Graph Sparsification by Effective Resistances

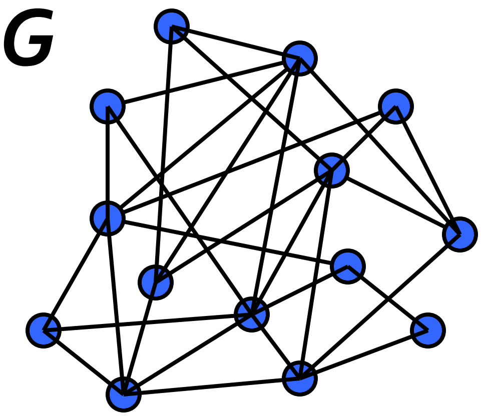
Daniel Spielman
Nikhil Srivastava
Yale

STOC 2008

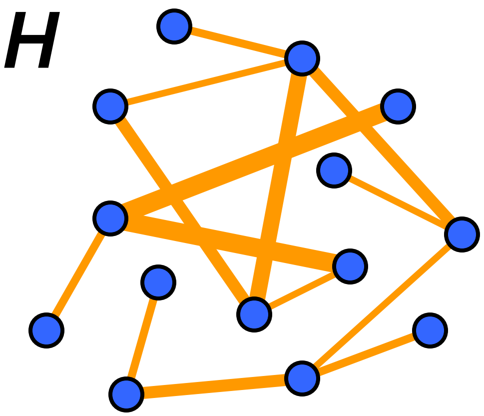
Sparsification

Approximate any graph ***G*** by a sparse graph ***H***.

G



H



– Nontrivial statement about ***G***

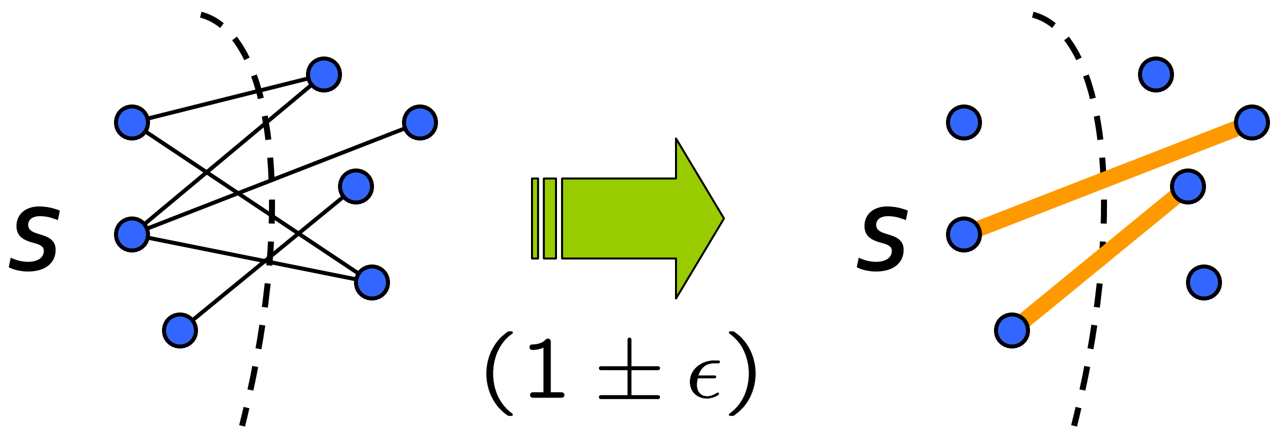
– ***H*** is faster to compute with than ***G***

Cut Sparsifiers [Benczur-Karger'96]

H approximates ***G*** if

for every cut ***S***$\subset$***V***

sum of weights of edges leaving ***S*** is preserved



Can find ***H*** with O(nlogn/ε²) edges in $\tilde{O}(m)$ time

The Laplacian (quick review)

$$L_G = D_G - A_G$$

Quadratic form

$$x : V \to \mathbb{R}$$

$$x^T L_G x = \sum_{uv \in E} c_{uv}(x(u) - x(v))^2$$

Positive semidefinite

Ker(L_G)=span($\mathbf{1}$) if ***G*** is connected

Cuts and the Quadratic Form

For characteristic vector $x_S \in \{0,1\}^n$ of $S \subseteq V$

$$\begin{aligned} x_S^T L_G x_S &= \sum_{uv \in E} c_{uv}(x(u) - x(v))^2 \\ &= \sum_{uv \in (S, \overline{S})} c_{uv} \\ &= wt_G(S, \overline{S}) \end{aligned}$$

So BK says:

$$1 - \epsilon \leq \frac{x^T L_H x}{x^T L_G x} \leq 1 + \epsilon \quad \forall x \in \{0,1\}^n$$

A Stronger Notion

For characteristic vector $x_S \in \{0,1\}^n, S \subseteq V$

$$
\begin{aligned}
x_S^T L_G x_S &= \sum_{uv \in E} c_{uv}(x(u) - x(v))^2 \\
&= \sum_{uv \in (S,\overline{S})} c_{uv} \\
&= wt_G(S, \overline{S})
\end{aligned}
$$

So BK says:

$$\forall x \in \mathbb{R}^n$$

$$1 - \epsilon \leq \frac{x^T L_H x}{x^T L_G x} \leq 1 + \epsilon \quad \cancel{\forall x \in \{0,1\}^n}$$

Why?

By Courant-Fischer,

$$(1-\epsilon)\lambda_i(G) \leq \lambda_i(H) \leq (1+\epsilon)\lambda_i(G)$$

stronger than matrix sparsifiers
[AM01,DK03,FKV04]

$$\|L_G - L_H\|_2 \leq \epsilon$$

Why?

