Spectral Sparsification of Graphs

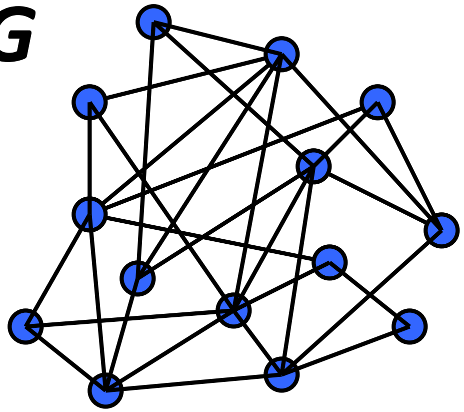
Nikhil Srivastava
Mathematics Department

Goals

1. Explain spectral sparsification, a form of lossy compression of graphs.

2. Introduce **physical** and **geometric** ways of thinking about graphs.

Graphs



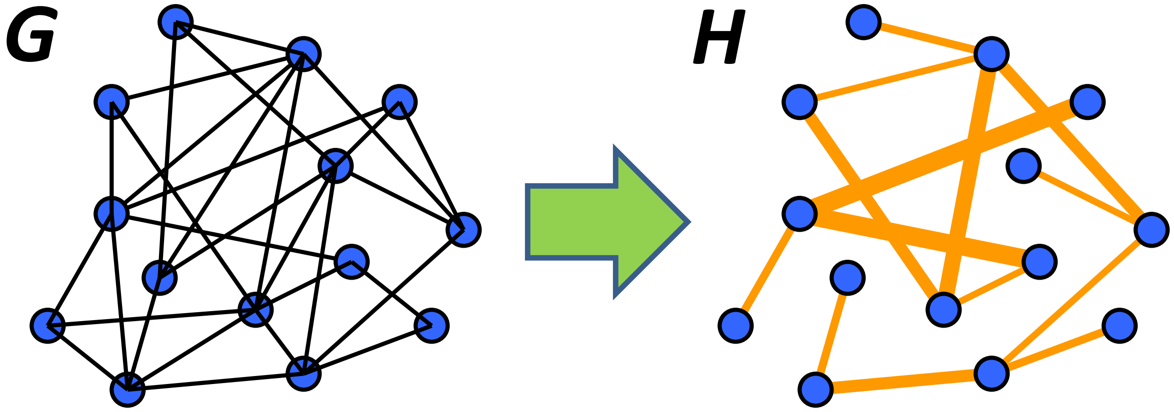
$\boldsymbol{G}=(V,E,w)$ *undirected*

$|V| = n$

$w: E \rightarrow \mathbf{R}_+$

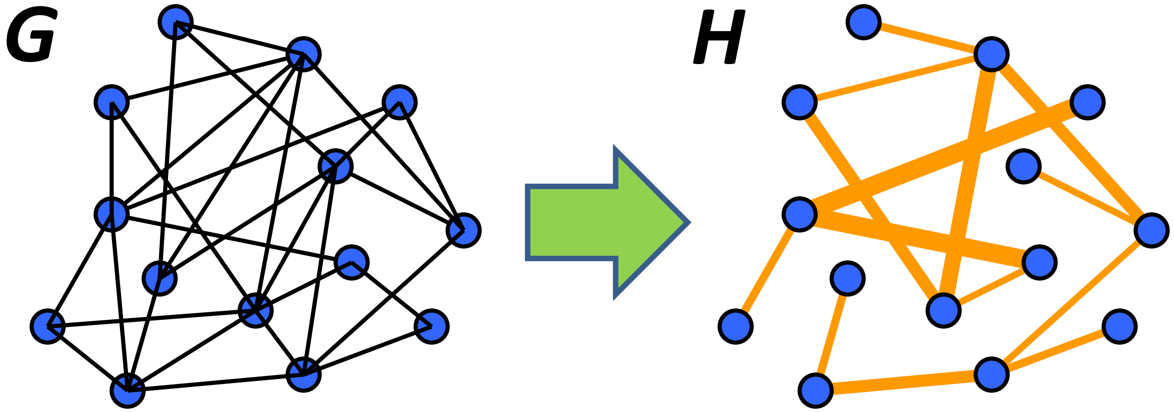
Sparsification

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



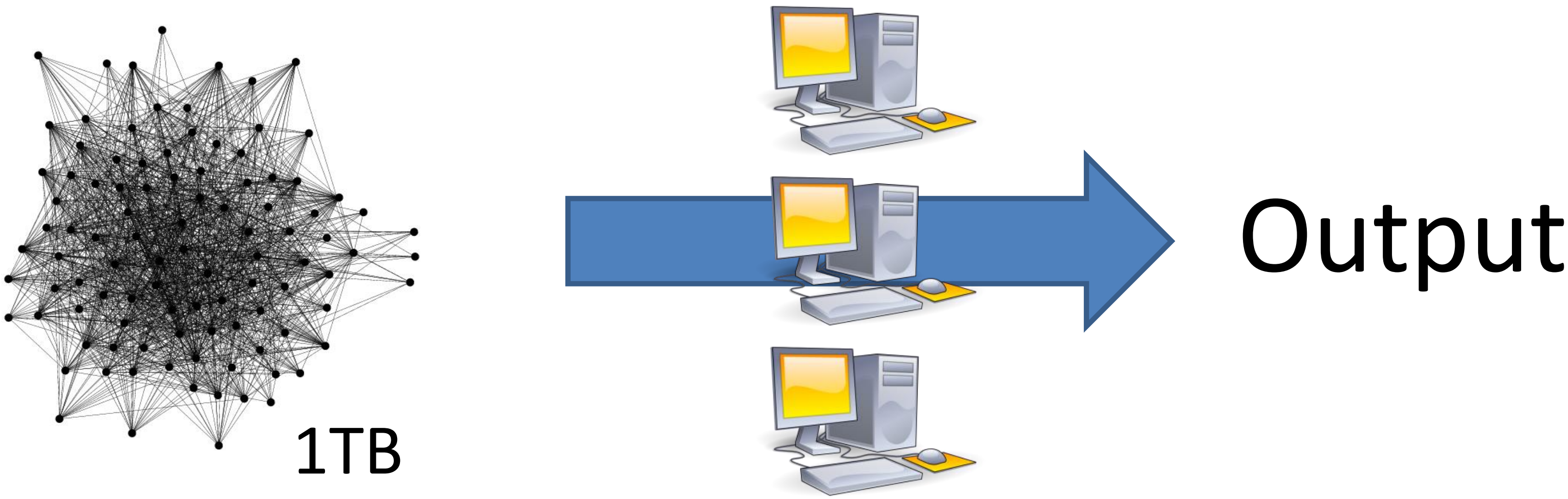
Sparsification

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

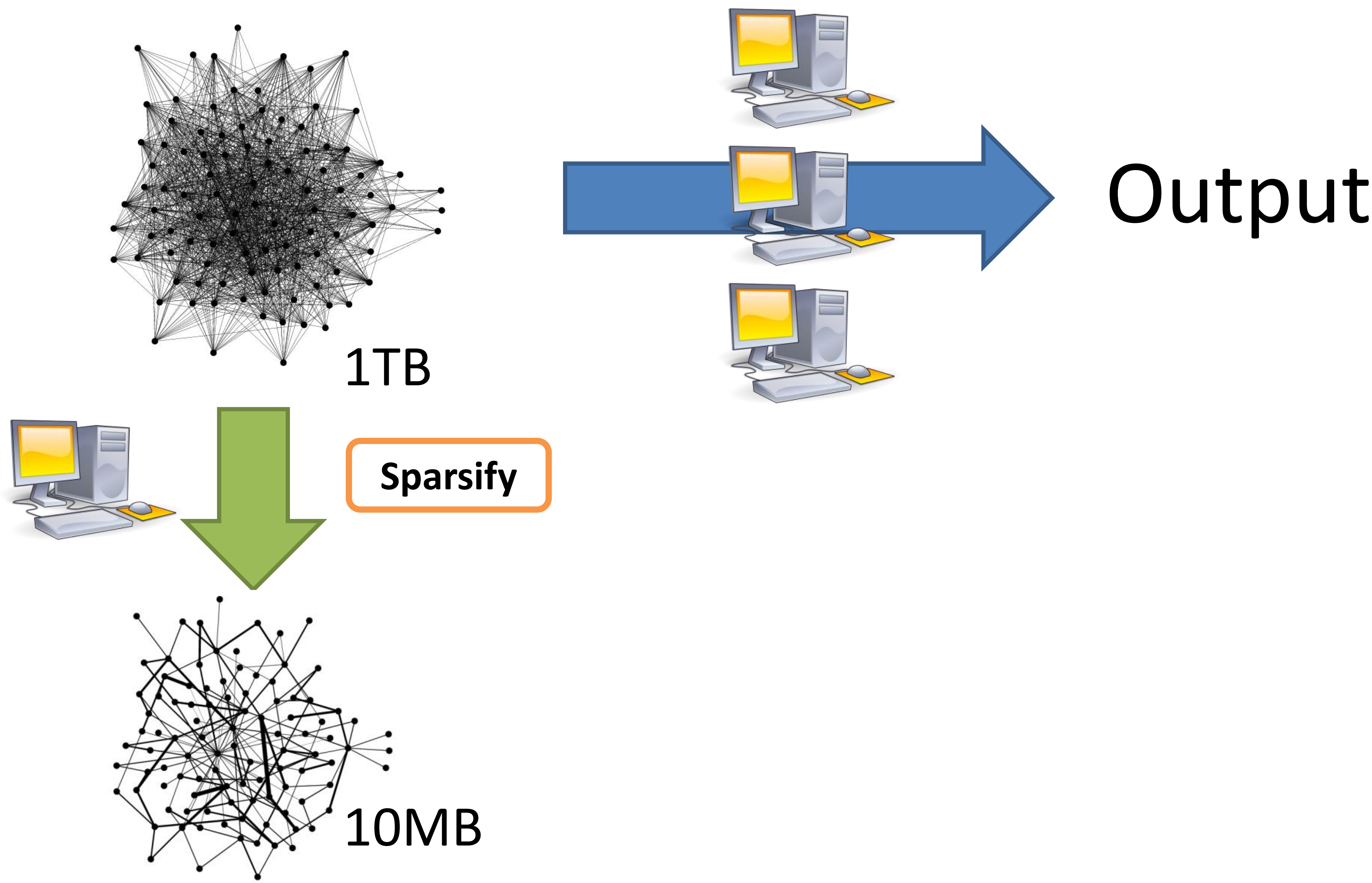


- ***H*** is faster to compute with than ***G***
- Nontrivial statement about ***G***

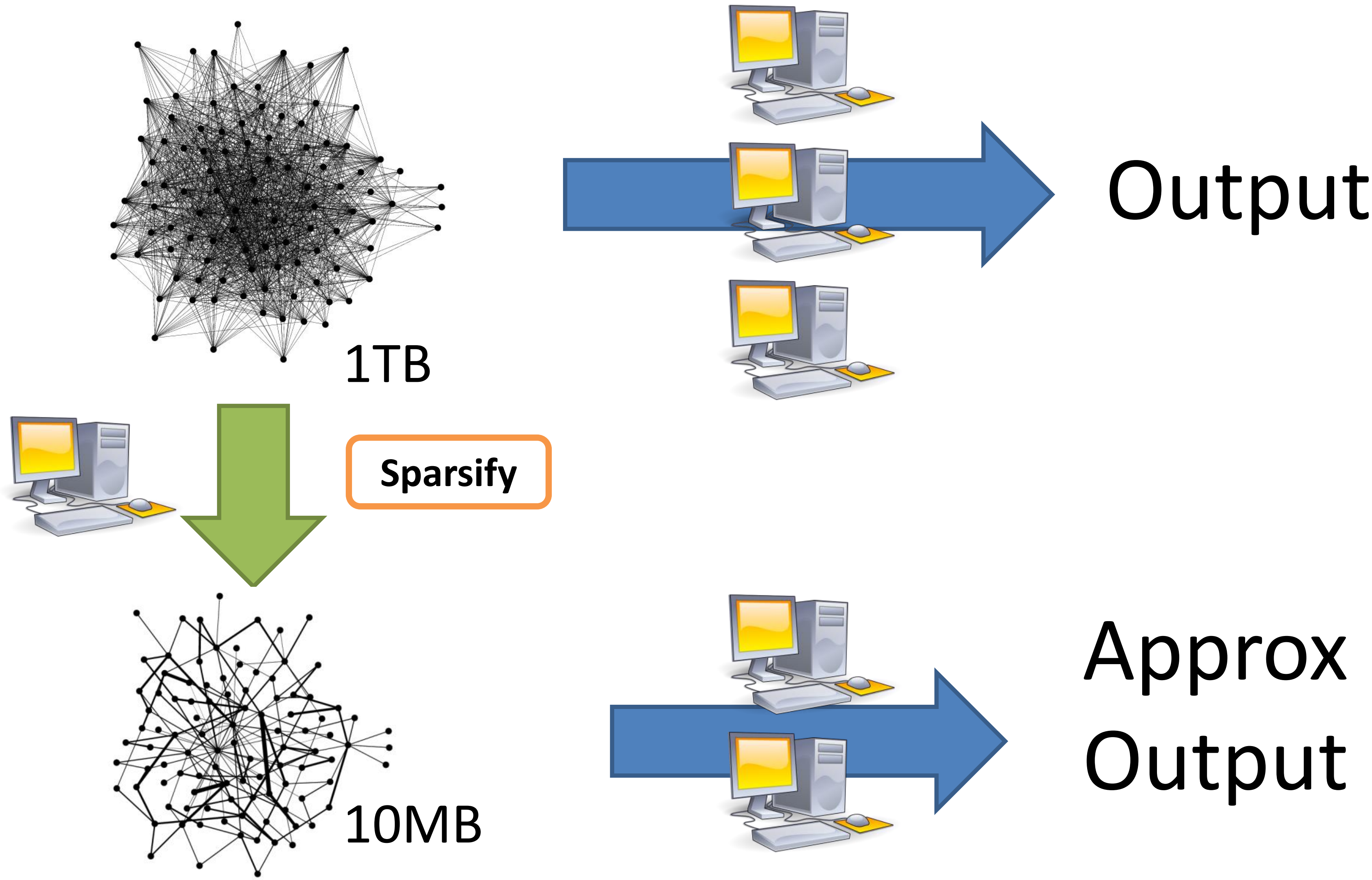
Sample Application



Sample Application



Sample Application



Some properties of interest

Sizes of cuts	“bottlenecks”
Clusters	“communities”
Distances	(shortest paths)
Random walks	
Single / multicommodity flows	
Electrical flows + other physical processes	
Coloring	
Hamiltonian / Eulerian cycle	
Subgraph counts	e.g. triangles

Some properties of interest

Sizes of cuts **"bottlenecks"**

Clusters **"communities"**

Distances

Random walks

Single / multicommodity flows

Electrical flows + other physical processes

Coloring

Hamiltonian / Eulerian cycle

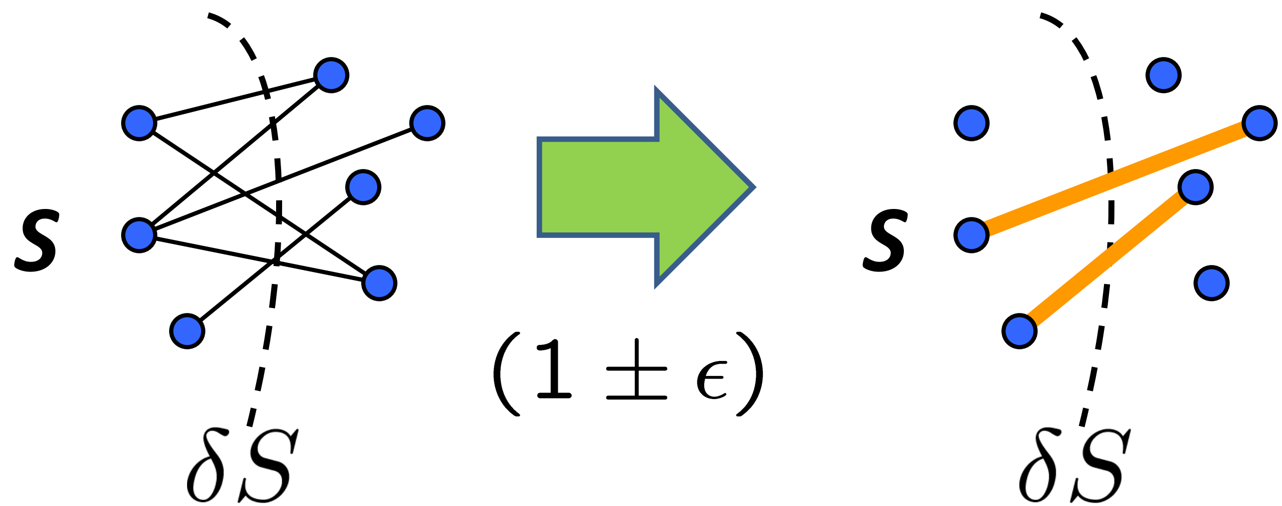
Subgraph counts e.g. triangles

Cut Approximation [Benczur-Karger'96]

H approximates ***G*** if

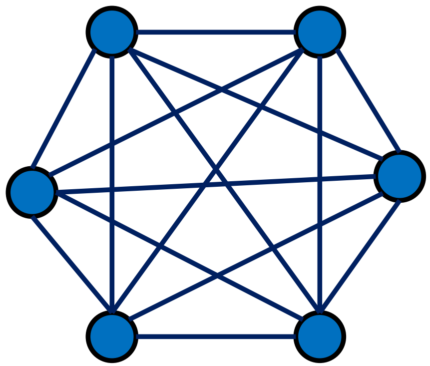
for every subset $S \subset V$

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



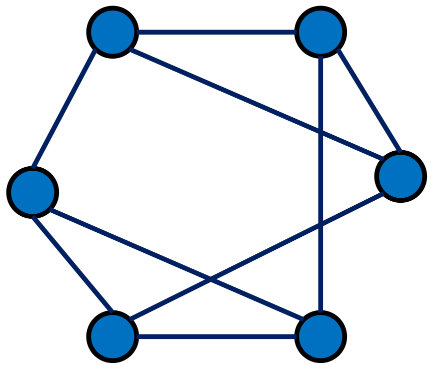
Example: The Complete Graph

$G=K_n$



$|E_G| = O(n^2)$

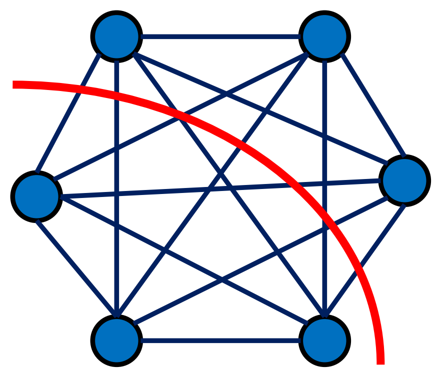
H = random d-regular



$|E_H| = O(dn)$

Example: The Complete Graph

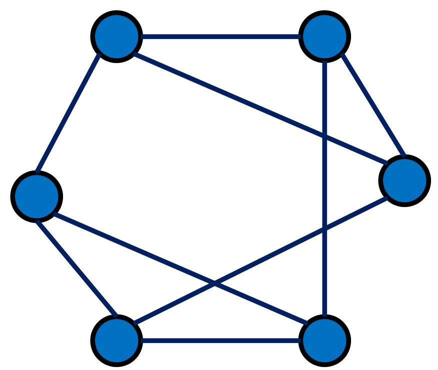
G=K$_n$



$|E_G| = O(n^2)$

$$wt_G(\delta S) = |S| \cdot |\overline{S}|$$

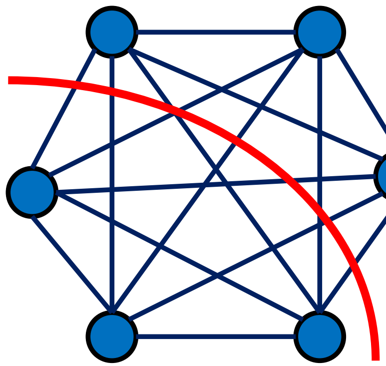
H = random d-regular



$|E_H| = O(dn)$

Example: The Complete Graph

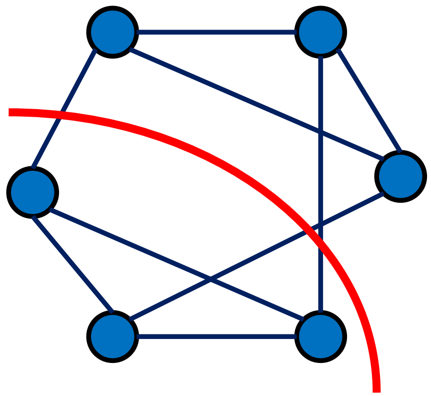
G=K$_n$



$|E_G| = O(n^2)$

$$wt_G(\delta S) = |S| \cdot |\overline{S}|$$

H = random d-regular

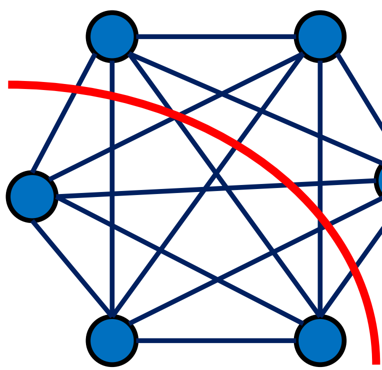


$|E_H| = O(dn)$

$$\mathbb{E} wt_H(\delta S) = (d/n)|S| \cdot |\overline{S}|$$

Example: The Complete Graph

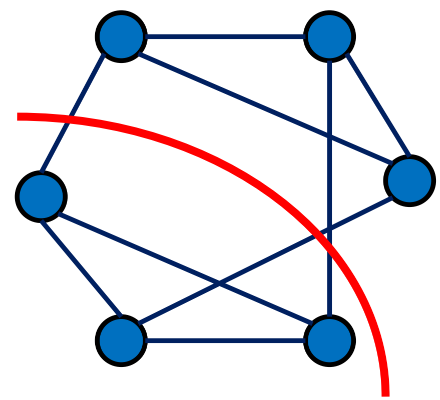
G=K$_n$



$|E_G| = O(n^2)$

$$wt_G(\delta S) = |S| \cdot |\overline{S}|$$

H = random d-regular



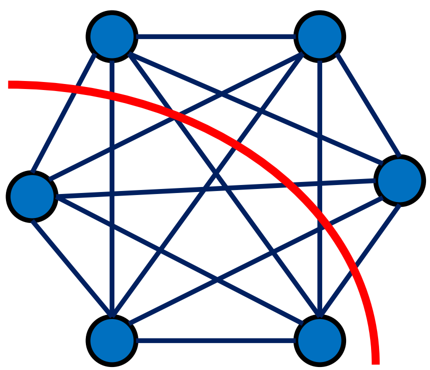
$|E_H| = O(dn)$

$$wt_H(\delta S) \simeq (d/n)|S| \cdot |\overline{S}|$$

with high probability

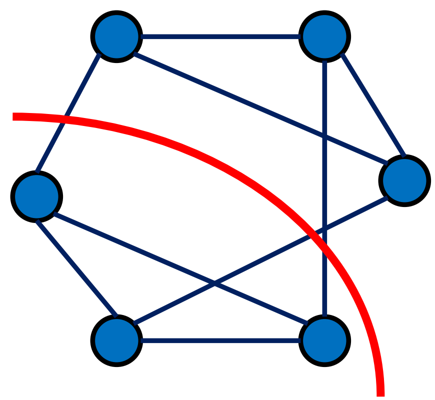
Example: The Complete Graph

G=K$_n$



$|E_G| = O(n^2)$

H = random d-regular



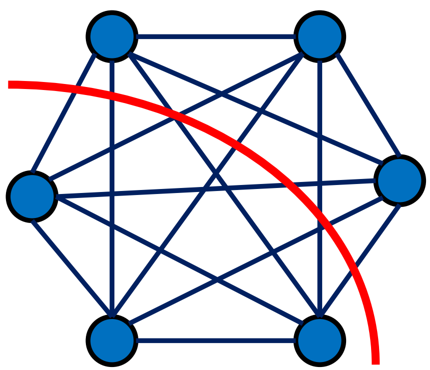
$|E_H| = O(dn)$

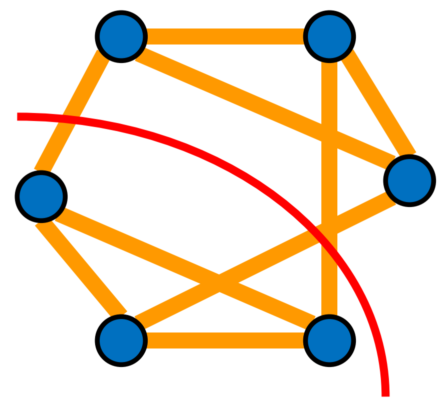
$$\forall S \subset V, \qquad \frac{wt_G(\delta S)}{wt_H(\delta S)} \simeq (n/d)$$

whp

Example: The Complete Graph

G=K$_n$ **H** = random d-regular x **(n/d)**





$|E_G| = O(n^2)$ $|E_H| = O(dn)$

$$\forall S \subset V, \quad \frac{wt_G(\delta S)}{wt_{(n/d)H}(\delta S)} \simeq 1$$

