

Sparsest Cut

Lecture 1

Probability

Events, probabilities,
conditional probabilities,
union bound

Random variables,
expected value,
linearity of expectation
Markov's inequality

Algorithms

Approx. 3-SAT,

Big idea: we only want the average

$$\begin{array}{l} \text{Linearity of Expectation} \\ E[X+Y] = E[X] + E[Y] \quad (\text{stochastic variable}) \\ \text{"average sum = sum of averages"} \end{array}$$

let $Y_i = \begin{cases} 1 & \text{if } i\text{th clause satisfied} \\ 0 & \text{otherwise} \end{cases} \quad (i=1, \dots, m)$

$$E[Y_i] = 7/8 \quad (1 \text{ clause case})$$

$$E[\text{\# clauses satisfied}] = E\left[\sum_{i=1}^m Y_i\right] = \sum_{i=1}^m E[Y_i] = 7/8 m$$

let $Y_i = \begin{cases} 1 & \text{if } i\text{th clause satisfied} \\ 0 & \text{otherwise} \end{cases}$

$$Z = \sum_{i=1}^m Y_i = \text{\# clauses satisfied}$$

$$E[Z] = \frac{1}{2} E[Z | x_1 = \text{true}] + \frac{1}{2} E[Z | x_1 = \text{false}]$$

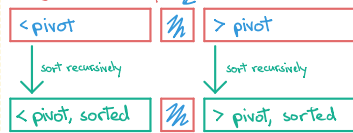
$\Rightarrow$ either $E[Z | x_1 = \text{true}]$ or $E[Z | x_1 = \text{false}]$ is $\geq E[Z]$!

QuickSort,

Random divide and conquer

① pick random 'pivot'

② Divide A into two parts



Expected running time of QuickSort

For each $i, j \in [n]$ w/ $i < j$,

$X_{ij} = \begin{cases} 1 & \text{if } i\text{th smallest element is compared to } j\text{th smallest} \\ 0 & \text{otherwise} \end{cases}$

$\sum_{i,j} X_{ij}$ = total # comparisons made by the algorithm
 $\approx$ total running time

$E[\sum_{i,j} X_{ij}]$ = average running time

$E[\sum_{i,j} X_{ij}] = \sum_{i,j} E[X_{ij}]$

now analyze $E[X_{ij}]$ in isolation

$$E[X_{ij}] = \Pr[X_{ij} = 1] = \Pr[i\text{th element is compared to } j\text{th}]$$

i th and j th elem selected before any element in between happens w/ probability $\frac{2}{j-i+1}$

$$E[X_{ij}] = \frac{2}{j-i+1}$$

$$E[\sum_{i,j} X_{ij}] = \sum_{i,j} E[X_{ij}]$$

$$= \sum_{i,j} \frac{2}{j-i+1}$$

$$= 2 \sum_{i,j} \frac{1}{j-i+1}$$

$$= 2 \sum_{i,j} \frac{1}{j-i+1}$$

$$= 2 \sum_{i,j} \frac{1}{j-i+1}$$

$$= 2 \sum_{i,j} \frac{1}{j-i+1}$$

$$= 2 \sum_{i,j} \frac{1}{j-i+1}$$

Quickselect (HW)

Lecture 2

Probability

Hash Functions,

Universal Hash Functions

$h: [n] \rightarrow k$ s.t. for all pairs $i, j \in [n]$,
 $\Pr[h(i) = h(j)] \leq \frac{1}{k}$ (much weaker than ideal hash)

e.g. $h(x) = (ax + b) \bmod p \bmod k$

where • p prime, $> k$

• a in $\{1, \dots, p-1\}$ unif. random

• b in $\{0, \dots, p-1\}$ unif. random



Markov's Inequality

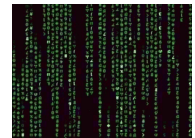
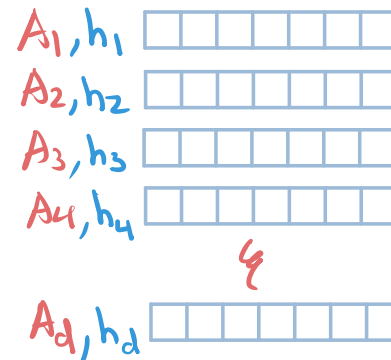
Markov's Inequality (special case)

Let $X \geq 0$ be nonnegative RV

$$\Pr[X \geq 2E[X]] \leq \frac{1}{2}$$

Algorithms

Frequency Estimation, Heavy Hitters (in streams)



estimate f_i w/
 $\min_{j=1, \dots, d} A_j[h_j(i)]$

Lecture 3 (last Tuesday)

Probability

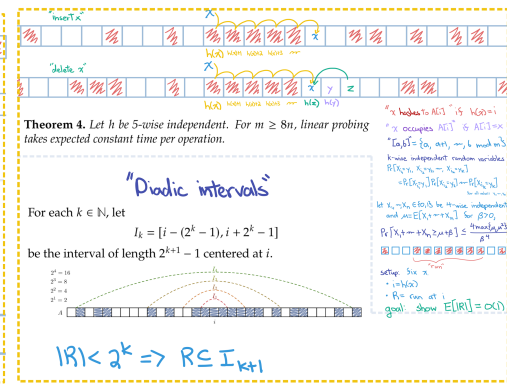
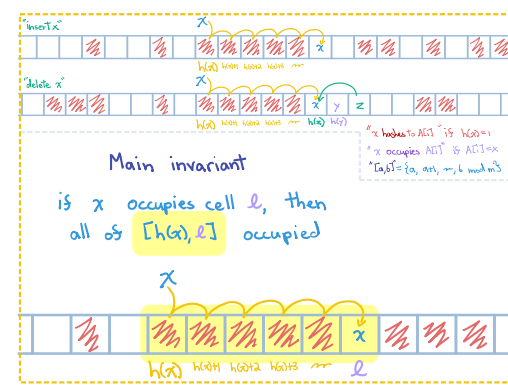
k-wise independence,
higher order moments
(generalizing universal hash
functions and Markov's inequality)

Algorithms

Hash tables w/
chaining +
linear probing

k-wise independent hash functions
 $h: [U] \rightarrow [m]$
s.t. for any k keys $x_1, x_2, \dots, x_k \in [U]$,
 k values $v_1, v_2, \dots, v_k \in [m]$,
 $P[h(x_1)=v_1, h(x_2)=v_2, \dots, h(x_k)=v_k] = \frac{1}{m^k}$
eg. $h(x) = a_0 + a_1x + a_2x^2 + \dots + a_{k-1}x^{k-1} \pmod p$
where $p \geq U$ prime,
 $a_0, a_1, \dots, a_{k-1} \in [p]$ indep, uniformly random

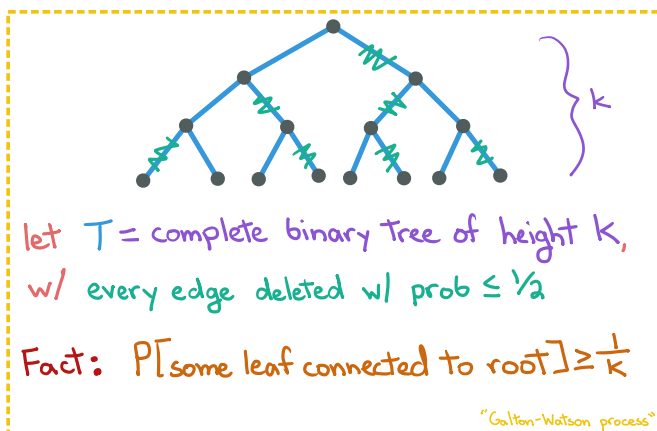
Important lemma
k-wise independent random variables
 $P[X_1=y_1, X_2=y_2, \dots, X_k=y_k]$
 $= P[X_1=y_1] P[X_2=y_2] \dots P[X_k=y_k]$
for all values $y_1, \dots, y_k$
let $X_1, \dots, X_n \in \{0,1\}$ be 4-wise independent
and $\mu = E[X_1 + \dots + X_n]$. then for $\beta > 0$,
 $P[X_1 + \dots + X_n \geq \mu + \beta] \leq \frac{4 \max\{\mu, \mu^3\}}{\beta^4}$



Lecture 4

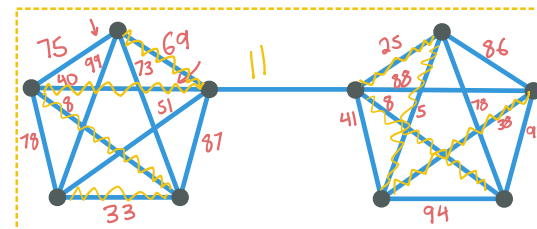
Probability

Galton-Watson processes



Algorithms

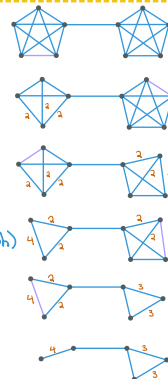
Random Contractions for min cut



Randomized contractions

Input: $G=(V,E)$, weights $c \in \mathbb{R}_{>0}^E$
(unweighted)

- ① while $|E| > 1$
 - Ⓐ sample $e \in E$ in proportion to c
 - Ⓑ "contract" e
- ② return set of edges (in original graph) contracted to remaining edge

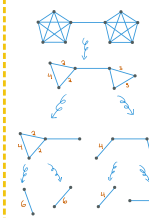


Amplification by branching

Instead of restarting at the beginning, restart partway through

High-level algorithm

- ① randomly contract until prob. of ruining the min-cut is $\sim 1/2$
- ② Branch: make two copies of current graph recurse on both copies



Lecture 5

Probability

Multiplicative Chernoff Bound

Multiplicative Chernoff bound

let $X_1, \dots, X_n \in [0, 1]$ be independent

$$\mu = E[X_1 + \dots + X_n]$$

$$\epsilon \in (0, 1)$$

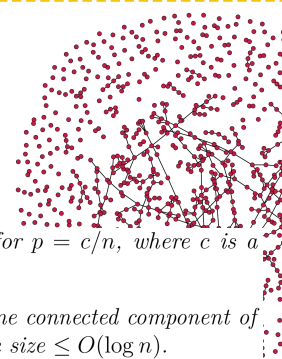
- $P[X_1 + \dots + X_n \geq (1 + \epsilon)\mu] \leq e^{-\epsilon^2 \mu / 3}$
- $P[X_1 + \dots + X_n \leq (1 - \epsilon)\mu] \leq e^{-\epsilon^2 \mu / 2}$

Algorithms (/ results)

Random graphs & the giant component

Erdős-Rényi graphs $G(n, p)$

- n vertices
- each edge sampled independently w/ probability p



Theorem 5.3. Consider a random graph $G \sim \mathcal{G}(n, p)$ for $p = c/n$, where c is a constant.

- If $c > 1$, then with high probability, there is exactly one connected component of G with $\Omega(n)$ vertices, and all other components have size $\leq O(\log n)$.
- For $c < 1$, then with high probability, all connected components of G has size $< O(\log n)$.

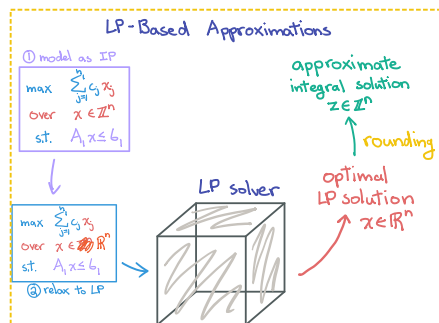
Lecture 6

Probability (& techniques)

Approximations via linear programming & randomized rounding

Linear programs (LPs)
Similar to IPs, except no integrality constraints

min/max $\sum_{j=1}^n c_j x_j$ linear objective
over $x \in \mathbb{R}^n$ linear integral variables
st. $A_1 x \leq b_1$ linear constraints
 $A_2 x = b_2$
 $A_3 x \geq b_3$



Continuous optimization vs Discrete optimization

randomization

Algorithms

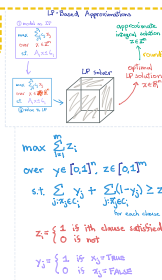
Max Sat Set Cover Covering Integer Programs

For each variable x_i , independently, randomly set $x_i = 1$ with probability p_i , and $x_i = 0$ otherwise.

$$E[\# \text{ sat. clauses}] \geq (1 - 1/e) \text{OPT}$$

k	oblivious	LP	average
1	1/2	$z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
2	3/4	$(3/4)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
3	7/8	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
4	15/16	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
5	31/32	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
...	...	...	...
k	$1 - 1/2^k$	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$

Can we get $\frac{3}{4}$ -APX?



Rounding LP solution
 $x \in [0, 1]^n \rightsquigarrow F \subseteq \{S_1, \dots, S_n\}$

For each set S_j , independently, take S_j with probability x_j .

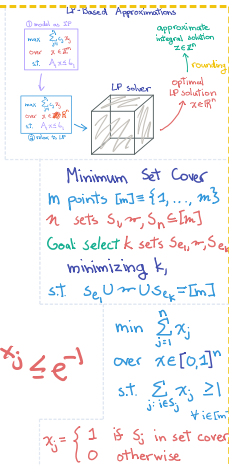
$$E[|F|] = \sum x_j \leq \text{OPT}$$

Fix a point i .

$$E[\# \text{ sets in } F \text{ covering } i] = \sum_{j \in S_i} x_j \geq 1$$

$$P[F \text{ not cover } i] = \prod_{j \in S_i} (1 - x_j) \leq e^{-\sum_{j \in S_i} x_j} \leq e^{-1}$$

$$P[F \text{ cover } i] \geq 1 - e^{-1}$$



Lecture 7

Probability

Continuous Random Variables

Variance, Chebyshev's Variance Reduction

Median trick

Variance

$$\text{Var}(X) = E[(X - E[X])^2] = E[X^2] - (E[X])^2$$

Chebyshev's inequality

$$P[|X - E[X]| \geq d] \leq \frac{\text{Var}(X)}{d^2}$$

"median trick", "median of means"

- ① take mean to reduce variance, get constant prob. of error
- ② take median to amplify success prob. exponentially

Algorithms

Distinct Elements



Goal: estimate # of distinct elements in the stream

Suppose you had an ideal hash function

$$h: [m] \rightarrow [0, 1]$$

and the following facts:

let $Y_1, \dots, Y_k \in [0, 1]$ independent and uniformly random

$$X = \min\{Y_1, \dots, Y_k\}$$

$$\text{then } E[X] = \frac{1}{k+1}$$

$$\textcircled{1} X \leftarrow 1$$

② for each element e

$$\textcircled{a} X \leftarrow \min\{X, h(e)\}$$

$$/* E[X] = \frac{1}{k+1}, k = (\text{\# distinct elements}) */$$

$$\textcircled{3} \text{ return } \frac{1}{X} - 1$$

$$\text{recall } E[X] = \frac{1}{k+1}$$

$$\text{goal: show } |X - E[X]| \leq \frac{\epsilon}{k+1}$$

w/ good prob.

$$P[|X - E[X]| \geq \frac{\epsilon}{k+1}]$$

$$\leq \text{Var}(X) \cdot \left(\frac{k+1}{\epsilon}\right)^2 \leftarrow \text{Chebyshev's inequality}$$

where

$$\text{Var}(X) = \frac{k}{(k+1)^2(k+2)} \approx \frac{1}{k^2}$$

Q: how to reduce $\text{Var}(X)$?

we can get $(1 \pm \epsilon)$ -APX

w/ 10% error prob. in $O(\frac{1}{\epsilon^2})$ space

what about high probability?