By Courant-Fischer,

$$(1-\epsilon)\lambda_i(G) \leq \lambda_i(H) \leq (1+\epsilon)\lambda_i(G)$$

stronger than matrix sparsifiers [AM01,DK03,FKV04]

$$\|L_G - L_H\|_2 \leq \epsilon$$

Linear systems in L_G ~ linear systems in L_H [ST04]

Examples

Example: Sparsify Complete Graph by Ramanujan Expander

G is complete on n vertices. $\lambda_i(L_G) = n$

H is d-regular Ramanujan graph. $\lambda_i(L_H) \sim d$

$$\lambda_i(\tfrac{n}{d} L_H) \sim n$$

Example: Sparsify Complete Graph by Ramanujan Expander

G is complete on n vertices. $\lambda_i(L_G) = n$

H is d-regular Ramanujan graph. $\lambda_i(L_H) \sim d$

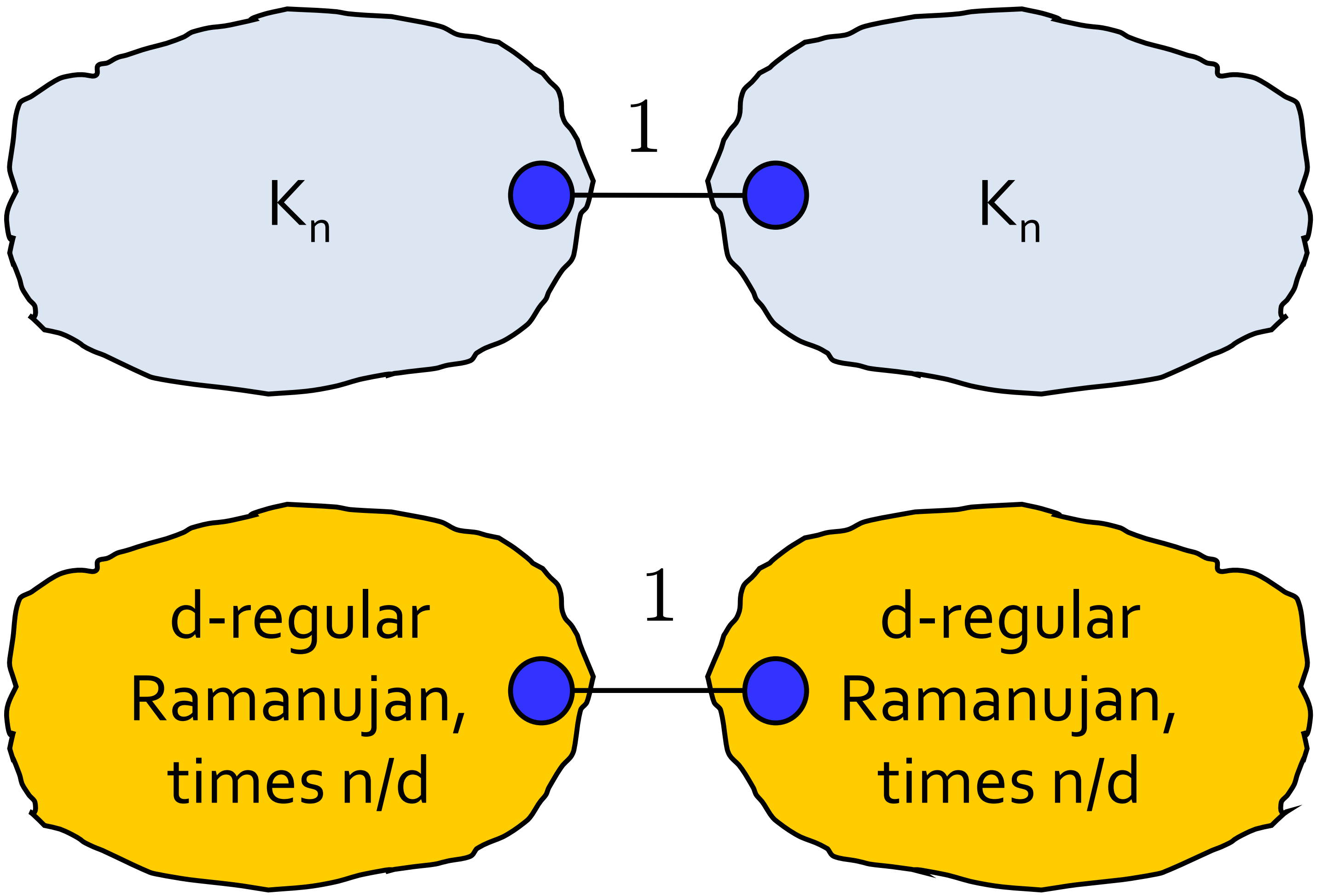
$$\lambda_i(\tfrac{n}{d} L_H) \sim n$$

$$\frac{x^T(\frac{n}{d} L_H)x}{x^T L_G x} \sim 1$$

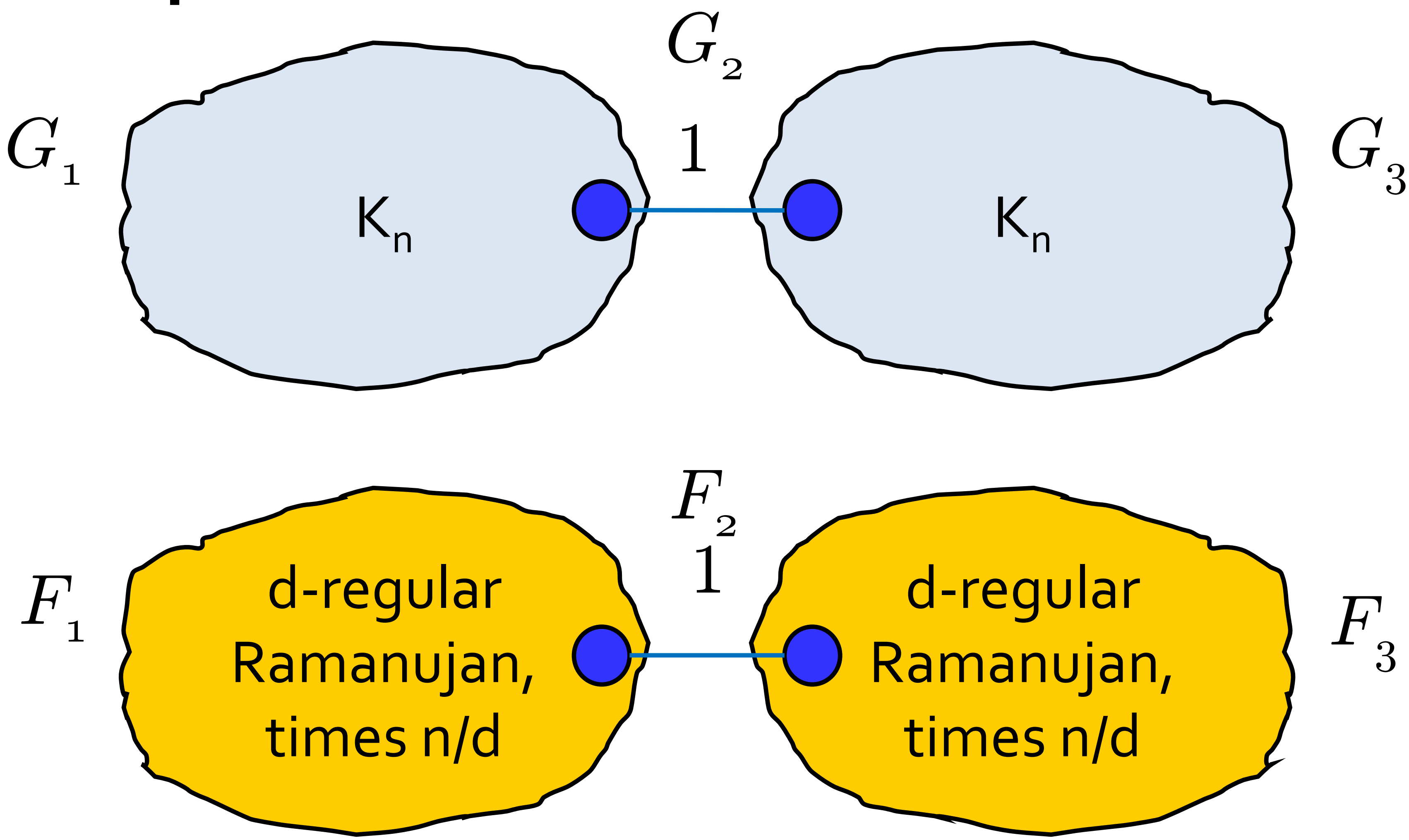
Each edge has weight (n/d)

So, $\frac{n}{d}H$ is a good sparsifier for ***G***.

Example: Dumbell



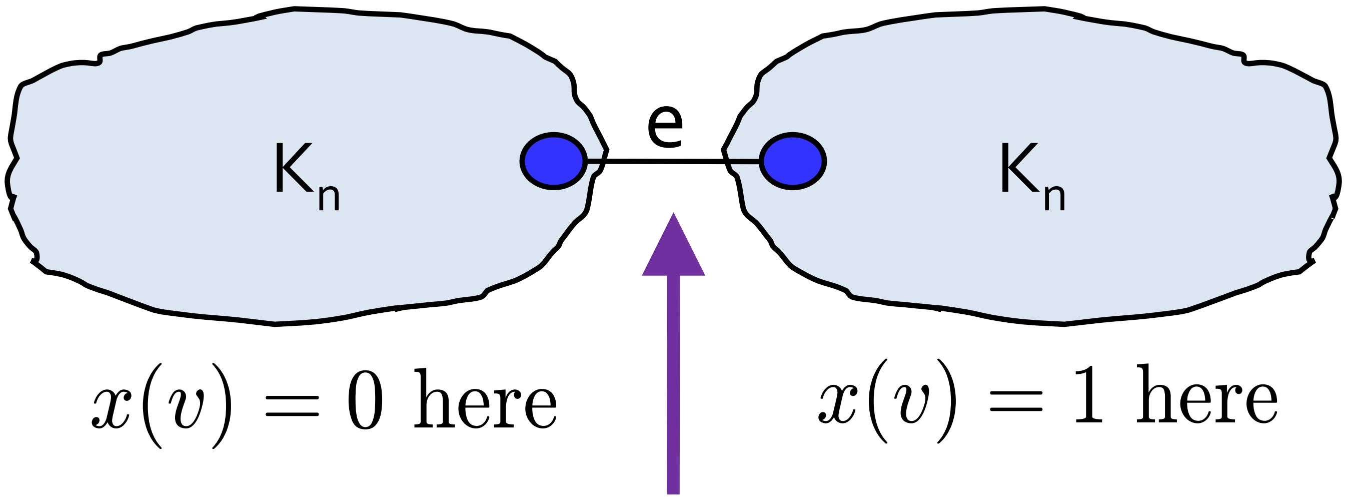
Example: Dumbell



$$G = G_1 + G_2 + G_3$$