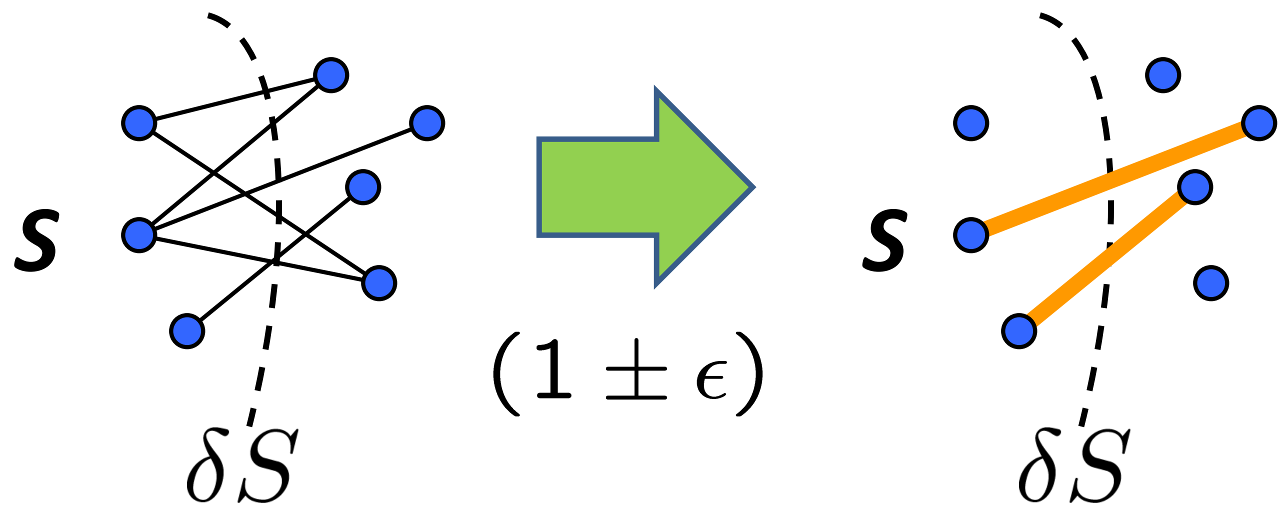
whp

Cut Approximation [Benczur-Karger'96]

H approximates ***G*** if

for every subset $S \subset V$

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



[Benczur-Karger'96]*:* For every ***G*** can quickly find ***H*** with **O(nlogn/ε^2)** edges.

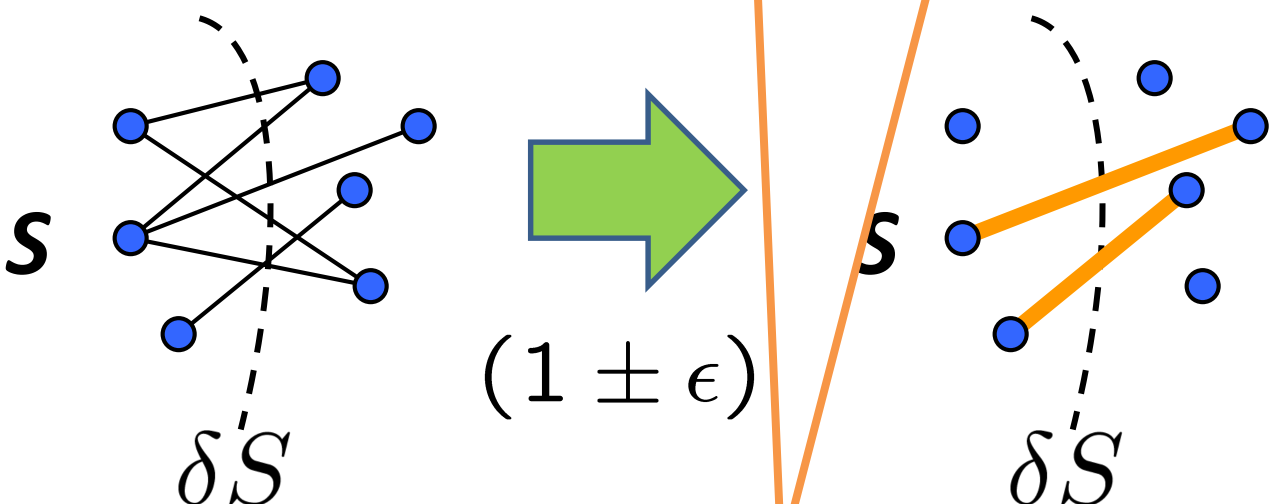
Cut Approximation [Benczur-Karger'96]

H approximates G

for every subs

sum of weights of edges leavi preserved

G and H have the same 'bottlenecks'.



[Benczur-Karger'96]*:* For every ***G*** can quickly find ***H*** with **O(nlogn/ε²)** edges.

Cut Approximation [Benczur-Karger'96]

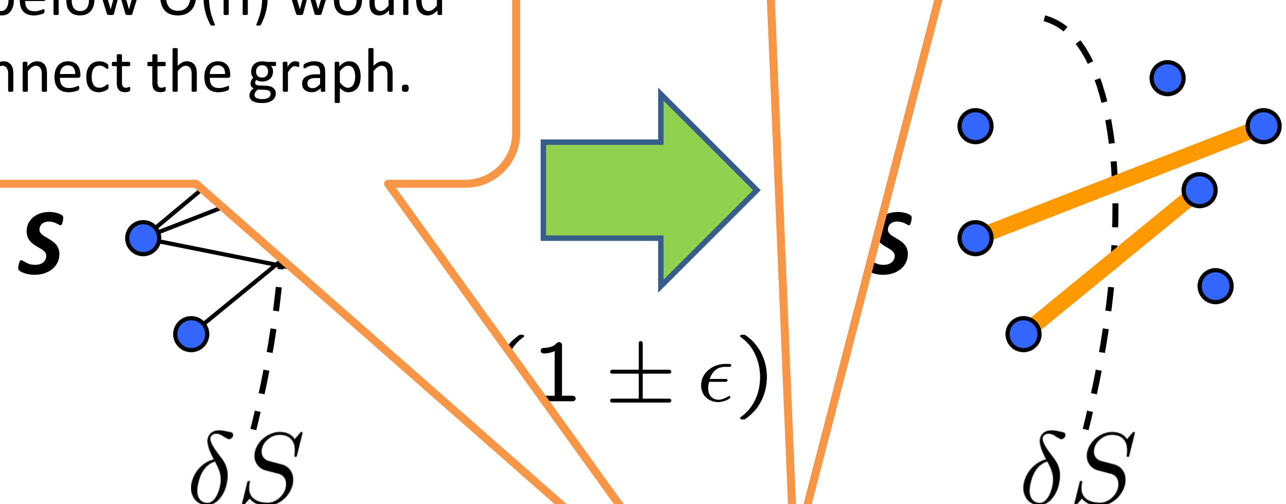
H approximates (

for every subs

ges leavi preserved

G and H have the same 'bottlenecks'.

Going below O(n) would disconnect the graph.

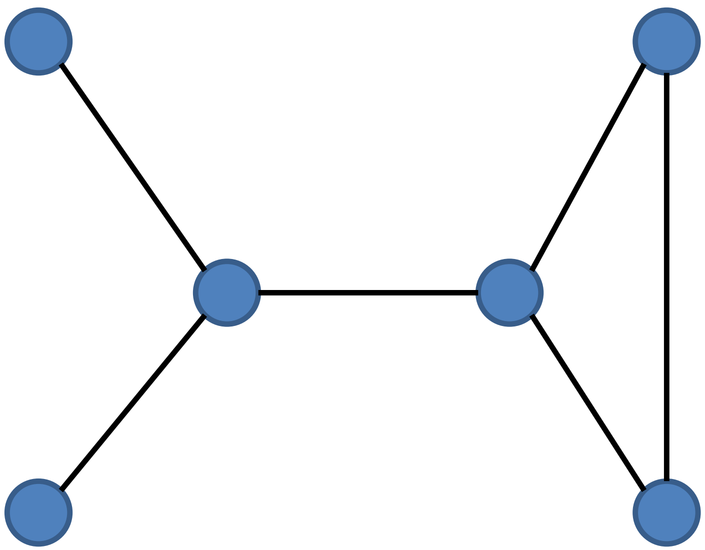


[Benczur-Karger'96]*:* For every ***G*** can quickly find ***H*** with **O(nlogn/ε²)** edges.

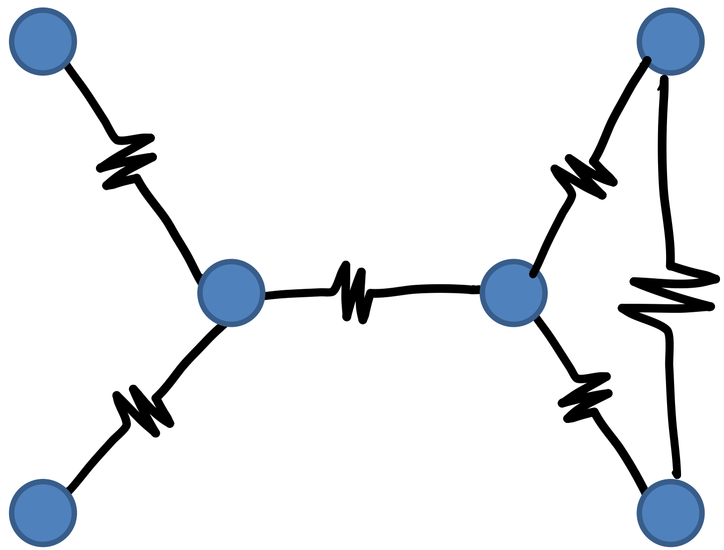
Physical Approximation [Spielman-Teng'04]

(i.e., spectral approximation)

Resistor Network Metaphor

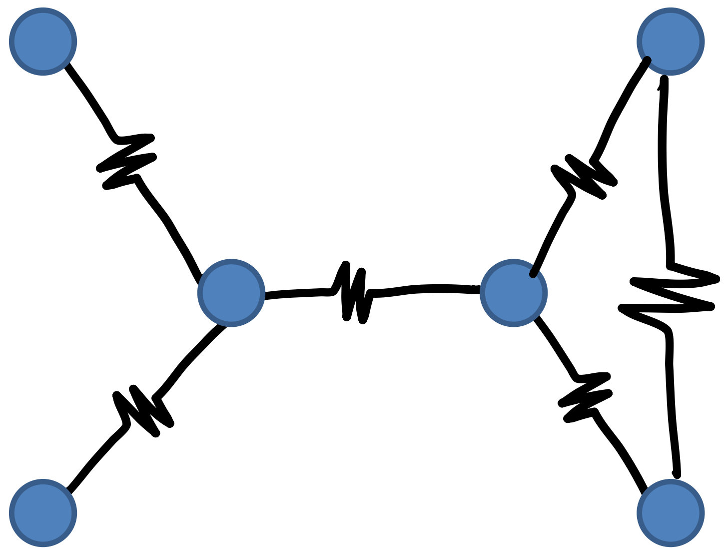


Resistor Network Metaphor



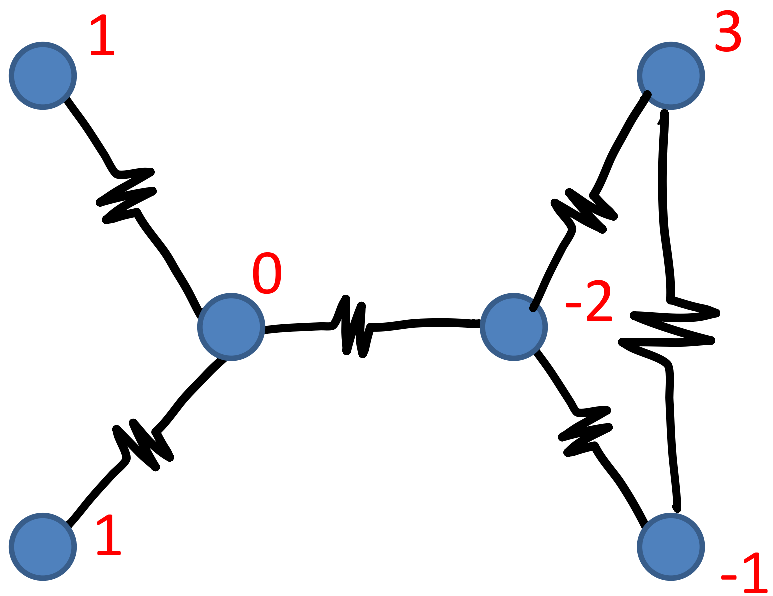
edge = 1Ω resistor

Resistor Network Metaphor



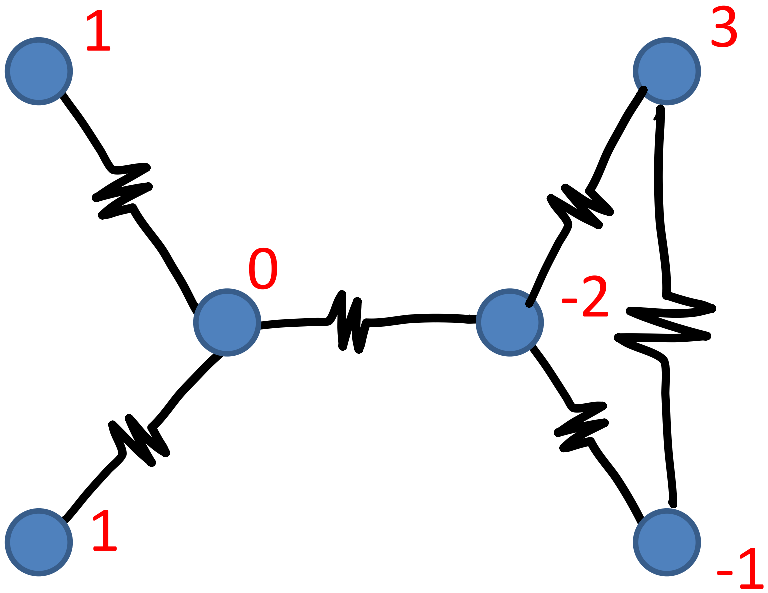
potentials $x: V \to \mathbb{R}$

Resistor Network Metaphor



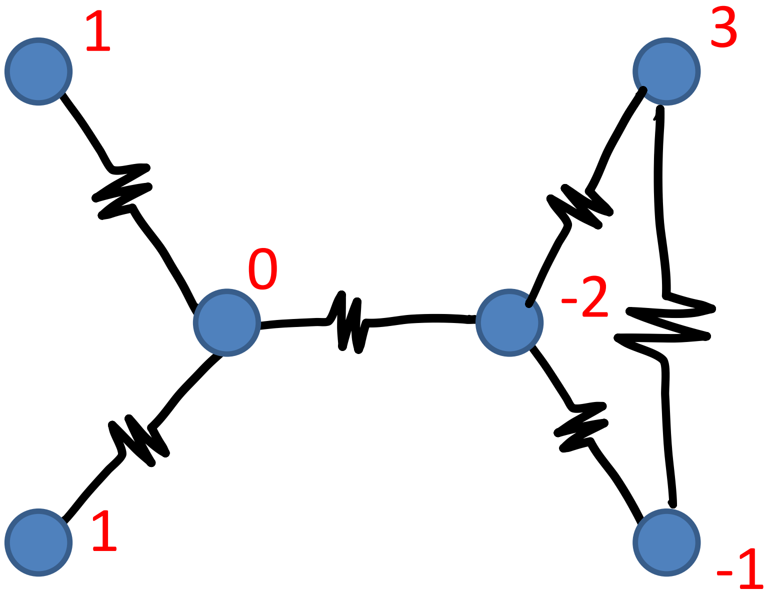
potentials $x: V \to \mathbb{R}$

Resistor Network Metaphor



$$\text{energy } \mathcal{E}_G(x) = \sum_{ij \in E} \left(x_i - x_j\right)^2$$

Resistor Network Metaphor



$$\text{energy } \mathcal{E}_G(x) = \sum_{ij \in E} \left(x_i - x_j\right)^2$$

$$= 1^2 + 1^2 + 2^2 + 5^2 + 1^2 + 2^2 = 36$$

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if for all potentials $x: V \to \mathbb{R}$:

$$\boxed{\mathcal{E}_H(x) \leq \mathcal{E}_G(x) \leq \kappa \cdot \mathcal{E}_H(x)}$$

"Electrically Equivalent"

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if for all potentials $x: V \to \mathbb{R}$:

$$\sum_{ij \in F} u_{ij}\left(x_i - x_j\right)^2 \leq \sum_{ij \in E} w_{ij}\left(x_i - x_j\right)^2 \leq \kappa \cdot \sum_{ij \in F} u_{ij}\left(x_i - x_j\right)^2$$

"Electrically Equivalent"

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if for all potentials $x: V \to \mathbb{R}$:

$$\sum_{ij \in F} u_{ij}(x_i - x_j)^2 \leq \sum_{ij \in E} w_{ij} (x_i - x_j)^2 \leq \kappa \cdot \sum_{ij \in F} u_{ij}(x_i - x_j)^2$$

Laplacian matrix $x^T L_G x$ $x^T L_H x$

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if for all potentials $x: V \to \mathbb{R}$:

$$x^T L_H x \leq x^T L_G x \leq \kappa \cdot x^T L_H x$$

where $$L_G = \sum_{ij} w_{ij} (\delta_i - \delta_j)(\delta_i - \delta_j)^T$$

is the **Laplacian** matrix of ***G***.

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if for all potentials $x: V \to \mathbb{R}$:

$$x^T L_H x \le x^T L_G x \le \kappa \cdot x^T L_H x$$

where $L_G = \sum_{ij} w_{ij} (\delta_i - \delta_j)(\delta_i - \delta_j)^T$

is the **Laplacian** matrix of ***G***.

$$\begin{array}{c|cc} & & \\ i & \multicolumn{2}{c}{\multirow{2}{*}{}} \\ \end{array}$$
$$\begin{array}{cc} \begin{matrix} i \\ j \end{matrix} & \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \\ & \begin{matrix} i & \quad j \end{matrix} \end{array}$$

Properties of the Laplacian

$$L_G = \sum_{ij \in E} w_{ij} \left(\delta_i - \delta_j\right)\left(\delta_i - \delta_j\right)^T = \sum_{ij \in E} w_{ij} L_{ij}$$

$x^T L_G x \geq 0$ so **positive semidefinite** $L_G \succcurlyeq 0$.

$A \succcurlyeq B$ means $x^T A x \geq x^T B x$

Properties of the Laplacian

$$L_G = \sum_{ij \in E} w_{ij} \left(\delta_i - \delta_j\right)\left(\delta_i - \delta_j\right)^T = \sum_{ij \in E} w_{ij} L_{ij}$$

$x^T L_G x \geq 0$ so **positive semidefinite** $L_G \succcurlyeq 0$.

nullspace = $\mathrm{span}\{(1,1,\ldots,1)\}$ for connected G.

$\sum_{ij \in E} w_{ij} \left(x_i - x_j\right)^2 = 0$ iff $x_i = x_j$ for every $ij \in E$

Properties of the Laplacian

$$L_G = \sum_{ij \in E} w_{ij} \left(\delta_i - \delta_j\right)\left(\delta_i - \delta_j\right)^T = \sum_{ij \in E} w_{ij} L_{ij}$$

$x^T L_G x \geq 0$ so **positive semidefinite** $L_G \succcurlyeq 0.$

nullspace = $\mathrm{span}\{(1,1,\ldots,1)\}$ for connected G.

Will talk about **inverse** $L_G^{-1} \succcurlyeq 0$ orthogonal to nullspace.

Properties of the Laplacian

$$L_G = \sum_{ij \in E} w_{ij} \left(\delta_i - \delta_j\right)\left(\delta_i - \delta_j\right)^T = \sum_{ij \in E} w_{ij} L_{ij}$$

$x^T L_G x \geq 0$ so **positive semidefinite** $L_G \succcurlyeq 0$.

nullspace = $\mathrm{span}\{(1,1,\ldots,1)\}$ for connected G.

Will talk about **inverse** $L_G^{-1} \succcurlyeq 0$ orthogonal to nullspace.

Can talk about **square root** $L_G^{-1/2}$ because $L_G^{-1} \succcurlyeq 0$.

Physical Approximation [ST'04]

Definition. $H = (V, F, u)$ is a κ −approximation of $G = (V, E, w)$ if:

$$L_H \preccurlyeq L_G \preccurlyeq \kappa \cdot L_H$$

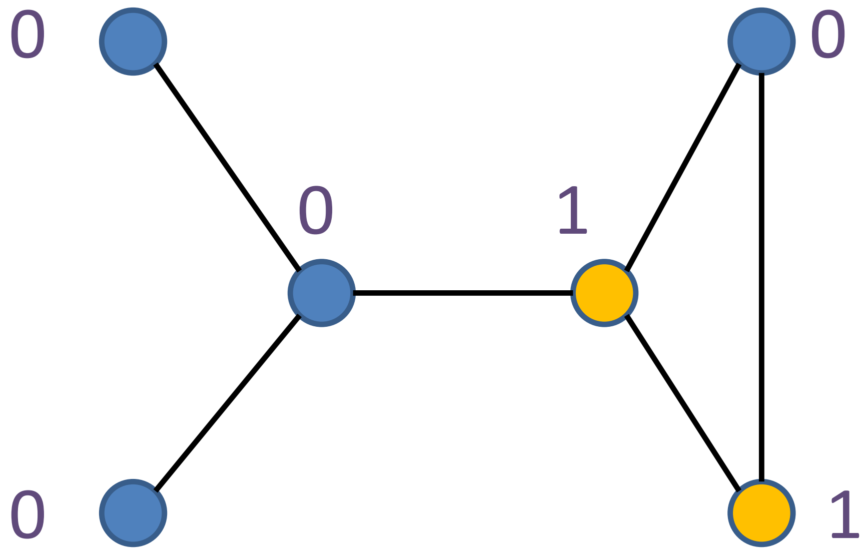
where $$L_G = \sum_{ij} w_{ij} (\delta_i - \delta_j)(\delta_i - \delta_j)^T$$

is the **Laplacian** matrix of ***G***.

Why?