① make q copies $Y_1, \dots, Y_q$ of

② return median of $Y_1, \dots, Y_q$

Algo correct $\Leftrightarrow \geq \frac{1}{2}$ of Y_i 's correct

let $Z = \# \text{correct } Y_i$'s $E[Z] = \frac{q}{2}$

$$P[Z \leq (1 - \frac{\epsilon}{4})E[Z]] \leq e^{-\left(\frac{\epsilon}{4}\right)^2 E[Z]/2} \leq \frac{1}{n^{O(n)}} \text{ for } q = O(\log n)$$

ideal hash $h: [m] \rightarrow [0, 1]$

let $Y_1, \dots, Y_k \in [0, 1]$

indep. and unifs. rand.

$$X = \min\{Y_1, \dots, Y_k\}$$

$$\text{then } E[X] = \frac{1}{k+1}$$

$$\text{and } E[X^2] = \frac{2}{(k+1)(k+2)}$$

$$\text{Var}(X) = E[(X - E[X])^2]$$

$$X_1, \dots, X_k \text{ indep. copies of } X$$

$$\bar{X} = \frac{1}{k}(X_1 + \dots + X_k)$$

$$\text{Var}(\bar{X}) = \frac{1}{k} \text{Var}(X)$$

$$P[X_1 + \dots + X_k \leq (1 - \epsilon)k] \leq e^{-\epsilon^2 k/4}$$

Lecture 8

Probability

Gaussian random variables

Gaussian random variables

$$x \sim \mathcal{N}(\mu, \sigma^2)$$

mean variance

Def: the nicest, most pleasant, convenient random variable ever

Fact let $x \sim \mathcal{N}(\mu, \sigma^2)$, $d \in \mathbb{R}$.

$$\text{then } x+d \sim \mathcal{N}(\mu+d, \sigma^2)$$

$$dx \sim \mathcal{N}(d\mu, d^2\sigma^2)$$

Fact let $x_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$, $x_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$

$$\text{then } x_1 + x_2 \sim \mathcal{N}(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$$

Fact let $x \sim \mathcal{N}(0, 1)$, $t < \frac{1}{2}$

$$\text{then } E[e^{tx^2}] = \frac{1}{\sqrt{1-2t}}$$

Algorithms

Dimensionality reduction

Bag-of-words

literally any sentence is fine

any ↑ is literally sentence English
five encodes text in high dimensions

Lemma 8.8. Let $A \sim \mathcal{N}^{k \times d}$ be a random matrix where each coordinate A_{ij} is an independently drawn sample from $\mathcal{N}$. Let $\epsilon > 0$ be sufficiently small. For any vector x ,

$$\mathbb{P}\left[(1-\epsilon)\|x\|^2 \leq \frac{1}{k}\|Ax\|^2 \leq (1+\epsilon)\|x\|^2\right] \geq 1 - 2e^{-k/8}$$

$$k \left\{ \left[\frac{1}{k} \|Ax\|^2 \right] \leftarrow \alpha_i \right.$$

$$E[\|Ax\|^2] = \sum_{i=1}^k E[(Ax)_i^2] = k\|x\|^2$$

$$E[(\sum_{i=1}^k A_{ij} x_j)^2] = \|x\|^2$$

$$A_{ij} x_j \sim \mathcal{N}(0, x_j^2)$$

$$\sum_{j=1}^n A_{ij} x_j \sim \mathcal{N}(0, \|x\|^2)$$

$$(Ax) \sim \mathcal{N}^k(0, \|x\|^2)$$

$$\Pr[\|Ax\|^2 \geq k\|x\|^2 \cdot (1+\epsilon)] \leq e^{-\epsilon^2 k/8}$$

Dimensionality Reduction - high level

$O(\log n)$ dimensions suffice

where $n = \# \text{ points}$

$$\mathbb{R}^{\text{very large}} \xrightarrow{f(x)} \mathbb{R}^{O(\log n)}$$

One can map n points into $\mathbb{R}^{O(\log n/\epsilon^2)}$ while preserving all distances up to $(1 \pm \epsilon)$ -APX

Lecture 9

Probability

Locality-sensitive hashing

Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \text{ w/ } g_i \sim N^d$$

$$k = \pi(\log(n)/2\theta)$$



d-dimensional sphere

when querying y :

• if there is $x \in P$ w/ $\angle(x, y) \leq \theta$, then it collides w/ probability $\geq 1/\sqrt{n}$

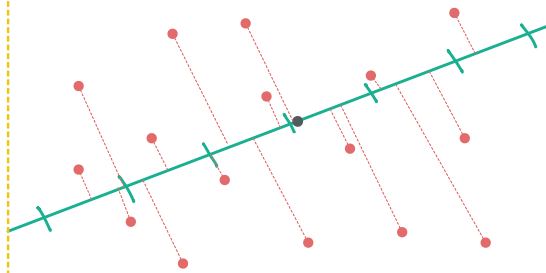
• expect to collide w/ at most one point $x \in P$ w/ $\angle(x, y) > 2\theta$

Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim N^d$
 then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

Random line embedding & Buckets

$$h(x) = \lfloor \langle g, x \rangle + d \rfloor \text{ w/ } g \sim N^d, d \in [0, 1] \text{ unifs. rand.}$$

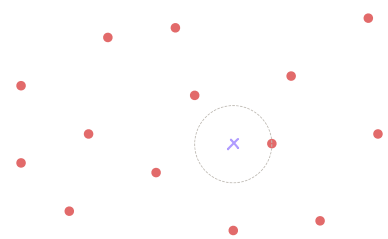


Algorithms

Approximate nearest neighbors

Nearest neighbor search

Input: n points $P \subset \mathbb{R}^d$



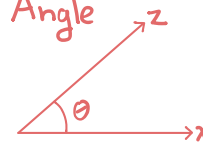
Query: given $q \in \mathbb{R}^d$, find closest point in P to q

Types of distances

• Euclidean distance

$$\|x - z\|_2 = \sqrt{\sum_i (x_i - z_i)^2}$$

• Angle



Lecture 9

Probability (and techniques)

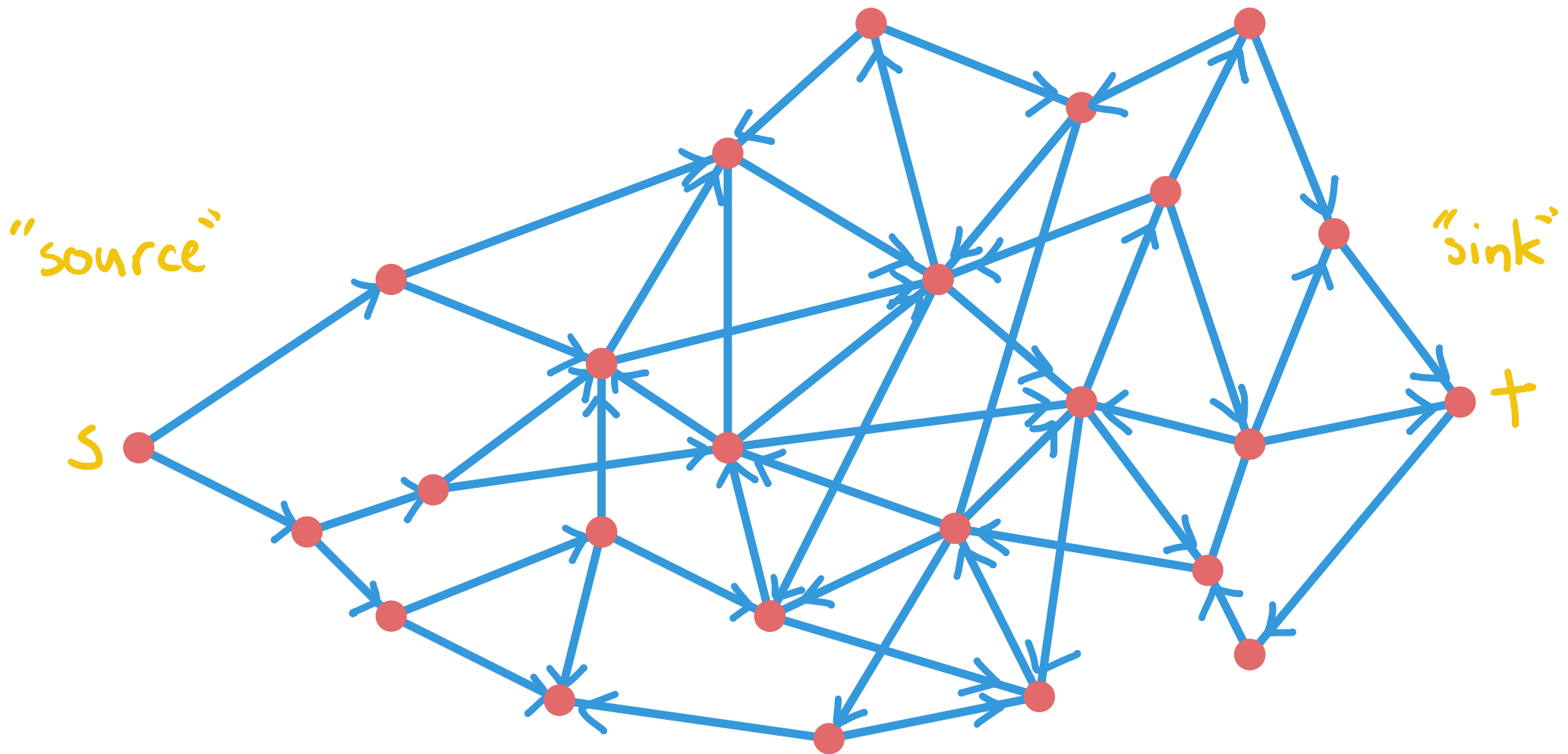
Algorithms

LP duality

Sparsest Cut

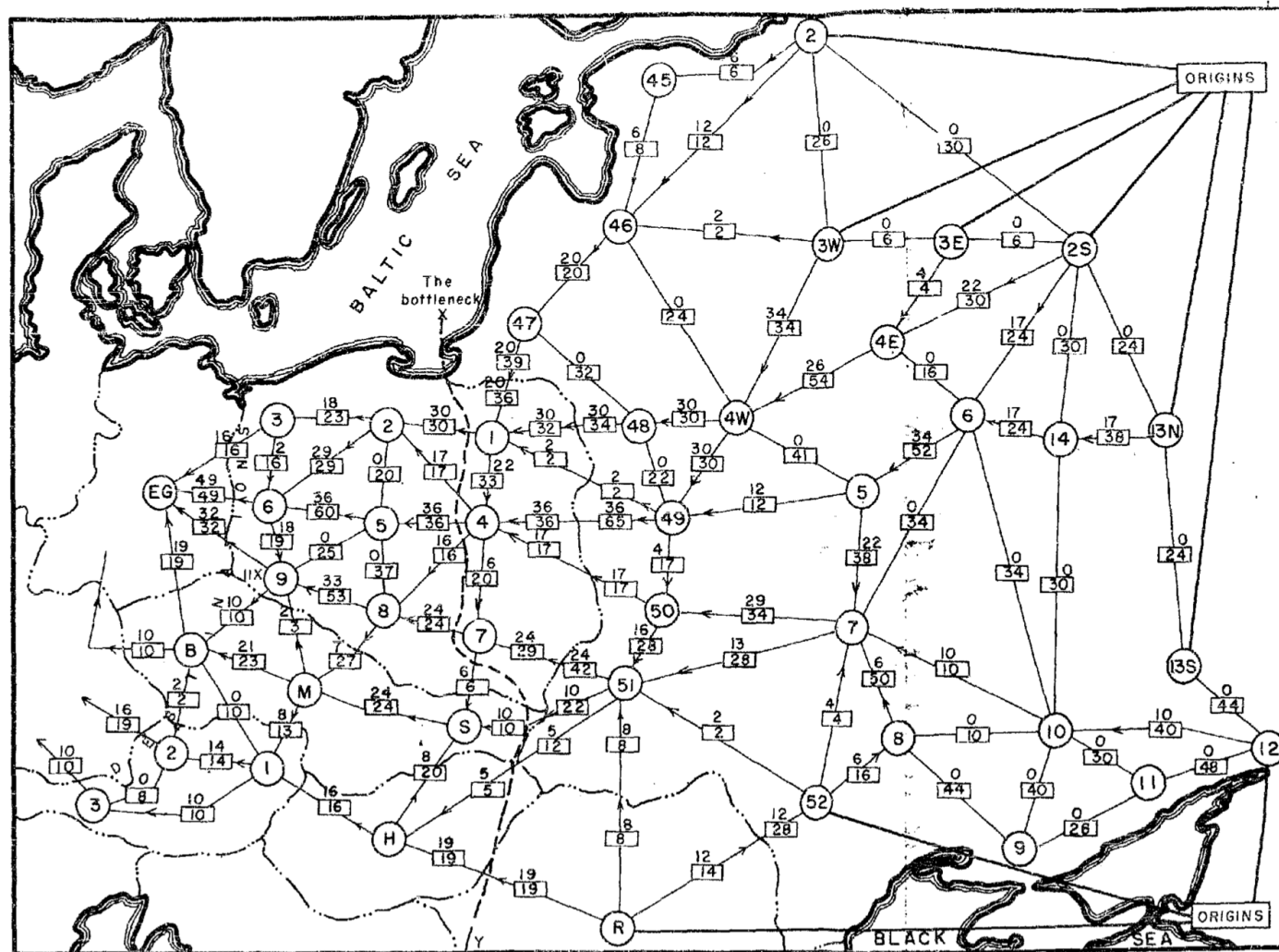
L_1 -metric embeddings

Measuring connectivity



how well connected are s and t?

Physical infrastructure



SECRET
RM-1373
10-24-55
-33-

Fig. 7 — Traffic pattern: entire network available

Legend:

— International boundary

(B) Railway operating division

← 9/12 → Capacity: 12 each way per day. Required flow of 9 per day toward destinations (in direction of arrow) with equivalent number of returning trains in opposite direction

All capacities in $\sqrt{1000}$'s of tons } each way per day

Origins: Divisions 2, 3W, 3E, 2S, 13N, 13S, 12, 52 (USSR), and Roumania

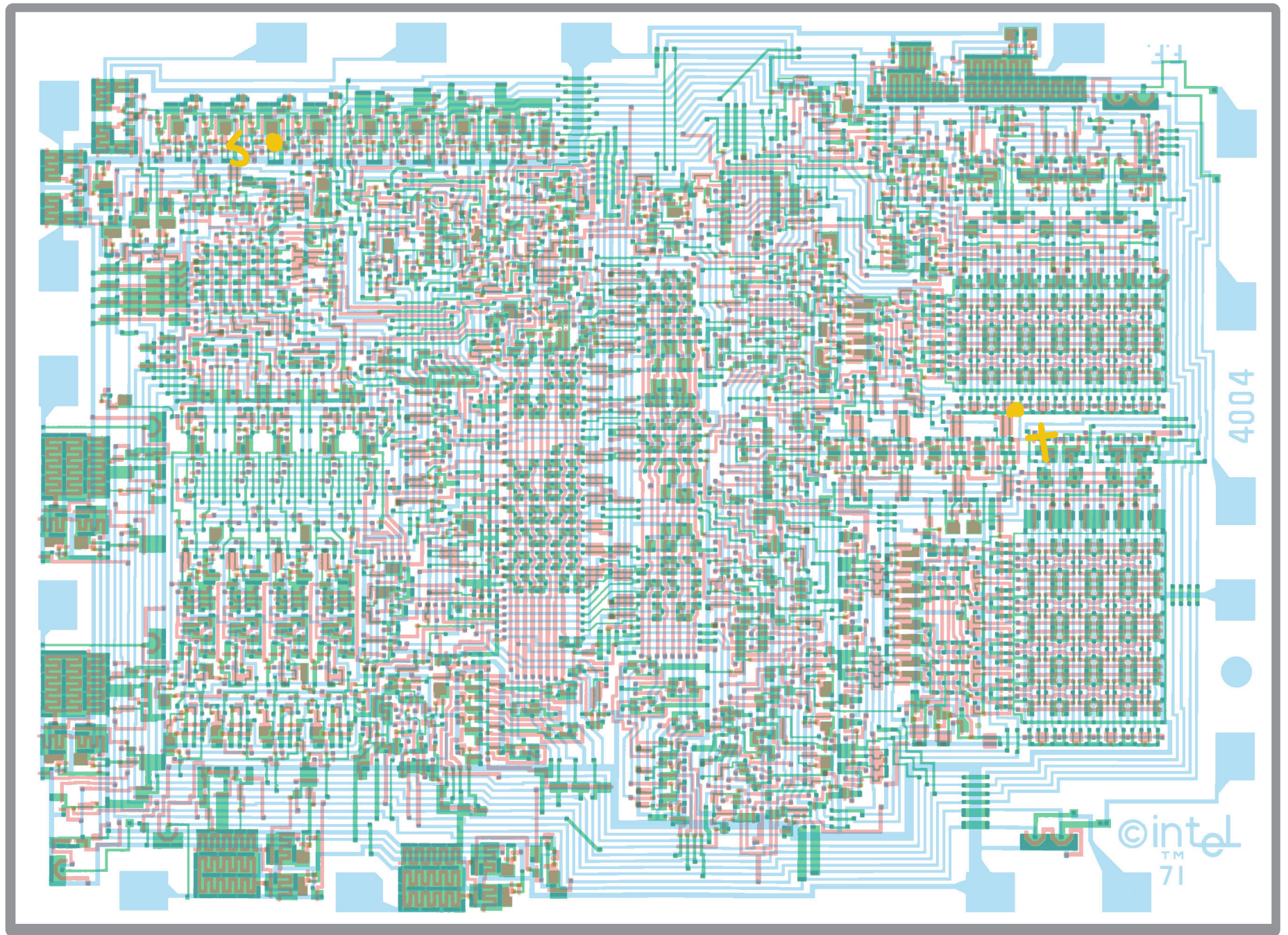
Destinations: Divisions 3, 6, 9 (Poland); B (Czechoslovakia); and 2, 3 (Austria)

Alternative destinations: Germany or East Germany

Note IIX of Division 9, Poland

(from Jeff's notes)

