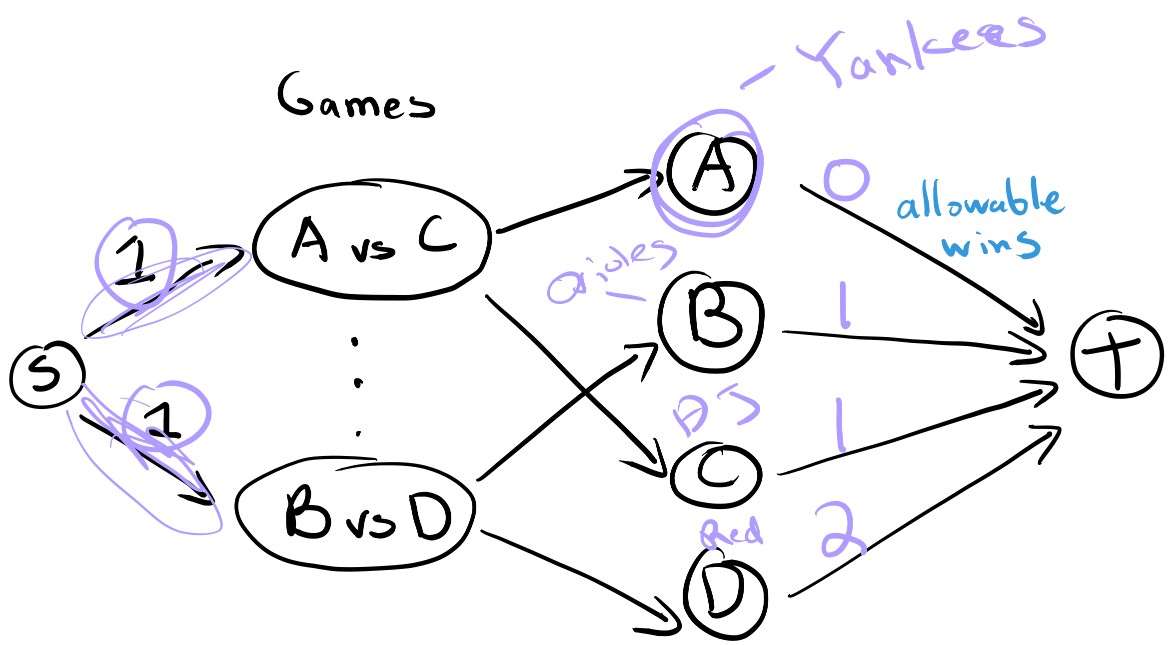
(wins)	(92) New York	92 (91) Baltimore	92 → 93 (91) Toronto	(90) Boston
New York	×	~~1~~	~~1~~	0
92 Baltimore	1	×	1	~~1~~
92 Toronto	1	1	×	~~1~~
Boston	0	1	1	×

Modeling w/ max flow

each win is 1 unit of flow.

each games gives 1 win (unit of flow)

to one of the two teams



Input: Set of teams S

1 for each team $x \in S$, w_x = # of wins of team x

2 for each pair of teams x, y, $g(x,y)$ = # games left between x and y

3 favorite team z

- z can win if: (for some m)