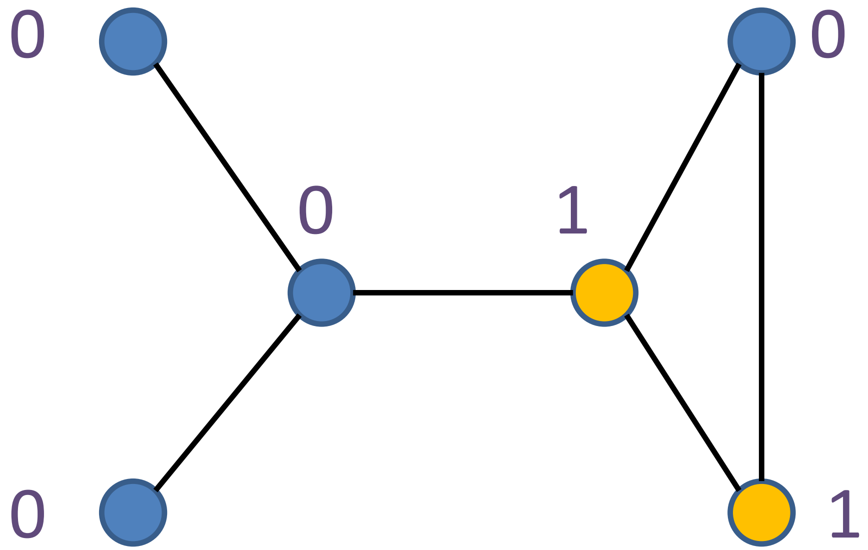
0. Energy Encodes Cuts

$$x: V \rightarrow \{0,1\}$$



0. Energy Encodes Cuts

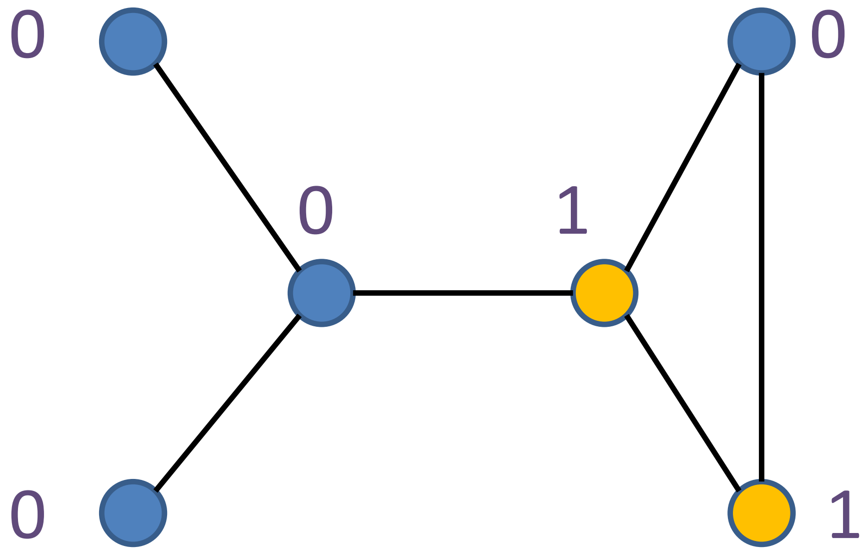
$$x : V \to \{0,1\}$$



$$\mathcal{E}_G(x) = 1^2 + 1^2 + 1^2 = 3$$

0. Energy Encodes Cuts

$$x: V \to \{0,1\}$$

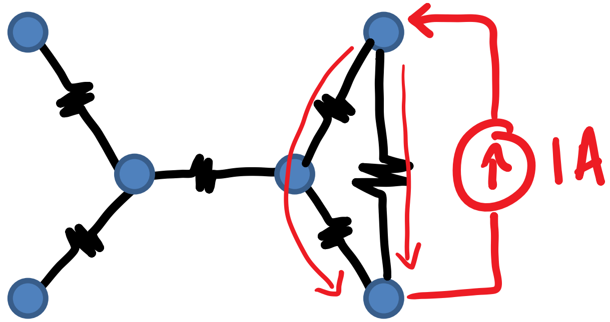


Physical approx. implies cut approx.

1. Energy controls physical processes

Electrical Flow:

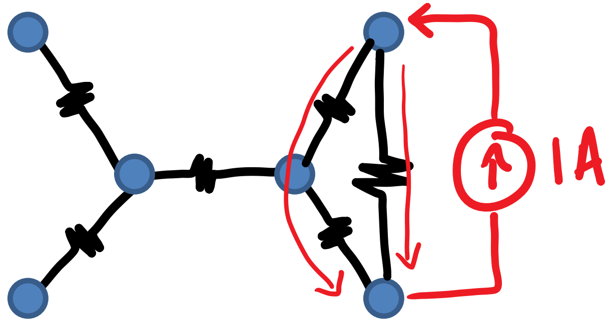
minimizes energy



1. Energy controls physical processes

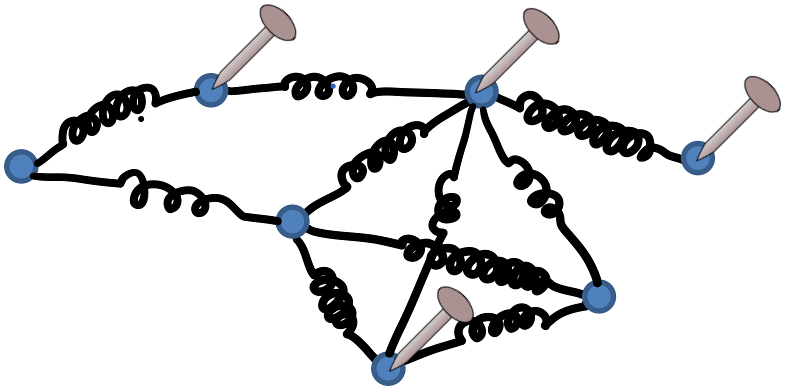
Electrical Flow:

minimizes energy



Spring Network:

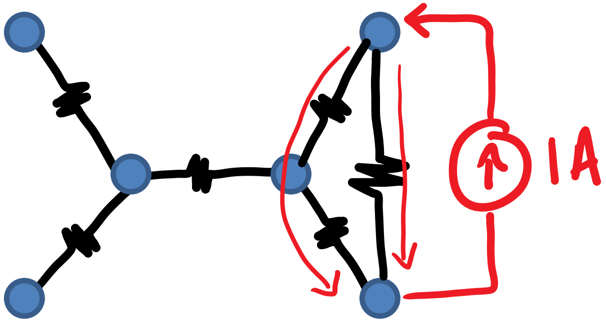
settles at min. energy



1. Energy controls physical processes

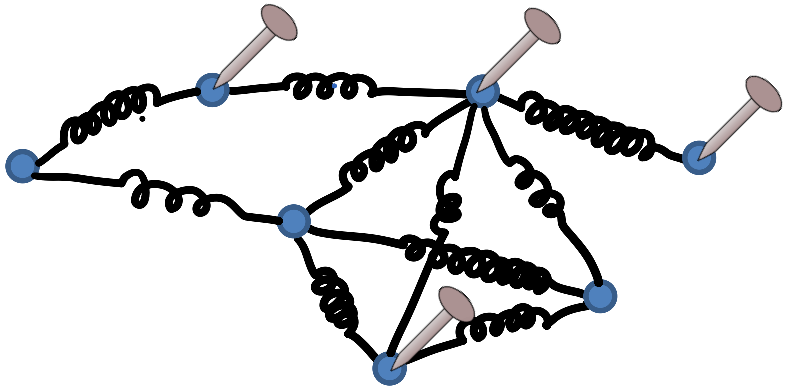
Electrical Flow:

minimizes energy

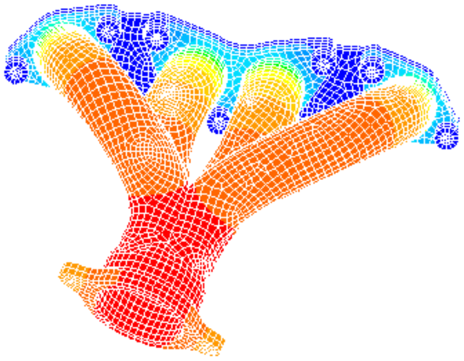


Spring Network:

settles at min. energy



Heat Flow:



1. Energy controls physical processes

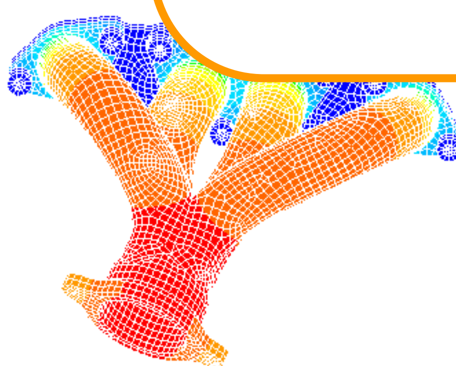
Electrical Flow:

minimizes ene

Spring Network:

settles at min

Heat Flow:



Solving any of these reduces to solving a Laplacian linear system

$$Lx = b$$

1. Solving $Lx = b$ fast [ST'04]

$\boldsymbol{x^T L_G x \sim x^T L_H x}$: can solve systems

in $\mathbf{L_G}$ by solving systems in $\mathbf{L_H}$.

$Lx = b$

Electrical Flow

Heat Flow

Spring Network

2. Spectral Graph Theory

Courant-Fischer Thm: $\boldsymbol{x^T L_G x}$ determines $\lambda_i(L_G)$

$$\boxed{\lambda_{max}(L) = \max \frac{x^T L x}{x^T x} \qquad \lambda_{min}(L) = \min \frac{x^T L x}{x^T x}}$$

Thus for physical approx. **H** of **G**:

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

Now **H** inherits many combinatorial properties:

random walks, colorings, spanning trees, etc.

3. Natural Setting

Spectral formulation more tractable:

$\boldsymbol{x^T L x}$ better behaved over $\mathbf{R}^n$ than $\{0,1\}^n$.

Cuts are discrete objects.

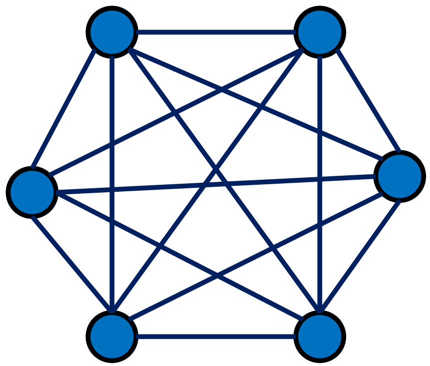
Quadratic forms are continuous objects,

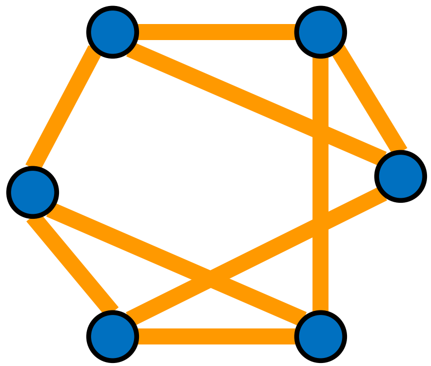
with a richer set of global transformations.

Examples

Example: The Complete Graph

G=K$_n$ **H** = random d-regular x **(n/d)**



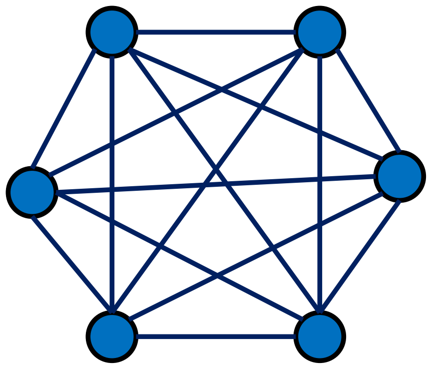


$|E_G| = O(n^2)$ $|E_H| = O(dn)$

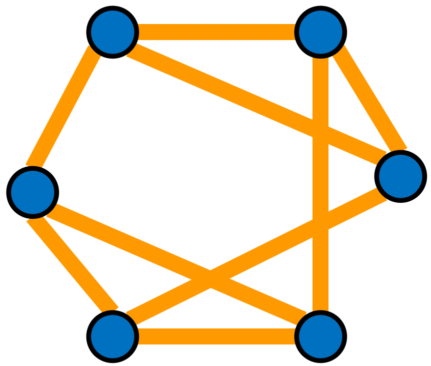
$$\forall S \subset V, \quad \frac{wt_G(\delta S)}{wt_{(n/d)H}(\delta S)} \simeq 1 \pm \epsilon$$

Example: The Complete Graph

G=K$_n$ **H** = random d-regular x **(n/d)**



$|E_G| = O(n^2)$



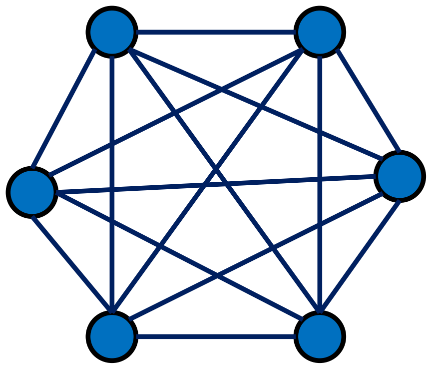
$|E_H| = O(dn)$

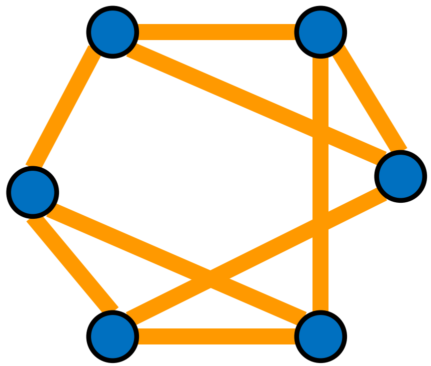
$$\forall x, \quad \frac{x^T L_G x}{x^T L_H x} \simeq 1 \pm \epsilon$$

Example: The Complete Graph

$G=K_n$

H = random d-regular x **(n/d)**





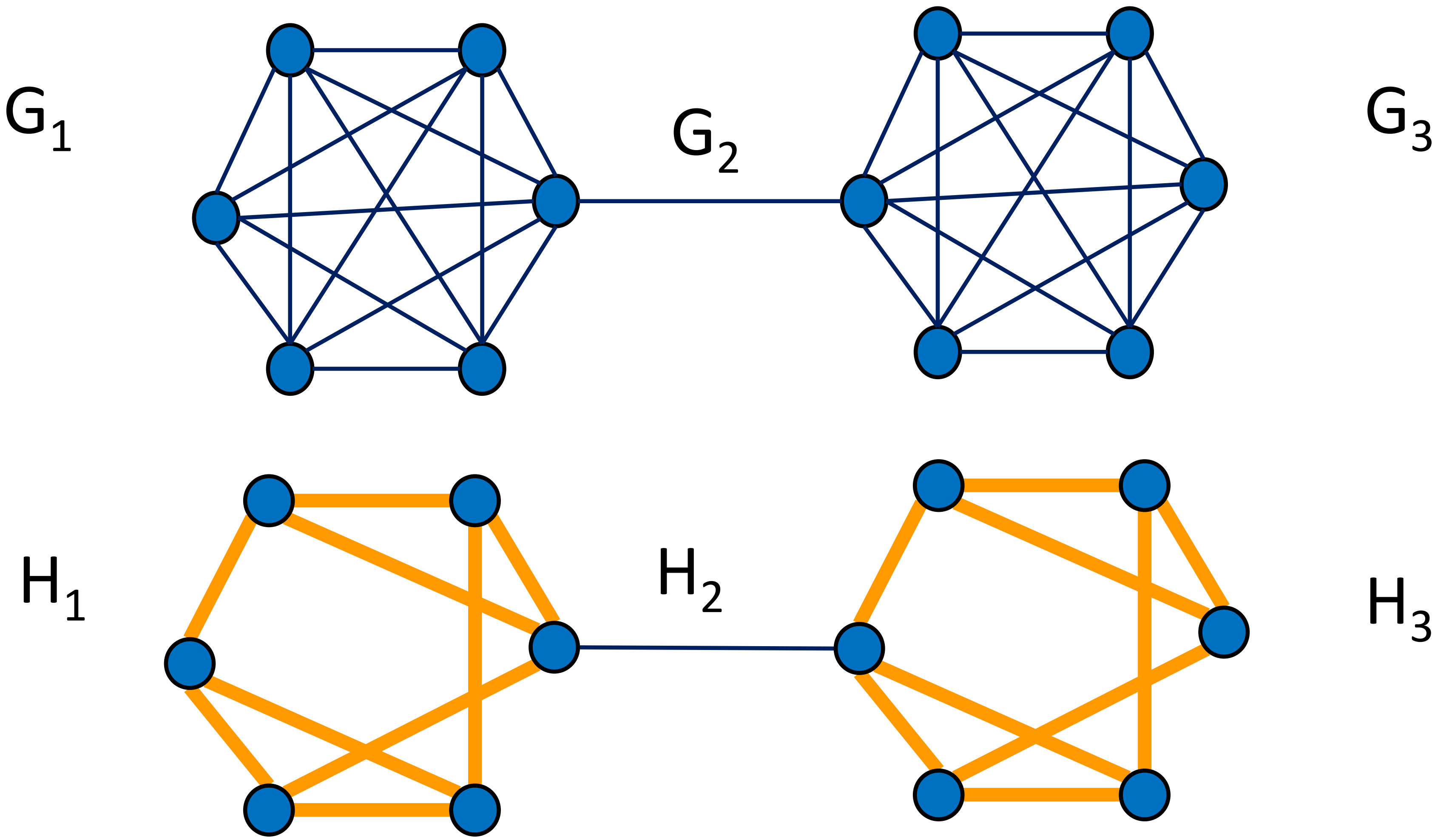
$|E_G| = O(n^2)$

$|E_H| = O(dn)$

$$d = 2/\epsilon^2$$

$$\forall x, \quad \frac{x^T L_G x}{x^T L_H x} \simeq 1 \pm \epsilon$$

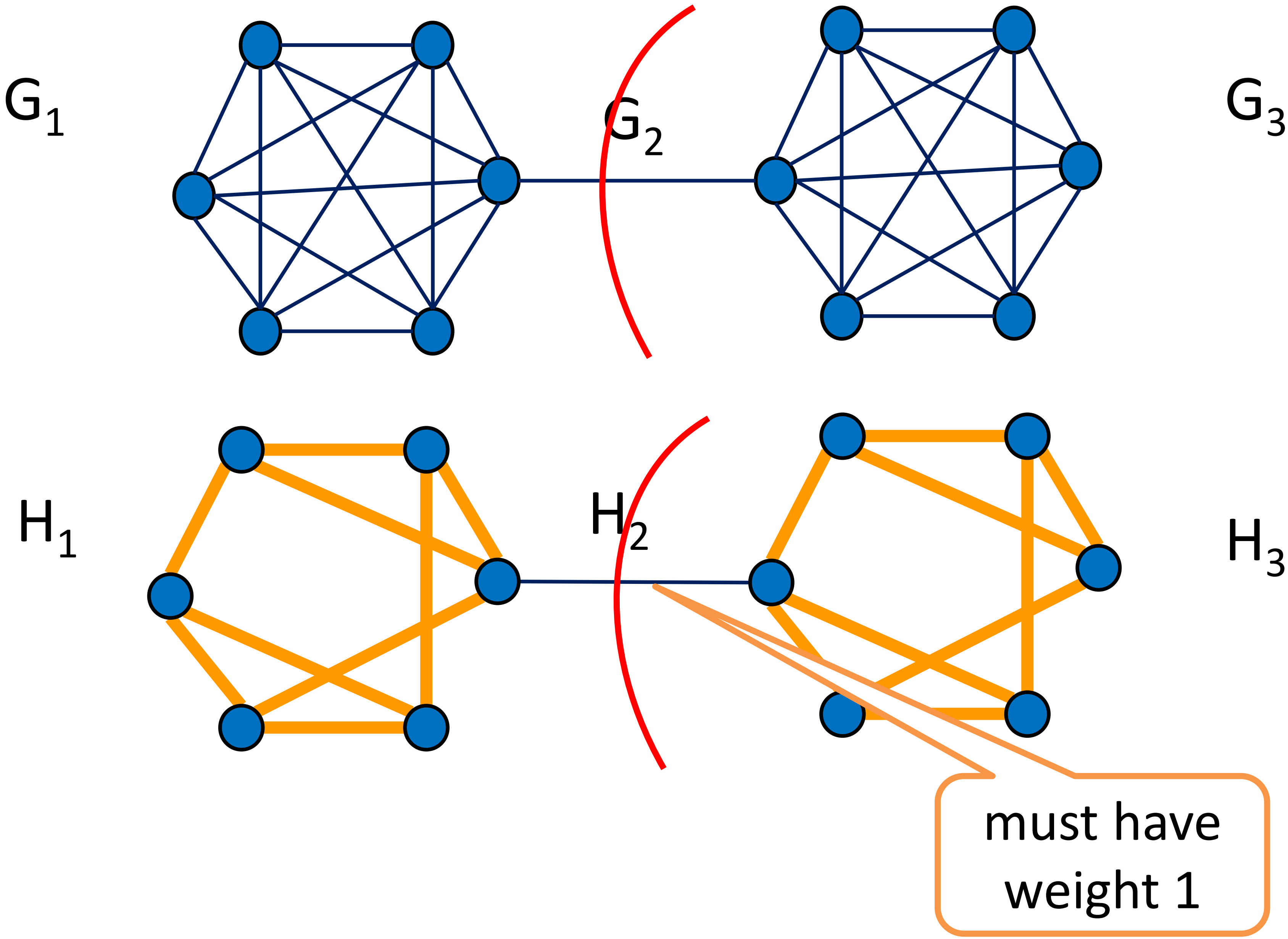
Example: Dumbell



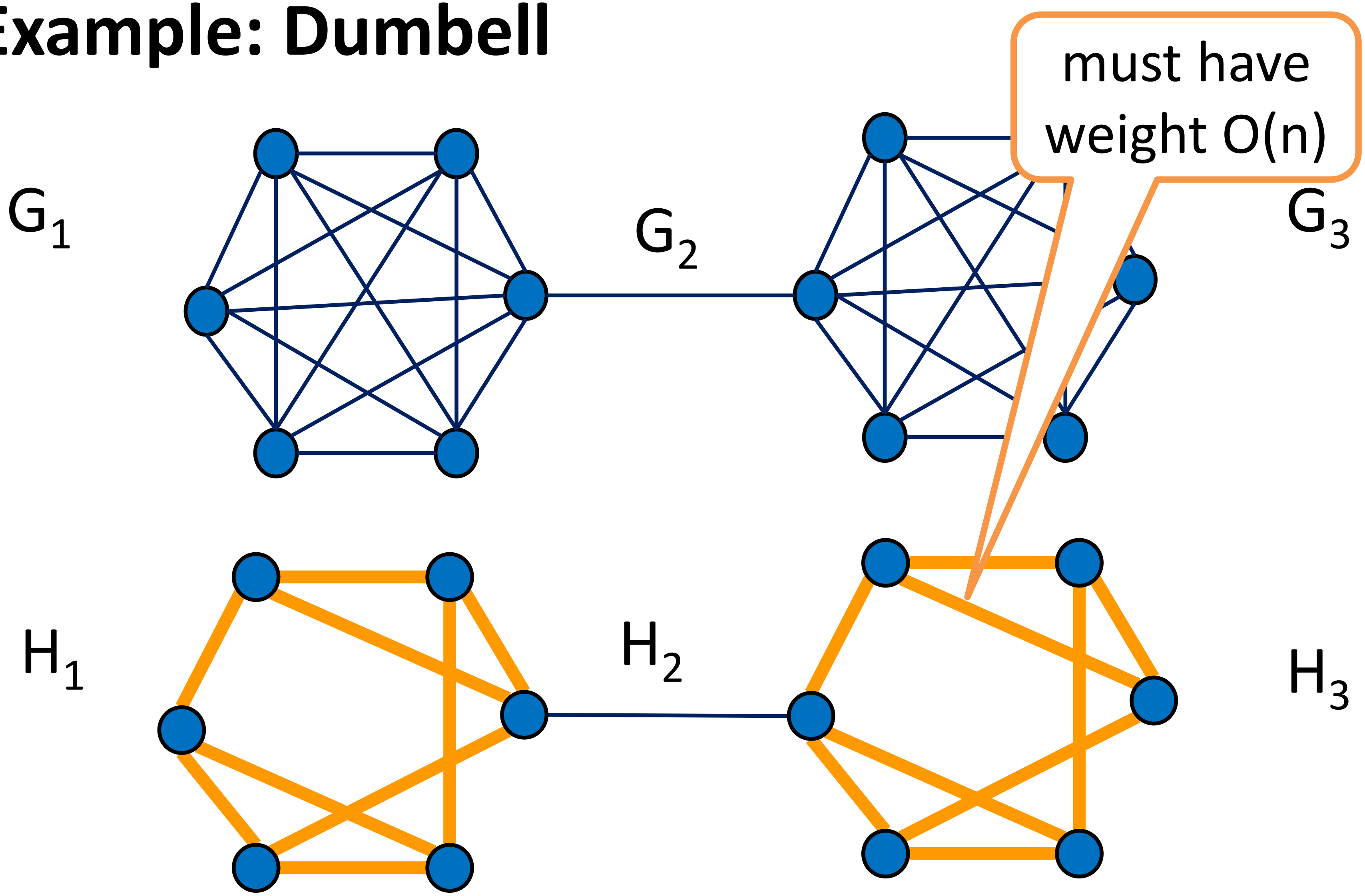
$$G = G_1 + G_2 + G_3$$

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

Example: Dumbell



Example: Dumbell



Will show how to do this for every graph…

Theorem. Every weighted graph **G** has a weighted subgraph **H** with at most $9n \log n / \epsilon^2$ edges s.t.

$$L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon) L_G.$$

Moreover, H can be found in time $O^{\sim}(m/\epsilon^2)$.