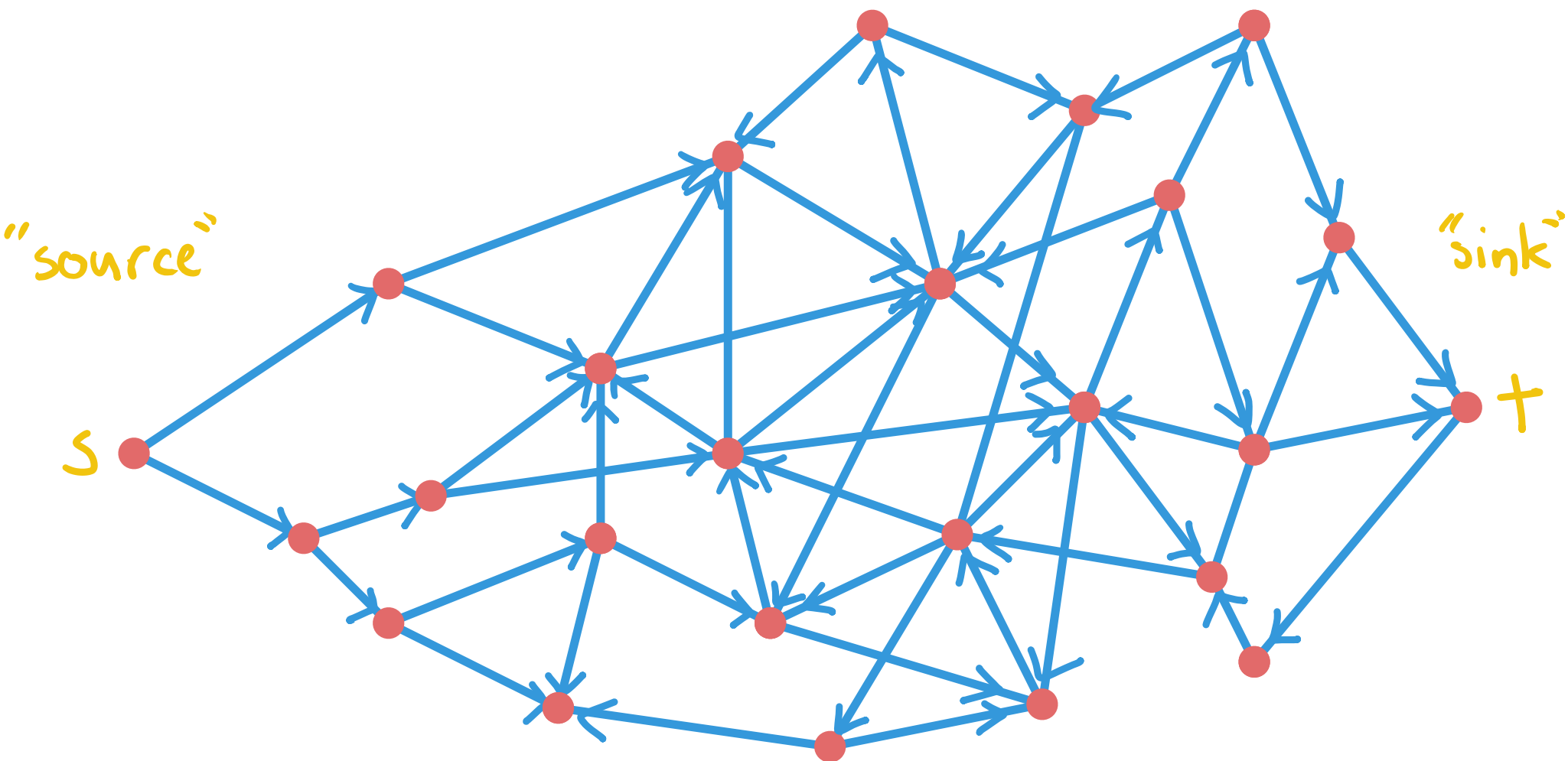
CPU's



social network

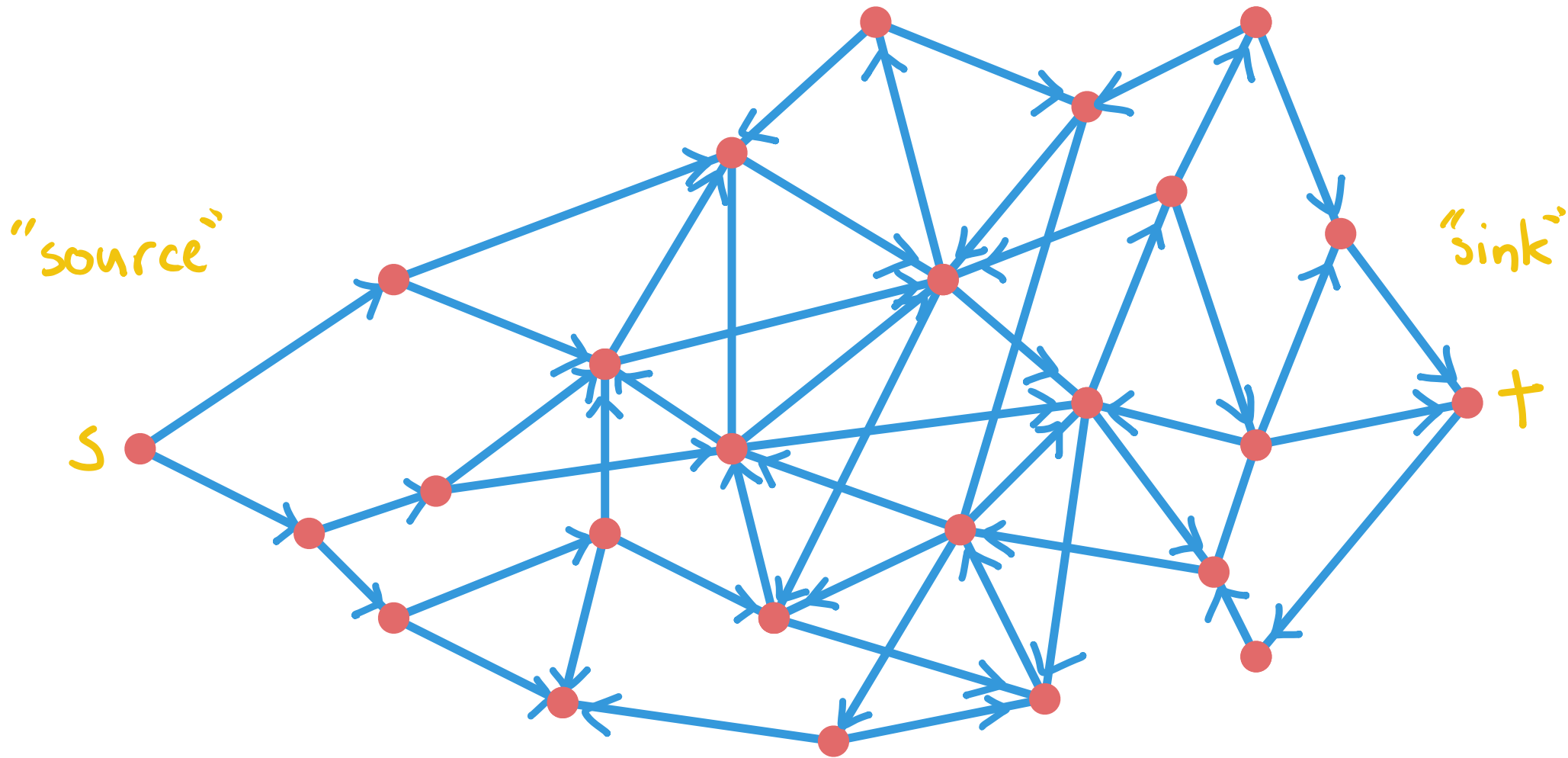


Measuring connectivity



① max # of edge-disjoint $s \rightsquigarrow t$ paths?

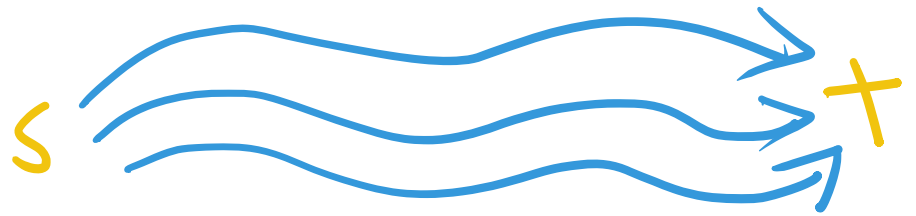
Measuring connectivity



② min # of edge deletions to disconnect $s \rightsquigarrow t$?

① max # of edge-disjoint $s \rightsquigarrow t$ paths?

"path packing"



collection of edge-disjoint paths

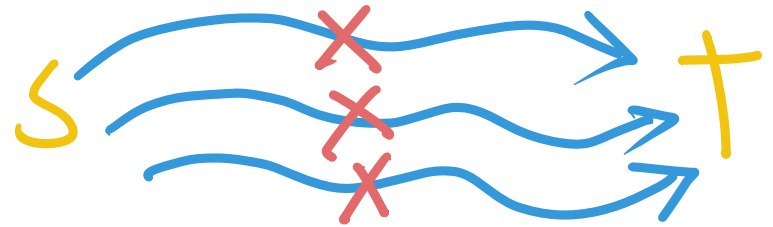
"packing paths into edges"

Question ① = max cardinality path packing

example of "packing problem"

path p "packs into" each edge $e \in p$

② min # of edge deletions to
disconnect $s \rightarrow t$?



" $s \rightarrow t$ cut"

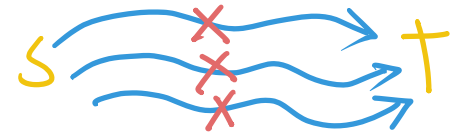
set of edges whose removal
disconnects t from s

Question ② = min s - t cut

② min # of edge deletions to disconnect $s \rightsquigarrow t$?

Question ② = min s - t cut

" $s \rightarrow t$ cut"



set of edges whose removal disconnects t from s

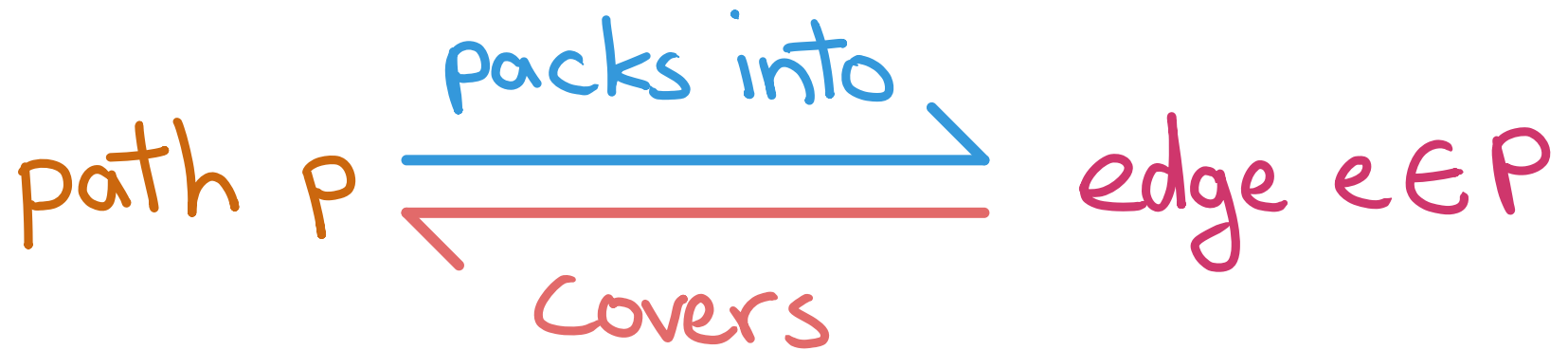
example of "covering problem"

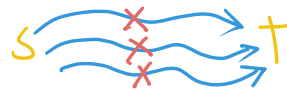
edge e "covers" path p containing e

goal:

min edge set covering all $s \rightsquigarrow t$ paths

Duality





Duality
 path p $\xrightarrow{\text{packs into}}$ edge $e \in P$
~~covers~~

Lemma

$$\left(\begin{array}{c} \text{max \#} \\ \text{disjoint} \\ s \rightsquigarrow t \text{ paths} \end{array} \right) \leq \left(\begin{array}{c} \text{min \# edges} \\ \text{in any } s \rightsquigarrow t \text{ cut} \end{array} \right)$$

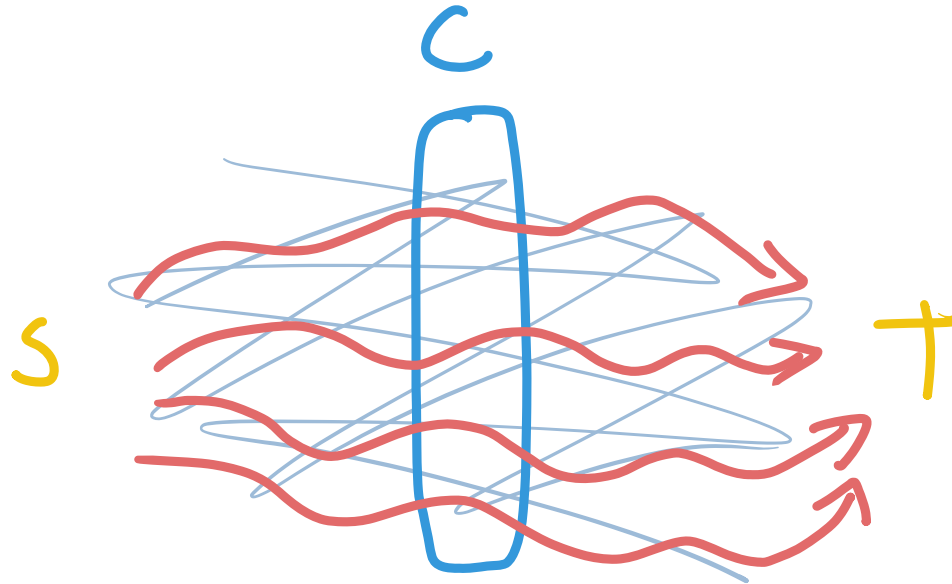
Proof let

Lemma

$$\left(\begin{array}{c} \text{max \#} \\ \text{disjoint} \\ s \rightsquigarrow t \text{ paths} \end{array} \right) \leq \left(\begin{array}{c} \text{min \# edges} \\ \text{in any } s \rightsquigarrow t \text{ cut} \end{array} \right)$$

P : collection of edge disjoint paths

$C \subseteq E$: s - t cut

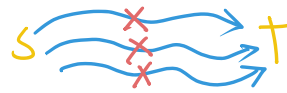


each path contains ≥ 1 edge from C

for each path choose 1 edge in
path and C

choices distinct

$$|P| \leq \sum_{p \in P} |p \cap C| = \left| \bigcup_{p \in P} (p \cap C) \right| \leq |C|$$



Duality
 path p $\xrightarrow{\text{packs into}}$ edge $e \in P$
~~covers~~

Lemma

$$\left(\begin{array}{c} \text{max \#} \\ \text{disjoint} \\ s \rightsquigarrow t \text{ paths} \end{array} \right) \leq \left(\begin{array}{c} \text{min \# edges} \\ \text{in any } s \rightsquigarrow t \text{ cut} \end{array} \right)$$

are they equal?

Notation

$$\langle x, y \rangle = \sum_{i=1}^n x_i y_i$$

$$x, y \in \mathbb{R}^n$$

A^T = transpose of A

$$A \in \mathbb{R}^{m \times n}$$

def. by $\langle y, Ax \rangle = \langle A^T y, x \rangle$

$$x \in \mathbb{R}^n, y \in \mathbb{R}^m$$

$x \leq y$ means $x_i \leq y_i$ for all i

$$x, y \in \mathbb{R}^n$$

★ LP Duality ★ Theorem

Packing LP

maximize $\langle b, x \rangle$
over $x \in \mathbb{R}_{\geq 0}^n$
s.t. $Ax \leq c$

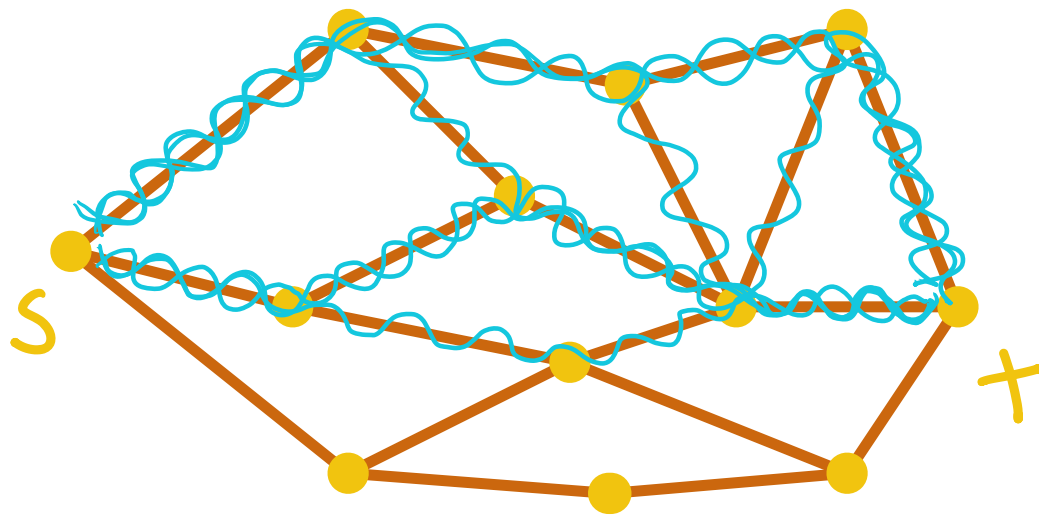
$\equiv$
(equal)

Covering LP

minimize $\langle c, y \rangle$
over $y \in \mathbb{R}_{\geq 0}^m$
s.t. $A^T y \geq b$

$A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^n, c \in \mathbb{R}^m$

Max Flow
Min Cut



Path Packing LP

maximize $\sum_{p \in P_{st}} x_p$
over $x: P_{st} \rightarrow \mathbb{R}_{\geq 0}$
(all $s \rightsquigarrow t$ paths)

s.t. $\sum_{p \ni e} x_p \leq c_e$ for all edges e
"Ax ≤ c" (capacity of e)

Path Covering LP
(i.e., cut)

(total capacity of "fractional cut")
minimize $\sum_{e \in E} c_e y_e$
over $y: E \rightarrow \mathbb{R}_{\geq 0}$

s.t. $\sum_{e \in p} y_e \geq 1$ for all $p \in P_{st}$
"A^Ty ≥ b" (continuous analog of $s \rightsquigarrow t$ cut)

goal:

$$\underbrace{(\max \text{ flow})}_{\text{(same as max fractional path packing)}} = (\min_{\text{(discrete)}} \text{ cut})$$

(same as max fractional path packing)

$$\begin{aligned} &\text{maximize } \sum_{p \in \mathcal{P}_{st}} x_p \\ &\text{over } x: \mathcal{P}_{st} \rightarrow \mathbb{R}_{\geq 0} \\ &\text{s.t. } \sum_{p \ni e} x_p \leq c_e \quad \text{for all edges } e \\ &\quad \text{"Ax} \leq \text{b"} \quad \text{(capacity of } e \text{)} \end{aligned}$$

$$\begin{aligned} &\text{minimize } \sum_{e \in E} c_e y_e \quad \text{(total capacity of "fractional cut")} \\ &\text{over } y: E \rightarrow \mathbb{R}_{\geq 0} \\ &\text{s.t. } \sum_{e \in P} y_e \geq 1 \quad \text{for all } p \in \mathcal{P}_{st} \\ &\quad \text{"Ay} \geq \text{b"} \quad \text{(continuous analog of srt cut)} \end{aligned}$$