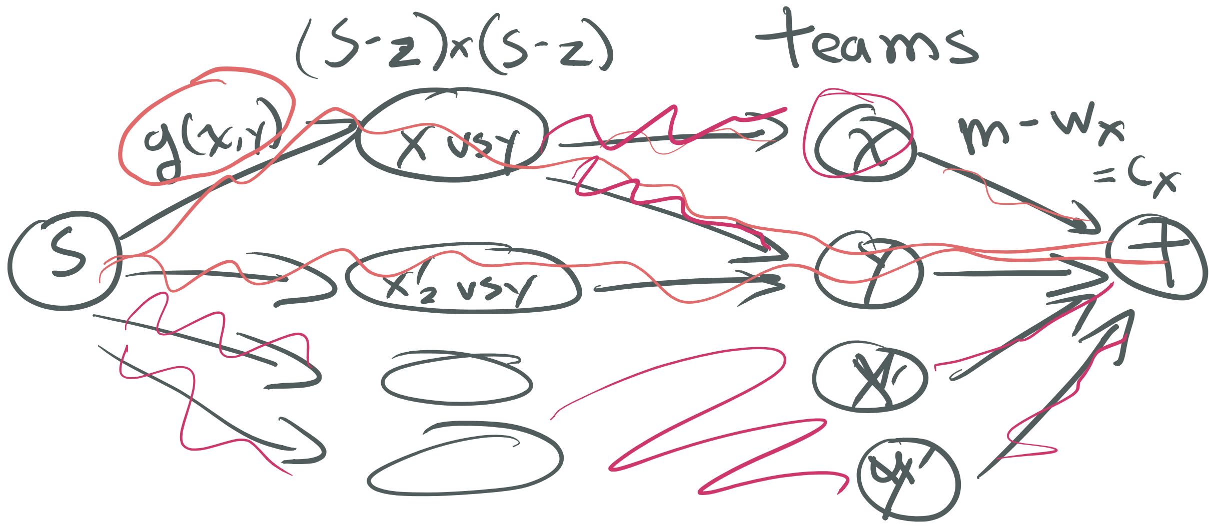
z wins at least m

all other teams win $\leq m$

$$m = \left(\sum_{y \in S-z} g(y,z) \right) + w_z$$

I want each other $y \in S-z$ to win

$$\leq m - w_y = c_y$$



1. compute max flow => value F

2. if $F = \sum_{x,y} g(x,y)$ then answer yes

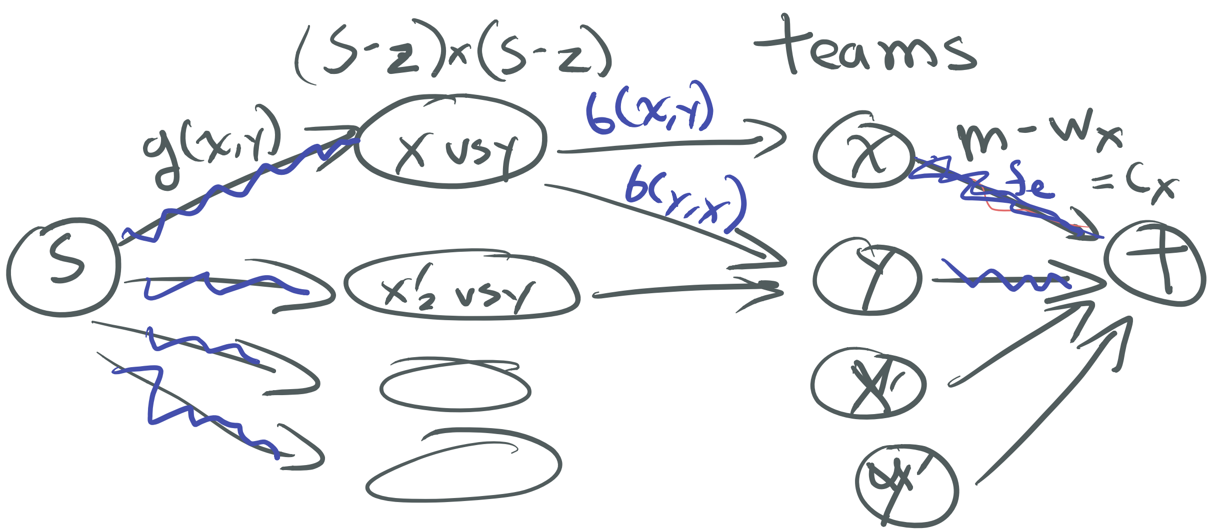
3. else answer no

team z is eliminated iff $F < \sum_{x,y} g(x,y)$

(a) suppose z is not eliminated.

let $b(x,y)$ be the # times x beats y, [e.g. $b(y,x) = g(x,y) - b(x,y)$]

construct a flow f



need to show that this flow is feasible and check it has value F

if $f_e > m - w_x$, $\Rightarrow \sum_y' b(x,y) > m - w_x$

$$w_x + \sum_y' b(x,y) > m \qquad \boxtimes$$

opposite direction:
give integral flow f w/ value F
then I can find $b(x,y)$ s.t. z is still in first