Basic idea: **Random Sampling**

Choose each edge e with some probability p_e.

take k independent samples.

If included, add to H with weight $1/kp_e$.

$$\mathbb{E}[L_H] = \mathbb{E}[L_e] = \sum_{e \in G} p_e \cdot \frac{b_e b_e^T}{p_e} = L_G.$$

Basic idea: **Random Sampling**

Choose each edge e with some probability p_e.

take k independent samples.

If included, add to H with weight $1/kp_e$.

$$\mathbb{E}[L_H] = \mathbb{E}[L_e] = \sum_{e \in G} p_e \cdot \frac{b_e b_e^T}{p_e} = L_G.$$

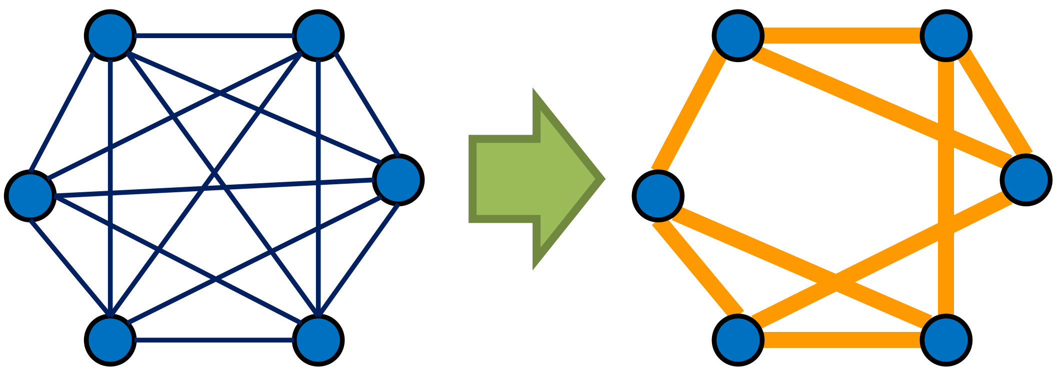
Law of large numbers: as $k \to \infty$,

$$L_H \to L_G$$

Question: how fast does this happen?

Attempt: Uniform Sampling

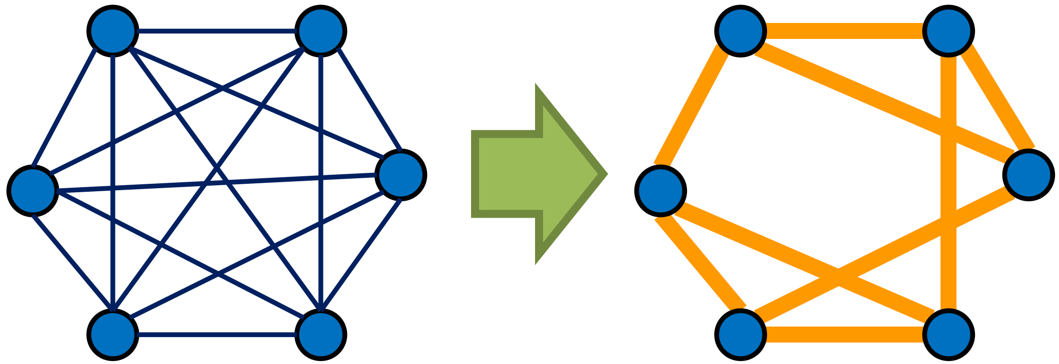
Works for K_n:



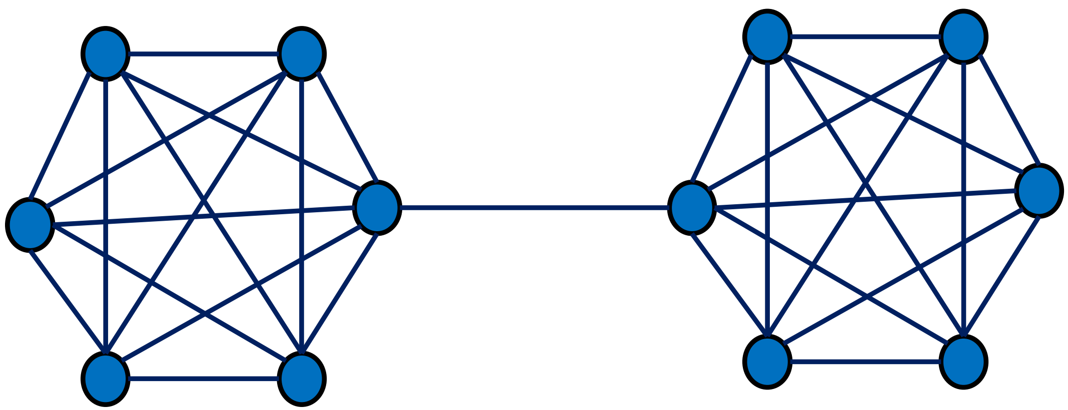
**O(nlogn) samples for i.i.d. edges*

Attempt: Uniform Sampling

Works for K_n:



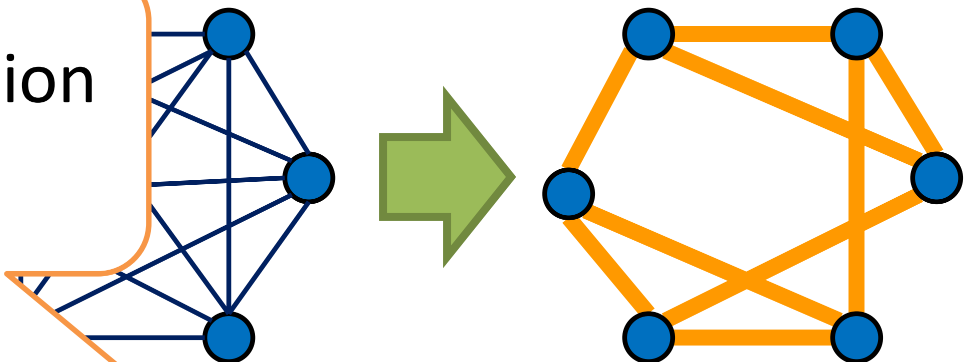
Bad for dumbbell:



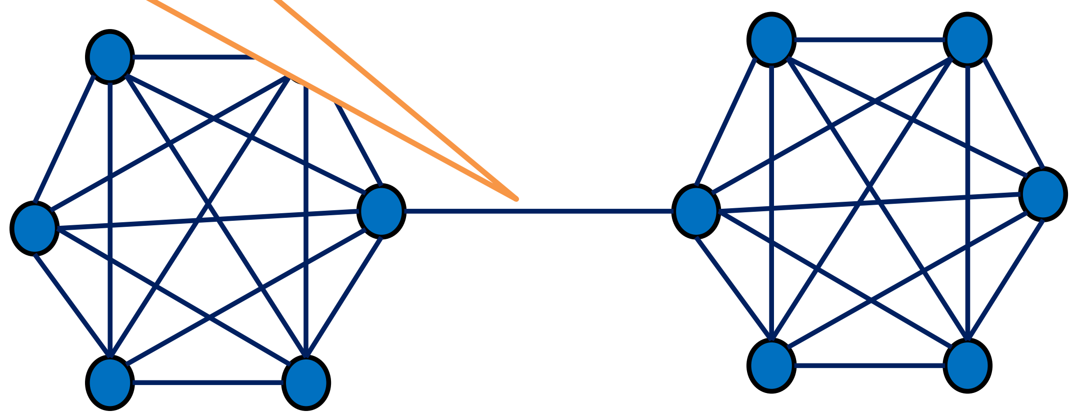
Need $\Omega(m)$ samples to catch the bridge edge.

Attempt: Uniform Sampling

Need to bias distribution towards this edge



Bad for dumbbell:



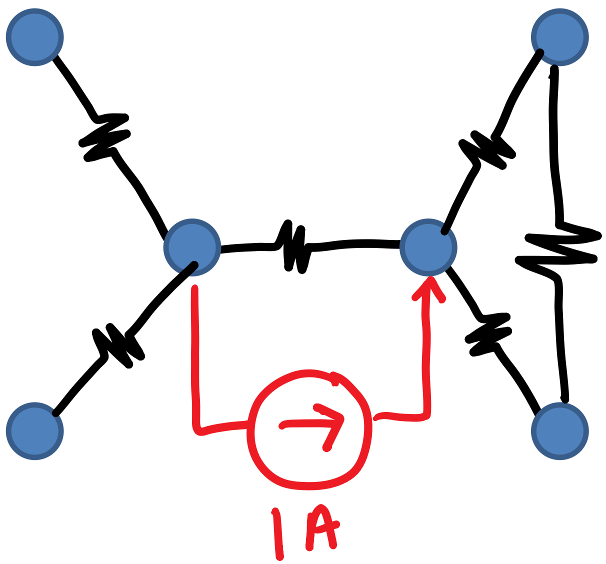
Need $\Omega(m)$ samples to catch the bridge edge.

Effective Resistance

$\mathbf{Reff}(e)$ **=** energy dissipation when a unit current is injected/removed across ends of e**.**

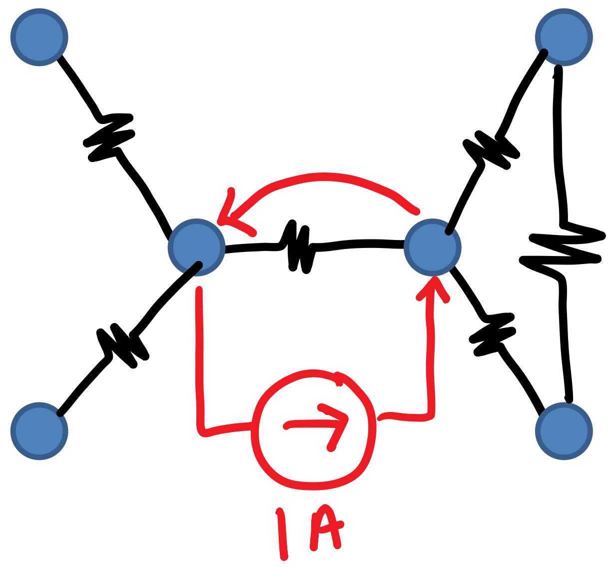
Effective Resistance

$\mathbf{Reff}(e)$ = energy dissipation when a unit current is injected/removed across ends of e.



Effective Resistance

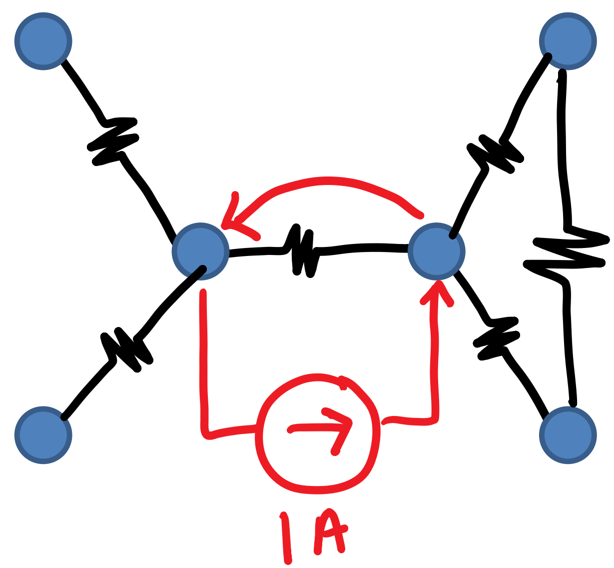
$\mathbf{Reff}(e)$ = energy dissipation when a unit current is injected/removed across ends of e.



electrical flow minimizes energy

Effective Resistance

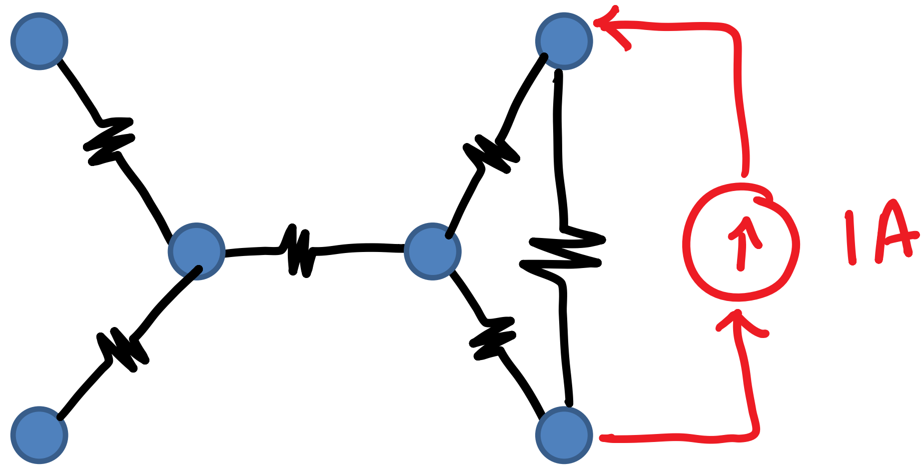
$\mathbf{Reff}(e)$ **=** energy dissipation when a unit current is injected/removed across ends of e**.**



$$\mathbf{Reff}(e) = 1^2 = 1$$

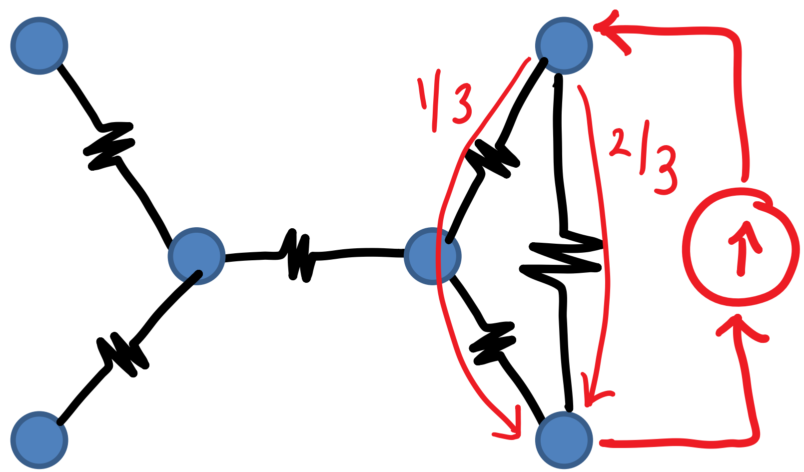
Effective Resistance

$\mathbf{Reff}(e)$ = energy dissipation when a unit current is injected/removed across ends of e.



Effective Resistance

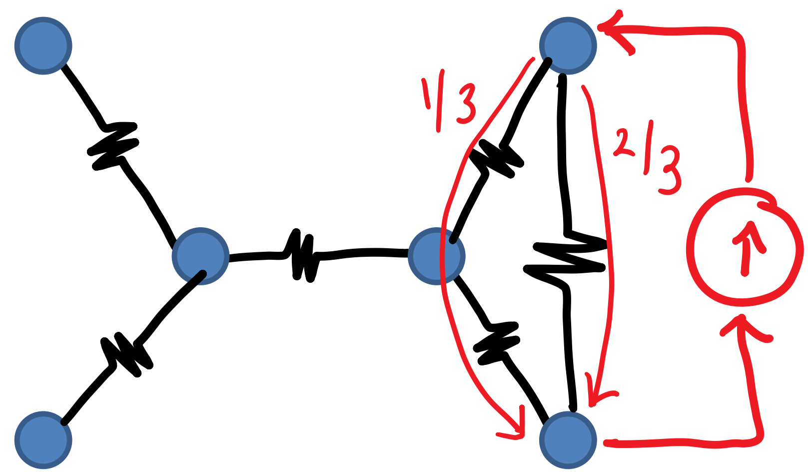
$\mathbf{Reff}(e)$ = energy dissipation when a unit current is injected/removed across ends of e.



$$\mathbf{Reff}(e) = (2/3)^2 + (1/3)^2 + (1/3)^2 = 2/3$$

Effective Resistance

$\mathbf{Reff}(e)$ **=** energy dissipation when a unit current is injected/removed across ends of e**.**



many alternate paths = lower resistance
= **electrically "redundant"**

Effective Resistance

$\mathbf{Reff}(e)$ **=** energy dissipation when a unit current is injected/removed across ends of e.

few alternate paths = high resistance
= **electrically "important"**

many alternate paths = lower resistance
= **electrically "redundant"**

Effective Resistance

$\mathbf{Reff}(e)$ **=** energy dissipation when a unit current is injected/removed across ends of e.

few alternate paths = high resistance
= **electrically "important"**

many alternate paths = lower resistance
= **electrically "redundant"**

Idea: sample edges according to effective resistances.

Theorem. Every weighted graph **G** has a weighted subgraph **H** with at most $9n\log n/\epsilon^2$ edges s.t.

$$L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon)L_G.$$

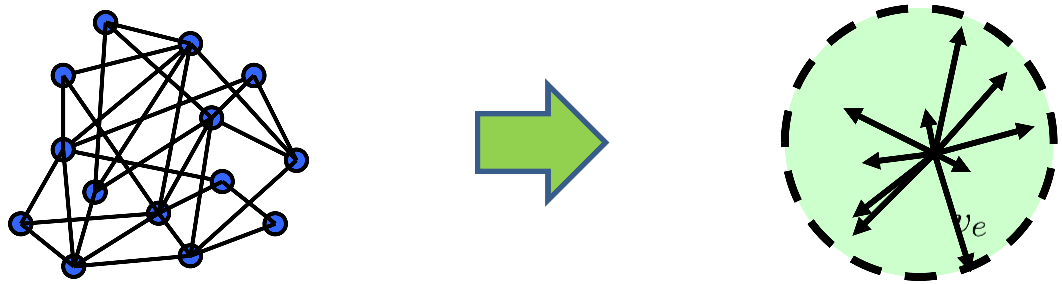
Moreover, H can be found in time $O^{\sim}(m/\epsilon^2)$.

Algorithm: sample $9n\log n/\epsilon^2$ edges independently according to effective resistances.

3 Step Proof

1. Reduction to a linear algebra problem
2. Solution of linear algebra problem by random matrix theory.
3. Fast computation of sampling probabilities

[Spielman-S'08]



Part 1: Reduction to Linear Algebra

Original Goal

Given G

Find sparse H

satisfying $L_G \preceq L_H \preceq \kappa \cdot L_G$

Outer Product Expansion

Recall:

$$L_G = \sum_{ij \in E} (\delta_i - \delta_j)(\delta_i - \delta_j)^T = \sum_{e \in E} b_e b_e^T .$$

Outer Product Expansion

Recall:

$$L_G = \sum_{ij\in E} (\delta_i - \delta_j)(\delta_i - \delta_j)^T = \sum_{e\in E} b_e b_e^T .$$

For a weighted subgraph ***H***:

$$L_H = \sum_{e\in E} s_e b_e b_e^T$$

where s_e=wt(e) in H.