★ LP-duality ★

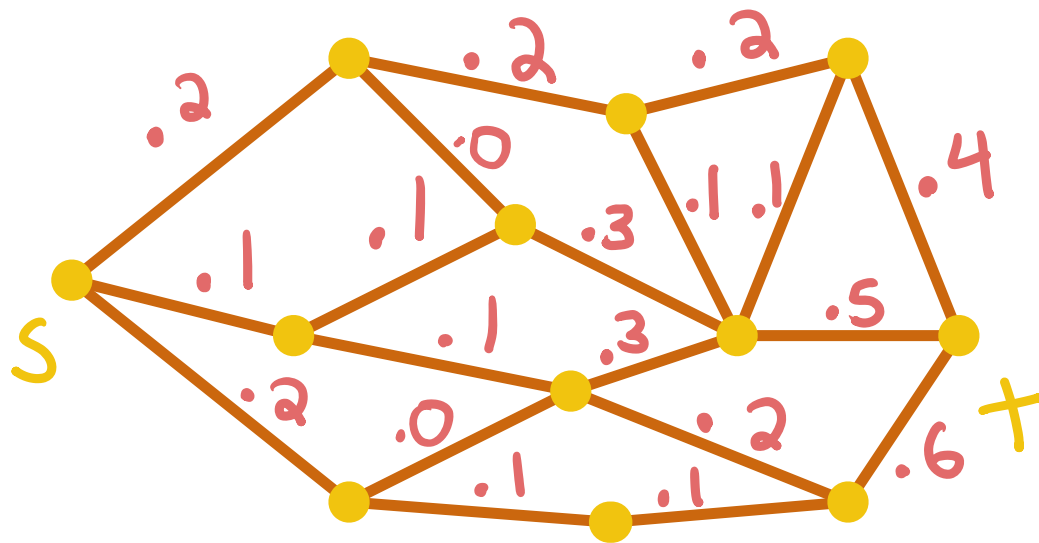
$$(\max \text{ flow}) = (\min \text{ fractional cut})$$

(i.e., the "path covering LP")

because the LP is a relaxation :

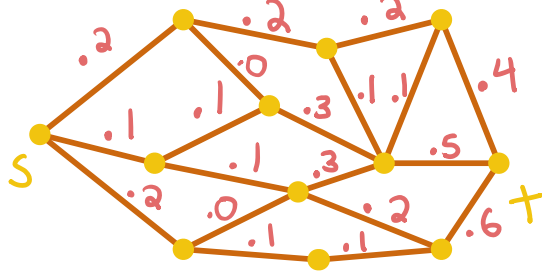
$$(\min \text{ fractional cut}) \leq (\min_{\text{(discrete)}} \text{ cut})$$

let $\gamma: E \rightarrow \mathbb{R}_{\geq 0}$ be a minimum
fractional cut

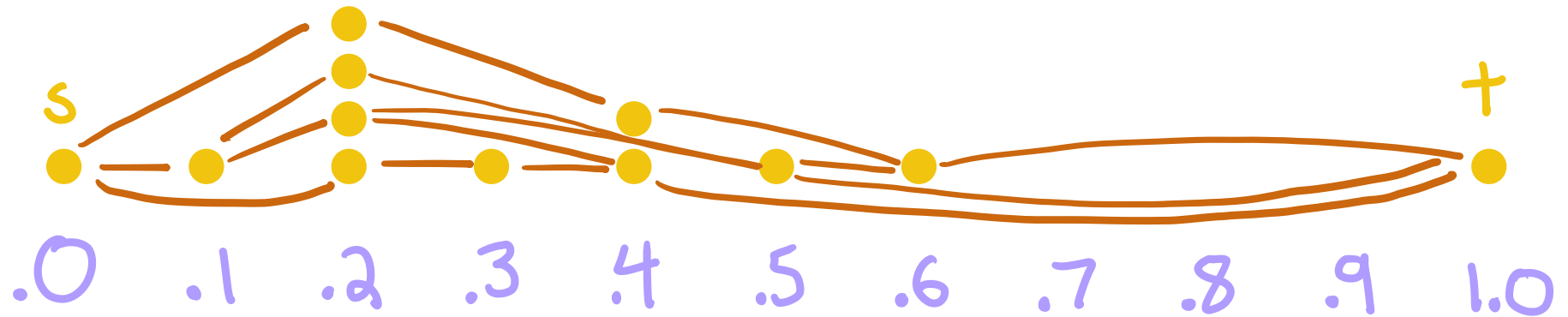


let $\gamma: E \rightarrow \mathbb{R}_{\geq 0}$ be a minimum

fractional cut

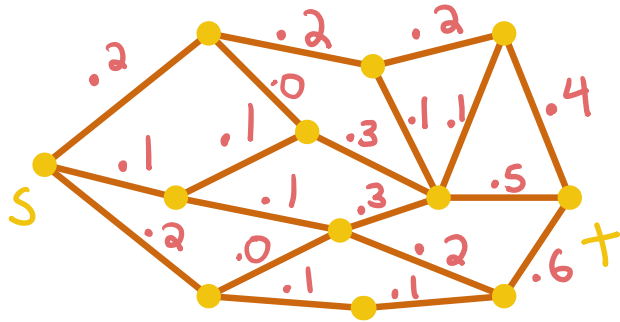


embed vertices on $\mathbb{R}$ based on
distance from s w/r/t lengths γ

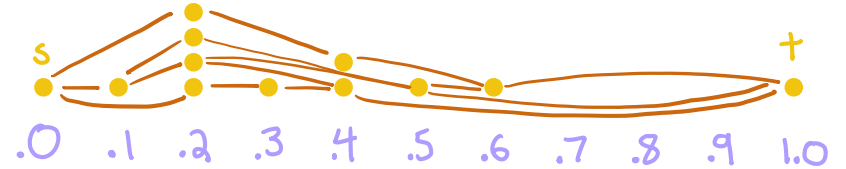


all $s \rightsquigarrow t$ paths have length ≥ 1 b/c of LP

let $\gamma: E \rightarrow \mathbb{R}_{\geq 0}$ be a minimum fractional cut

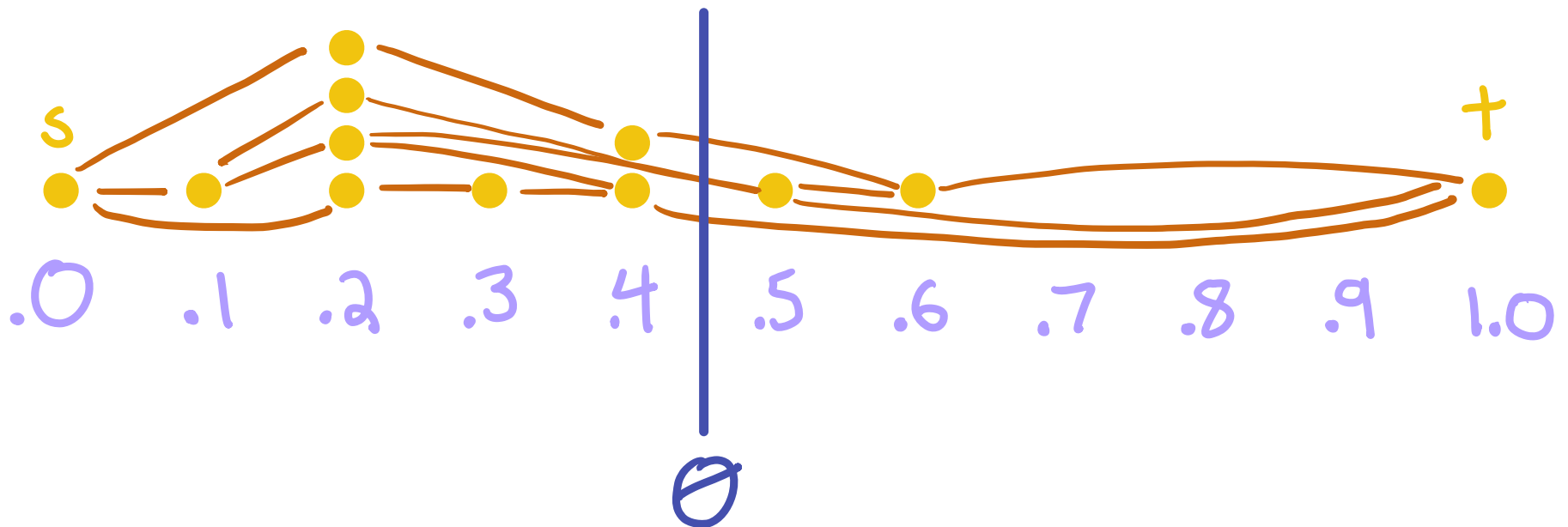


embed vertices on $\mathbb{R}$ based on distance from s w/r/t lengths γ

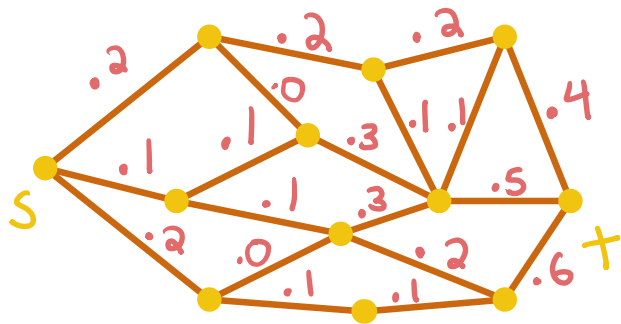


all $s \rightsquigarrow t$ paths have length ≥ 1 b/c of LP

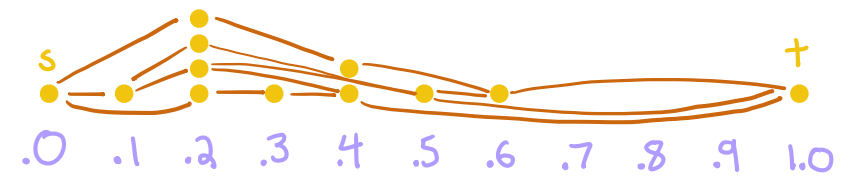
pick $\theta \in [0, 1]$ uniformly at random



let $\gamma: E \rightarrow \mathbb{R}_{\geq 0}$ be a minimum fractional cut



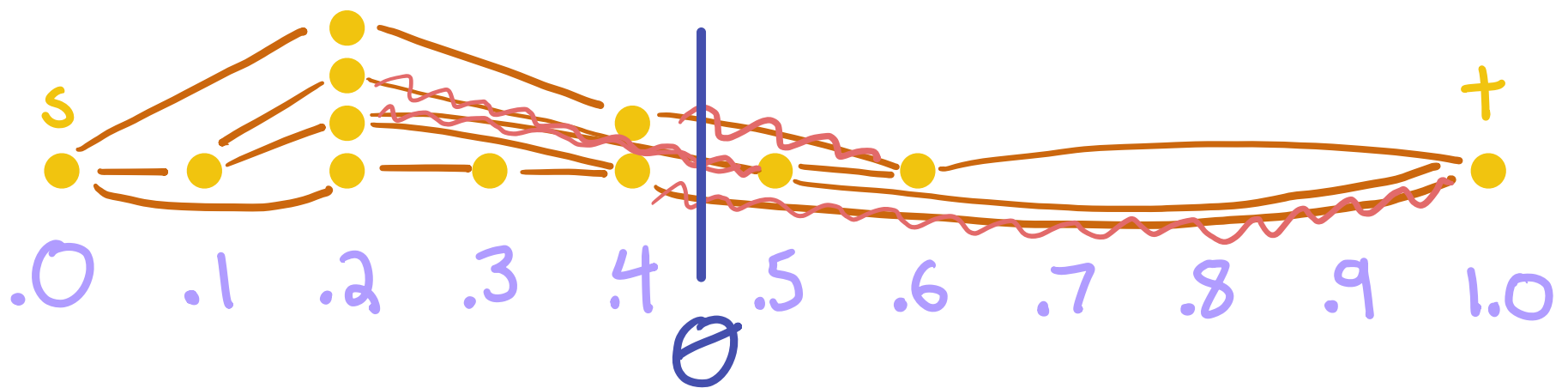
embed vertices on $\mathbb{R}$ based on distance from s w/r/t lengths γ

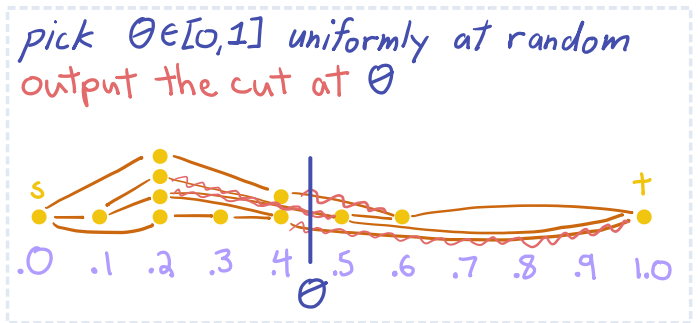
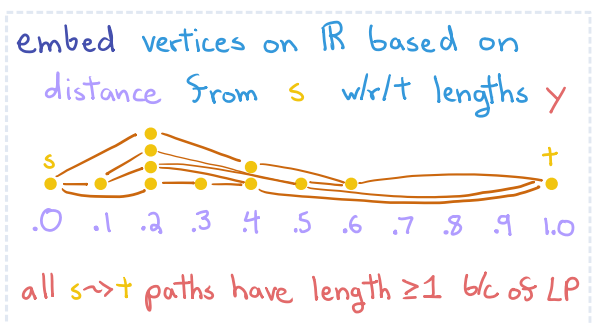
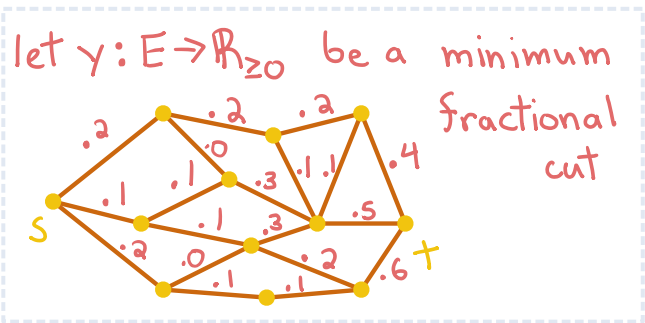


all $s \rightsquigarrow t$ paths have length ≥ 1 b/c of LP

pick $\theta \in [0, 1]$ uniformly at random

output the cut at θ





$$\begin{aligned}
 & E[\text{capacity of cut}] \\
 & \stackrel{\text{(linearity of expectation)}}{=} \sum_{e \in E} c_e \Pr[e \text{ cut}] \leq \sum_{e \in E} c_e \gamma_e
 \end{aligned}$$

s

$\bullet$

0

a

e

b

$b \leq a + \gamma_e$ b/c shortest paths

paths

Probabilistic Method

Paul Erdős



(we showed)

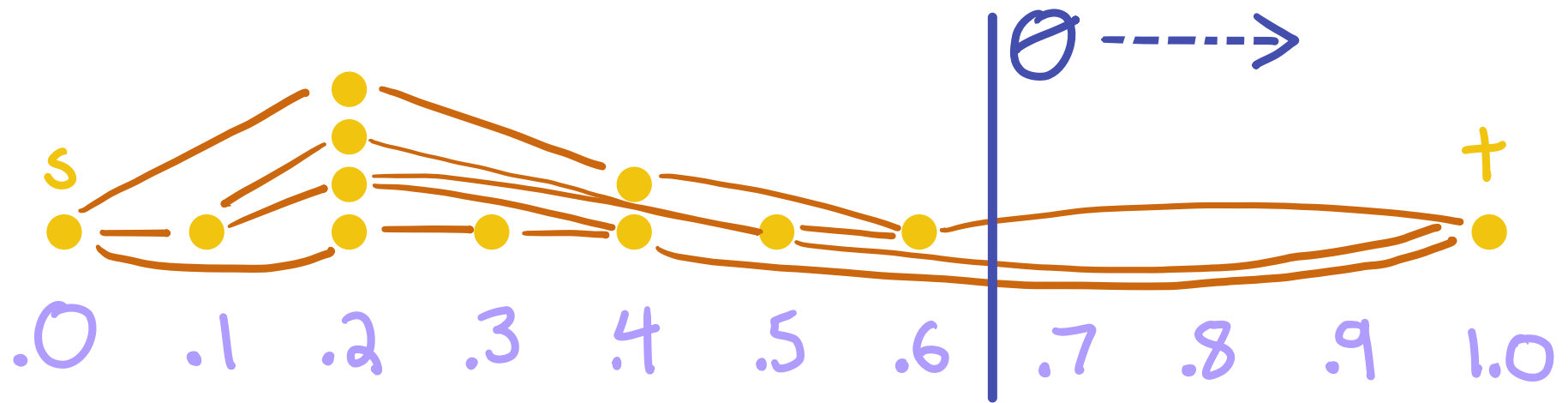
$$E \left[\begin{array}{c} \text{capacity} \\ \text{of random} \\ \text{cut} \end{array} \right] \leq \left(\begin{array}{c} \text{minimum} \\ \text{capacity} \\ \text{of any} \\ \text{fractional} \\ \text{cut} \end{array} \right)$$

if , then there exists a cut w/ capacity $\leq$

more precisely:
random cut is good
with probability > 0 .



to find this lucky cut



sweep through all θ from 0 to 1

Overall argument

$$\left(\begin{array}{c} \text{minimum cut} \\ \text{along line} \\ \text{embedding} \end{array} \right) \overset{\text{probabilistic method}}{\leq} \left(\begin{array}{c} \text{minimum} \\ \text{fractional} \\ \text{cut} \end{array} \right) \overset{\text{LP Duality}}{=} \left(\begin{array}{c} \text{max} \\ \text{flow} \end{array} \right)$$