Project scheduling

n projects (numbered) $1, 2, \ldots, n$

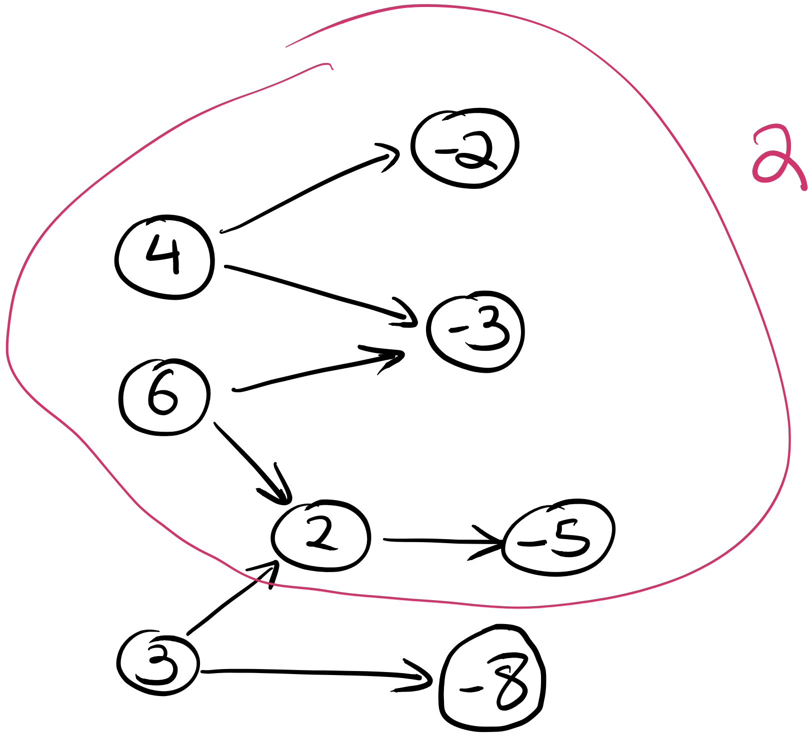
dependencies of the form

"project i depends on project j."

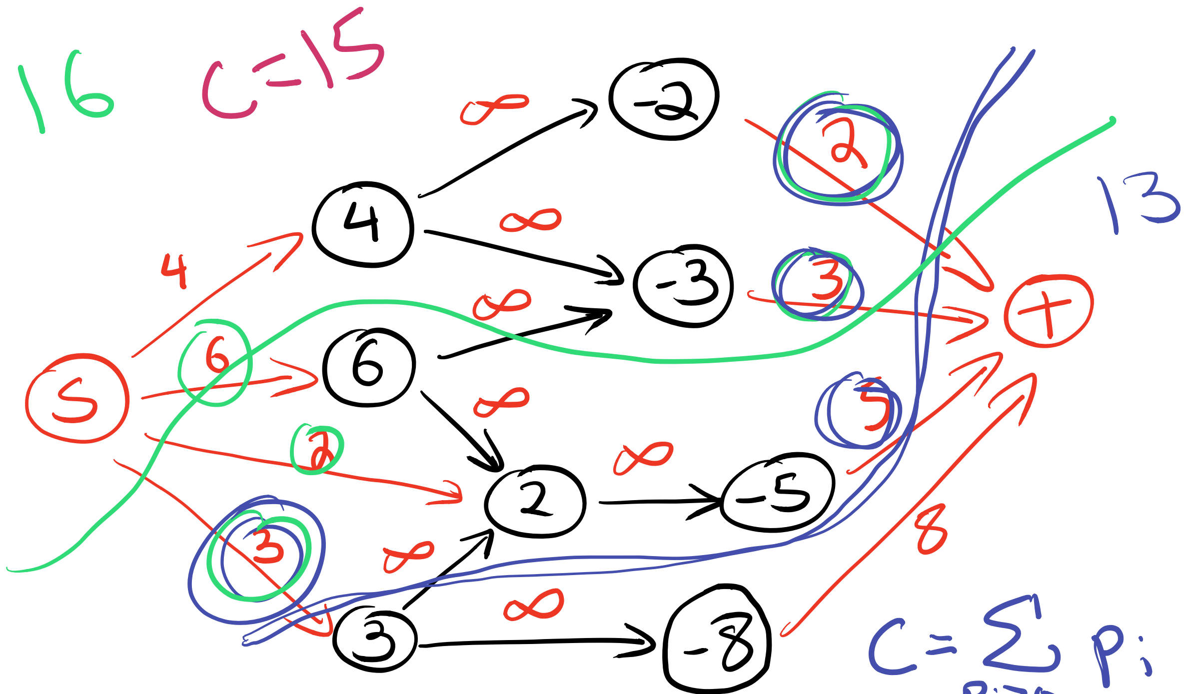
(Assume dependencies are acyclic)