$$x^T G x = x^T G_1 x + x^T G_2 x + x^T G_3 x$$

Example: Dumbell. Must include cut edge



Only this edge contributes to

$$x^T L_G x = \sum_{(u,v) \in E} c_{(u,v)} (x(u) - x(v))^2$$

If $\quad e \notin H, \quad x^T L_H x = 0$

Results

Main Theorem

Every ***G=(V,E,c)*** contains ***H=(V,F,d)*** with $O(n\log n/\varepsilon^2)$ edges such that:

$$(1-\epsilon)x^T L_G x \le x^T L_H x \le (1+\epsilon)x^T L_G x \quad \forall x \in \mathbb{R}^n$$

Main Theorem

Every ***G=(V,E,c)*** contains ***H=(V,F,d)*** with $O(n\log n/\varepsilon^2)$ edges such that:

$$(1-\epsilon)x^T L_G x \le x^T L_H x \le (1+\epsilon)x^T L_G x \quad \forall x \in \mathbb{R}^n$$

Can find ***H*** in $\tilde{O}(m)$ time by random sampling.

Main Theorem

Every ***G=(V,E,c)*** contains ***H=(V,F,d)*** with $O(n\log n/\varepsilon^2)$ edges such that:

$$(1-\epsilon)x^T L_G x \leq x^T L_H x \leq (1+\epsilon)x^T L_G x \quad \forall x \in \mathbb{R}^n$$

Can find ***H*** in $\tilde{O}(m)$ time by random sampling.