(we already knew that $\left(\begin{array}{c} \text{max} \\ \text{flow} \end{array} \right) \leq \left(\begin{array}{c} \text{min} \\ \text{(discrete)} \\ \text{cut} \end{array} \right) \leq \left(\begin{array}{c} \text{minimum cut} \\ \text{along line} \\ \text{embedding} \end{array} \right)$)

thus $\left(\begin{array}{c} \text{max} \\ \text{flow} \end{array} \right) = \left(\begin{array}{c} \text{min} \\ \text{(discrete)} \\ \text{cut} \end{array} \right)$

Lecture 9

Probability (and techniques)

LP duality

L_1 -metric embeddings

Algorithms

Sparsest Cut

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$ (symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$

← cut few edges

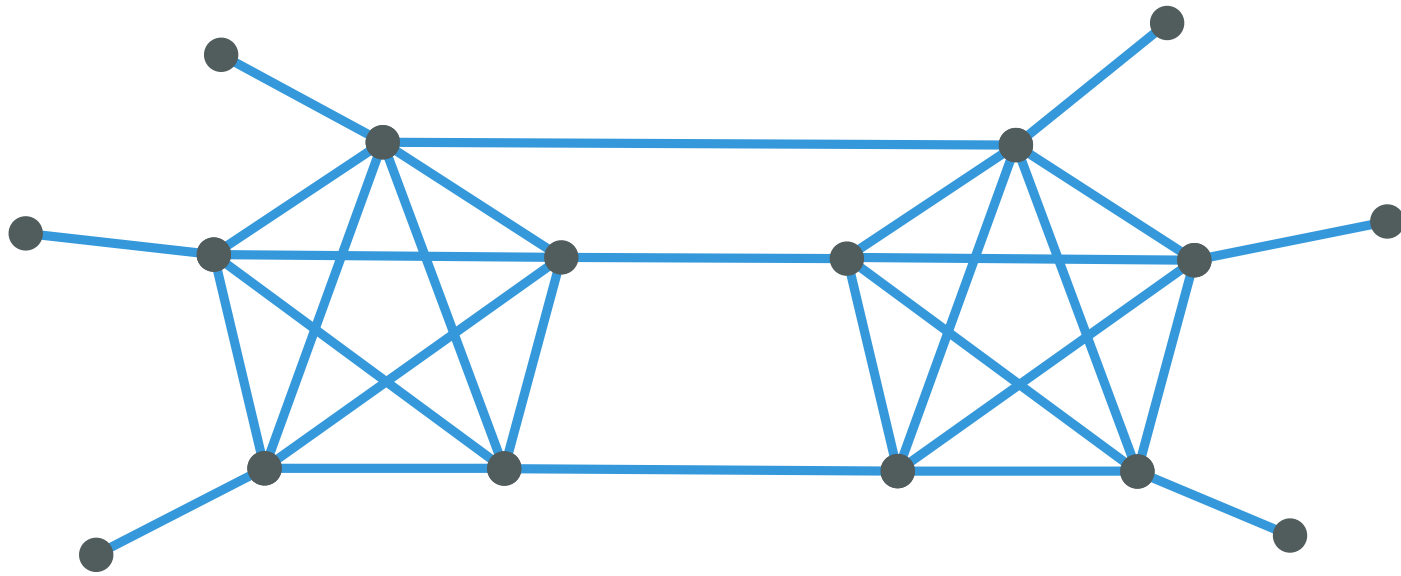
← split up many pairs

" $\delta(S)$ " = edges cut by S

Special case: uniform sparsest cut

w/ uniform demands $b(u,v)=1 \quad \forall u,v$

$$\text{"sparsity"} = \frac{c(\delta(S))}{|S| |\bar{S}|}$$



useful primitive for graph partitioning, divide + conquer

Designing an LP for sparsest cut

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

Q: what would be a linear relaxation for cuts?

Idea 1: sparsest metric

$$\min \frac{\sum_{e=\{u,v\}} c(e) d(u,v)}{\sum_{\{u,v\}} b(u,v) d(u,v)} \quad \text{over metric } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

Linear constraints

- $d(u,v) \geq 0 \quad \forall u,v$
- $d(u,v) = d(v,u) \quad \forall u,v$
- $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Q: how to make objective linear?

Idea 2: min-cost metric

$$\min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \quad \text{over all metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

Taking the dual of the LP

min-cost metric

$$\begin{aligned} \min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over all metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0} \\ \text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1 \end{aligned}$$

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \notin S} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

① rewrite as covering LP

let $P_{st} = \{\text{all } (s,t)\text{-paths}\}$, $P_{\star} = \prod_{\{s,t\}} P_{s,t}$ "path bundles"

$$\min \sum_e c(e) \gamma(e) \text{ over } \gamma: E \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) \sum_{e \in P_{s,t}} \gamma(e) \geq 1 \text{ for all } p \in P_{\star}$$

Separability and ellipsoid algorithm

let $P_{st} = \{\text{all } (s,t)\text{-paths}\}$, $P_{\star} = \prod_{\{s,t\}}^{\text{"path bundles"}} P_{s,t}$

$$\min \sum_e c(e) \gamma(e) \quad \text{over } \gamma: E \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) \sum_{e \in P_{s,t}} \gamma(e) \geq 1 \quad \text{for all } p \in P_{\star}$$

- exponentially many constraints!
 - but given γ , we can verify constraints in polytime by computing all-pairs shortest paths
- $\Rightarrow$ LP can be solved by "ellipsoid algorithm"

Taking the dual of the LP

① rewrite as covering LP $\mathcal{P}_{st} = \{\text{all } (s,t)\text{-paths}\}$, $\mathcal{P}_\star = \prod_{\{s,t\}} \mathcal{P}_{st}$ "path bundles"

$$\min \sum_e c(e) \gamma(e) \text{ over } \gamma: E \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) \sum_{e \in \mathcal{P}_{st}} \gamma(e) \geq 1 \text{ for all } p \in \mathcal{P}_\star$$

② X covers $Y \iff Y$ packs into X

$$\max \sum_{p \in \mathcal{P}_\star} x(p) \text{ over } x: \mathcal{P}_\star \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{p \in \mathcal{P}_\star} \sum_{\{s,t\}: e \in \mathcal{P}_{st}} b(s,t) x(p) \leq c(e) \quad \forall e \in E$$

↗ "demand multicommodity flow"

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

$$\text{"sparsity"} \frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)} \leftarrow \text{cut few edges}$$

$\leftarrow \text{split up many pairs}$

min-cost metric

$$\min \sum_{e \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

★ LP Duality ★

Packing LP

$$\begin{aligned} &\text{maximize } \langle b, x \rangle \\ &\text{over } x \in \mathbb{R}_{\geq 0}^n \\ &\text{s.t. } Ax \leq c \end{aligned}$$

Covering LP

$$\begin{aligned} &\text{minimize } \langle c, y \rangle \\ &\text{over } y \in \mathbb{R}_{\geq 0}^m \\ &\text{s.t. } A^T y \geq b \end{aligned}$$

$\downarrow$
(equal)

LP for sparsest cut

$$\begin{aligned} \min \quad & \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0} \\ \text{s.t.} \quad & \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1 \end{aligned}$$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

Q: how to round metric to sparse cut?

Q: how to round LP to sparse cut?

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \notin S} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

min $\sum_{e=\{u,v\} \in E} c(e) d(u,v)$ over metrics $d: V \times V \rightarrow \mathbb{R}_{\geq 0}$

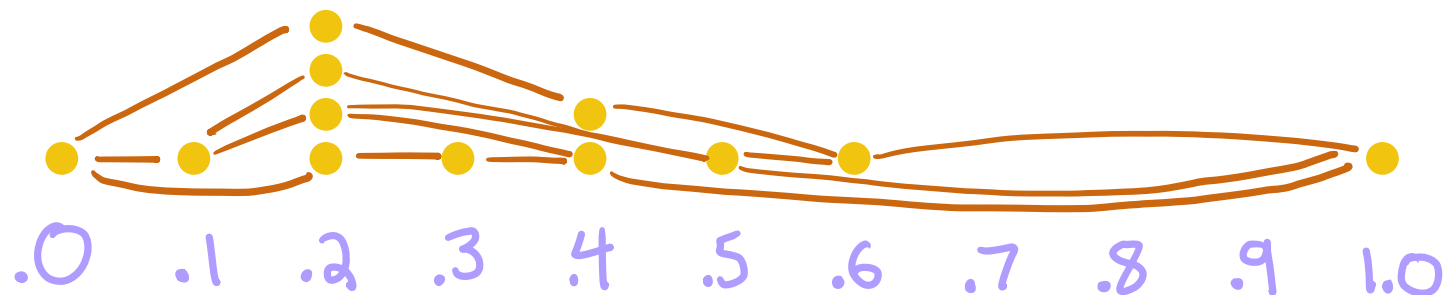
s.t. $\sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Suppose $d(u,v)$ was of the form

$d(u,v) = |f(u) - f(v)|$ for $f: V \rightarrow [0,1]$

(assume $\min_u f(u) = 0, \max_u f(u) = 1$)



min $\sum_{e=\{u,v\} \in E} c(e) d(u,v)$ over metrics $d: V \times V \rightarrow \mathbb{R}_{\geq 0}$

s.t. $\sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

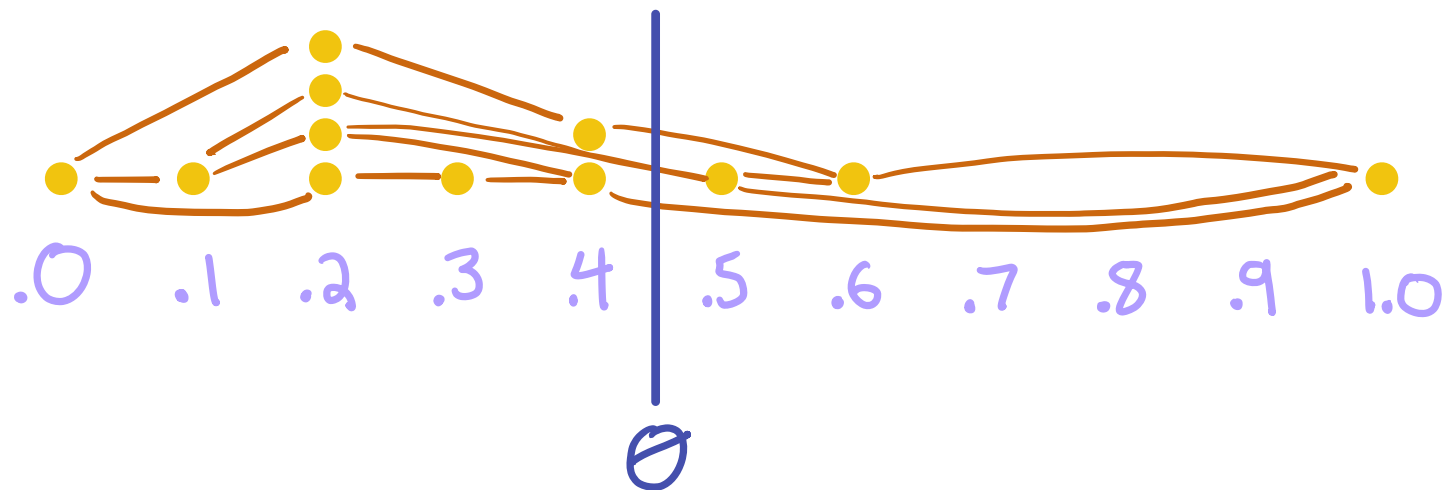
"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

Suppose $d(u,v)$ was of the form

$$d(u,v) = |f(u) - f(v)| \quad \text{for } f: V \rightarrow [0,1]$$

(assume $\min_u f(u) = 0, \max_u f(u) = 1$)

pick $\theta \in [0,1]$ uniformly at random



$$\min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

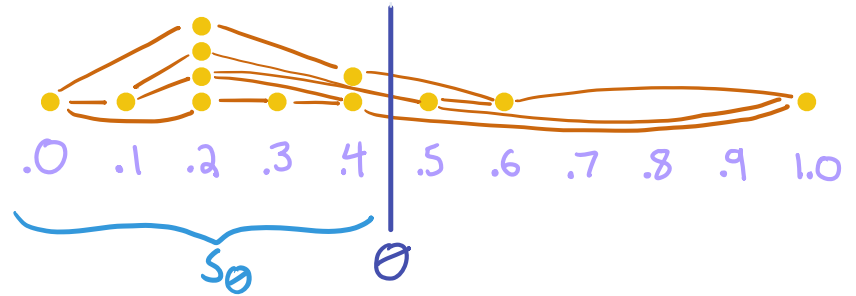
Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$
 ← cut few edges
 ← split up many pairs

Suppose $d(u,v)$ was of the form

$$d(u,v) = |f(u) - f(v)| \text{ for } f: V \rightarrow [0,1]$$

pick $\theta \in [0,1]$ uniformly at random



$$E[c(\delta(S_\theta))] = \sum_e c(e) \Pr[e \in S_\theta] = \sum_{e=\{u,v\}} c(u,v) d(u,v)$$

$$E\left[\sum_{\substack{s \in S_\theta \\ t \in \bar{S}_\theta}} b(s,t)\right] = \sum_{s,t} d(s,t) b(s,t)$$

$$\text{OPT} = \frac{E[c(\delta(S_\theta))]}{E\left[\sum_{s \in S_\theta, t \in \bar{S}_\theta} b(s,t)\right]} \neq E[\text{sparsity}]$$

$$E[c(\delta(s_\theta))] = \sum_{e=\{u,v\}} c(u,v) d(u,v), \quad E\left[\sum_{\substack{s \in S_\theta \\ t \in \bar{S}_\theta}} b(s,t)\right] = \sum_{s,t} d(s,t) b(s,t),$$

$$\text{But } E[\text{sparsity}] \neq \frac{E[c(\delta(s_\theta))]}{E\left[\sum_{\substack{s \in S_\theta \\ t \in \bar{S}_\theta}} b(s,t)\right]} = \text{OPT}$$

let $\lambda = \text{OPT sparsity}$

$$E[c(\delta(s_\theta))] \leq \lambda E\left[\sum_{\substack{s \in S_\theta \\ t \in \bar{S}_\theta}} b(s,t)\right]$$

$\Rightarrow \exists \theta \in (0,1)$ s.t.

$$c(\delta(s_\theta)) \leq \lambda \sum_{\substack{s \in S_\theta \\ t \in \bar{S}_\theta}} b(s,t)$$

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

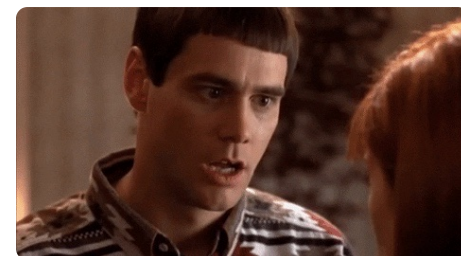
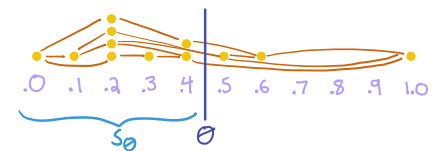
min $\sum_{e=\{u,v\} \in E} c(e) d(u,v)$ over metrics $d: V \times V \rightarrow \mathbb{R}_{\geq 0}$
s.t. $\sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Suppose $d(u,v)$ was of the form

$d(u,v) = |\xi(u) - \xi(v)|$ for $\xi: V \rightarrow [0,1]$

pick $\theta \in [0,1]$ uniformly at random



Q: how to round LP to sparse cut?