Original Goal

Given G

Find **sparse** H

satisfying

$$L_G \preceq L_H \preceq \kappa \cdot L_G$$

Original Goal

Given $$L_G = \sum_{e \in G} b_e b_e^T \qquad b_{ij} = \delta_i - \delta_j$$

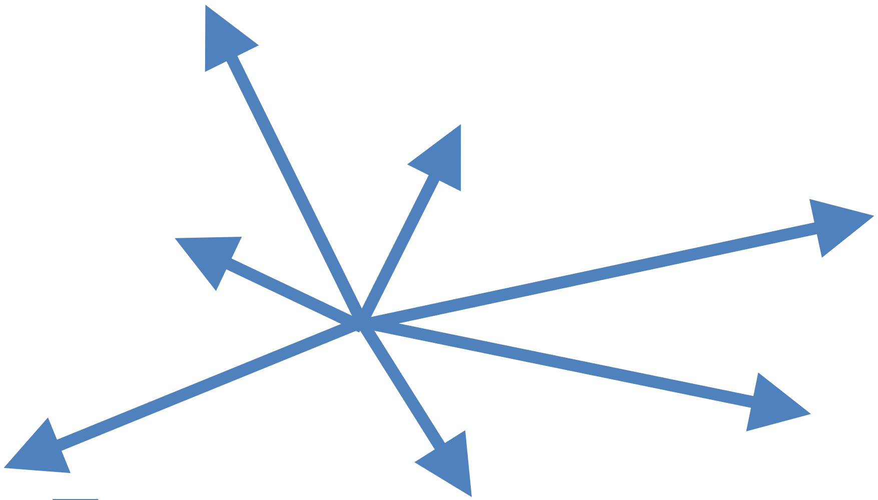
Find **sparse**

$$s_e \geq 0$$

satisfying

$$L_G \preceq L_H = \sum_{e \in G} s_e b_e b_e^T \preceq \kappa \cdot L_G$$

Quadratic Forms as Ellipsoids

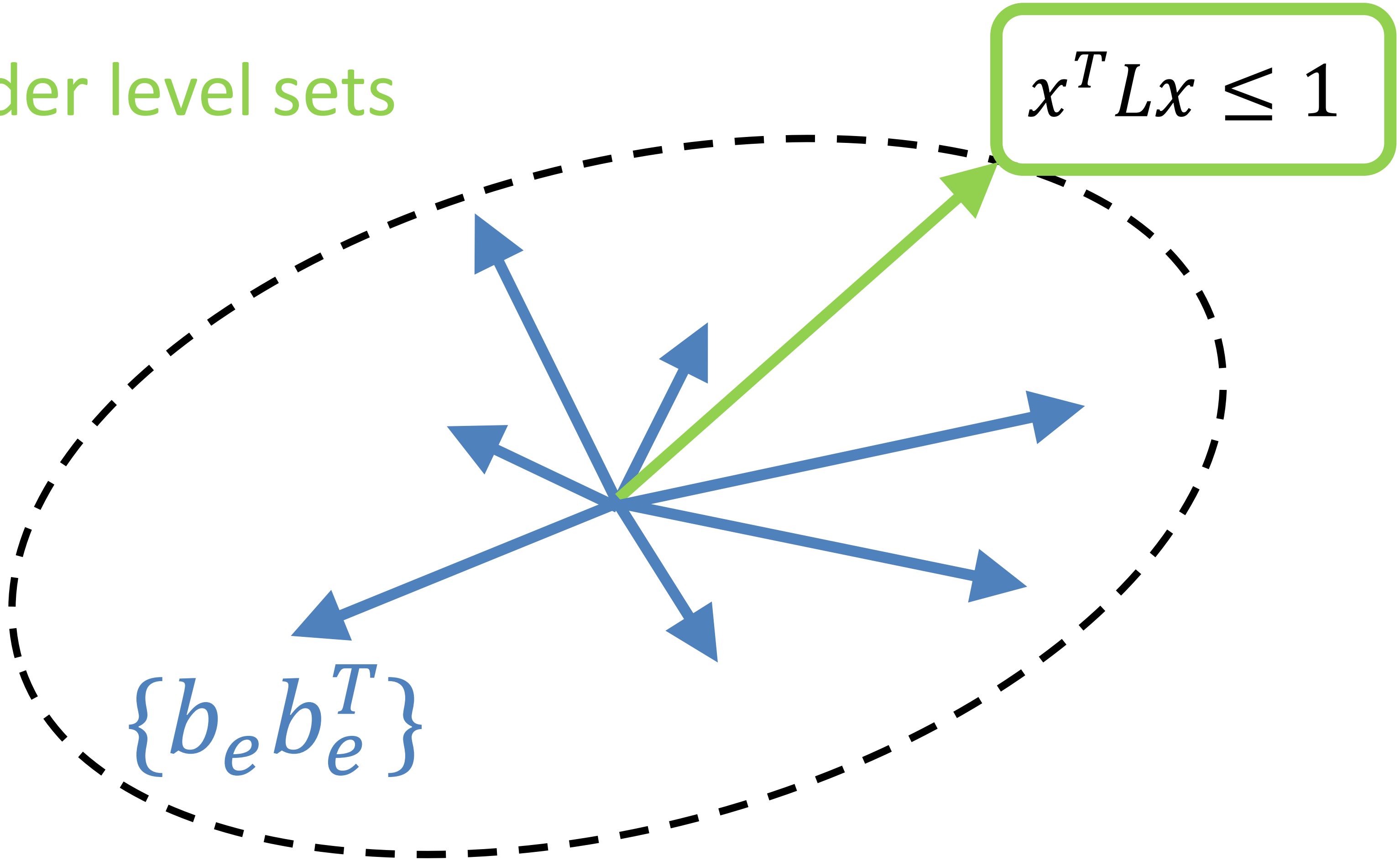


$\{b_e b_e^T\}$

$$L_G = \sum_{e \in G} b_e b_e^T \qquad b_{ij} = \delta_i - \delta_j$$

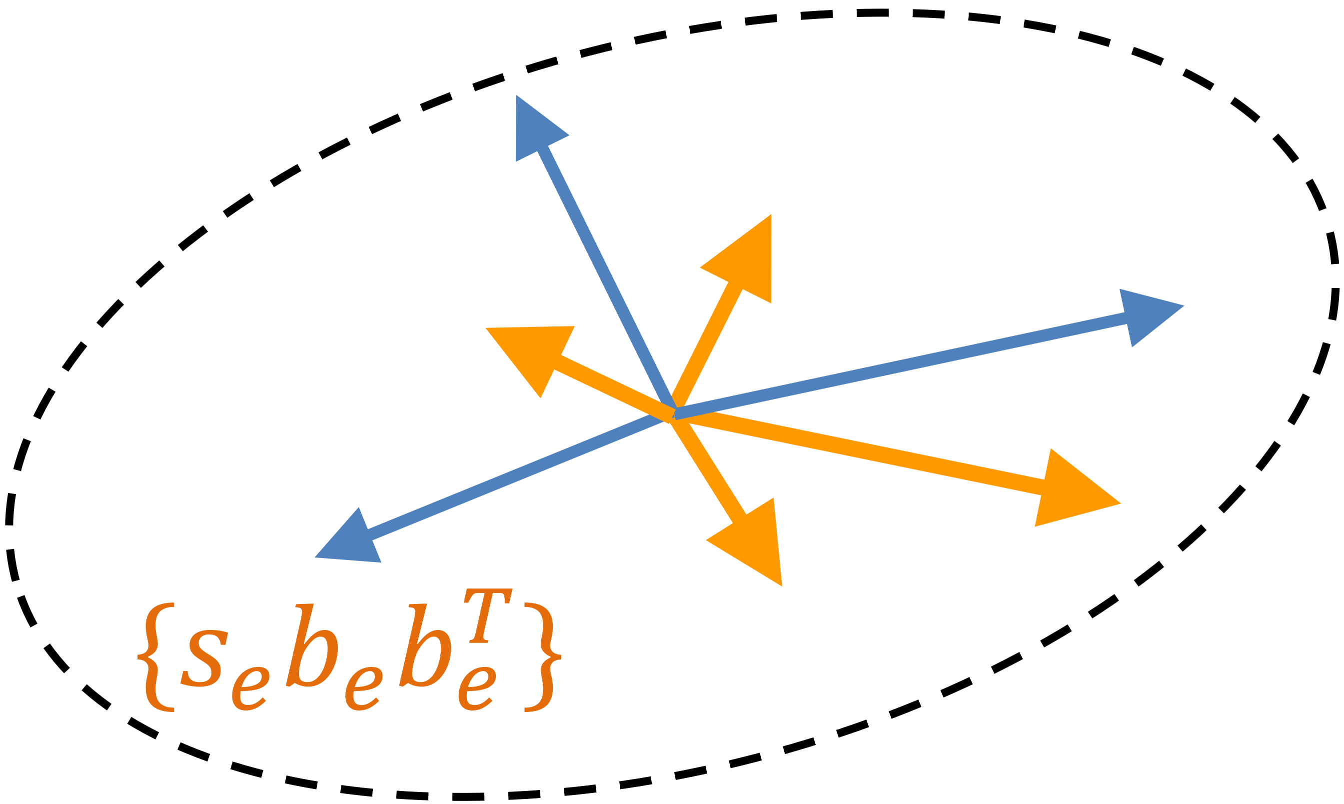
Quadratic Forms as Ellipsoids

Consider level sets



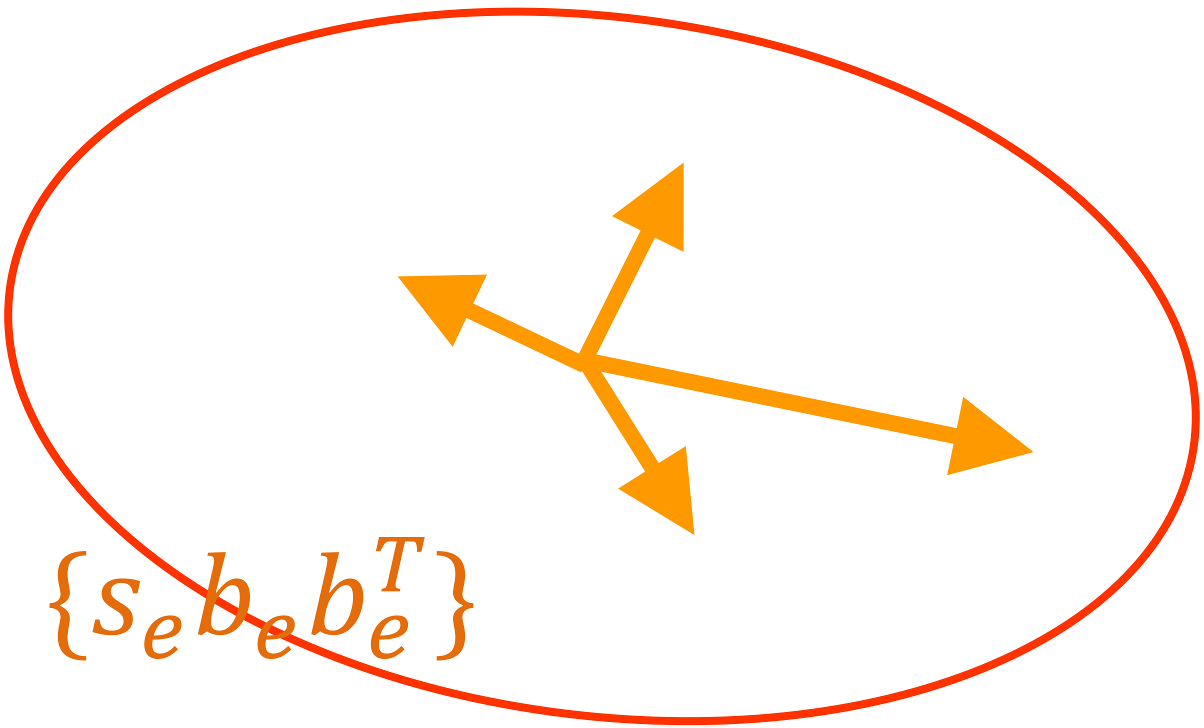
$$L_G = \sum_{e \in G} b_e b_e^T \qquad b_{ij} = \delta_i - \delta_j$$

Quadratic Forms as Ellipsoids



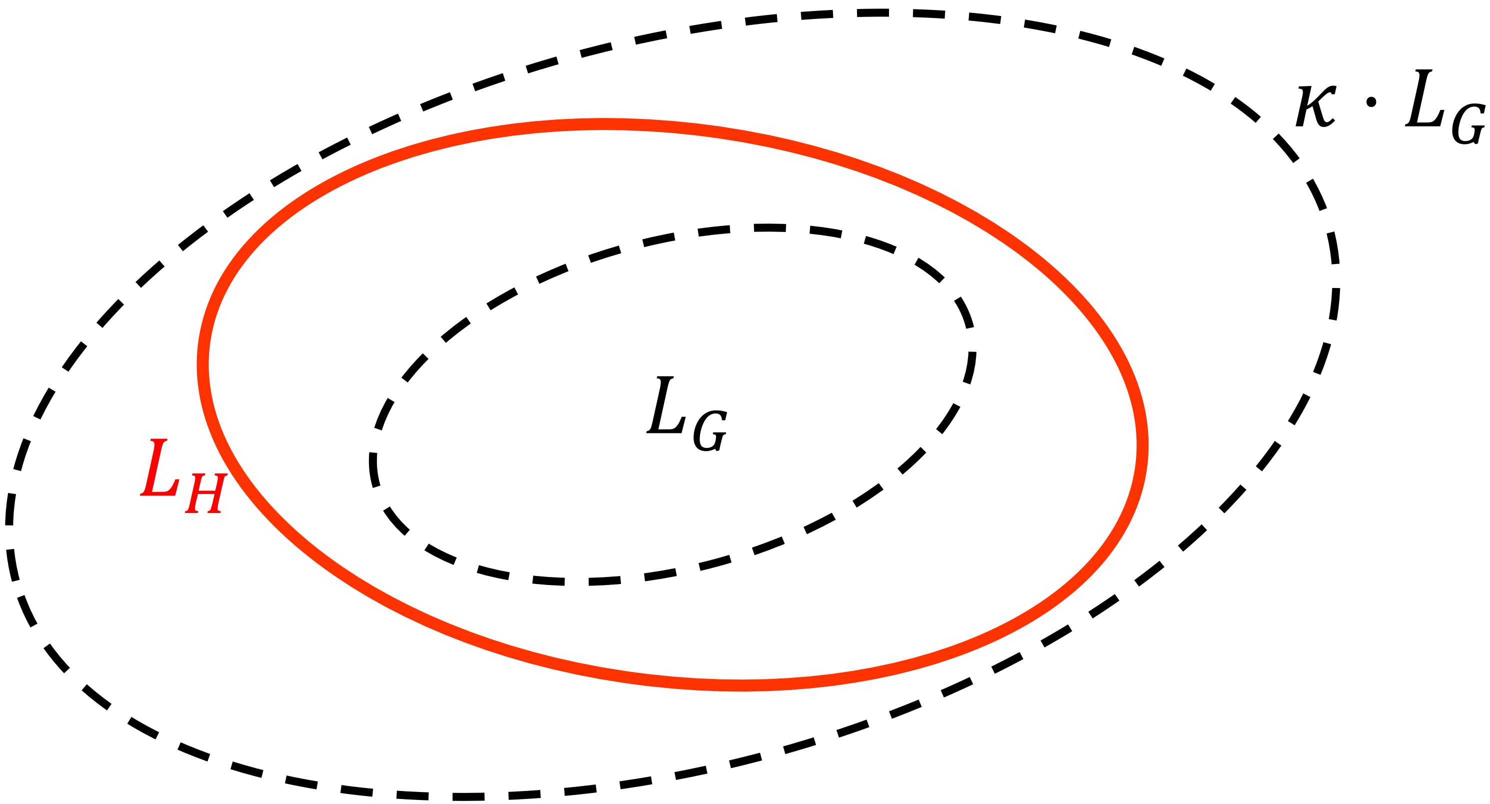
$$L_H = \sum_{e \in E} s_e b_e b_e^T$$

Quadratic Forms as Ellipsoids



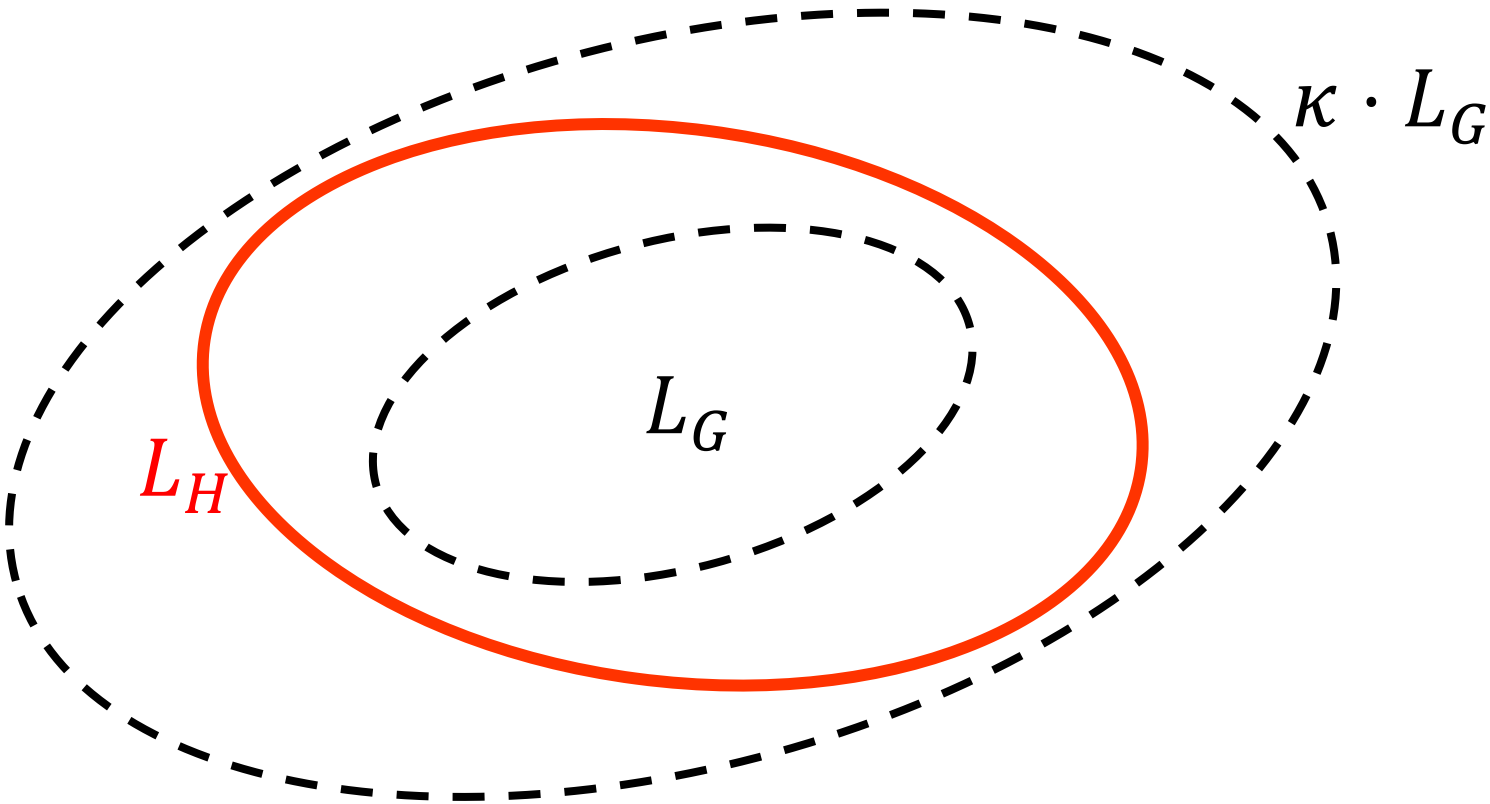
$$L_H = \sum_{e \in E} s_e b_e b_e^T$$

Containment of Ellipsoids



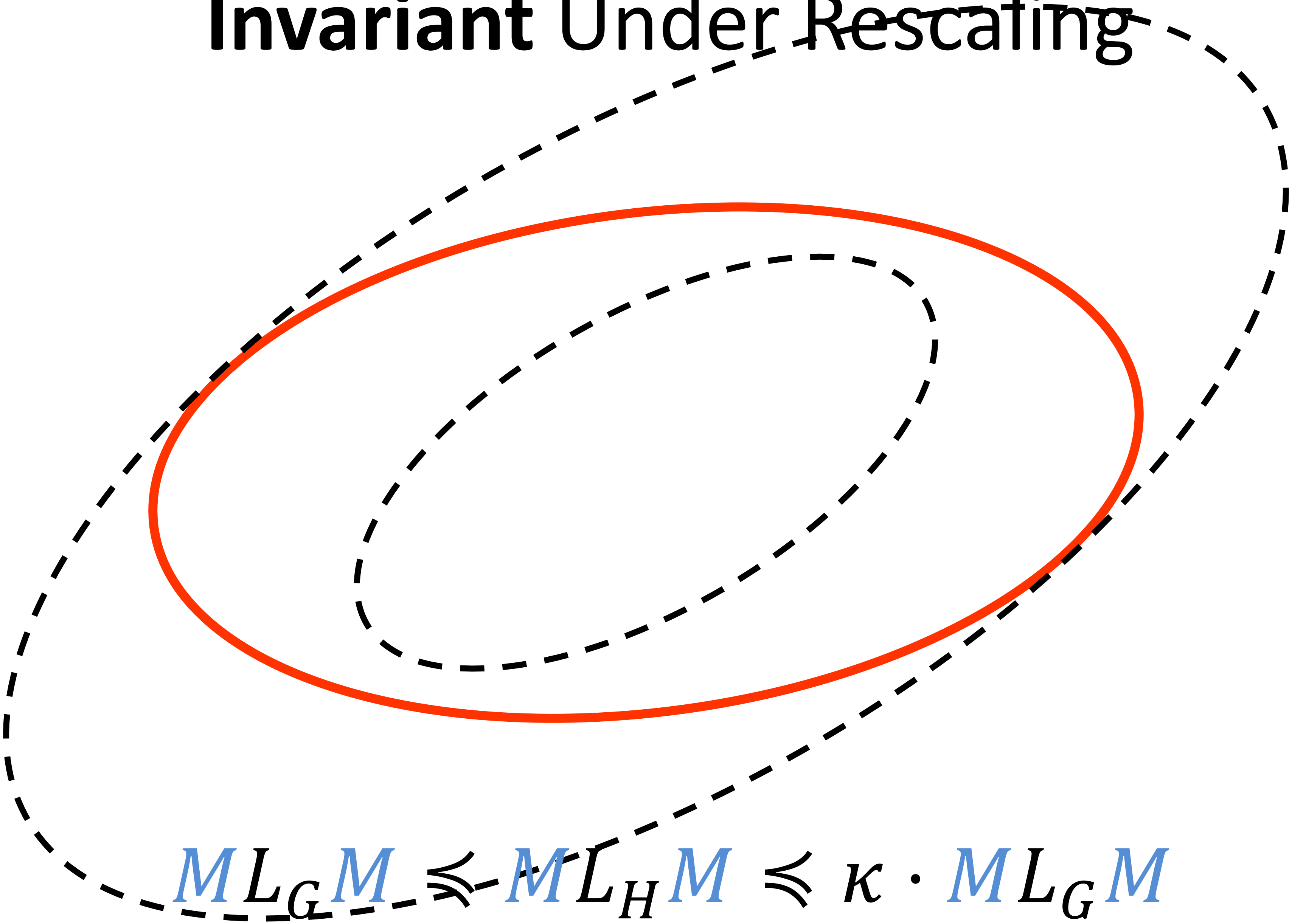
$$L_G \preccurlyeq L_H \preccurlyeq \kappa \cdot L_G$$

Invariant Under Rescaling



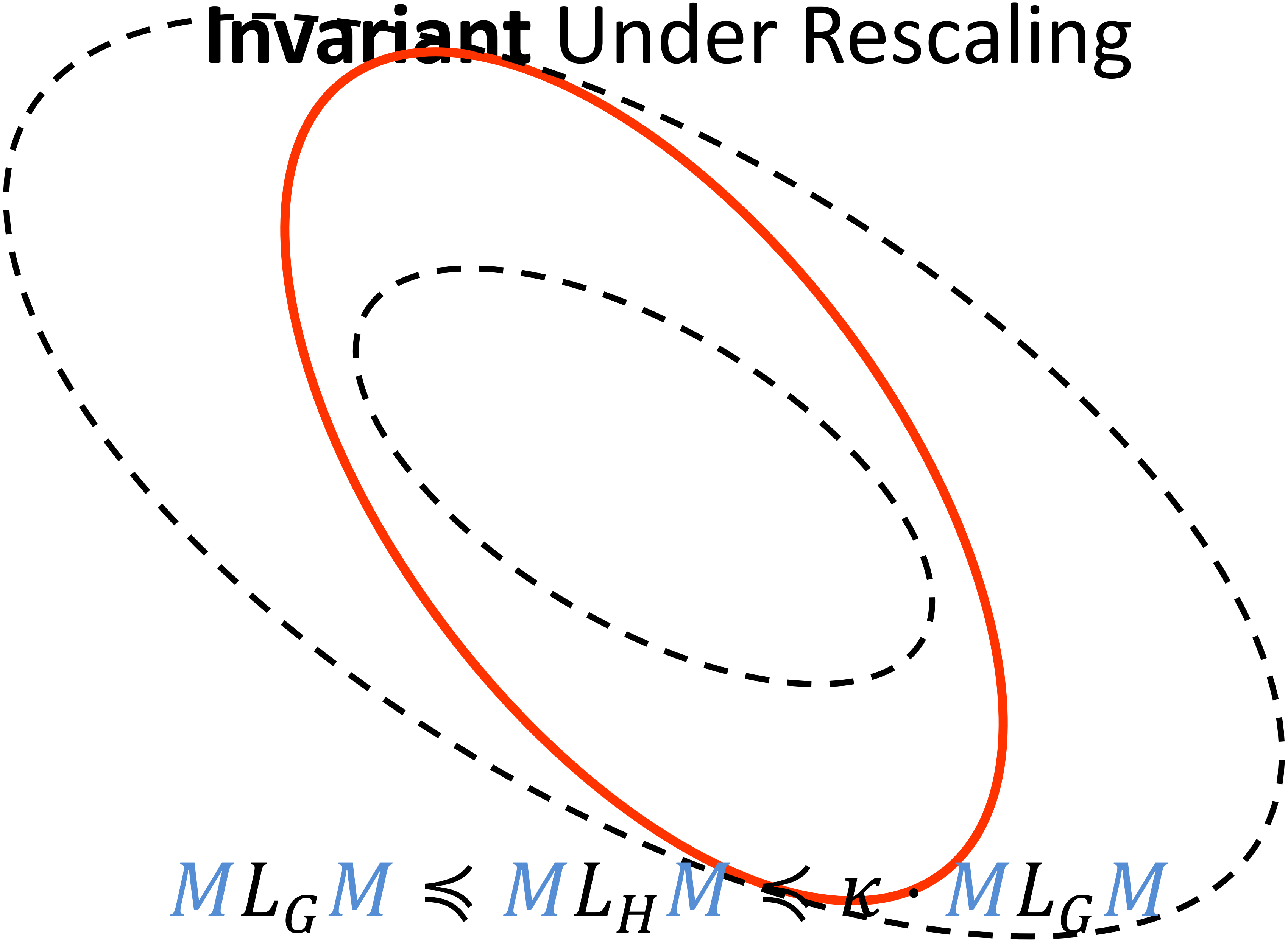
$$M L_G M \preccurlyeq M L_H M \preccurlyeq \kappa \cdot M L_G M$$