Improves [BK'96]

Improves $O(n\log^c n)$ sparsifiers [ST04]

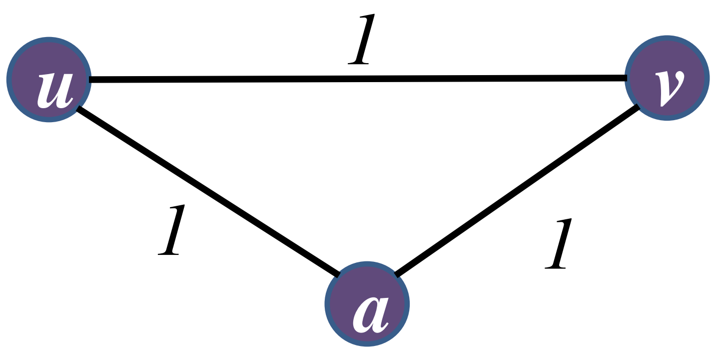
How?

Electrical Flows.

Effective Resistance

Identify each edge of G with a unit resistor

$R_{\text{eff}}(e)$ is resistance between endpoints of e

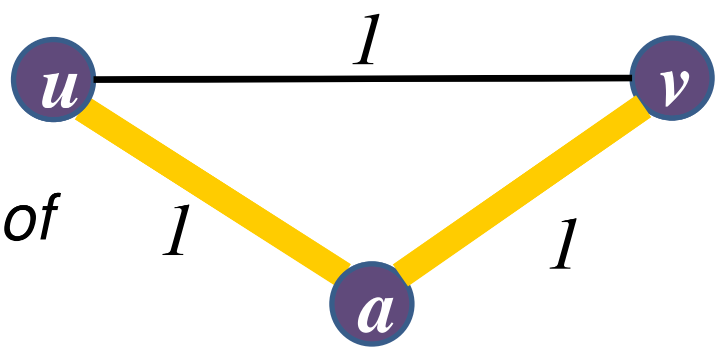


Effective Resistance

Identify each edge of G with a unit resistor

$R_{\text{eff}}(e)$ is resistance between endpoints of e

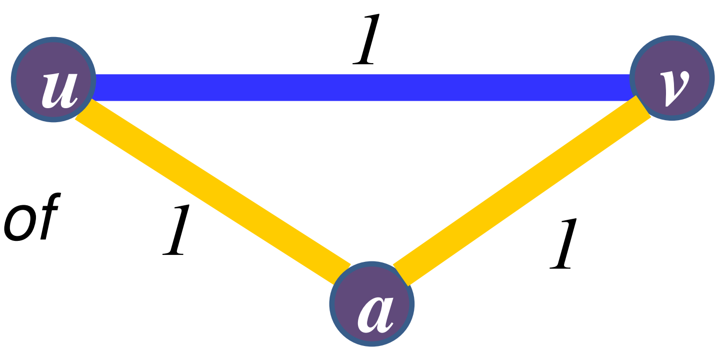
Resistance of path is 2



Effective Resistance

Identify each edge of G with a unit resistor

$R_{\text{eff}}(e)$ is resistance between endpoints of e



Resistance of path is 2

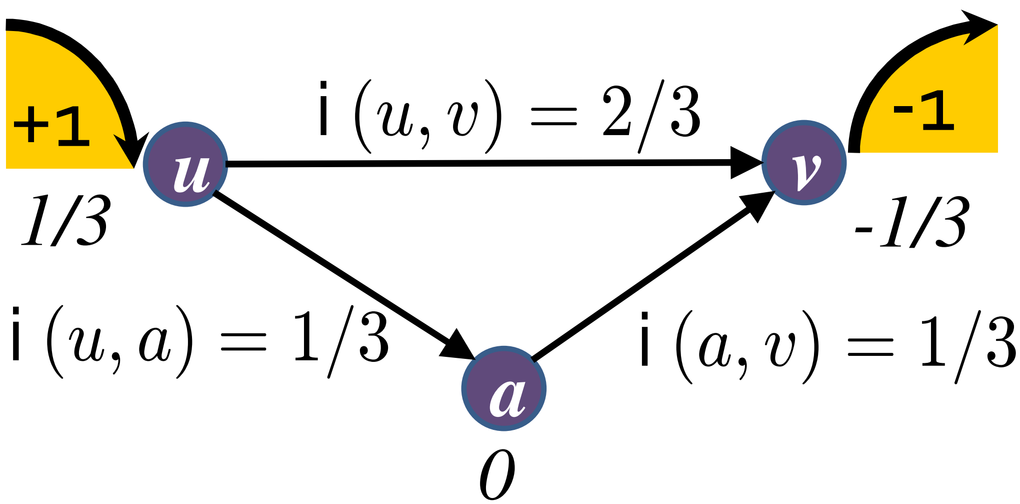
Resistance from u to v is

$$\frac{1}{1/2 + 1/1} = 2/3$$

Effective Resistance

Identify each edge of G with a unit resistor

$R_{\text{eff}}(e)$ is resistance between endpoints of e

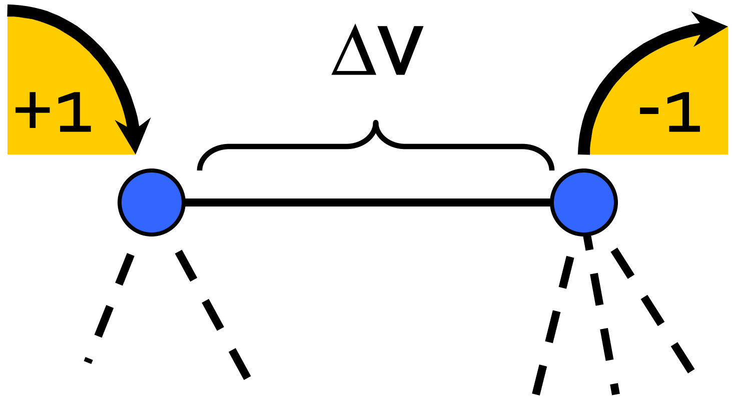


v $=$ i r

Effective Resistance

Identify each edge of G with a unit resistor

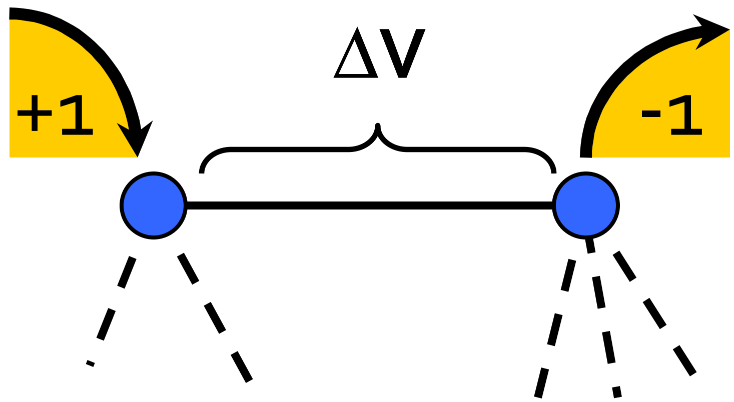
$R_{\text{eff}}(e)$ is resistance between endpoints of e