Each project has cost/profit p_i
(negative p_i is a cost)

goal: find maximum profit set of projects



4
6
3
-2
-3
2
-5
-8
2



$$C = \sum_{p_i > 0} p_i$$

say cut value $D \Rightarrow$ total profit = [redacted]

look for min- $s \to t$ cut; component containing s is the projects we take

how to make the aux.

- for each i w/ $p_i > 0$, add an edge from s to i w/ capacity p_i
- for each i w/ $p_i < 0$, add an edge from i to t w/ capacity $-p_i$

- for each dependency $i \to j$, add an edge from i to j w/ capacity ∞

<u>Proof</u>

given a cut defined by $S \subset V$. ($\partial(S)$ = edges leaving S)

take all projects in S.

if cost of $\partial(S) < \infty$, then (I claim) I didn't break any dependencies.

in opp. direction,

if I have "feas. projects"

and $S = \{s\,t : \ldots\}$.

Given a feasible set of projects $P \subset [n]$ of total profit α, there is an s-t cut of weight

$\alpha = C - D$

sum of all profits   cut weight

$D = \cancel{\alpha}\ C - \alpha$

---

given cut of weight D,

we can make ~~set~~ set of feas. projects w/ sum profit $C - D$

More formally:

[reduction, slowly]

applications ✓

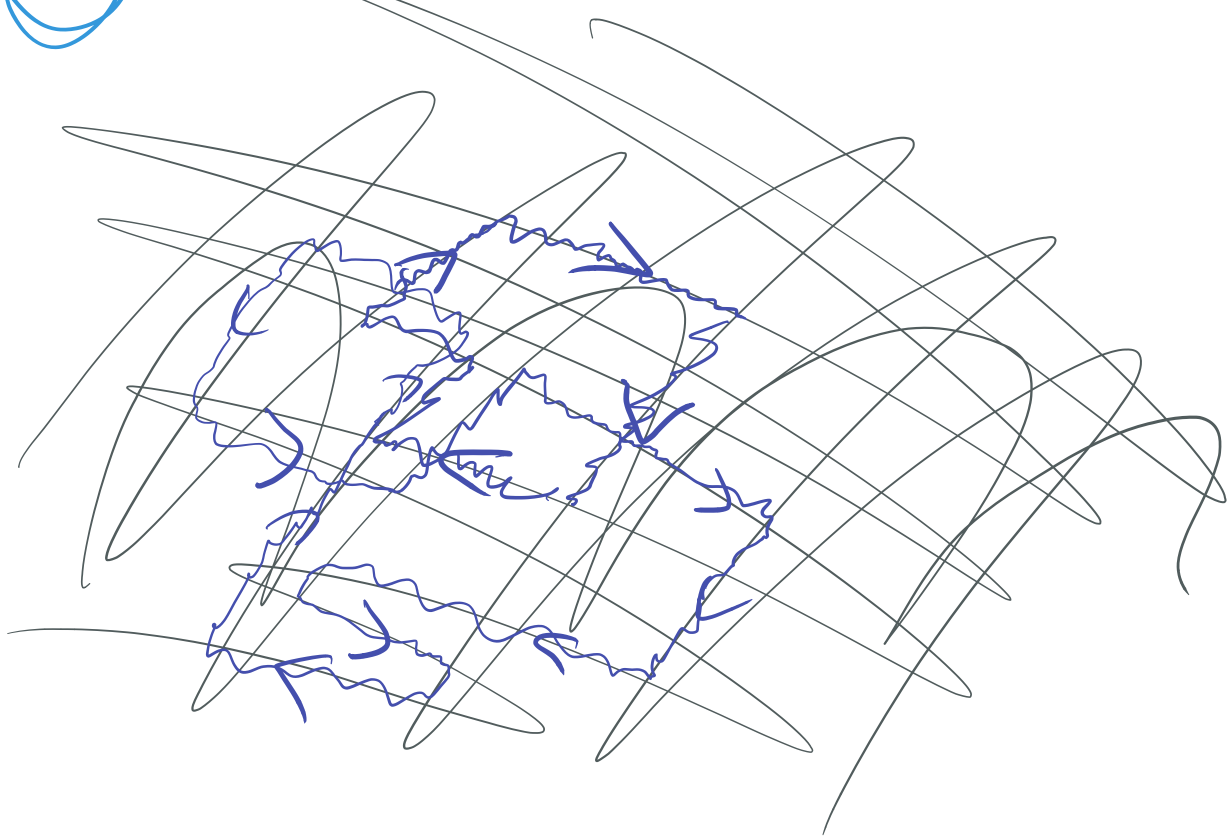
1. circulations
2. min cost max flow

Circulations $G=(V,E)$ w/ capacities c.

"circulation": $f: E \to \mathbb{R}_{>0}$

(a) $\sum_{e \text{ into } v} f(e) = \sum_{e \text{ out of } v} f(e)$ for each $v \in V$

(b) $f(e) \le c(e)$ for each $e \in E$



circulation problem

capacities lowerbounds

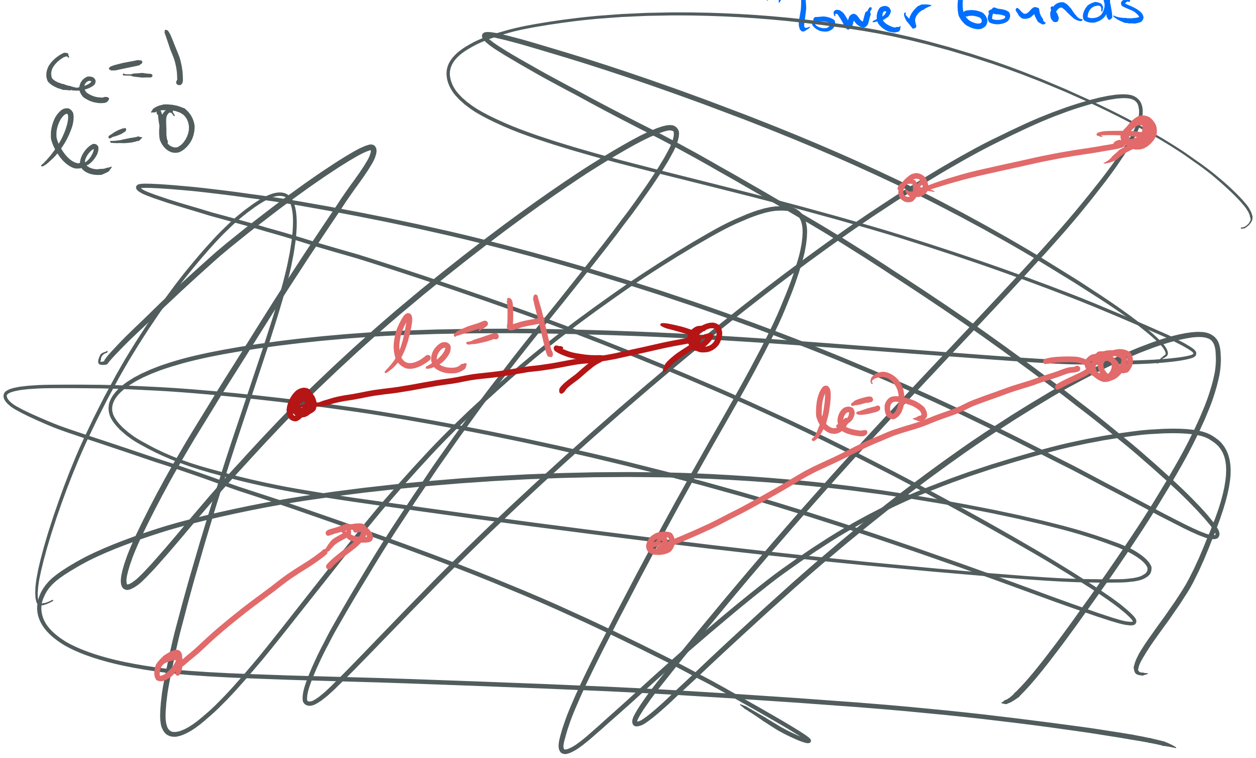
input: $G=(V,E)$, $c: E \to \mathbb{R}_{>0}$, $\ell: E \to \mathbb{R}_{>0}$