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

$$\min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

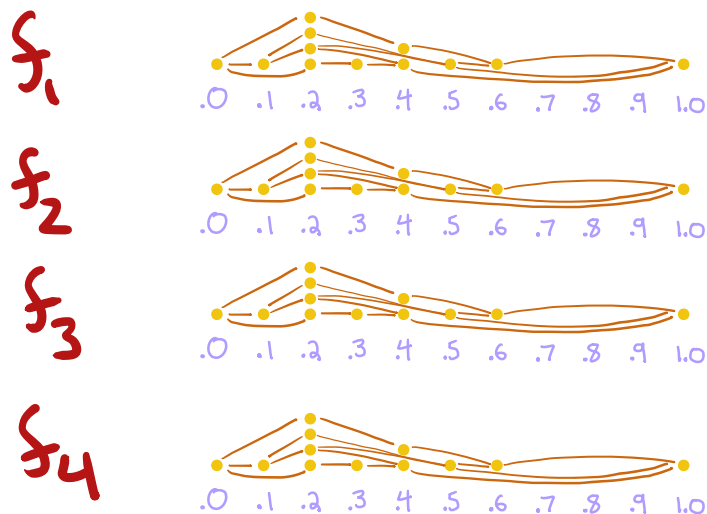
$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z$

Suppose $d(u,v)$ was of the form

$$d(u,v) = \sum_{i=1}^k |f_i(u) - f_i(v)| \text{ for } f_i: V \rightarrow [0,1]^k$$

(assume $\min_u f_i(u) = 0, \max_u f_i(u) = 1$)



" L_1 -metric"

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

$$\min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y)$

Suppose $d(u,v)$ was of the form

$$d(u,v) = \sum_{i=1}^k |f_i(u) - f_i(v)| \text{ for } f: V \rightarrow [0,1]^k$$

then

$$\sum_{i=1}^k \sum_{e=\{u,v\} \in E} c(e) |f_i(u) - f_i(v)| \leq \lambda \sum_{i=1}^k \sum_{\{s,t\}} b(s,t) |f_i(u) - f_i(v)|$$

$\Rightarrow$ for some i ,

$$\sum_{e=\{u,v\} \in E} c(e) |f_i(u) - f_i(v)| \leq \lambda \sum_{\{s,t\}} b(s,t) |f_i(u) - f_i(v)|$$

which can be rounded as before

min $\sum_{e=\{u,v\} \in E} c(e) d(u,v)$ over metrics $d: V \times V \rightarrow \mathbb{R}_{\geq 0}$

s.t. $\sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$

Linear constraints
 $d(u,v) \geq 0 \quad \forall u,v$
 $d(u,v) = d(v,u) \quad \forall u,v$
 $d(x,y) \leq d(x,z) + d(z,y)$

Sparsest cut

Input: undirected graph $G=(V,E)$
 edge capacities $c(e) > 0$ for $e \in E$
 demands $b(u,v) > 0$ for $u,v \in V$

Goal: compute SCV minimizing
 "sparsity" $\frac{C(S(S))}{\sum_{u \in S, v \in S} b(u,v)}$ ← cut few edges
 ← split up many pairs

Suppose $d(u,v)$ was of the form

$$d(u,v) = \sum_{i=1}^k |f_i(u) - f_i(v)| \text{ for } f_i: V \rightarrow [0,1]^k$$

then

$$\sum_{i=1}^k \sum_{e=\{u,v\} \in E} c(e) |f_i(u) - f_i(v)| \leq \lambda \sum_{i=1}^k \sum_{\{s,t\}} b(s,t) |f_i(u) - f_i(v)|$$

$\Rightarrow$ for some i ,

$$\sum_{e=\{u,v\} \in E} c(e) |f_i(u) - f_i(v)| \leq \lambda \sum_{\{s,t\}} b(s,t) |f_i(u) - f_i(v)|$$

which can be rounded as before

morale:

L_1 -metrics can be rounded w/out loss

LP for sparsest cut

$$\min \sum_{e=\{u,v\} \in E} c(e) d(u,v) \text{ over metrics } d: V \times V \rightarrow \mathbb{R}_{\geq 0}$$

$$\text{s.t. } \sum_{\{s,t\}} b(s,t) d(s,t) \geq 1$$

Linear constraints

$$\begin{cases} d(u,v) \geq 0 \quad \forall u,v \\ d(u,v) = d(v,u) \quad \forall u,v \\ d(x,y) \leq d(x,z) + d(z,y) \quad \forall x,y,z \end{cases}$$

Sparsest cut

Input: undirected graph $G=(V,E)$
edge capacities $c(e) > 0$ for $e \in E$
demands $b(u,v) > 0$ for $u,v \in V$
(symmetric)

Goal: compute SCV minimizing

"sparsity" $\frac{c(\delta(S))}{\sum_{u \in S, v \in \bar{S}} b(u,v)}$ $\leftarrow$ cut few edges
 $\leftarrow$ split up many pairs

Q: how to round metric to sparse cut?

New idea: convert $d(u,v)$ to an L_1 -metric!

Theorem 10.7. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log(n) \log(1/\delta))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that for all $u, v \in V$, we have*

$$\|f(u) - f(v)\|_1 \leq O(\log(1/\delta))d(u, v)$$

deterministically, and

$$\mathbf{P}[\|f(u) - f(v)\|_1 \leq d(u, v)] \leq \delta \tag{10.6}$$

Corollary 10.8. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log^2(n))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that with high probability, for all $u, v \in V$,*

$$d(u, v) \leq \|f(u) - f(v)\|_1 \leq O(\log n)d(u, v).$$

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

for ease of notation, let $u_i \stackrel{\text{def}}{=} f_i(u)$

Lemma for $u, v \in V$, index i ,

$$|u_i - v_i| \leq d(u, v)$$

(each coordinate is a contraction)

proof

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Lemma for $u, v \in V$,

$$E[\|u - v\|_1] \geq c d(u, v)$$

proof

for some
constant $c > 0$

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Lemma for $u, v \in V$, index i ,
 $|u_i - v_i| \leq d(u, v)$

all put together:

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Lemma for $u, v \in V$, index i ,
 $|u_i - v_i| \leq d(u, v)$

Lemma for $u, v \in V$,
 $\mathbf{E}[\|u - v\|_1] \geq c d(u, v)$
for some constant $c > 0$

Theorem 10.13. randomized-Fréchet embeds V into $\mathbb{R}^{\lceil \log n \rceil}$ such that

$$\|u - v\|_1 \leq \lceil \log n \rceil d(u, v) \text{ and } \mathbf{E}[\|u - v\|_1] \geq c d(u, v) \text{ for all } u, v \in V,$$

for some absolute constant $c > 0$.

Theorem 10.13. randomized-Fréchet embeds V into $\mathbb{R}^{\lceil \log n \rceil}$ such that

$$\|u - v\|_1 \leq \lceil \log n \rceil d(u, v) \text{ and } \mathbf{E}[\|u - v\|_1] \geq cd(u, v) \text{ for all } u, v \in V,$$

for some absolute constant $c > 0$.

how?

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$
 - A. let S_i sample each $v \in V$ independently with probability 2^{-i}
 - B. for each $v \in V$
 1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$
2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Lemma for $u, v \in V$, index i ,
 $|u_i - v_i| \leq d(u, v)$

Lemma for $u, v \in V$,
 $\mathbf{E}[\|u - v\|_1] \geq cd(u, v)$
 for some constant $c > 0$

Corollary 10.8. Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log^2(n))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that with high probability, for all $u, v \in V$,

$$d(u, v) \leq \|f(u) - f(v)\|_1 \leq O(\log n)d(u, v).$$

Theorem 10.13. randomized-Fréchet embeds V into $\mathbb{R}^{\lceil \log n \rceil}$ such that

$$\|u - v\|_1 \leq \lceil \log n \rceil d(u, v) \text{ and } \mathbf{E}[\|u - v\|_1] \geq c d(u, v) \text{ for all } u, v \in V,$$

for some absolute constant $c > 0$.

take average of $O(\log n)$ embeddings

fix u, v .

each coordinate diff $|u_i - v_i|$

is indep. RV w/ $|u_i - v_i| \leq d(u, v)$

Expected sum (over all coordinates of all embeddings)
 $\geq 10 \log(n) d(u, v)$

Chernoff $\Rightarrow \geq \log(n) d(u, v)$ w/ high probability
 (scaled by $d(u, v)$)

Corollary 10.8. Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log^2(n))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that with high probability, for all $u, v \in V$,

$$d(u, v) \leq \|f(u) - f(v)\|_1 \leq O(\log n) d(u, v).$$

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Lemma for $u, v \in V$, index i ,
 $|u_i - v_i| \leq d(u, v)$

Lemma for $u, v \in V$,
 $\mathbf{E}[\|u - v\|_1] \geq c d(u, v)$
 for some constant $c > 0$

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$

A. let S_i sample each $v \in V$ independently with probability 2^{-i}

B. for each $v \in V$

1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$

2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Theorem 10.14. *With probability of error $1/\text{poly}(n)$, the average of $O(\log n)$ embeddings produced by randomized-Frechet is an embedding V into $\mathbb{R}^{O(\log^2 n)}$ such that*

$$cd(u, v) \leq \|u - v\|_1 \leq C \log(n) d(u, v) \text{ for all } u, v \in V$$

for absolute constants $c, C > 0$.

Theorem 10.9. *There is a $O(\log n)$ randomized approximation to (non-uniform) sparsest cut (on undirected graphs).*

Chapter 10

Sparsest cut

Let G be undirected, and let $b : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a set of nonnegative demands. Given a set S , the *sparsity of S* is defined as the ratio

$$\frac{c(\delta(S))}{\sum_{u \in S, v \notin S} b(u, v)}.$$

(For $S = \emptyset$ or $S = V$, we treat the sparsity of S as $+\infty$). The sparsest cut problem is to compute the set S of minimum sparsity. An important special case is *uniform sparsest cut* where the demands are uniformly $b(u, v) = 1$. Then the sparsity has the simpler form

$$\frac{c(\delta(S))}{|S||\bar{S}|}.$$

To minimize the uniform sparsity, we (of course) want to minimize the numerator and maximize the denominator. Minimizing the numerator is to find small cuts (as usual). The denominator (by AM-GM) is maximized by choosing $|S| \approx n/2$. So the uniform sparsest cut is looking for a tradeoff between the capacity of the cut and how “balanced” the cut is. (In fact, sparsest cut is used as a subroutine for the *balanced cut* problem, as we will discuss.)

To drive this point further, observe that

$$\frac{1}{n} \cdot \frac{c(\delta(S))}{\min\{|S|, |\bar{S}|\}} \leq \frac{c(\delta(S))}{|S||\bar{S}|} \leq \frac{2}{n} \cdot \frac{c(\delta(S))}{\min\{|S|, |\bar{S}|\}}$$

because $n/2 < \max\{|S|, |\bar{S}|\} \leq n$. That is, up to a constant factor, we are trying to minimize the ratio

$$\frac{c(\delta(S))}{\min\{|S|, |\bar{S}|\}}.$$

Here we clearly see the expense of choosing a very small set S .

In this chapter, we describe a very influential result of Leighton and Rao [LR99] that obtains a $O(\log(n))$ approximation ratio for the uniform sparsest cut problem. Their algorithm is based on applying region growing to the metric induced by the dual LP. We then discuss a randomized $O(\log(n))$ approximation algorithm for general demands based on ℓ_1 -metric embeddings. We will also discuss lower bounds and some applications of sparsest cut.

10.1 The LP

Leighton and Rao's algorithm [LR99] is based on rounding an LP relaxation of the sparsest cut problem with region growing. However, since the sparsest cut optimizes a ratio, obtaining the linear relaxation is not as obvious. As a step in this general direction, consider the following (nonlinear) relaxation of the sparsest cut problem.

Compute a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$ minimizing the ratio

$$\frac{\sum_{e=\{u,v\}} c(e)d(u,v)}{\sum_{\{u,v\}} b(u,v)d(u,v)}.$$

Now, we can scale the distances up or down with no effect on the ratio. In particular, we can fix the denominator to be 1, which gives the following optimization problem which *is* a linear program.

Find the minimum c -cost metric such that the b -weighted sum of distances is at least 1.

That is:

$$\begin{aligned} & \text{minimize} && \sum_{e=\{u,v\} \in E} c(e)d(u,v) \\ & \text{over all metrics } d : V \times V \rightarrow \mathbb{R}_{\geq 0} && (10.1) \\ & \text{s.t.} && \sum_{\{s,t\}} b(s,t)d(s,t) \geq 1. \end{aligned}$$

10.1.1 The dual

To obtain the dual, it is helpful to rewrite (10.1) as a pure covering problem. Recall the correspondence between metrics and edge lengths, via shortest path distances. Then (10.1) is the same as:

Find the minimum cost edge lengths such that the b -weighted sum of shortest path distances is at least 1.

To make this more explicit, for $s, t \in V$, let $\mathcal{P}_{s,t}$ denote the family of all (s, t) -paths. Let us define a *path bundle* as a collection of paths P consisting of an (s, t) -path $P_{s,t} \in \mathcal{P}_{s,t}$ for every pair (s, t) . We let $\mathcal{P}_* \stackrel{\text{def}}{=} \prod_{\{s,t\}} \mathcal{P}_{s,t}$ denote the family of all path bundles. Then we can express the problem above as follows.

$$\begin{aligned} & \text{minimize } \sum_{e \in E} c(e)y(e) \text{ over } y : E \rightarrow \mathbb{R}_{\geq 0} \\ & \text{s.t. } \sum_{\{s,t\}} b(s, t) \sum_{e \in P_{s,t}} y(e) \geq 1 \text{ for all } P \in \mathcal{P}_*. \end{aligned} \tag{10.2}$$

We can separate this LP by computing the shortest (s, t) -path for every pair (s, t) , and verifying the covering constraint for this bundle of shortest paths.

(10.2) covers path bundles with edges; thus, the dual packing LP packs path bundles into edges.

$$\begin{aligned} & \text{maximize } \sum_{P \in \mathcal{P}_*} x(P) \text{ over } x : \mathcal{P}_* \rightarrow \mathbb{R}_{\geq 0} \\ & \text{s.t. } \sum_{P \in \mathcal{P}_*} x(P) \sum_{\{s,t\}: e \in P_{s,t}} b(s, t) \leq c(e) \text{ for all } e \in E. \end{aligned} \tag{10.3}$$

In (10.3), each path bundle P represents a choice of paths for every (s, t) -pair to *concurrently* route the demands b . So (10.3) is trying to concurrently route b as much as possible subject to the capacity constraints. This problem is called *concurrent flow* or *demand multicommodity flow*.

10.2 Rounding via region growing (for uniform demands)

Leighton and Rao [LR99] addressed the setting of uniform demands $b = \mathbb{1}$, as well as “product demands” where the demands are of the form $b(u, v) = \pi(u)\pi(v)$ for a fixed vector $\pi \in \mathbb{R}_{\geq 0}^V$. In this section we assume uniform demands, $b = \mathbb{1}$. Then the sparsity of a set S has the cleaner form of

$$\frac{c(\delta(S))}{|S||\bar{S}|}.$$

Let d be an optimum metric to the metric relaxation (10.1). For uniform demands we may assume that $d(u, v) \leq 1$ for all u, v . The goal is to compute a set of vertices S

with

$$\frac{c(\delta(S))}{|S||\bar{S}|} \leq O(\log(n)) \text{OPT}_{(10.1)}.$$

Our analysis is divided into two cases. The first, dubbed the “concentrated” case, is when there is a high concentration of points within a small ball of radius $r_0 \approx 1/n^2$. The second “non-concentrated” case is when this does not occur. In the concentrated case we show that the line embedding from the center of the concentrated ball induces a sparse cut. In the non-concentrated case we partition the graph into pieces with a region-growing technique; the non-concentrated setting implies that these pieces are individually small enough to be reassembled to give a sparse cut.

For the remainder of this section, let $r_0 = 1/2n^2$.

10.2.1 Concentrated case.

First we consider the concentrated case.

Lemma 10.1. *Suppose there exists a vertex s such that $|B(s, r_0)| \geq 2n/3$. Then there exists $r \geq r_0$ such that $B(s, r)$ has sparsity $O(1) \text{OPT}_L$.*

Proof. Suppose not. Then for all $q > 0$, we have

$$\begin{aligned} c(\delta(B(s, r_0 + q))) &\geq c \text{OPT}_L |B(s, r_0 + q)| |V \setminus B(s, r_0 + q)| \\ &\stackrel{(a)}{\geq} \frac{cn \text{OPT}_L}{3} |V \setminus B(s, r_0 + q)| \end{aligned}$$

for a fixed universal constant $c > 0$ (to be determined). Here (a) observes that $|B(s, r_0 + q)| \geq |B(s, r_0)| \geq 2n/3$.

Define a region-growing weight function $W(r)$ differentially by

$$\begin{aligned} W(0) &= 0 \\ W'(0) &= c(\delta(B(s, r))) \end{aligned}$$

$W(r)$ is nondecreasing, bounded below by 0, and bounded above OPT_L (by a similar calculation as in multicut). Now, we have

$$\begin{aligned} 1 &\geq \frac{W(1) - W(r_0)}{\text{OPT}_L} = \int_0^{1-r_0} \frac{c(\delta(B(s, r_0 + q)))}{\text{OPT}_L} dq \\ &\geq \frac{cn}{3} \int_0^{1-r_0} |V \setminus B(s, r_0 + q)| dq \stackrel{(b)}{=} \frac{cn}{3} \left(\sum_{v \in V \setminus B(s, r_0)} d(v, s) - r_0 \right) \\ &\stackrel{(c)}{\geq} \frac{cn}{3} \sum_{v \in V} (d(v, s) - r_0) = \frac{cn}{3} \sum_{v \in V} d(v, s) - \frac{cnr_0}{3}. \end{aligned} \tag{10.4}$$

(b) implicitly interchanges changes. In (c), the inequality holds because the sum is extended only by nonpositive terms.

We also have

$$1 \stackrel{(d)}{\leq} \sum_{\{u,v\}} d(u,v) \stackrel{(e)}{\leq} \sum_{\{u,v\}} d(u,s) + d(s,v) < n \sum_v d(v,s),$$

where (d) is by the distance constraint in the LP and (e) is by the triangle inequality.

Plugging back into (10.4), we have $1 > c/6$. For $c \geq 6$, then, we have a contradiction. $\square$

10.2.2 Non-concentrated case.

Now we address the remaining non-concentrated case.

Lemma 10.2. *Suppose $|B(s, r_0)| \leq 2n/3$ for all $s \in V$. Then one can compute, a cut with sparsity $O(\log n) \text{OPT}_L$.*

Proof. We will apply a region growing technique (similar to multicut) to partition V into sets $S_1, \dots, S_k \subset V$ such that

1. $c(\cup_i \delta(S_i)) \leq O(n^2 \log n) \text{OPT}_L$.
2. $|S_i| \leq 2n/3$ for all i .

Assuming this holds, then, we obtain the desired sparse cut as follows. Since $|S_i| \leq 2n/3$ for all i , we can group the S_i 's into two sets $T \subset V$ and $\bar{T} = V \setminus T$ such that $n/3 \leq |T|, |\bar{T}| \leq 2n/3$. Then T has sparsity

$$\frac{c(\delta(T))}{|T||\bar{T}|} \leq \frac{c(\cup_i \delta(S_i))}{(2n/3)(n/3)} \leq O(\log n) \text{OPT}_L.$$

It remains to obtain the sets $S_1, \dots, S_k$. As alluded to above, we will use region growing to iteratively remove sets of the form $B(s, r)$; the key is to show that there are always “good” radii r .