= potential difference between endpoints when flow one unit from one endpoint to other

Effective Resistance

$$R_{\text{eff}}(e) =$$



$$R_{\text{eff}}(e) = \mathbb{P}_{\text{spanning T}}[e \in T]$$

$$R_{\text{eff}}(uv) \propto \mathbb{E}_v T_u + \mathbb{E}_u T_v$$

[Chandra et al. STOC '89]

The Algorithm

Sample edges of G with probability

$$p_e \propto R_{\mathsf{eff}}(e)$$

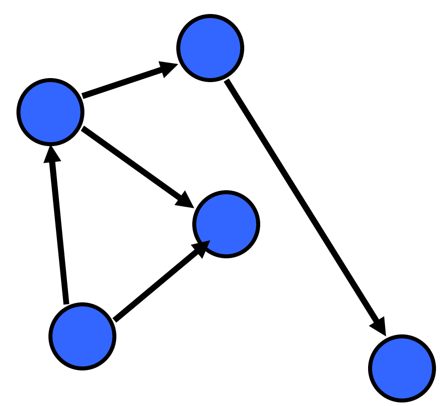
If chosen, include in H with weight $\frac{1}{p_e}$

Take $q=O(n\log n/\varepsilon^2)$ samples with replacement

Divide all weights by *q*.

An algebraic expression for R_{eff}

Orient G arbitrarily.



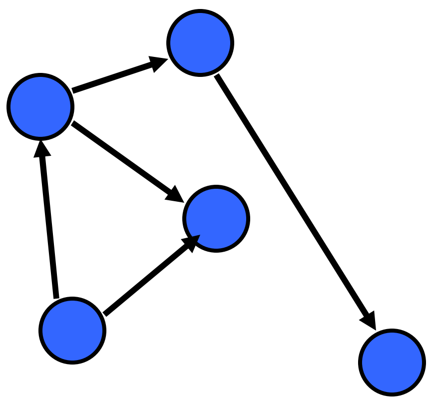
An algebraic expression for R_{eff}

Orient G arbitrarily.

Signed incidence matrix $B_{m \times n}$:

$$B(e,v) = \begin{cases} +1 & \text{if } v \text{ is head of } e \\ -1 & \text{if } v \text{ is tail of } e \\ 0 & \text{otherwise} \end{cases}$$

$$\text{i.e.,} B(uv, \cdot) = \chi_u - \chi_v.$$



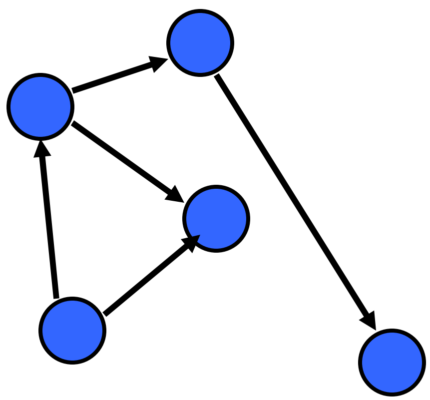
An algebraic expression for R_{eff}

Orient G arbitrarily.

Signed incidence matrix $B_{m \times n}$:

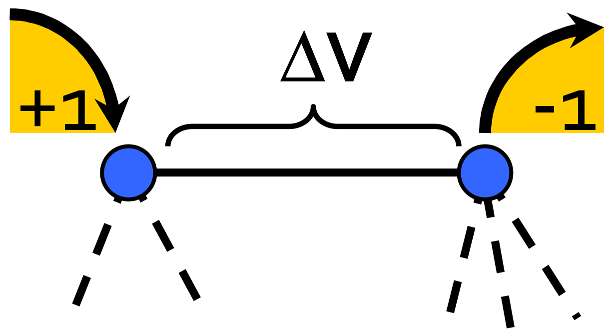
$$B(e,v) = \begin{cases} +1 & \text{if } v \text{ is head of } e \\ -1 & \text{if } v \text{ is tail of } e \\ 0 & \text{otherwise} \end{cases}$$

$$\text{i.e.,} B(uv, \cdot) = \chi_u - \chi_v.$$



Write Laplacian as $L = B^T B$

An algebraic expression for R_{eff}



$$R_{\text{eff}}(uv) = (\chi_u - \chi_v)^T L^{-1} (\chi_u - \chi_v)$$
$$= B(uv,\cdot) L^{-1} B(uv,\cdot)^T$$

An algebraic expression for R_{eff}

$$\text{Let } \Pi = BL^{-1}B^T.$$

Then

$$\begin{aligned} R_{\text{eff}}(e) &= B(e,\cdot)L^{-1}B(e,\cdot)^T \\ &= BL^{-1}B^T(e,e). \end{aligned}$$