goal: find a circulation $f: E \to \mathbb{R}_{>0}$

$f(e) \leq c_e$

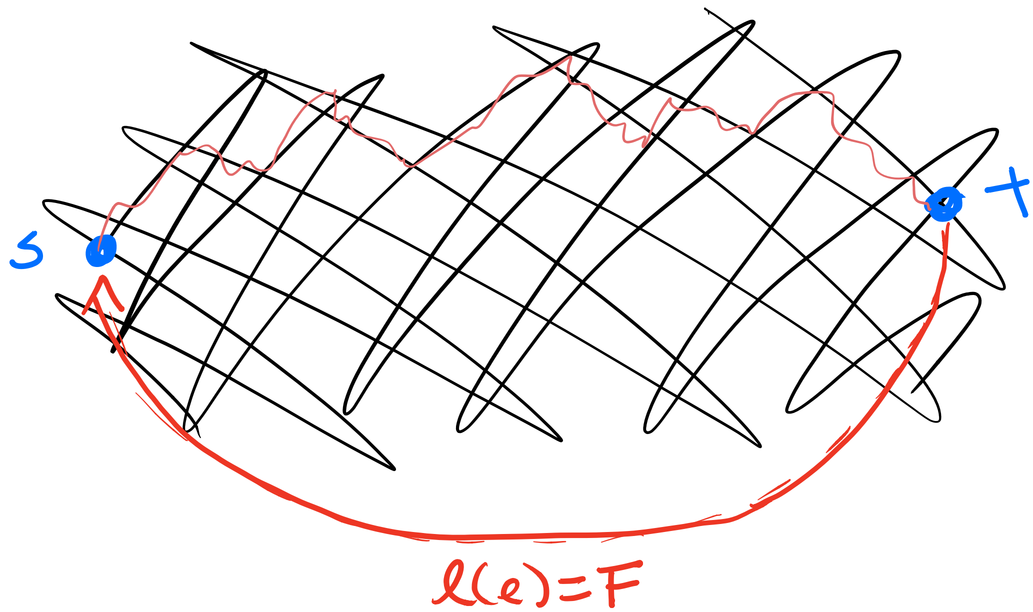
s.t. for each $e \in E$, $f(e) \geq \ell(e)$

"lower bounds"