Invariant Under Rescaling



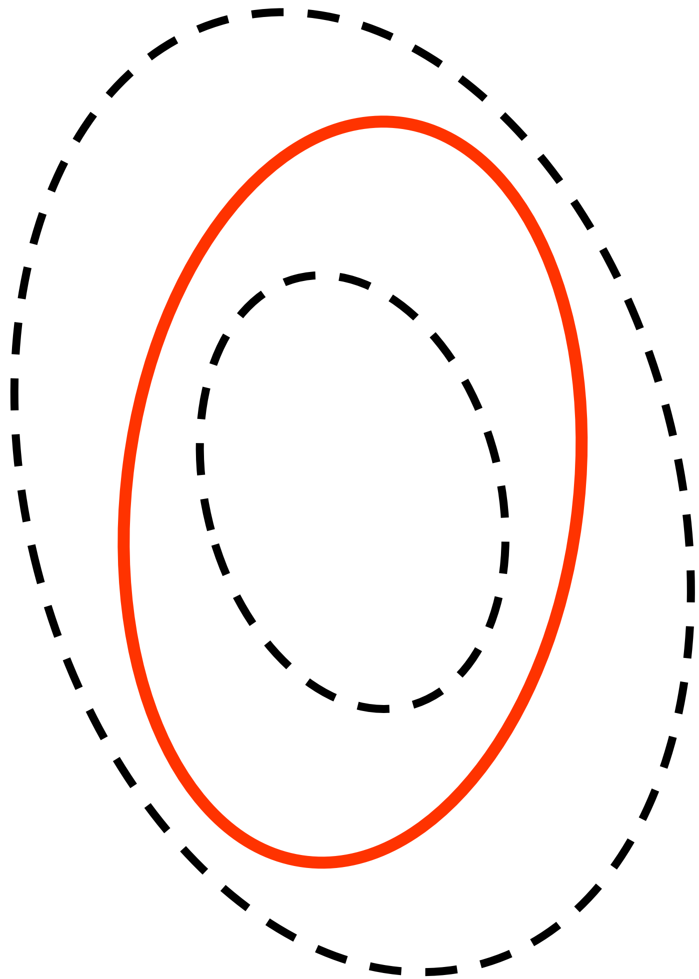
$$ML_GM \preccurlyeq ML_HM \preccurlyeq \kappa \cdot ML_GM$$

Invariant Under Rescaling



$$M L_G M \preccurlyeq M L_H M \preccurlyeq \kappa \cdot M L_G M$$

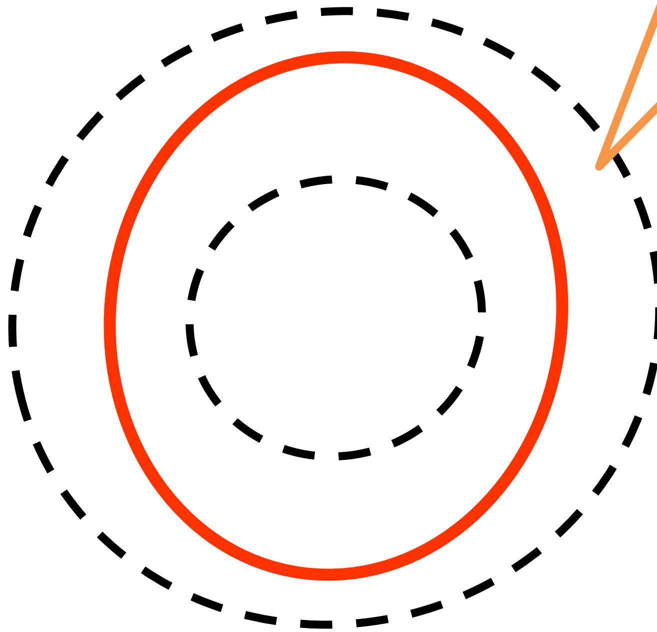
Invariant Under Rescaling



$$M L_G M \preccurlyeq M L_H M \preccurlyeq \kappa \cdot M L_G M$$

Invariant Under Re

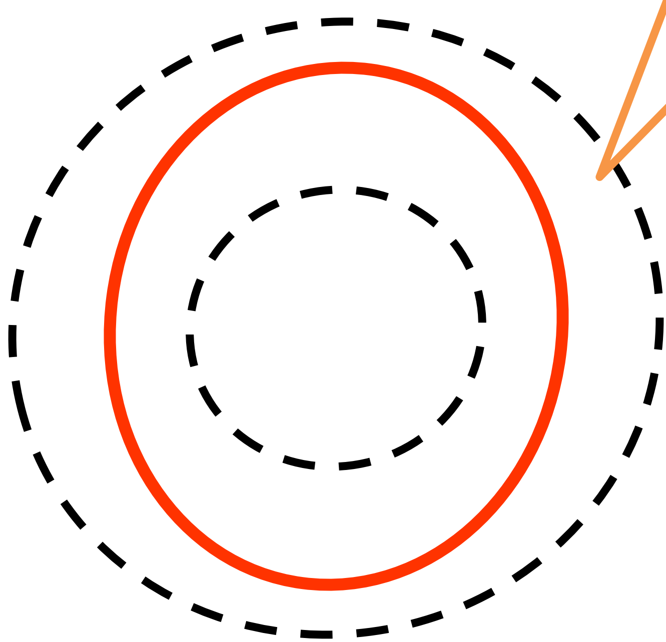
Choose $M = L_G^{-1/2}$



$$L_G^{-1/2} L_G L_G^{-1/2} \preccurlyeq L_G^{-1/2} L_H L_G^{-1/2} \preccurlyeq \kappa \cdot L_G^{-1/2} L_G L_G^{-1/2}$$

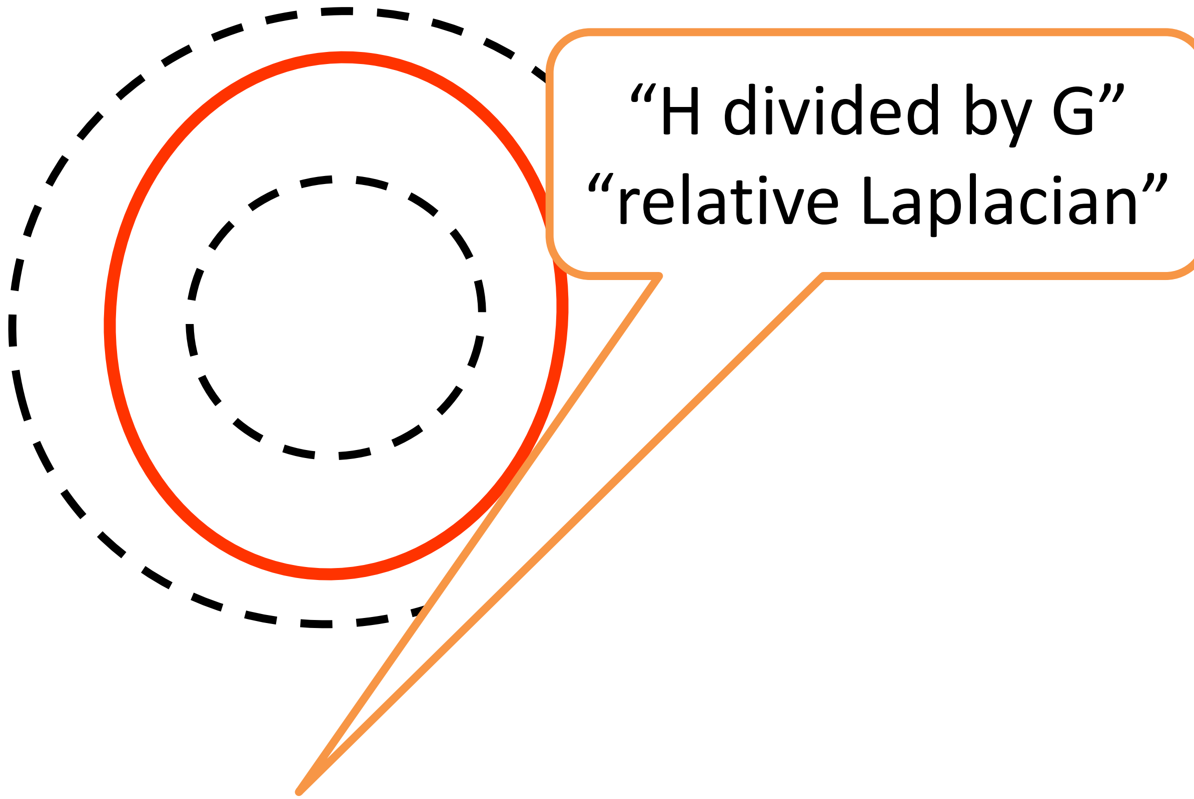
Invariant Under Re

Choose $M = L_G^{-1/2}$



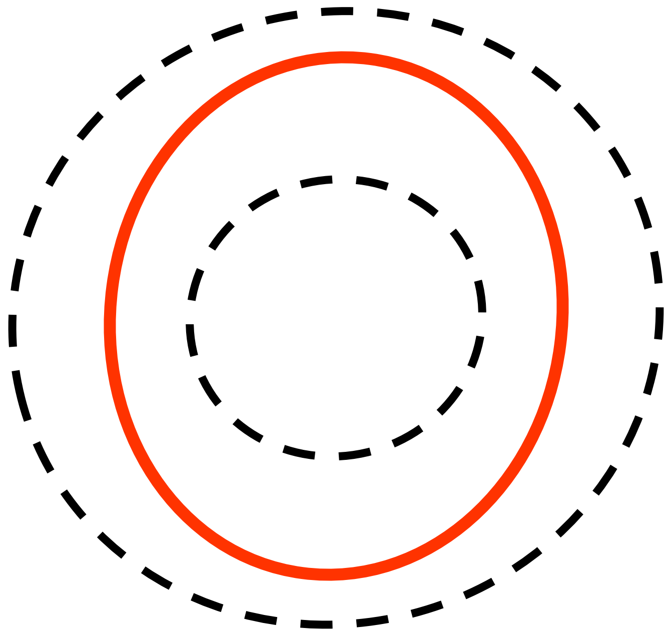
$$I \preccurlyeq L_G^{-1/2} L_H L_G^{-1/2} \preccurlyeq \kappa \cdot I$$

Invariant Under Rescaling



$$I \preccurlyeq L_G^{-1/2} L_H L_G^{-1/2} \preccurlyeq \kappa \cdot I$$

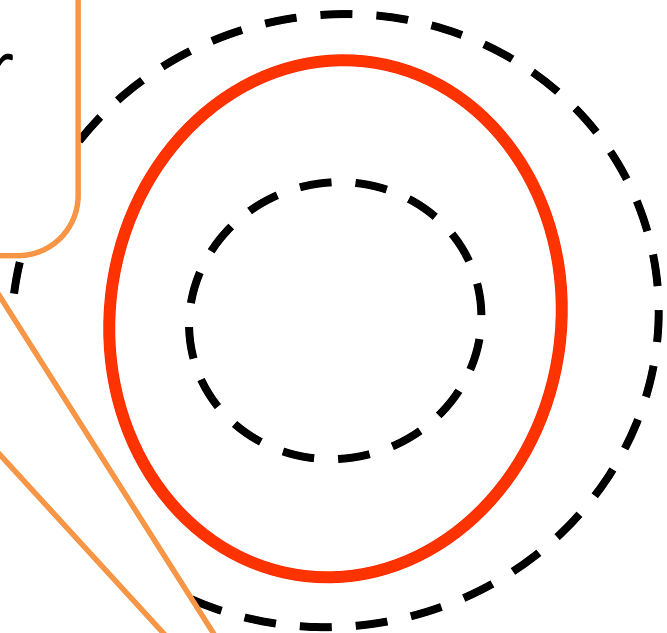
Invariant Under Rescaling



$$I \preccurlyeq \sum_e s_e L_G^{-1/2} b_e b_e^T L_G^{-1/2} \preccurlyeq \kappa \cdot I$$

Invariant Under Rescaling

Rescaled
incidence vector
$v_e \coloneqq L_G^{-1/2} b_e$



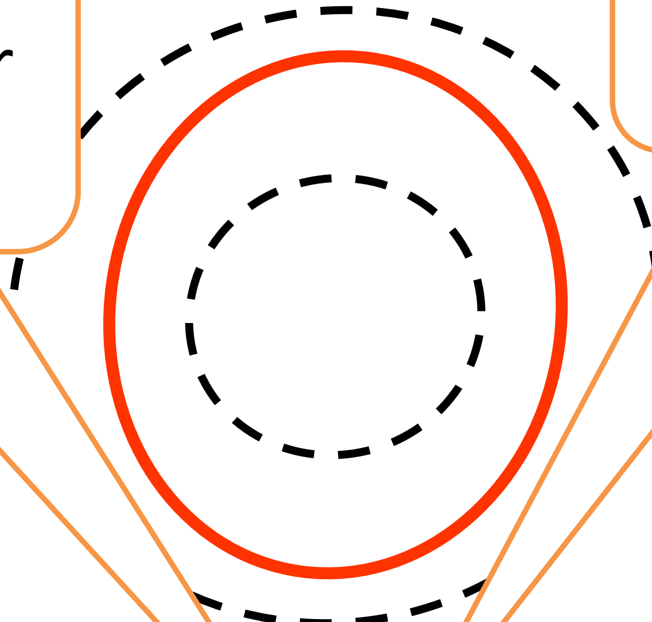
$$I \preccurlyeq \sum_e s_e v_e v_e^T \preccurlyeq \kappa \cdot I$$

Invariant Under Rescaling

Rescaled incidence vector

$v_e := L_G^{-1/2} b_e$

$$\sum_e v_e v_e^T = I$$



$$I \preccurlyeq \sum_e s_e v_e v_e^T \preccurlyeq \kappa \cdot I$$

Equivalent Problem

Given $$I = \sum_e v_e v_e^T$$

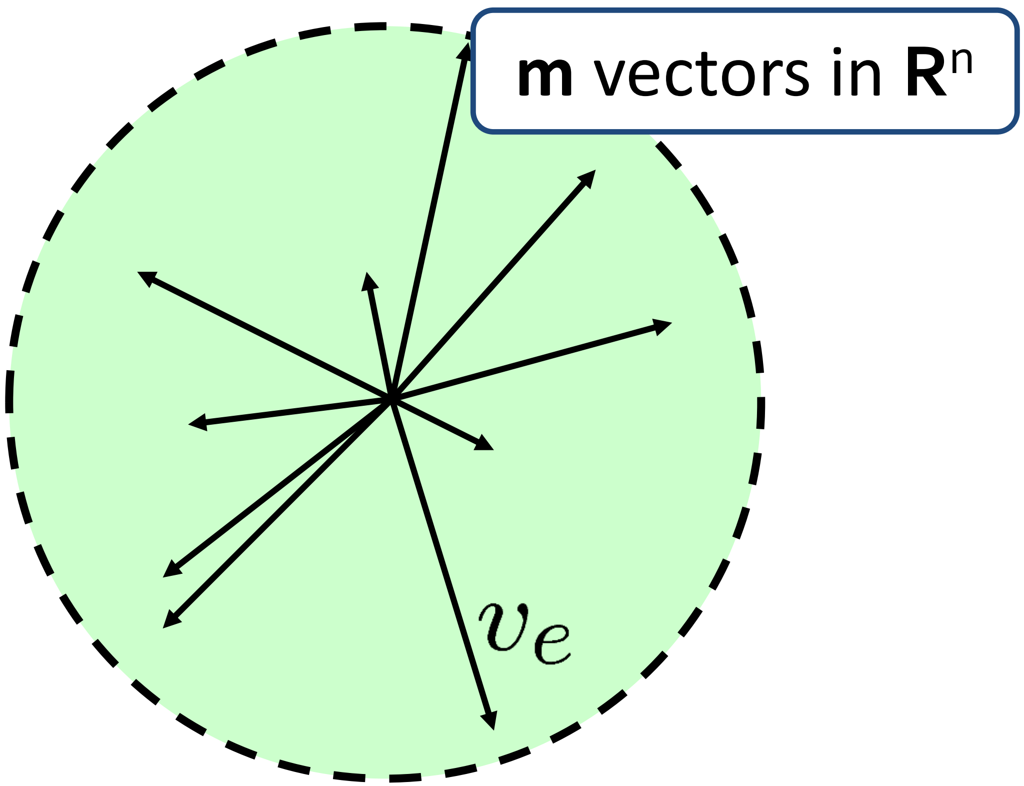
Find **sparse**

$$s_e \geq 0$$

satisfying

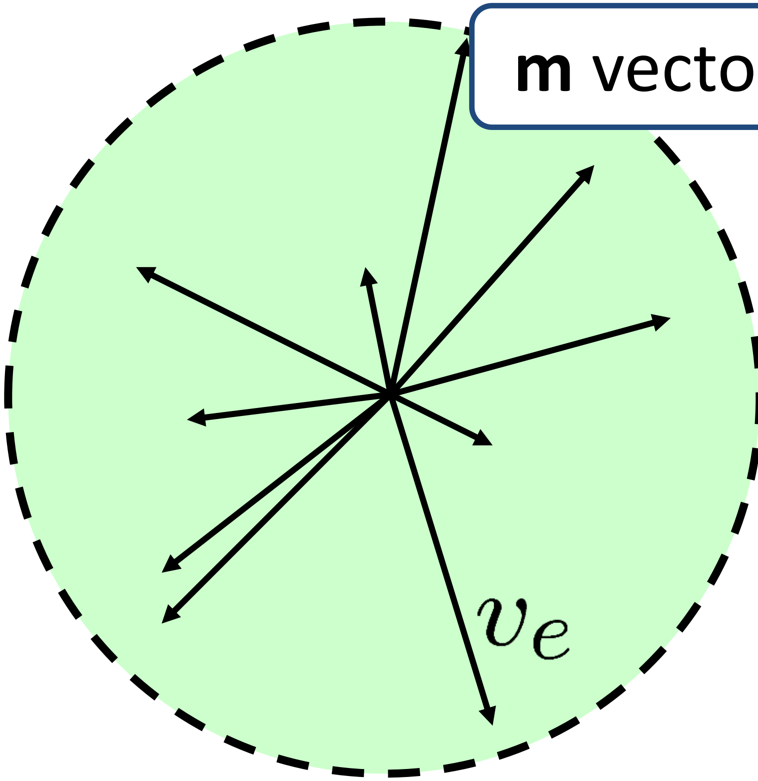
$$I \preceq \sum_{e \in G} s_e v_e v_e^T \preceq \kappa \cdot I$$