Fix a center vertex $s \in V$. Consider the region-growing weight function $W(r)$ defined differentially by

$$\begin{aligned} W(0) &= \text{OPT}_L / n \\ W'(r) &= c(\delta(B(s, r))). \end{aligned}$$

$W(r)$ is nondecreasing in r and bounded by the range $[\text{OPT}_L/n, (1 + 1/n) \text{OPT}_L]$. Consequently, for a sufficiently large constant c_0 , there always exists a radius $r < r_0$ such that

$$W'(0) \leq c_0 n^2 \log(n) W(r),$$

since otherwise the differential inequality $W'(0) \geq c_0 \log(n) n^2 W(r)$ implies that

$$W(r_0) \geq e^{c_0 n^2 \log(n) r_0} (\text{OPT}_L/n) > (1 + 1/n) \text{OPT}_L,$$

a contradiction. Thus there exists a radius $r \leq r_0$ that, in particular, satisfies

$$\begin{aligned} c(\delta(B(s, r))) &= W'(r) \leq O(n^2 \log(n)) W(r) \\ &\leq O(n^2 \log(n)) \left(\text{OPT}_L/n + \sum_{v \in B(s, r)} \sum_{e \in \delta(v)} y_e c_e \right). \end{aligned}$$

Moreover, by assumption, we have $|B(s, r)| \leq 2n/3$.

We repeat the following steps until V is empty. Here we let i be the index of the iteration.

1. Pick $s_i \in V$ arbitrarily.
2. Find $r_i \in [0, r_0]$ such that

$$c(\delta(B(s, r))) \leq O(n^2 \log n) \left(\frac{\text{OPT}_L}{n} + \sum_{v \in B(s, r)} \sum_{e \in \delta(v)} c(e) y(e) \right).$$

3. Set $S_i = B(s, r)$ and $V = V \setminus S_i$.

We claim that these steps produce sets $S_1, \dots, S_k$ satisfying the two requirements above.

Clearly we have $|S_i| \leq 2n/3$. We also have

$$\begin{aligned} c\left(\bigcup_i \delta(S_i)\right) &\leq \sum_i c(\delta(S_i)) \\ &\leq O(n^2 \log n) \left(\text{OPT}_L + \sum_i \sum_{v \in S_i} \sum_{e \in \delta(v)} c(e) y(e) \right) \\ &= O(n^2 \log(n)) \left(\text{OPT}_L + \sum_{v \in V} \sum_{e \in \delta(v)} c(e) y(e) \right) \\ &= O(n^2 \log(n)) \text{OPT}_L, \end{aligned}$$

which completes the proof. □

10.2.3 Putting it all together.

Theorem 10.3 (Leighton and Rao [LR99]). *The region-growing algorithm described above gives a $O(\log n)$ -approximation to the uniform sparsest cut.*

Proof. Let y be an optimum solution to (10.2). It is easy to identify if y falls in the concentrated or non-concentrated case. For the former, we obtain a $O(1)$ -approximation; for the latter, a $O(\log n)$ -approximation. $\square$

10.3 Rounding via L_1 -metric embeddings

We now analyze a different approach to rounding the sparsest-cut metric, based on *randomized embeddings*. This version readily generalizes to general demands $b : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$.

Let d be any metric that is a feasible solution to LP (10.1). Our goal is to convert d to a cut with sparsity comparable to the cost of d .

10.3.1 Rounding line embeddings

We first observe that some special cases of metrics are very easy to round – namely, those related to line embeddings. Suppose there is a function $f : V \rightarrow \mathbb{R}$ such that

$$d(u, v) = |f(u) - f(v)| \quad \text{for all } u, v \in V.$$

(Such a function f , placing V on the real line, is called a *line embedding*.) Rescaling and translating (which does not effect the sparsity), we may assume that $\min_u f(u) = 0$, and $\max_v f(v) = 1$.

Consider the following random cut S (which we have seen before in proving the max-flow min-cut theorem). Pick $\theta \in (0, 1)$ uniformly at random, and let $S = \{u : f(u) \leq \theta\}$. Observe that for each edge $e = \{u, v\}$, we have

$$\mathbf{P}[e \in \delta(S)] = |f(u) - f(v)| = d(u, v).$$

Thus we can rewrite the sparsity of d as

$$(\text{sparsity of } d) = \frac{\sum_{\{u,v\} \in E} c(u, v) d(u, v)}{\sum_{\{s,t\}} d(s, t) b(s, t)} = \frac{\mathbf{E}[\sum_{e \in \delta(S)} c(e)]}{\mathbf{E}[\sum_{s \in S, t \in \bar{S}} b(s, t)]}. \quad (10.5)$$

Note that the RHS is *not* the expected sparsity of S . The expected sparsity of S is the quantity

$$\mathbf{E}[\text{sparsity of } S] = \mathbf{E}\left[\frac{\sum_{e \in \delta(S)} c(e)}{\sum_{s \in S, t \in \bar{S}} b(s, t)}\right],$$

is not the same as the quantities in (10.5). This is in contrast to our proof of max-flow min-cut, where S is a minimum (s, t) -cut on average, and the existence of a minimum (s, t) -cut follows immediately from the probabilistic method.

Still the probabilistic approach can be salvaged with a little more work. Observe that S can only be one of $n - 1$ different sets $S_1, \dots, S_{n-1}$ where $\emptyset \subsetneq S_1 \subset S_2 \subset S_3 \cdots \subset S_{n-1} \subsetneq V$. For each i , let $p_i = \mathbf{P}[S = S_i]$. Then

$$(10.5) = \frac{\sum_{i=1}^{n-1} p_i \sum_{e \in \delta(S_i)} c(e)}{\sum_{i=1}^{n-1} p_i \sum_{s \in S_i, t \in \bar{S}_i} b(s, t)}.$$

Now we apply the following elementary fact. (The proof is left as exercise 10.1.)

Lemma 10.4. *Let $a_1, \dots, a_h, b_1, \dots, b_h > 0$. Then*

$$\min_i \frac{a_i}{b_i} \leq \frac{\sum_i a_i}{\sum_i b_i} \leq \max_i \frac{a_i}{b_i}.$$

It follows that for some S_i , the sparsity of S_i is at most the sparsity of d . In conclusion, we have shown the following.

Lemma 10.5. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}$ be a metric induced by a line embedding. Then one can partition the vertices into two sets $(S, \bar{S})$ such that*

$$\frac{\sum_{e \in \delta(S)} c(e)}{\sum_{s \in S, t \in \bar{S}} b(s, t)} \leq \frac{\sum_{\{u, v\} \in E} c(u, v) d(u, v)}{\sum_{\{s, t\}} d(u, v) b(s, t)}.$$

Rounding L_1 -metrics So much for metrics given by line embeddings. How about a metric obtained as a sum of line metrics? Recall that the L_1 -metric on $\mathbb{R}^h$ is defined by

$$\|x - y\|_1 = \sum_{i=1}^h |x_i - y_i|.$$

Suppose d was the L_1 -metric of an embedding $f : V \rightarrow \mathbb{R}^h$. That is,

$$d(u, v) = \|f(u) - f(v)\|_1 = \sum_{i=1}^h |f_i(u) - f_i(v)|$$

for a function $f : V \rightarrow \mathbb{R}^h$. We can think of this as the sum of h line metrics $f_1, \dots, f_h$. The sparsity of d expands out to

$$(\text{sparsity of } d) = \frac{\sum_{i=1}^h \sum_{e=\{u, v\}} c(e) |f_i(u) - f_i(v)|}{\sum_{i=1}^h \sum_{\{u, v\}} b(u, v) |f_i(u) - f_i(v)|}.$$

Applying lemma 10.4 again, we see that one of these line embeddings, say f_j , has sparsity no worse than d . From the line embedding $f_j : V \rightarrow \mathbb{R}$, we can extract a cut with sparsity at most that of f_j . This establishes the following.

Lemma 10.6. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}$ be the L_1 -metric over an explicit embedding of V . Then one can partition the vertices into two sets $(S, \bar{S})$ such that*

$$\frac{\sum_{e \in \delta(S)} c(e)}{\sum_{s \in S, t \in \bar{S}} b(s, t)} \leq \frac{\sum_{\{u, v\} \in E} c(u, v) d(u, v)}{\sum_{\{s, t\}} d(u, v) b(s, t)}.$$

To sum up: L_1 metrics can be rounded without loss. We can find an L_1 -metric with sparsity within a factor α of the sparsest metric d , then we can convert that into an α -approximate sparsest cut.

10.3.2 Randomized L_1 -metric embeddings

We now know that L_1 -metrics can be rounded to sparse cuts without any loss. But the LP for sparsest cut produces a generic metric d , that is not an L_1 -metric.

Our new strategy, given a generic metric d , is to try to find an L_1 -metric d_1 with sparsity comparable to d . We then invoke lemma 10.6 to round obtain a sparse cut from d_1 . Our L_1 -metric d_1 will be defined by a mapping $f : V \rightarrow \mathbb{R}^h$ (for some $h \in \mathbb{N}$), so that

$$d_1(u, v) = \|f(u) - f(v)\|_1.$$

We will prove the following theorem.

Theorem 10.7. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log(n) \log(1/\delta))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that for all $u, v \in V$, we have*

$$\|f(u) - f(v)\|_1 \leq O(\log(1/\delta))d(u, v)$$

deterministically, and

$$\mathbf{P}[\|f(u) - f(v)\|_1 \leq d(u, v)] \leq \delta \tag{10.6}$$

For $\delta = 1/\text{poly}(n)$ and $h = O(\log^2 n)$, we can apply the union bound to (10.6) over all pairs u, v . This gives the following theorem.

Corollary 10.8. *Let $d : \binom{V}{2} \rightarrow \mathbb{R}_{\geq 0}$ be a metric and $\delta \in (0, 1)$. For $h = O(\log^2(n))$, one can construct a randomized embedding $f : V \rightarrow \mathbb{R}^h$ such that with high probability, for all $u, v \in V$,*

$$d(u, v) \leq \|f(u) - f(v)\|_1 \leq O(\log n)d(u, v).$$

The embedding $f : V \rightarrow \mathbb{R}^h$ described in corollary 10.8 is said to be a $O(\log n)$ -distortion metric embedding as it maps points in one metric space into another while preserving all distances up to a $O(\log n)$ -multiplicative factor.

The $O(\log n)$ -distortion embedding into L_1 is the last ingredient for the following algorithm for sparsest cut.

1. Solve the LP (10.1) to obtain a sparsest metric d .
2. Invoke corollary 10.8 to obtain a $O(\log n)$ -distortion embedding $f : \binom{V}{2} \rightarrow \mathbb{R}^h$ from d into the L_1 -metric. The L_1 -metric via f has sparsity at most a $O(\log n)$ factor greater than d .
3. Invoke lemma 10.6 to round the L_1 -metric to a cut with sparsity at most the metric, which is a factor $O(\log n)$ greater than the sparsity of d (and the optimum of (10.1)).

This algorithm is due to Linial, London, and Rabinovich [LLR95], and establishes the following.

Theorem 10.9. *There is a $O(\log n)$ randomized approximation to (non-uniform) sparsest cut (on undirected graphs).*

Actually, Linial, London, and Rabinovich [LLR95] observed that one can do slightly better when there are demands for only k commodity pairs. (i.e., k pairs (u, v) with $b(u, v) > 0$.)

Theorem 10.10. *There is a $O(\log k)$ randomized approximation to sparsest cut, where k is the number of commodity pairs with nonzero demand.*

This approximation factor is obtained by building on the ideas in ??, and left as exercise 10.2.

Now we describe the randomized algorithm of Linial, London, and Rabinovich [LLR95] that computes the embedding in theorem 10.14. We note that previously Bourgain [Bou85] had obtained a deterministic embedding but the output dimension h was exponential. Linial, London, and Rabinovich's algorithm [LLR95] can be

random-Fréchet($d : V \times V \rightarrow \mathbb{R}_{\geq 0}$)

1. for $i = 1, \dots, \lceil \log n \rceil$
 - A. let S_i sample each $v \in V$ independently with probability 2^{-i}
 - B. for each $v \in V$
 1. $f_i(v) \leftarrow \min_{s \in S_i} d(s, v)$
2. return $f : V \rightarrow \mathbb{R}_{\geq 0}^{\lceil \log n \rceil}$

Figure 10.1: A $O(\log n)$ dimension, randomized Fréchet embedding with $O(\log n)$ distortion in expectation

interpreted as an efficient randomized implementation of Bourgain’s embedding [Bou85].

The algorithm, which we call **random-Fréchet**, is extremely simple. We generate $\lceil \log n \rceil$ coordinates. For $i = 1, \dots, n$, we sample a set S_i where each point is sampled independently with probability $1/2^i$. For each vertex v , we find the distance between v and (the closest point in) S_i . This gives the i th coordinate of v . Pseudocode is described in fig. 10.1.

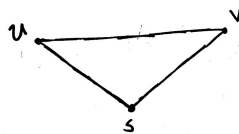
Each coordinate of the randomized embedding is given by distances from a set. There is a name for this class of embeddings: *Fréchet embeddings*. fig. 10.2 attempts to visualize a single coordinate generated in this manner.

10.3.3 Low-distortion in expectation

We now turn to proving theorem 10.14. Consider an instance of the **random-Fréchet** algorithm, which computes a randomized embedding $f : V \rightarrow \mathbb{R}_{\geq 0}^{O(\log(n))}$.

For ease of notation, for a vertex v and coordinate i , we let $v_i \stackrel{\text{def}}{=} f_i(v)$ denote the i th coordinate of the embedding of v .

Lemma 10.11. *For $u, v \in V$ and $i \in \mathbb{N}$, $|u_i - v_i| \leq d(u, v)$.*



Proof. By the triangle inequality, we have both

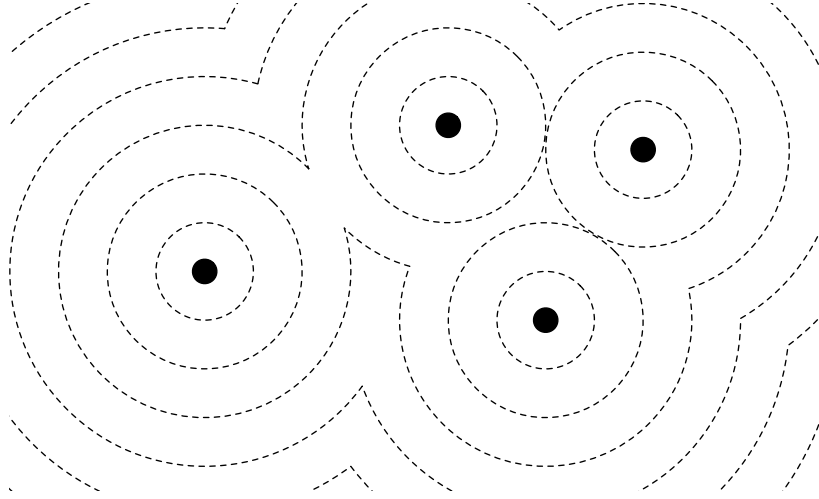


Figure 10.2: Level sets by distance from a set of points S , encoding one coordinate of a Frechét embedding.

$$d(s, u) - d(s, v) \leq d(u, v) \text{ and } d(s, v) - d(s, u) \leq d(u, v).$$

for all $s \in S_i$. □

lemma 10.11 immediate implies the $\|u - v\|_1 \leq O(\log n)d(u, v)$, since there are $O(\log n)$ dimensions and each can contribute at most $d(u, v)$. The harder part is showing the lower bound: informally, we want to show that $\|u - v\|_1 \geq d(u, v)$, up to constant factors. This lower bound is too strong; instead, we settle for the same inequality but only in expectation.

Lemma 10.12. *For $u, v \in V$, we have $\mathbf{E}[\|u - v\|_1] \geq cd(u, v)$ for some constant $c > 0$.*

Proof. For ease of notation, let $\delta = d(u, v)$. For each i , let r_i be the minimum length r such that there are at least 2^i points at distance $\leq r$ from u , and 2^i points at distance $\leq r$ from v ; i.e.,

$$r_i = \arg \min_{r > 0} \left\{ |\{x : d(u, x) \leq r\}| \geq 2^i, |\{x : d(v, x) \leq r\}| \geq 2^i \right\}$$

We claim that

For each index i , we have

$$|u_{i+1} - v_{i+1}| \geq (\min\{r_i, \delta/2\} - \min\{r_{i-1}, \delta/2\})$$

with constant probability $c > 0$.

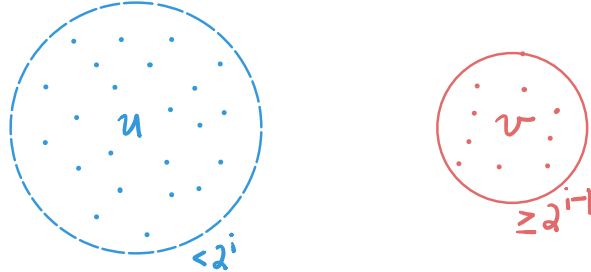
Before proving the claim, suppose it holds true. Let k be the largest index such that $r_{k-1} \leq \delta/2$ (for which the claim applies). We have

$$\begin{aligned} \mathbf{E}[\|u - v\|_1] &\geq \sum_{i=0}^k \mathbf{E}[\|u_{i+1} - v_{i+1}\|] \\ &\stackrel{(a)}{\geq} c \sum_{i=0}^k (\min\{r_i, \delta/2\} - \min\{r_{i-1}, \delta/2\}) \\ &\stackrel{(b)}{\geq} \frac{c\delta}{4}, \end{aligned}$$

as desired. Here (a) applies the claim and (b) is by telescoping sums and recalling that $r_{k+1} > \delta/2$.

It remains to prove the claim. We have two cases: (a) $r_i \leq \delta/2$, and (b) $r_{i-1} < \delta/2 \leq r_i$. We assume without loss of generality that r_i is defined by u ; i.e., $|\{x : d(u, x) < r_i\}| < 2^i$.