An algebraic expression for R_{eff}

$$\text{Let } \Pi = BL^{-1}B^T.$$

Then

$$\begin{aligned} R_{\text{eff}}(e) &= B(e,\cdot)L^{-1}B(e,\cdot)^T \\ &= BL^{-1}B^T(e,e). \end{aligned}$$

$$\boxed{\Pi\Pi = \Pi}$$

An algebraic expression for R_{eff}

$$\text{Let } \Pi = BL^{-1}B^T.$$

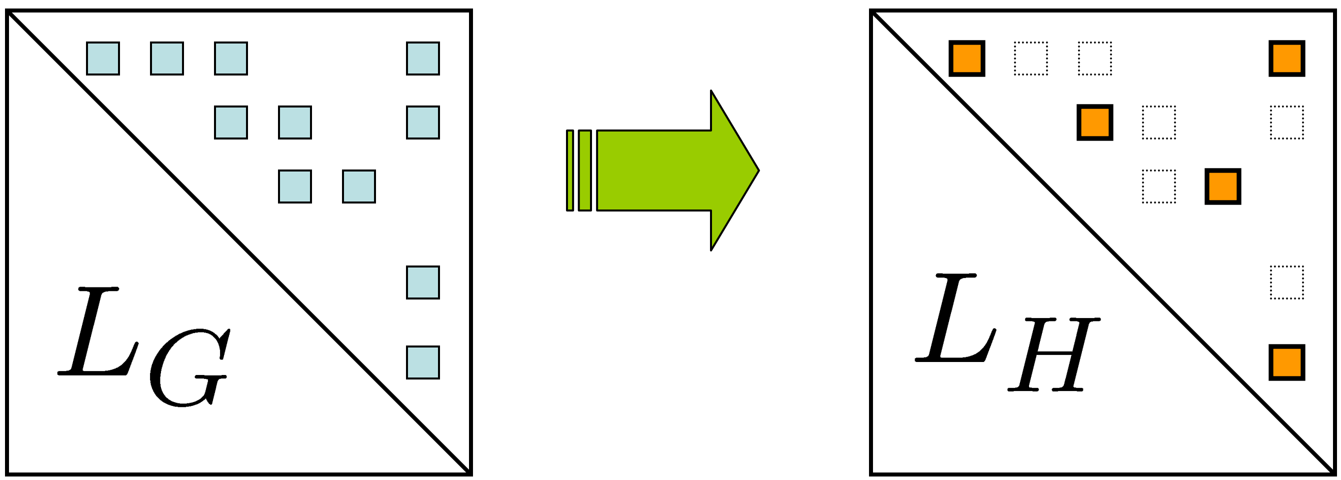
Then

$$R_{\text{eff}}(e) = B(e,\cdot)L^{-1}B(e,\cdot)^T$$
$$= BL^{-1}B^T(e,e).$$

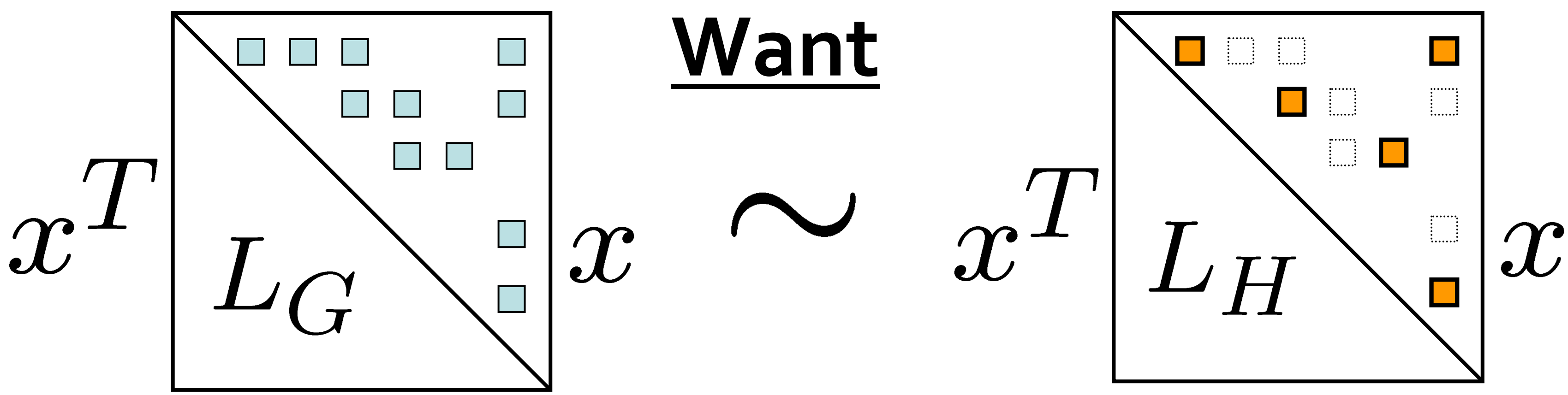
$$\boxed{\Pi\Pi = \Pi}$$

Reduce thm.
to statement
about Π

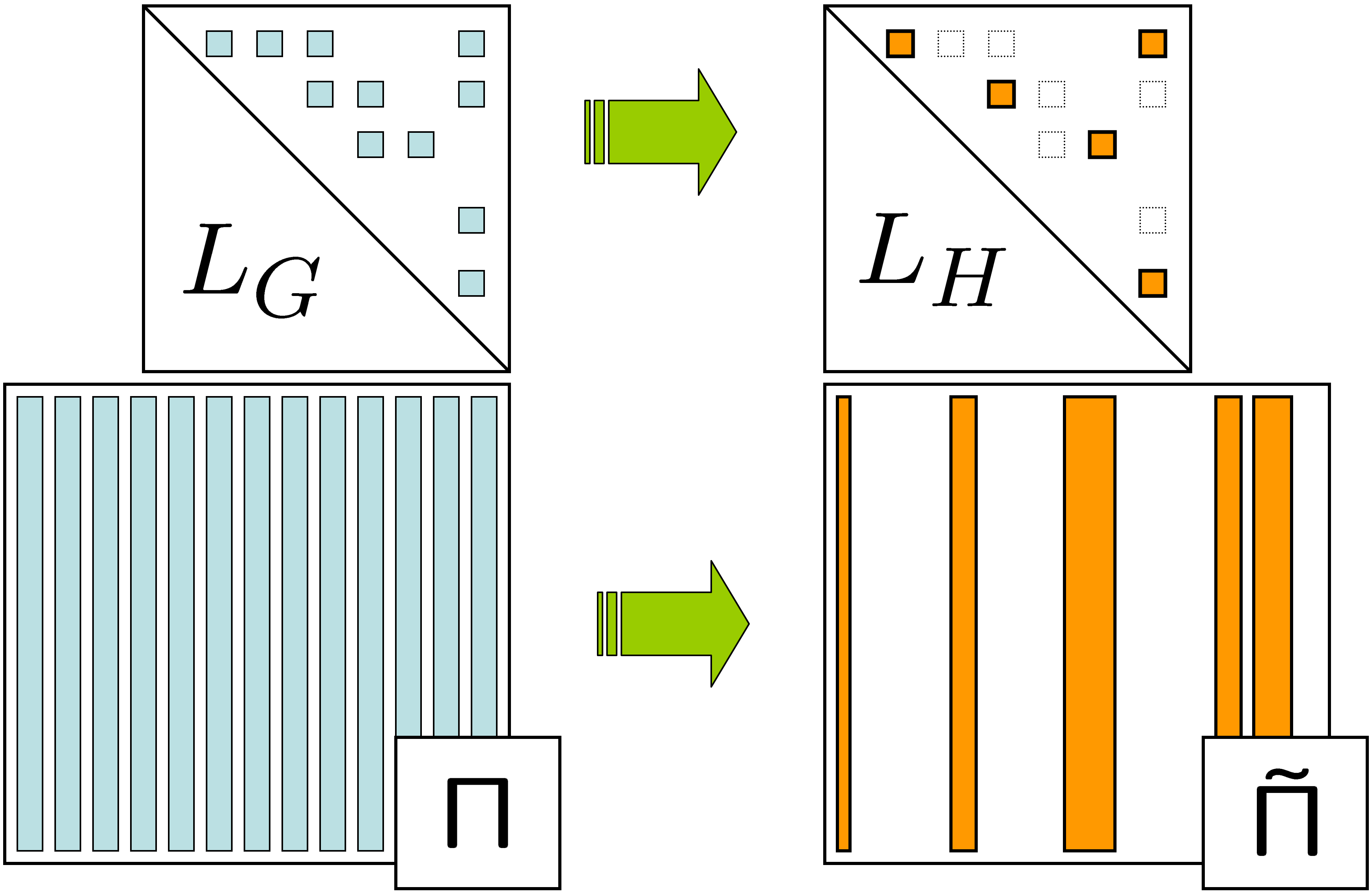
Goal



Want



Sampling in Π



Reduction to Π

Lemma.

$$1 - \epsilon \leq \frac{x^T L_H x}{x^T L_G x} \leq 1 + \epsilon \quad \forall x \in \mathbb{R}^n$$

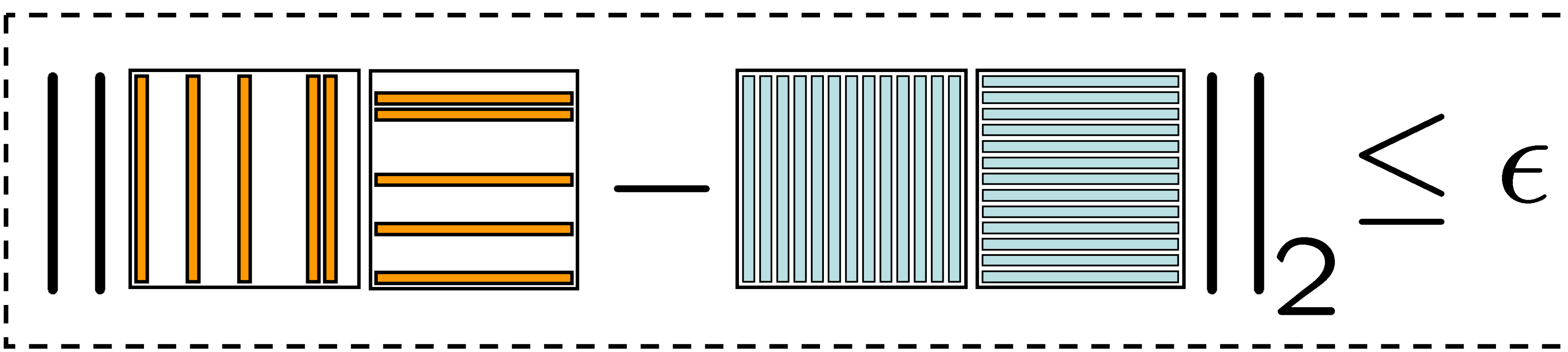
$$\iff \|\tilde{\Pi}\tilde{\Pi}^T - \Pi\Pi^T\|_2 \leq \epsilon$$

New Goal

Lemma.

$$1 - \epsilon \leq \frac{x^T L_H x}{x^T L_G x} \leq 1 + \epsilon \quad \forall x \in \mathbb{R}^n$$

$$\iff \|\tilde{\Pi}\tilde{\Pi}^T - \Pi\Pi^T\|_2 \leq \epsilon$$



The Algorithm

Sample edges of ***G*** with probability

$$p_e \propto R_{\text{eff}}(e)$$

If chosen, include in ***H*** with weight $\frac{1}{p_e}$

Take q=O(nlogn/ε²) samples with replacement

Divide all weights by *q*.

The Algorithm

Sample $\boxed{\text{columns of } \Pi}$ with probability

$$p_e \propto R_{\mathrm{eff}}(e)$$

If chosen, include in $\boxed{\tilde{\Pi}}$ with weight $\dfrac{1}{p_e}$

Take q=O(nlogn/ε²) samples with replacement

Divide all weights by *q*.

The Algorithm

Sample $\boxed{\text{columns of } \Pi}$ with probability

$$p_e \propto \boxed{\Pi(e,e)}$$

If chosen, include in $\boxed{\tilde{\Pi}}$ with weight $\frac{1}{p_e}$

Take q=O(nlogn/ε^2) samples with replacement

Divide all weights by *q*.