Core Problem



$$I = \sum_e v_e v_e^T$$

Core Problem



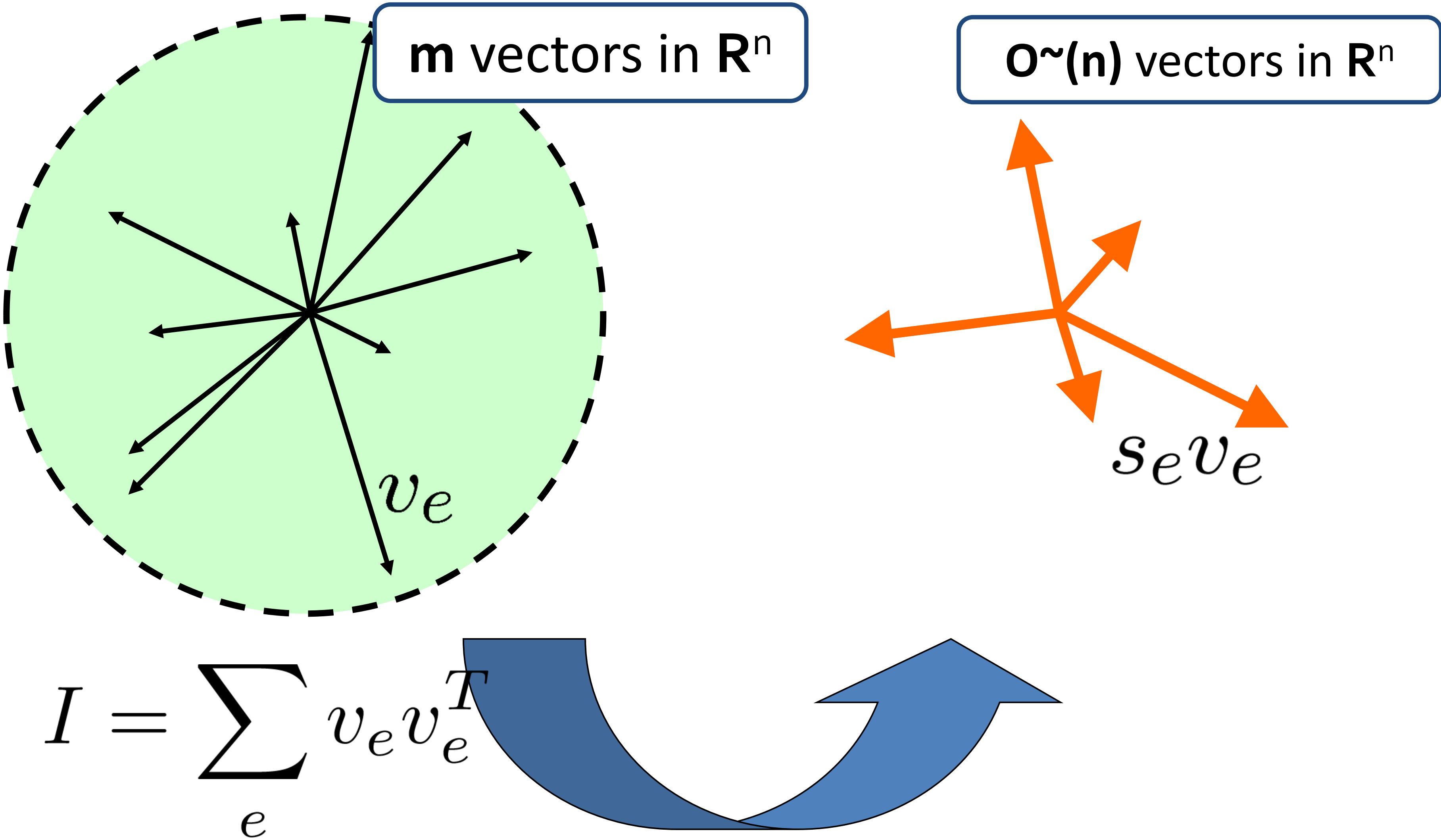
m vectors in **R**^{n}

$$\forall u \quad \sum_e \langle u, v_e \rangle^2 = 1$$

variance is the same in every direction

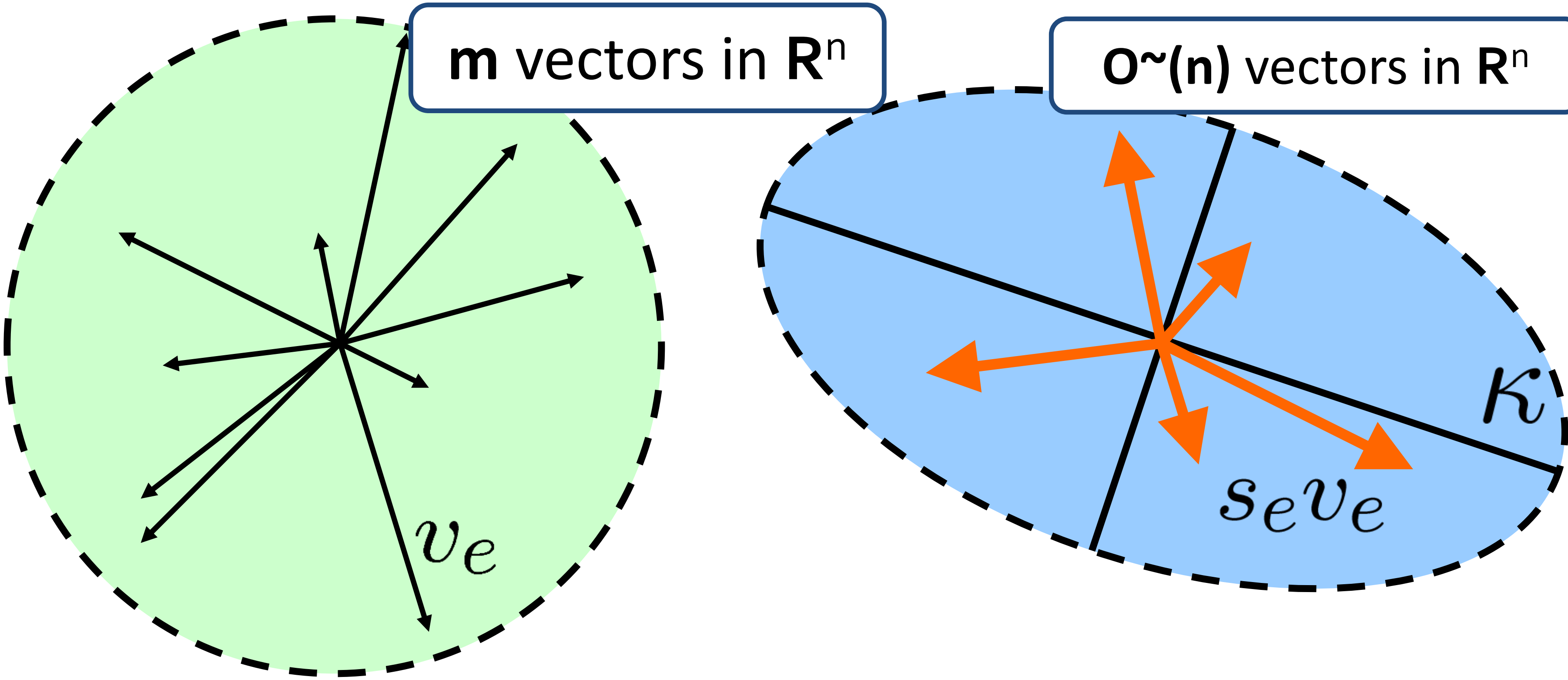
$$I = \sum_e v_e v_e^T$$

Core Problem



$$I = \sum_e v_e v_e^T$$

Core Problem

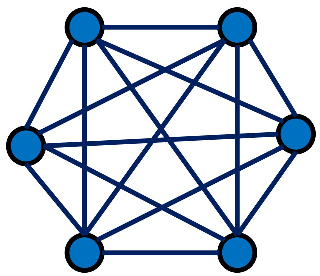
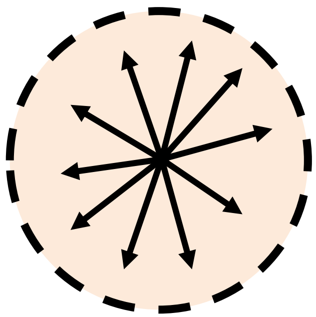
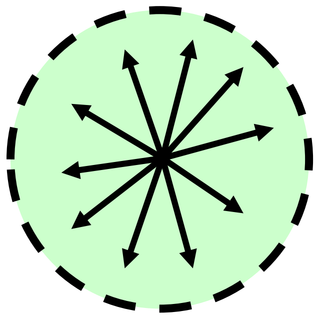


$$I = \sum_e v_e v_e^T \qquad I \preceq \sum_e s_e v_e v_e^T \preceq \kappa I$$

Examples of the Reduction

Graph

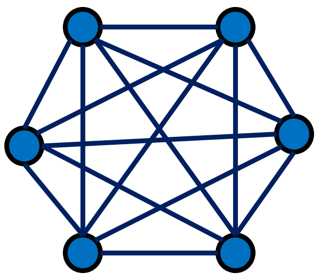
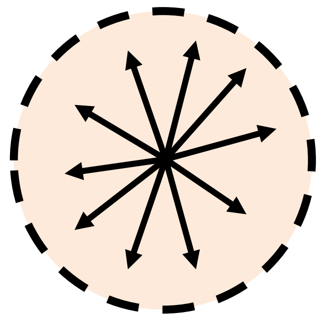
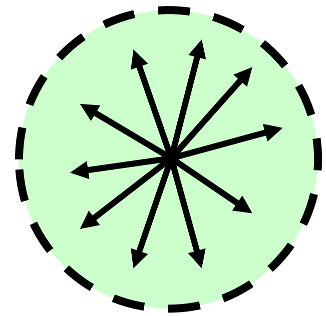
$$L_G = \sum_e b_e b_e^T \qquad I = \sum_e v_e v_e^T$$

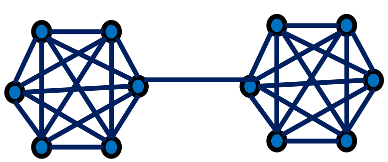
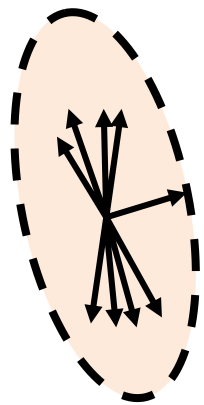
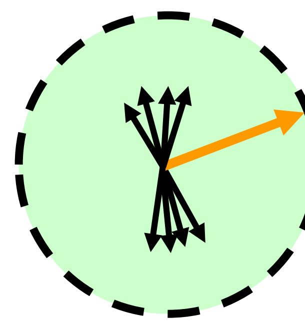




$$v_e = L_G^{-1/2} b_e$$

Examples of the Reduction

Graph $L_G = \sum_e b_e b_e^T$ $I = \sum_e v_e v_e^T$

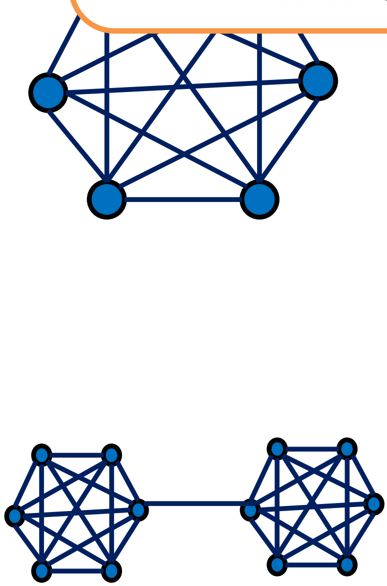




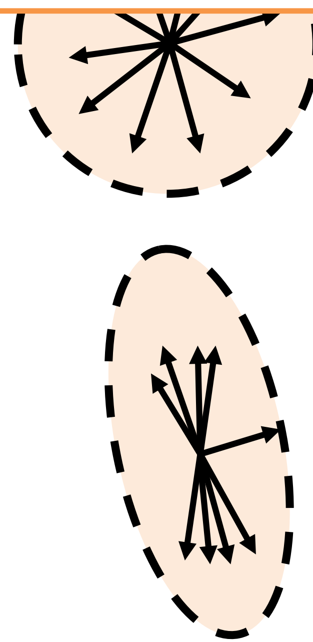
$$v_e = L_G^{-1/2} b_e$$

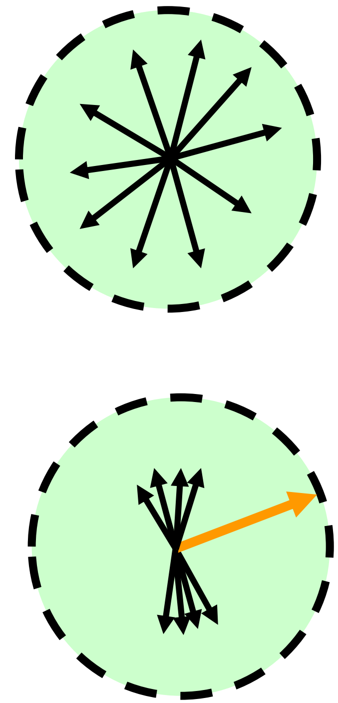
Examples of the Reduction

$$I = \sum_e v_e v_e^T$$

Q: Why rescale to identity?
A: All test directions are equally important in multiplicative approximation.





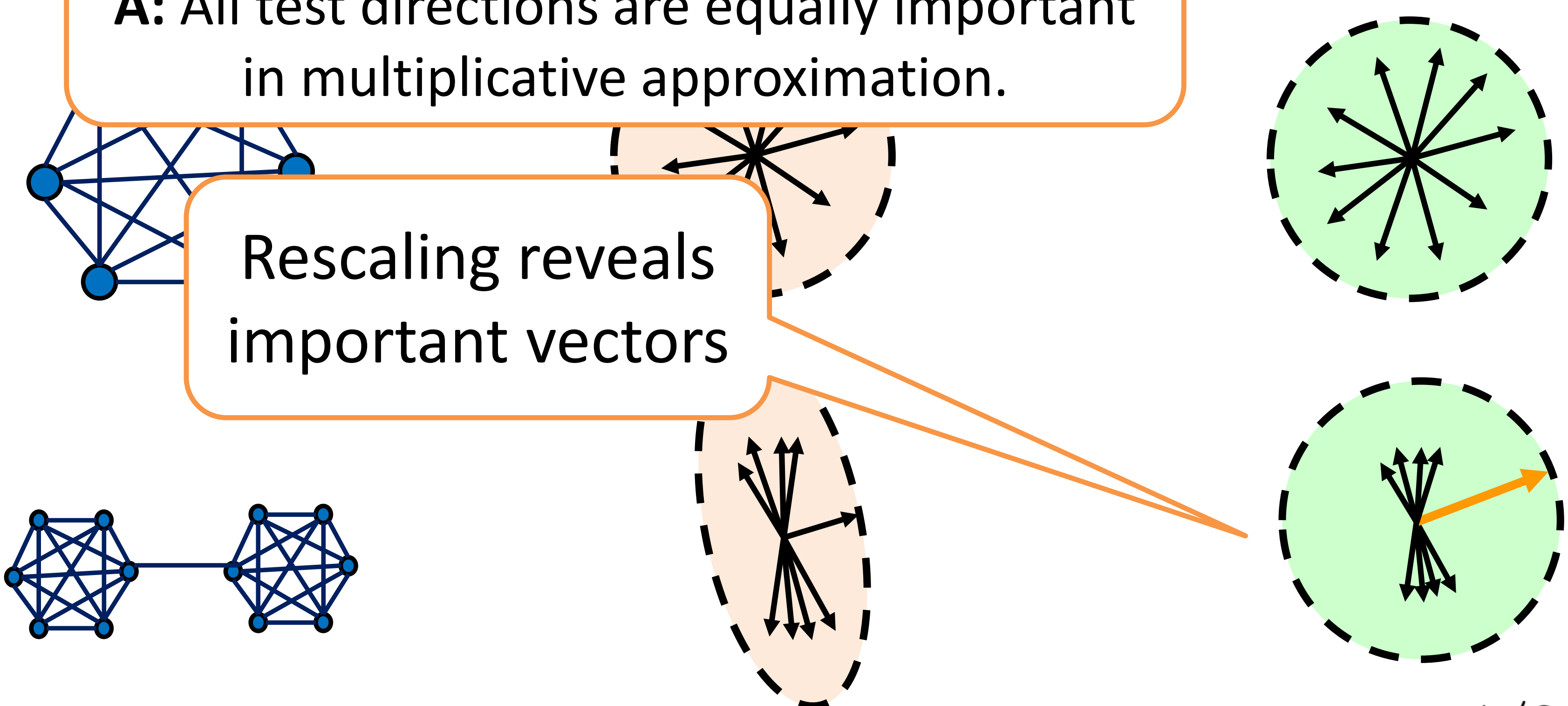


$$v_e = L_G^{-1/2} b_e$$

Examples of the Reduction

$I = \sum_e v_e v_e^T$

Q: Why rescale to identity?
A: All test directions are equally important in multiplicative approximation.



Rescaling reveals important vectors

$$v_e = L_G^{-1/2} b_e$$

Effective Resistance View

For a graph ***G***, the vectors are $v_e = L_G^{-1/2} b_e$

Lengths of vectors are:

$$\boxed{\|v_e\|^2 = \|L_G^{-1/2} b_e\|^2 = b_e^T L_G^{-1} b_e}$$

Effective Resistance View

For a graph ***G***, the vectors are $v_e = L_G^{-1/2} b_e$

Lengths of vectors are:

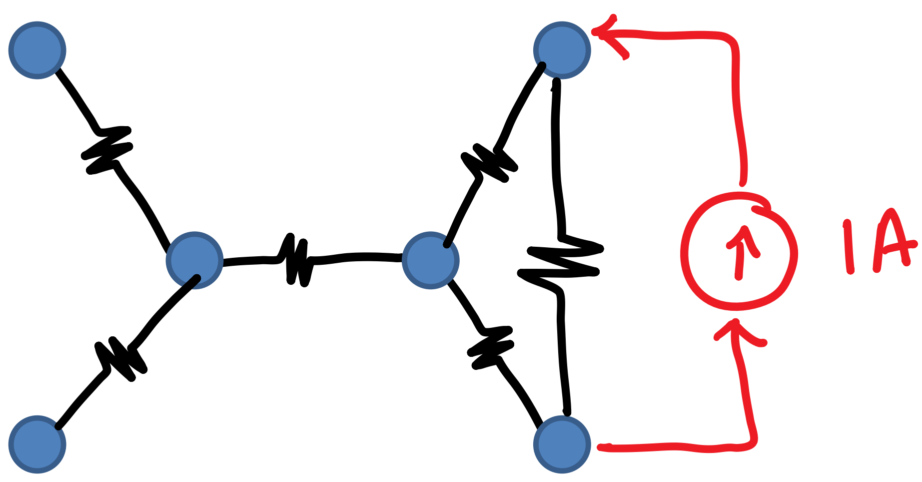
$$\boxed{\|v_e\|^2 = \|L_G^{-1/2} b_e\|^2 = b_e^T L_G^{-1} b_e = \mathbf{Reff}_G(e)}$$

Effective Resistance View

For a graph ***G***, the vectors are $v_e = L_G^{-1/2} b_e$

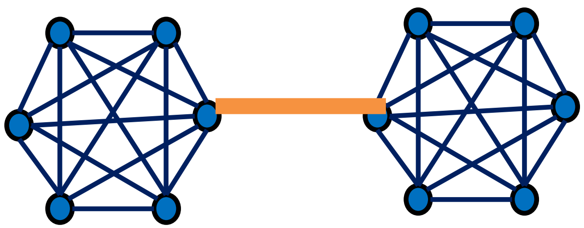
Lengths of vectors are:

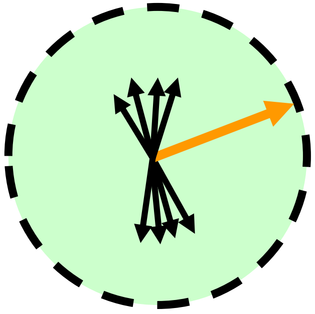
$$\|v_e\|^2 = \|L_G^{-1/2} b_e\|^2 = b_e^T L_G^{-1} b_e = \mathbf{Reff}_G(e)$$



Confirmation of Electrical Intuition

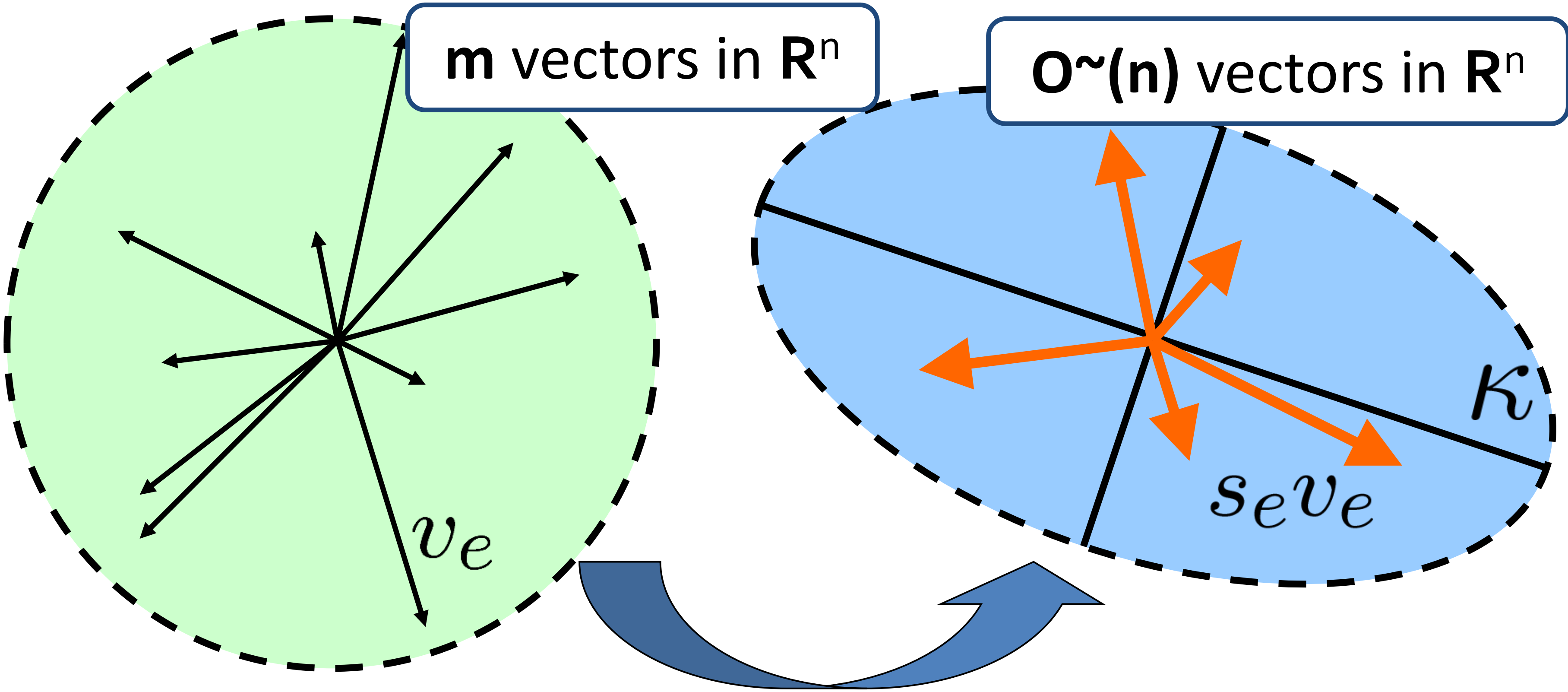
- Want ***G*** an ***H*** to be electrically equivalent
- Edges with higher **Reff** are more electrically significant = have higher norm after rescaling



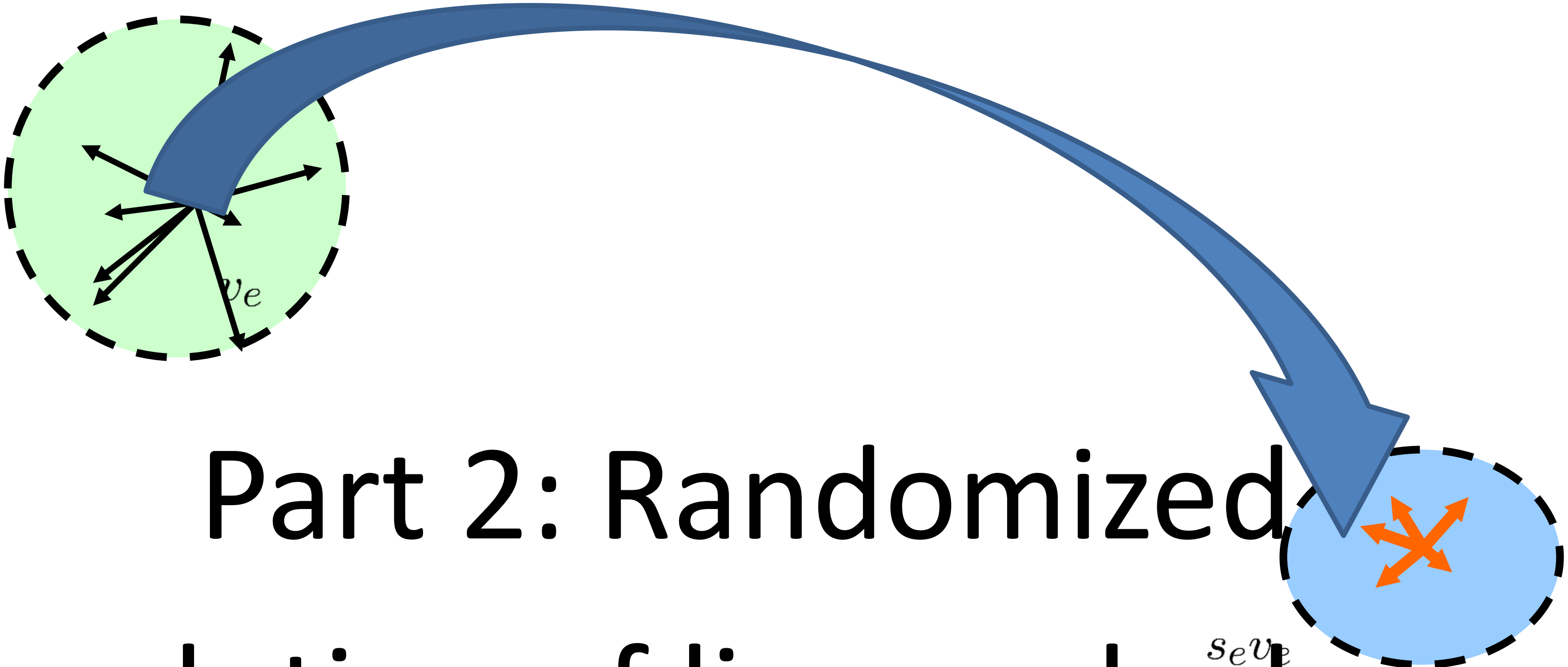


$$v_e = L_G^{-1/2} b_e$$

Core Problem

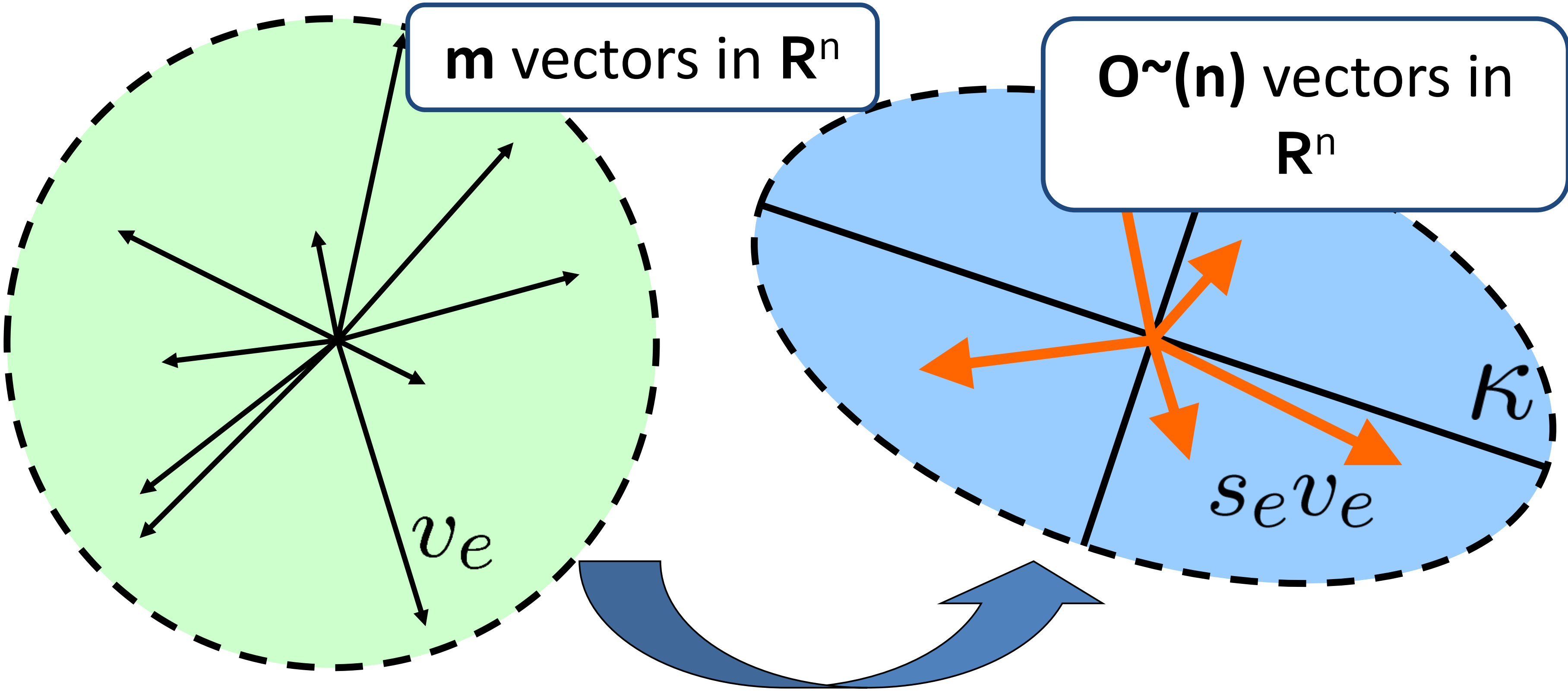


$$I = \sum_{i \le m} v_i v_i^T \qquad I \preceq \sum_i s_i v_i v_i^T \preceq \kappa I$$



Part 2: Randomized solution of linear algebra problem

Core Problem



$$I = \sum_{i \le m} v_i v_i^T \qquad I \preceq \sum_i s_i v_i v_i^T \preceq \kappa I$$

Approximating the Identity

Given $\sum_i v_i v_i^T = I$, consider the random matrix

$$X = \frac{v_i v_i^T}{p_i} \qquad \text{with probability } \; p_i$$

Then $\mathbb{E}X = \sum_i v_i v_i^T = I$.