all black edges have capacity 1

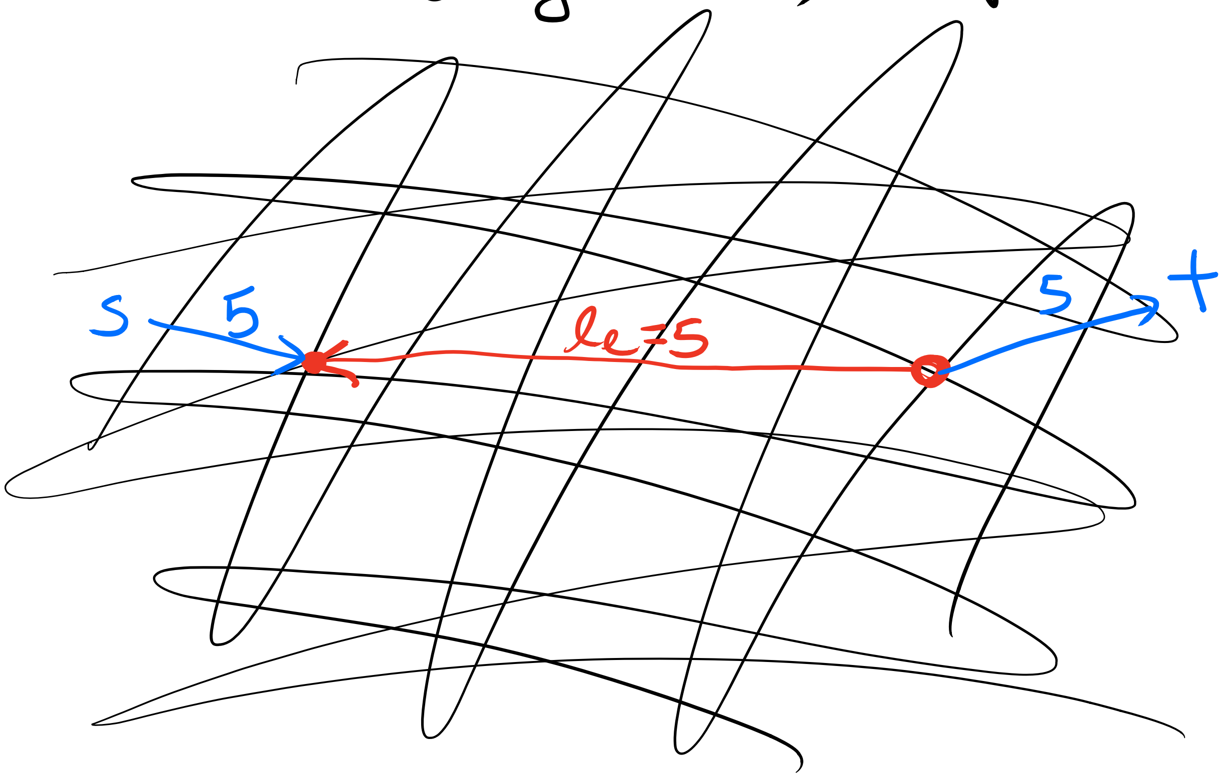
max flow → circulation



circulation → max flow

circulation → max flow

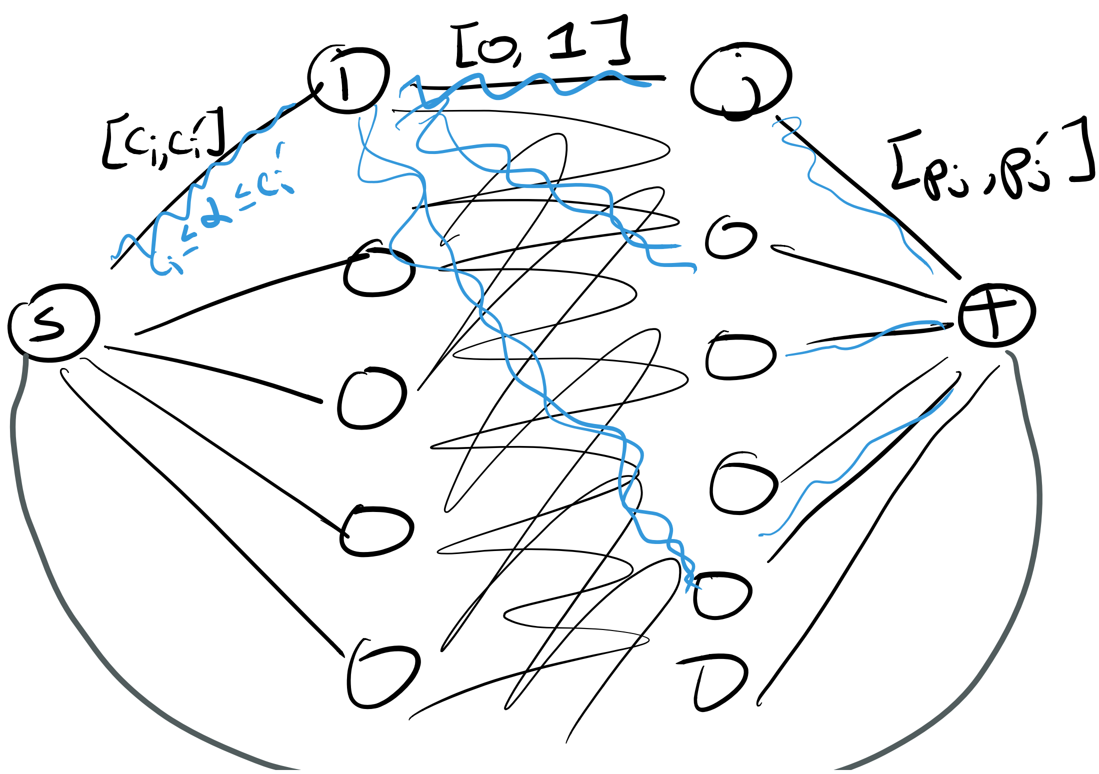
(see Kleinberg-Tardos, Chap 7)