The Algorithm

Sample columns of Π with probability

$$p_e \propto \Pi(e,e) = \|\Pi(\cdot,e)\|^2$$

If chosen, include in $\tilde{\Pi}$ with weight $\frac{1}{p_e}$

Take q=O(nlogn/ε^2) samples with replacement

Divide all weights by *q*.

$$\Pi^T\Pi = \Pi$$

The Algorithm

Sample $\boxed{\text{columns of } \Pi}$ with probability

$$p_e \propto \Pi(e,e) = \boxed{\|\Pi(\cdot,e)\|^2}$$

If chosen, include in $\boxed{\tilde{\Pi}}$ with weight $\frac{1}{p_e}$

Take $q = O(n \log n/\epsilon^2)$ samples with replacement

Di

$$\Pi^T \Pi = \Pi$$

cf. low-rank approx.
[FKV04,RV07]

A Concentration Result

Lemma. (Rudelson '99)
If we sample $n \log n/\epsilon^2$ cols of Π with $p_e \propto \|\Pi(\cdot, e)\|^2$, then

$$\mathbb{E}\|\tilde{\Pi}\tilde{\Pi} - \Pi\Pi\|_2 \leq \epsilon.$$

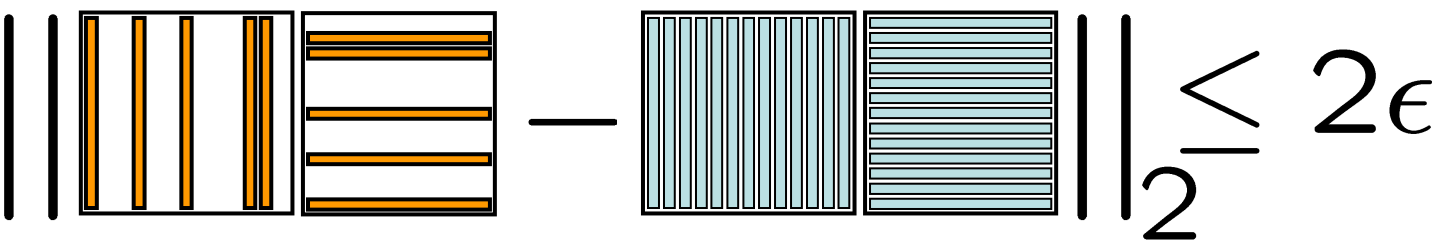
A Concentration Result

Lemma. (Rudelson '99)

If we sample $n \log n/\epsilon^2$ cols of Π with $p_e \propto \|\Pi(\cdot, e)\|^2$, then

$$\mathbb{E}\|\tilde{\Pi}\tilde{\Pi} - \Pi\Pi\|_2 \le \epsilon.$$

So with prob. ½:



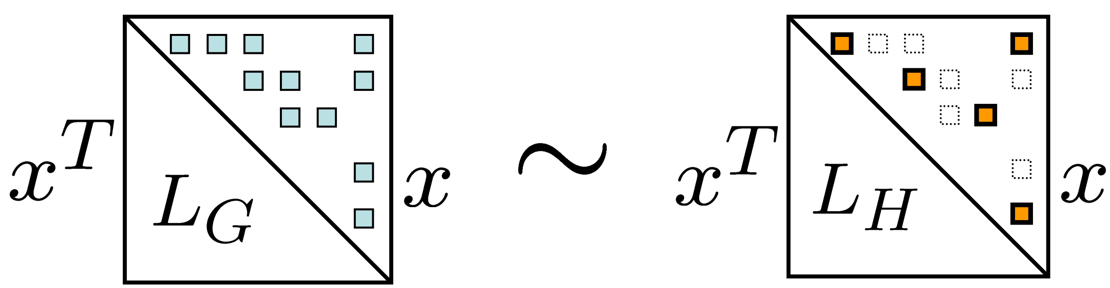
A Concentration Result

Lemma. (Rudelson '99)

If we sample $n \log n/\epsilon^2$ cols of Π with $p_e \propto \|\Pi(\cdot, e)\|^2$, then

$$\mathbb{E}\|\tilde{\Pi}\tilde{\Pi} - \Pi\Pi\|_2 \leq \epsilon.$$

So with prob. ½:



$\square$

Nearly Linear Time

The Algorithm

Sample edges of G with probability

$$p_e \propto R_{\mathsf{eff}}(e)$$

If chosen, include in H with weight $\frac{1}{p_e}$

Take $q=O(n\log n/\varepsilon^2)$ samples with replacement

Divide all weights by q.

The Algorithm

Sample edges of G with probability

$$p_e \propto R_{\text{eff}}(e)$$

If chosen, include in H with weight $\frac{1}{p_e}$

Take q=O(nlogn/ε²) samples with replacement

Divide all weights by *q*.

Nearly Linear Time

$$\boxed{R_{\text{eff}}(uv) = \|BL^{-1}(\chi_u - \chi_v)\|_2^2}$$

So care about distances between cols. of $\boldsymbol{BL^{-1}}$

Nearly Linear Time

$$\boxed{R_{\text{eff}}(uv) = \|BL^{-1}(\chi_u - \chi_v)\|_2^2}$$

So care about distances between cols. of $\boldsymbol{BL^{-1}}$

Johnson-Lindenstrauss! Take random $\boldsymbol{Q_{\log n \times m}}$

Set $\boldsymbol{Z=QBL^{-1}}$

$$\underset{(\log n \times m)}{Q} \quad \underset{(m \times n)}{BL^{-1}} \quad = \quad \underset{(\log n \times n)}{Z}$$

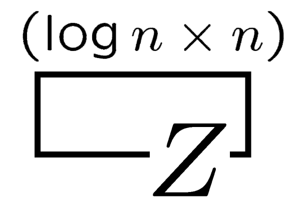
Nearly Linear Time

$(\log n \times n)$

Z

$$\boxed{R_{\text{eff}}(uv) \sim \|Z(\chi_u - \chi_v)\|^2}$$

Nearly Linear Time



Find **rows** of $\boldsymbol{Z_{\log n \times n}}$ by

$$\boldsymbol{Z = QBL^{-1}}$$

$$\boxed{R_{\text{eff}}(uv) \sim \|Z(\chi_u - \chi_v)\|^2}$$

$$\boldsymbol{ZL = QB}$$

$$\boldsymbol{z_i L = (QB)_i}$$

Nearly Linear Time

$(\log n \times n)$

Z

Find **rows** of $\boldsymbol{Z_{\log n \times n}}$ by

$\boldsymbol{Z = QBL^{-1}}$

$\boldsymbol{ZL = QB}$

$\boldsymbol{z_i L = (QB)_i}$

$$R_{\text{eff}}(uv) \sim \|Z(\chi_u - \chi_v)\|^2$$

Solve $\boldsymbol{O(\log n)}$ linear systems in $\boldsymbol{L}$ using Spielman-Teng '04 solver

which uses combinatorial $O(n\log^c n)$ sparsifier.

Can show approximate $\boldsymbol{R_{eff}}$ suffice.

$\square$

Main Conjecture

Sparsifiers with O(n) edges.