Case 1: $r_i \leq \delta/2$. Let $U = \{x : d(u, x) < r_i\}$, and let $V = \{x : d(v, x) \leq r_{i-1}\}$. We have $|U| < 2^i$ and $|V| \geq 2^{i-1}$. Since $r_{i-1} < r_i \leq \delta/2$, U and V are disjoint.

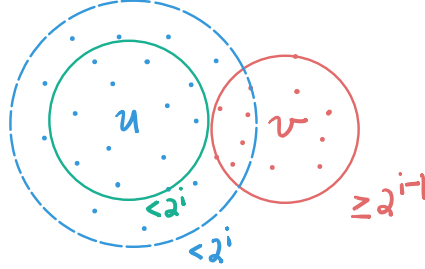


S_{i+1} samples each point with probability 2^{-i-1} . By direct calculation, S_{i+1} samples no points from U with constant probability, and at least one point from V with constant probability. Since U and V are disjoint, whether any point from U is sampled and whether any point from V is sampled is independent. Thus S_{i+1} samples a point from V and no points from U simultaneously with some constant probability $c > 0$. In this event, we have $u_{i+1} \geq r_i$ and $v_{i+1} \leq r_{i-1}$, so $u_{i+1} - v_{i+1} \geq r_i - r_{i-1}$. In expectation, we have

$$\mathbf{E}[\|u_{i+1} - v_{i+1}\|] \geq c(r_i - r_{i-1}),$$

as desired.

Case 2: $r_i > \delta/2 > r_{i-1}$. Let $U = \{x : d(u, x) \leq \delta/2\}$ (with $\delta/2$ in place of r_i) and let $V = \{x : d(u, v) \leq r_{i-1}/2\}$.



By the same argument as above, we have that S_{i+1} samples a point from V and no points in U with some constant probability $c > 0$. In this event, $u_{i+1} - v_{i+1} \geq \delta/2 - r_{i-1}$. $\square$

Theorem 10.13. *randomized-Frechet embeds V into $\mathbb{R}^{\lceil \log n \rceil}$ such that*

$$\|u - v\|_1 \leq \lceil \log n \rceil d(u, v) \text{ and } \mathbf{E}[\|u - v\|_1] \geq cd(u, v) \text{ for all } u, v \in V,$$

for some absolute constant $c > 0$.

10.3.4 Amplification

Theorem 10.13 shows that a single instance of **randomized-Fréchet** obtains $O(\log n)$ distortion “in expectation”, so to speak, for each pair of vertices. In particular the embedded distance is bounded above deterministically but below only in expectation. We want to strengthen this so that the lower bound holds with high probability.

Theorem 10.14. *With probability of error $1/\text{poly}(n)$, the average of $O(\log n)$ embeddings produced by **randomized-Frechet** is an embedding V into $\mathbb{R}^{O(\log^2 n)}$ such that*

$$cd(u, v) \leq \|u - v\|_1 \leq C \log(n) d(u, v) \text{ for all } u, v \in V$$

for absolute constants $c, C > 0$.

Proof sketch. Fix $u, v \in V$. We treat each coordinate difference $|u_i - v_i|$ (for $O(\log^2 n)$ coordinates over $O(\log n)$ independent calls to **random-Frechet**) as an independent random variable bounded above by $d(u, v)$. The expected sum of the $|u_i - v_i|$'s is $\Omega(n \log(n))$. By standard Chernoff inequalities, the sum is strongly concentrated at the mean; scaling down by $\log n$ (from averaging) gives the desired result. $\square$

10.4 Expander lower bounds

Recall that the *integrality gap* of an LP relaxation is the maximum (multiplicative) difference between the value of the LP and the discrete problem. The rounding algorithms above show that the integrality gap of the metric LP relaxation (10.1) is at most $O(\log n)$ for uniform sparsest cut (or $O(\log k)$ for k commodity pairs). We want to find a lower bound on the integrality gap as tight as possible.

Consider sparsest cut. The LP asks for the sparsest metric. The sparsest cut is at most a $O(\log n)$ -multiplicative factor greater than the sparsity of the sparsest metric. We want to find a family of graphs where the sparsest cut is at least a $\Omega(\log n)$ -multiplicative factor greater than the sparsest metric. We will show this for a family of graphs called *constant-degree expanders*.

An unweighted, undirected graph G is a φ -*expander* if for all sets $S \subset V$,

$$|\delta(S)| \geq \varphi \min\{|S|, |\bar{S}|\}.$$

The definition extends to weighted graphs by replacing the LHS with the weight of the cut. We only need to consider unweighted graphs for our lower bound.

An obvious example of a graph with high expansion is a clique, i.e., a complete graph. The clique on n vertices has expansion $n/2$. An obvious example of a graph with low expansion is a tree; a tree on n vertices has expansion $\Omega(1/n)$. As noted in the introduction, expansion and uniform sparsity are, up to scaling, within a constant factor of one another. Sparse cuts imply low expansion and high expansion imply no sparse cuts.

The clique and tree examples reflects a general intuition that high expansion is correlated with many edges in the graph. But in fact this is not necessarily so: there exists graphs with constant expansion (much higher than a tree) but constant maximum degree (like a tree, and unlike a clique).

Theorem 10.15. *There exists constants c, d such that for all sufficiently large n , there exists a c -expander with n vertices and maximum degree d .*

The fact that a very small (constant degree) graph can be strongly connected (i.e., have large expansion) has vast and profound consequences. We will see this phenomena realized in a few different places in this course. Below we will prove theorem 10.15 by showing that the union a constant number of uniformly random, perfect matchings is an expander with nonzero probability. First, we use theorem 10.15 to lower bound the integrality gap.

10.4.1 Integrality gap for uniform sparsest cut

Theorem 10.16. *Integrality gap for uniform sparsest cut is $O(\log n)$.*

Proof. By theorem 10.15, let G be an n -vertex c -expander with maximum degree d , where c and d are universal constants. For every set of vertices S , we have

$$(\text{sparsity of } \delta(S)) = \frac{|\delta(S)|}{|S||\bar{S}|} \stackrel{(a)}{\geq} \frac{c \min\{|S|, |\bar{S}|\}}{|S||\bar{S}|} \geq \frac{c}{n}$$

where (a) invokes the definition of a c -expander.

By LP duality, the sparsest metric has sparsity equal to the maximum throughput λ of any concurrent flow. We claim that any concurrent flow has throughput λ at most $O(1/n \log n)$. Then multiplicative difference between c/n and $O(1/n \log n)$ is $\Omega(\log n)$, as desired.

To upper bound λ , we argue that, since G has constant maximum degree, for almost every (s, t) -pair, the length of the shortest (s, t) -path is at least (roughly) $\log n$. This implies that all the flow paths in the concurrent flow has at last $\log n$, and almost every (s, t) -pair requires a total capacity of λ . Meanwhile the total capacity of the graph is limited because G is sparse, which leads to an upper bound on λ .

To this end, fix s . Since G has maximum degree d , there are at most d vertices at distance 1 from s , at most d^2 vertices at distance 2 from s , and so forth. In general there are $O(d^i)$ vertices at distance i or less from s . For $h = \log_d(n)/2$, there are at most $O(\sqrt{n})$ vertices at distance h or less from s . This leaves $n - O(\sqrt{n})$ vertices at distance $h = \Omega(\log n)$ or greater from s . Over all s , there are $(1 - o(1))\binom{n}{2}$ pairs (s, t) at distance $\Omega(\log n)$.

It follows that any concurrent flow with through λ – which, in particular, sends λ units of flow between $\Omega(n^2)$ (s, t) -pairs at distance $\Omega(\log n)$ – requires a total capacity of $O(n^2 \log(n) \lambda)$. But the total capacity in the constant degree expander is $O(n)$. So $\lambda \leq O(1/n \log n)$. $\square$

Remark 10.17. The proof above used to constant maximum degree to argue that most (s, t) -pairs are at distance $\Omega(\log n)$. Meanwhile one can show that a graph with constant expansion has diameter $O(\log n)$ ¹. See exercise 10.6.

10.4.2 A few random matchings make an expander

We now turn to proving theorem 10.15. While a proof of existence would suffice, we will prove theorem 10.15 via a very simple, randomized construction (for n even):

¹The diameter of a graph is the maximum (s, t) -distance among all (s, t) pairs.

namely, for a fixed set of n vertices, the union of a few (independently sampled) uniformly random perfect matchings over these n vertices has constant expansion. We remark that there are many other constructions and in particular there is interest in questions such as deterministic constructions, implicit constructions of very large expanders with compact oracle access, and extreme families such as Ramanujan graphs.

Theorem 10.18. *Let n be even and sufficiently large. There exists constants $c, d > 0$ such that the union of d independent and uniformly random matchings of a set of n vertices form a c -expander with high probability.*

Proof. Let $d \in \mathbb{N}$ be a parameter to be determined, and consider the union of d perfect matchings over a fixed set of n vertices. We want to show that for every set S of size $|S| \leq n/2$, $|\delta(S)| \geq c|S|$ for some constant c .

We reframe this claim as follows. Let $S \subset V$ be of size $k \stackrel{\text{def}}{=} |S| \leq n/2$. Let $T \subseteq V \setminus S$ be of size $|T| = k/6$.

Claim. *With nonzero probability, for all such S and T , some vertex in S is matched to a vertex outside $S \cup T$.*

The claim directly implies that, for all sets S with at most $n/2$ vertices, $|N(S) \cup S| \geq (7/6)|S|$. (Otherwise take $T = N(S) \setminus S$.) In turn, for all such S , we have

$$|\delta(S)| \geq |N(S) \cup S| - |S| \geq |S|/6,$$

as desired.

We will prove the claim probabilistically: our first goal is to *fix S and T , and bound the probability that all of S is matched to $S \cup T$.*

To this end, for fixed S and T , consider the following randomized procedure for generating a perfect matching.

1. Index the vertices $v_1, \dots, v_n$ such that $S = \{v_1, \dots, v_k\}$ and $T = \{v_{k+1}, \dots, v_{7k/6}\}$.
2. Repeat $n/2$ times:
 - (a) Let v_i be the unmatched vertex of smallest index
 - (b) Sample v_j among the remaining unmatched vertices uniformly at random.
 - (c) Match v_i with v_j

While this procedure might appear biased by S and T , we argue that it is not.

Claim. The above steps generate a uniformly random perfect matching.

Indeed, observe that each random choice in the procedure leads to a different matching. There are $(n-1)(n-3)\cdots 1$ possible outcomes and all are equally likely. There are also $(n-1)(n-3)\cdots 1$ possible matchings. This proves the claim.

Thus we may assume that our d perfect matchings are generated by the procedure above. Now, for a single matching, the probability that S is matched to $(S \cup T)$ is bounded above by probability that first $k/2$ choices of v_j 's are in $S \cup T$. The probability of this is bounded above by

$$\prod_{i=1}^{k/2} \frac{(7/6)k - 2i + 1}{n - 2i + 1} \stackrel{(a)}{\leq} \prod_{i=k/3+1}^{k/2} \left(\frac{k/2}{n/2} \right) = \left(\frac{k}{n} \right)^{k/6} \leq \left(\frac{n}{k} \right)^{-1/6}.$$

In (a) we observe that

$$(7/6)k - 2i + 1 \leq k/2 \text{ iff } i \geq k/3 + 1/2$$

and that

$$n - 2i + 1 \geq n/2 \text{ iff } i \leq n/4 + 1/2, \text{ if } i \leq k/2 + 1/2$$

For d matchings, the probability that S is matched to $(S \cup T)$ in all d matchings is bounded above by $\left(\frac{n}{k} \right)^{-d/6}$. Now we take a union bound over all choices of S and T . We have

$$\binom{n}{7k/6} \binom{7k/6}{k} \left(\frac{n}{k} \right)^{-d/6} = \binom{n}{k}^{1-d/6} \leq \binom{n}{k}^{-\Omega(d)}$$

for a sufficiently large constant d . Taking a union bound over $k = 1, 2, \dots, n/2$, we conclude that with high probability, all sets S with $|S| \leq n/2$ have at least $|S|/6$ neighbors. So the union of a constant number of random matchings is a $(1/6)$ -expander. $\square$

10.5 Application: Minimum bisection

A *bisection* is a partition of the vertices V into $(S, \bar{S})$ of (essentially) equal size: $\lfloor n/2 \rfloor \leq |S|, |\bar{S}| \leq \lceil n/2 \rceil$. Alternatively a bisection can be defined in terms of cuts as a set of edges whose removal leaves the graph with connected components of at most $\lceil n/2 \rceil$ vertices each. In any case the minimum bisection problem is to compute a vertex set $S \subset V$ of size $|S| = \lfloor n/2 \rfloor$ minimizing the cost of the cut, $c(\delta(S))$.

There is a natural connection between minimum bisection and the sparsest cut – the minimum bisection problem can be recast as the restricting the sparsest cut problem to vertex sets with exactly half the vertices.

The following algorithm uses a $O(\log n)$ -approximation for uniform sparsest to obtain a *bicriteria*-approximation algorithm. In particular, it returns a $(1/3)$ -balanced partition $(S, \bar{S})$ – that is, $n/3 \leq |S| \leq 2n/3$ – with cost at most $O(\log n)$ times the cost of the minimum bisection. The algorithm is very simple. It repeatedly computes the sparsest cut and removes the smaller side from the graph, until the number of vertices removed is at least $n/3$ (and necessarily at most $2n/3$).

1. For $i = 1, 2, \dots$
 - A. $S_i \leftarrow$ smaller side of a $O(\log n)$ -approximate uniform sparsest cut.
 - B. If $|S_1| \cup \dots \cup |S_i| \geq n/3$
 1. return $(S_1 \cup \dots \cup S_i, V - (S_1 \cup \dots \cup S_i))$.
 - C. Else remove S_i and all incident edges from the graph, and repeat.

It is (relatively) easy to see why the algorithm returns a $(1/3)$ -balanced cut; it remains to show that the cost is comparable to that of the minimum bisection. The intuition is as follows. Suppose for simplicity we have an exact algorithm for the sparsest cut. The sparsest cut is very close to the minimum (weighted) expansion, which we recall is the cost of the cut divided by the number of vertices on the smaller side of the cut. In particular this ratio for the sparsest cut is no worse than that of the minimum bisection. If the sparsest cut is balanced, then its cost is comparable to the minimum cost bisection. While its not balanced, we can interpret the sparsest cut as removing some vertices from the graph at the cost of the edges being cut. The ratio of vertices removed per unit cost – the bang-for-buck, so to speak – is at least as good. So we are gradually removing vertices while paying a favorable rate compared to the minimum cost bisection.

There are some additional details to take care of – for one, the minimum bisection in the input graph may no longer be a minimum bisection in the residual graphs, although it will still be somewhat balanced as long as we haven't removed $n/3$ vertices yet. Also we only have a $O(\log n)$ -approximation for the sparsest cut, which will imply that we pay an additional $O(\log n)$ factor throughout the argument. exercise 10.3 guides the reader through a formal proof of the argument.

Theorem 10.19. *In polynomial time, one can compute $(1/3)$ -balanced cut with total cost at most a $O(\log n)$ -factor greater than the minimum bisection.*

10.6 Exercises

Exercise 10.1. Prove lemma 10.4.

Exercise 10.2. Recall the randomized $O(\log n)$ approximation algorithm for sparsest cut based on L_1 -embeddings. Show how to adjust the algorithm and analysis to obtain a randomized $O(\log k)$ approximation factor where k is the number of commodities with nonzero demand.

Exercise 10.3. Recall the bicriteria approximation algorithm for the minimum bisection problem from section 10.5.

1. Show that the algorithm returns a $1/3$ -balanced cut.
2. For each iteration i , w/r/t the graph remaining at iteration i , we have

$$\frac{c(\delta(S_i))}{|S_i|} \leq O(\log(n)) \frac{\text{OPT}}{n}.$$

3. Combine the two parts above to prove that the algorithm returns a $1/3$ -balanced cut of size $O(\log n)$ OPT.

Exercise 10.4. One generalization of uniform sparsest cut is to directed graphs. Here one is given a directed graph G with edge costs c ; the (directed, out-) sparsity of a set S is defined

$$\frac{c(\delta^+(S))}{|S||\bar{S}|}$$

where we recall that $\delta^+(S)$ is the directed cut of edges leaving S . The directed uniform sparsest cut problem to find the set S minimizing the directed sparsity just defined. Extend Leighton and Rao's algorithm [LR99] to obtain a $O(\log n)$ approximation for the directed uniform sparsest cut².

Exercise 10.5. One can also consider the bisection problem in directed graphs. Here the goal is to find a vertex set S of size $\lfloor n/2 \rfloor \leq |S| \leq \lceil n/2 \rceil$ minimizing the cost of the directed cut $c(\delta^+(S))$. Suppose one had access to a $O(\log n)$ approximation algorithm for uniform directed sparsest cut (as described in exercise 10.4). Using this as a subroutine, design and analyze an algorithm that obtains a bicriteria approximation algorithm for the minimum directed bisection problem with essentially the same approximation bicriteria for the undirected setting: compute a set S with $n/3 \leq |S| \leq 2n/3$ with cost $c(\delta^+(S))$ at most a $O(\log n)$ -factor greater than that of the minimum directed bisection.

²This is not an easy exercise.

Exercise 10.6. Prove that any graph G with constant expansion has diameter $O(\log n)$.

10.7 Additional notes and references

One can do better than a $O(\log n)$ approximation for uniform sparsest cut – Arora, Rao, and Vazirani [ARV09] gave a $O(\sqrt{\log n})$ via semi-definite programming and ideas from high-dimensional geometry. We may discuss this result later in the course; in the meantime we refer the reader to lecture notes by Rothvoss [Rot16].

We refer to Spielman [Spi19] and Trevisan [Tre16] for further background on expanders, especially in the context of spectral graph theory. ?? follows the proof in [Tre16].