- Important properties
  - O(mn) time
  - integral capacities, lower bounds ⇒ integer valued circulation
  - Hoffman's circulation theorem equiv. to max-flow min cut
  - circulation decomposes to m cycles

Survey design

- k products, ℓ customers
- ask customers about purchased products
- ask a customer i about at least c_i products and at most c_i' products
- ask about a product j at least p_j times and at most p_j' products



$[0,\infty]$

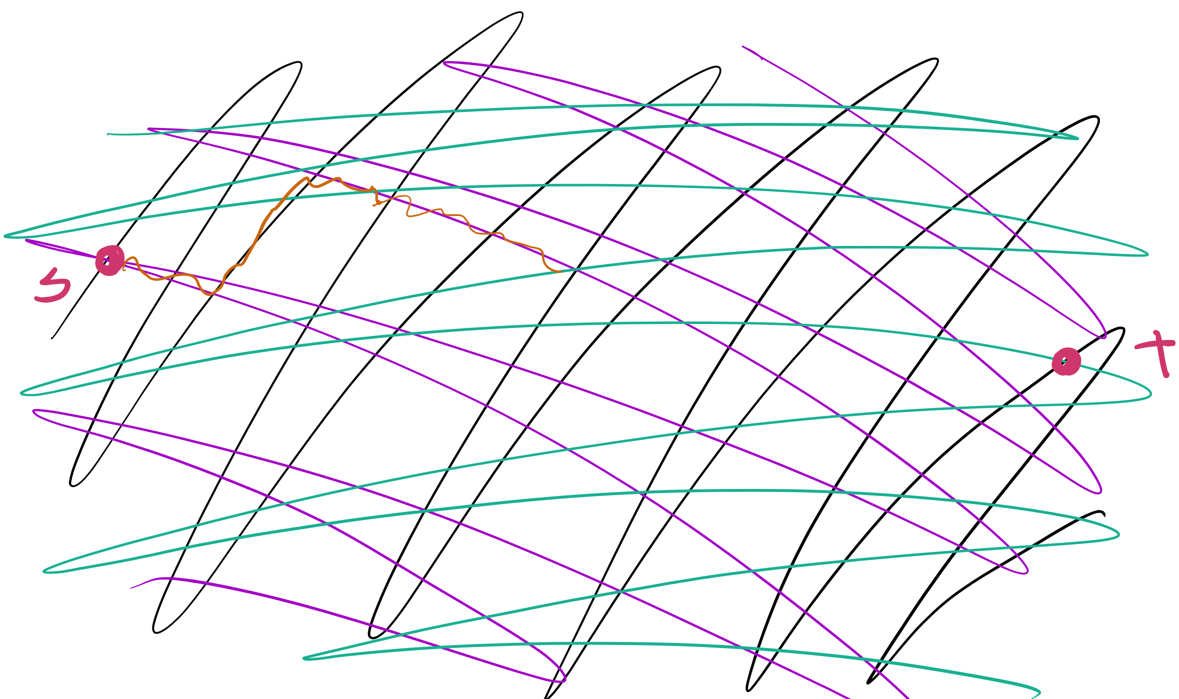
Min-cost flow

costs

Input: $G=(V,E)$, $c: E \to \mathbb{R}_{\geq 0}$, $w: E \to \mathbb{R}$, target F

Goal: find min-cost flow of size F from s to t

cost of $f: E \to \mathbb{R}_{\geq 0}$ = $\sum_e w_e f_e$



— = cost 1 — = cost 2 — = cost 3

Facts about min-cost max flow

$O(mn \log(C) \log(nW))$ time

[C = max capacity, W = max weight]

$O(m^2 \log n + n^2 \log^2 n)$ "strongly polynomial time"

Today: 2 algorithms

- cycle canceling
- shortest augmenting paths