Take k i.i.d. samples $X_1, \dots, X_k$. Would like

$$(1-\epsilon)I \preccurlyeq \frac{1}{k}\sum_i X_i \preccurlyeq (1+\epsilon)I$$

The Chernoff Bound

Suppose $X_1, \ldots, X_k$ are i.i.d. random variables with

$$0 \le X_i \le M \qquad \text{and} \qquad \mathbb{E}X_i = 1.$$

Then

$$\mathbb{P}\left[\left|\frac{1}{k}\sum_i X_i - 1\right| \ge \epsilon\right] \le 2\exp\left(-\frac{k\epsilon^2}{4M}\right)$$

The Chernoff Bound

$k = 4M/\epsilon^2$ samples give

$$\frac{1}{k}\sum_i X_i \approx_\epsilon 1$$

with constant probability.

dom variables with

$$\mathbb{E}X_i = 1.$$

Then

$$\mathbb{P}\left[\left|\frac{1}{k}\sum_i X_i - 1\right| \geq \epsilon\right] \leq 2\exp\left(-\frac{k\epsilon^2}{4M}\right)$$

The Chernoff Bound

Suppose $X_1, \ldots, X_k$ are i.i.d. random variables with

$0 \le X_i \le M$ and $\mathbb{E}X_i = 1.$

Then

$$\mathbb{P}\left[\left|\frac{1}{k}\sum_i X_i - 1\right| \ge \epsilon\right] \le 2\exp\left(-\frac{k\epsilon^2}{4M}\right)$$

The **Matrix** Chernoff Bound
[Rudelson'99, AW'02, Tropp'11]

Suppose $X_1, \dots, X_k$ are i.i.d. random $d \times d$ **matrices** with

$0 \preccurlyeq X_i \preccurlyeq M \cdot I$ and $\mathbb{E} X_i = I$.

Then

$$\mathbb{P}\left[\left\|\frac{1}{k}\sum_i X_i - I\right\| \geq \epsilon\right] \leq 2\boldsymbol{d}\exp\left(-\frac{k\epsilon^2}{4M}\right)$$

The **Matrix** Chernoff Bound
[Rudelson'99, AW'02, Tropp'11]

$k = 4M \log d / \epsilon^2$ samples give

$$\frac{1}{k}\sum_i X_i \approx_\epsilon I$$

with constant probability.

dom $d \times d$

$\mathbb{E}X_i = I$.

Then

$$\mathbb{P}\left[\left\|\frac{1}{k}\sum_i X_i - I\right\| \geq \epsilon\right] \leq 2\boldsymbol{d} \exp\left(-\frac{k\epsilon^2}{4M}\right)$$

The **Matrix** Chernoff Bound
[Rudelson'99, AW'02, Tropp'11]

Suppose $X_1, \ldots, X_k$ are i.i.d. random $d \times d$ **matrices** with

$0 \preccurlyeq X_i \preccurlyeq M \cdot I$ and $\mathbb{E} X_i = I$.

Then

$$\mathbb{P}\left[\left\|\frac{1}{k}\sum_i X_i - I\right\| \geq \epsilon\right] \leq 2\boldsymbol{d} \exp\left(-\frac{k\epsilon^2}{4M}\right)$$

In our case

$$X = \frac{v_i v_i^T}{p_i} \qquad \text{with prob. } p_i, \qquad \mathbb{E}X = I.$$

Want to minimize $M = \max_i \left\| \frac{v_i v_i^T}{p_i} \right\| = \max_i \frac{||v_i||^2}{p_i}$

To make this this tight for all v_i set $p_i = \frac{||v_i||^2}{M}$.

In our case

$$X = \frac{v_i v_i^T}{p_i} \quad \text{with prob. } p_i, \qquad \mathbb{E}X = I.$$

Want to minimize $M = \max_i \left\| \frac{v_i v_i^T}{p_i} \right\| = \max_i \frac{||v_i||^2}{p_i}$

To make this this tight for all v_i set $p_i = \frac{||v_i||^2}{M}$.

But $\sum_i p_i = \sum_i \frac{||v_i||^2}{M}$

In our case

$$X = \frac{v_i v_i^T}{p_i} \quad \text{with prob. } p_i, \qquad \mathbb{E}X = I.$$

Want to minimize $M = \max_i \left\| \frac{v_i v_i^T}{p_i} \right\| = \max_i \frac{||v_i||^2}{p_i}$

To make this this tight for all v_i set $p_i = \frac{||v_i||^2}{M}$.

But $\sum_i p_i = \sum_i \frac{||v_i||^2}{M} = \sum_i \frac{Tr\left(v_i v_i^T\right)}{M}$

In our case

$$X = \frac{v_i v_i^T}{p_i} \qquad \text{with prob. } p_i, \qquad \mathbb{E}X = I.$$

Want to minimize $M = \max_i \left\| \frac{v_i v_i^T}{p_i} \right\| = \max_i \frac{||v_i||^2}{p_i}$

To make this this tight for all v_i set $p_i = \frac{||v_i||^2}{M}$.

But $\sum_i p_i = \sum_i \frac{||v_i||^2}{M} = \sum_i \frac{Tr\left(v_i v_i^T\right)}{M} = \frac{Tr(\sum_i v_i v_i^T)}{M}$

In our case

$$X = \frac{v_i v_i^T}{p_i} \quad \text{with prob. } p_i, \quad \mathbb{E}X = I.$$

Want to minimize $M = \max_i \left\| \frac{v_i v_i^T}{p_i} \right\| = \max_i \frac{||v_i||^2}{p_i}$

To make this this tight for all v_i set $p_i = \frac{||v_i||^2}{M}$.

But $\sum_i p_i = \sum_i \frac{||v_i||^2}{M} = \sum_i \frac{Tr\left(v_i v_i^T\right)}{M} = \frac{Tr(\sum_i v_i v_i^T)}{M} = \frac{n}{M}$

In our case

$$X = \frac{v_i v_i^T}{p_i} \quad \text{with prob. } p_i, \qquad \mathbb{E}X = I.$$

Must have $M = n$

Thm: $4n \log n/\epsilon^2$ samples suffice.

$$\ldots \Big\| = \max_i \frac{||v_i||^2}{p_i}$$

To make this this tigh... v_i set $p_i = \frac{||v_i||^2}{M}$.

But $\sum_i p_i = \sum_i \frac{||v_i||^2}{M} = \sum_i \frac{Tr\left(v_i v_i^T\right)}{M} = \frac{Tr(\sum_i v_i v_i^T)}{M} = \frac{n}{M}$

How to Approximate the Identity

Given $\sum_i v_i v_i^T = I$

Sample $n \log n / \epsilon^2$ vectors randomly with replacement, by $\boxed{p_i \propto \|v_i\|^2}$.

Set $s_i = 1/p_i$ for chosen vectors.

Rudelson'99: This works whp:

$$1 - \epsilon \preceq \sum_i s_i v_i v_i^T \preceq 1 + \epsilon$$

How to Approximate the Identity

For a graph, $p_e \propto \boldsymbol{Reff_G(e)}$

Given $\sum_i v_i v_i^T = I$

Sample $n \log n / \epsilon^2$ vectors randomly with replacement, by $p_i \propto \|v_i\|^2$.

Set $s_i = 1/p_i$ for chosen vectors.

Rudelson'99: This works whp:

$$1 - \epsilon \preceq \sum_i s_i v_i v_i^T \preceq 1 + \epsilon$$

Theorem. Every weighted graph **G** has a weighted subgraph **H** with at most $4n \log n / \epsilon^2$ edges s.t.

$$L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon) L_G.$$

Algorithm: sample $4n \log n / \epsilon^2$ edges independently according to effective resistances.

Theorem. Every weighted graph **G** has a weighted subgraph **H** with at most $9n \log n / \epsilon^2$ edges s.t.

$$L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon) L_G.$$

Moreover, H can be found in time $O^{\sim}(m/\epsilon^2)$.

Algorithm: sample $9n \log n / \epsilon^2$ edges independently according to **approximate** effective resistances.

[Spielman-S'08]

Part 3: Fast Calculation of Sampling Probabilities

Computing A Single Resistance

Recall

$$Reff_G(e) = b_e^T L_G^{-1} b_e$$

So can compute a resistance by solving $L_G x = b_e$

Computing A Single Resistance

Recall

$$Reff_G(e) = b_e^T L_G^{-1} b_e$$

So can compute a resistance by solving $L_G x = b_e$

Amazing Theorem [Spielman-Teng'04…] Can solve linear systems in L_G in time $O(mlogn)$.

Computing A Single Resistance

Recall

$$Reff_G(e) = b_e^T L_G^{-1} b_e$$

So can compute a resistance by solving $L_G x = b_e$

Amazing Theorem [Spielman-Teng'04…] Can solve linear systems in L_G in time $O(mlogn)$.

But need to compute all resistances…

Resistances are Distances

Outer product expansion:

$$L_G = \sum_e b_e b_e^T = B^T B \qquad \text{for rows}(B) = \{b_e^T\}$$

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 \\ & & \vdots & & \\ & & \vdots & & \\ 1 & 0 & -1 & 0 & 0 \end{pmatrix}$$

Signed edge-vertex incidence matrix

Resistances are Distances

Outer product expansion:

$$L_G = \sum_e b_e b_e^T = B^T B \qquad \text{for rows}(B) = \{b_e^T\}$$

Sampling probabilities:

$$\begin{aligned} \|v_e\|^2 &= b_e^T L_G^{-1} b_e \\ &= b_e^T L_G^{-1} B^T B L_G^{-1} b_e \\ &= \|B L_G^{-1}(\delta_i - \delta_j)\|^2 \qquad \text{for } e = ij. \end{aligned}$$

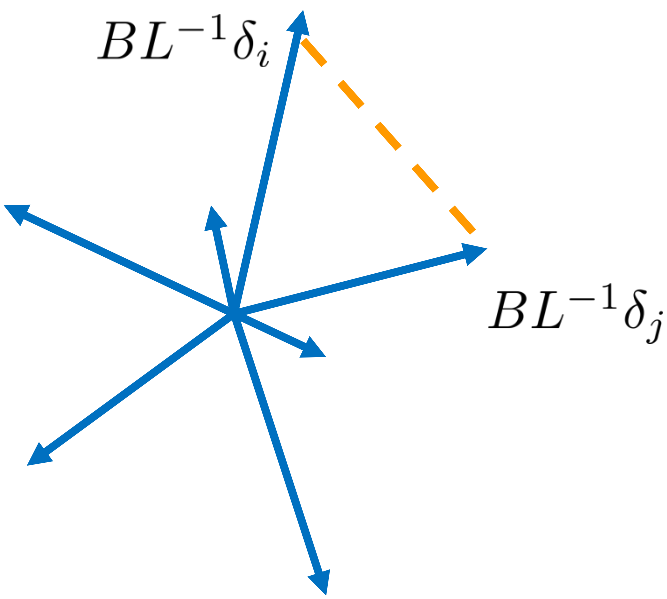
Nearly Linear Time

$$\boxed{\mathbf{Reff}(ij) = \|BL^{-1}(\delta_i - \delta_j)\|^2}$$

Nearly Linear Time

$$\boxed{\mathbf{Reff}(ij) = \|BL^{-1}(\delta_i - \delta_j)\|^2}$$

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



Dimension Reduction

Johnson-Lindenstrauss Lemma [JL'84]:

Suppose $x_1, \dots, x_n$ are points in $\mathbb{R}^d$.

Let $Q_{k\times n}$ be a random k $-$dimensional projection.

Then

$$||Qx_i - Qx_j||_2 = (1 \pm \epsilon)||x_i - x_j||_2$$

With high probability as long as

$$k \geq 10 \log n / \epsilon^2$$

Dimension Reduction

Johnson-Lindenstrauss Lemma [JL'84]:

Suppose $x_1, \ldots, x_n$ are points in $\mathbb{R}^d$.

Let $Q_{k \times n}$ be a random Bernoulli matrix.

Then

$$||Qx_i - Qx_j||_2 \propto (1 \pm \epsilon)||x_i - x_j||_2$$

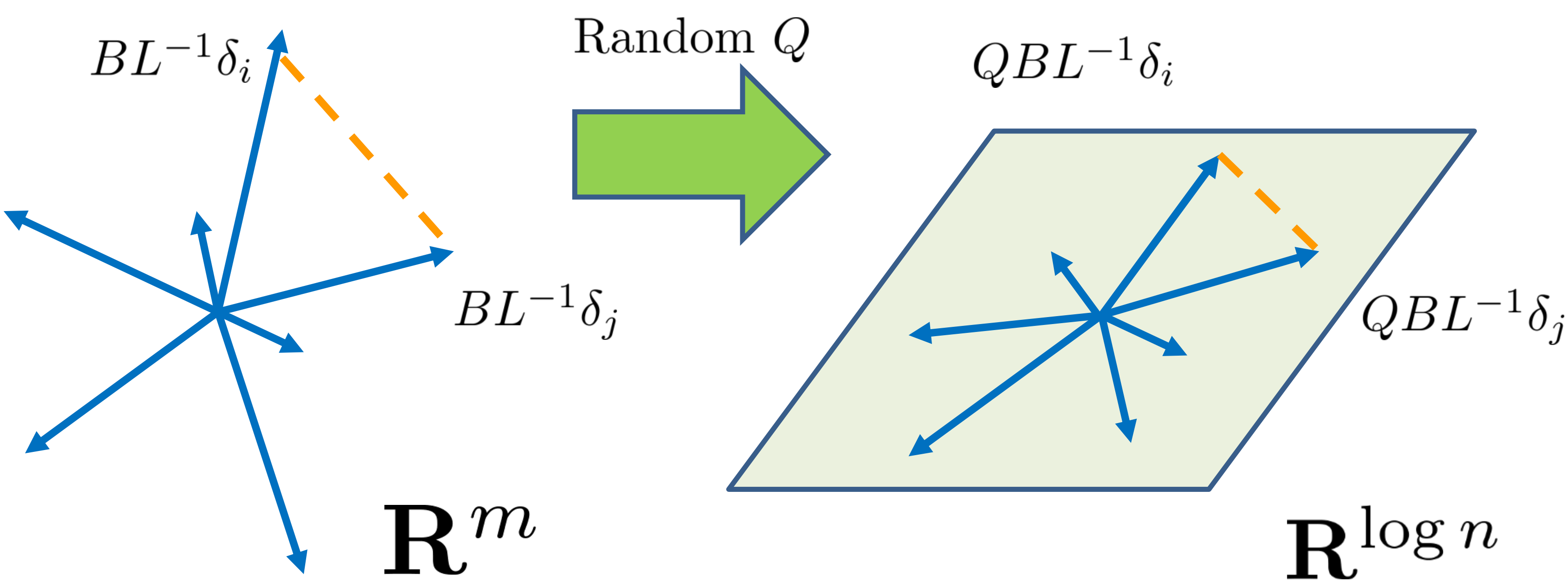
With high probability as long as

$$k \geq 10 \log n / \epsilon^2$$

Johnson-Lindenstrauss with $\epsilon = 1/2$

$$\boxed{\mathbf{Reff}(ij) = \|BL^{-1}(\delta_i - \delta_j)\|^2}$$

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



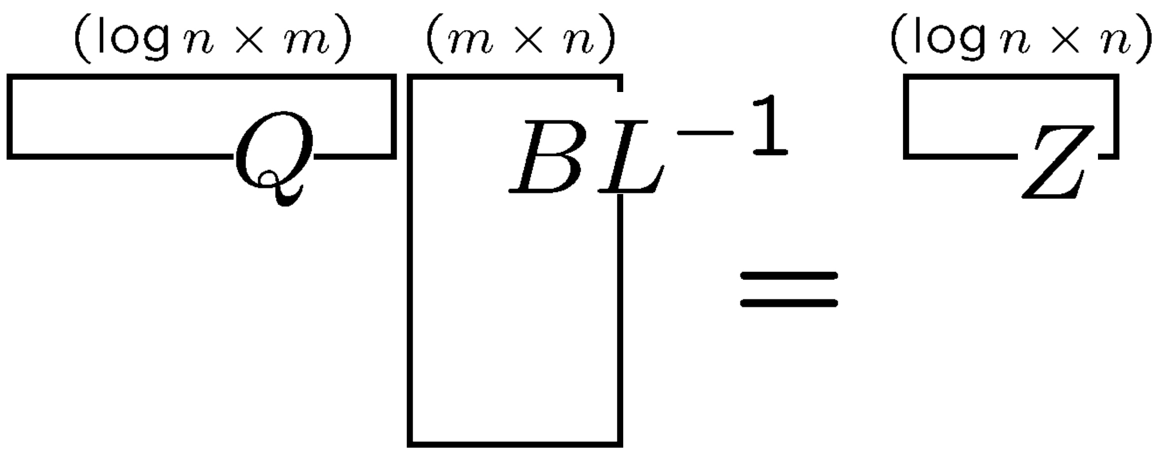
Nearly Linear Time

$$\boxed{\mathbf{Reff}(ij) = \|BL^{-1}(\delta_i - \delta_j)\|^2}$$

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

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

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

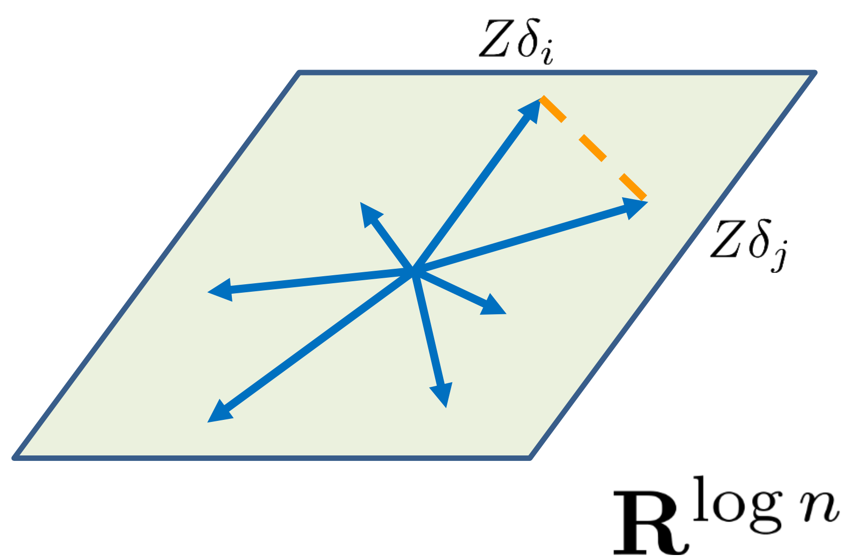


Nearly Linear Time

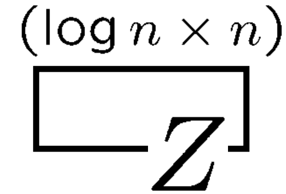
$(\log n \times n)$

Z

$$\mathbf{Reff}(ij) \sim \|Z(\delta_i - \delta_j)\|^2$$



Nearly Linear Time



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

$Z=QBL^{-1}$

$$\boxed{\mathbf{Reff}(ij) \sim \|Z(\delta_i - \delta_j)\|^2}$$

$ZL=QB$

$z_iL=(QB)_i$

Nearly Linear Time

$(\log n \times n)$

Z

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

$Z=QBL^{-1}$

$ZL=QB$

$z_i L=(QB)_i$

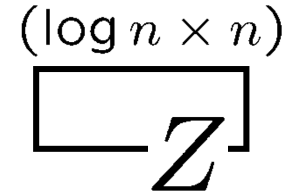
$$\boxed{\mathbf{Reff}(ij) \sim \|Z(\delta_i - \delta_j)\|^2}$$

Solve ***O(logn)*** linear systems in ***L*** using

fast Laplacian solver solver

learns all pairwise resistances by probing a few random electrical flows.

Nearly Linear Time



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

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

$$\boxed{\mathbf{Reff}(ij) \sim \|Z(\delta_i - \delta_j)\|^2}$$

$\boldsymbol{ZL=QB}$

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

Solve ***O(logn)*** linear systems in ***L*** using

fast Laplacian solver solver

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

(only change M by a constant factor)

□

Actual Algorithm

Input: undirected graph $G = (V, E, w)$

Output: subgraph **H** with $L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon)L_G$

1. Let $Q_{\log n \times m}$ be a scaled random projection.

   Compute approximate resistance matrix

   $$Z = QBL^{+}$$

   by solving $\log n$ Laplacian systems

2. Repeat the following $9nlogn/\epsilon^2$ times:

   choose edge $e = ij$ w.p. $p_e \propto ||Z(\delta_i - \delta_j)||^2$

   add e to H with weight $s_e = 1/p_e$

Actual Algorithm

Input: undirected graph $G = (V, E, w)$

Output: subgraph **H** with $L_G \preccurlyeq L_H \preccurlyeq (1+\epsilon)L_G$

1. Let $Q_{\log n \times m}$ be a scaled random projection.

   Compute approximate resistance matrix

$$Z = QBL^{+}$$

   by solving $\log n$ Laplacian systems

2. Repeat the following $9nlogn/\epsilon^2$ times:

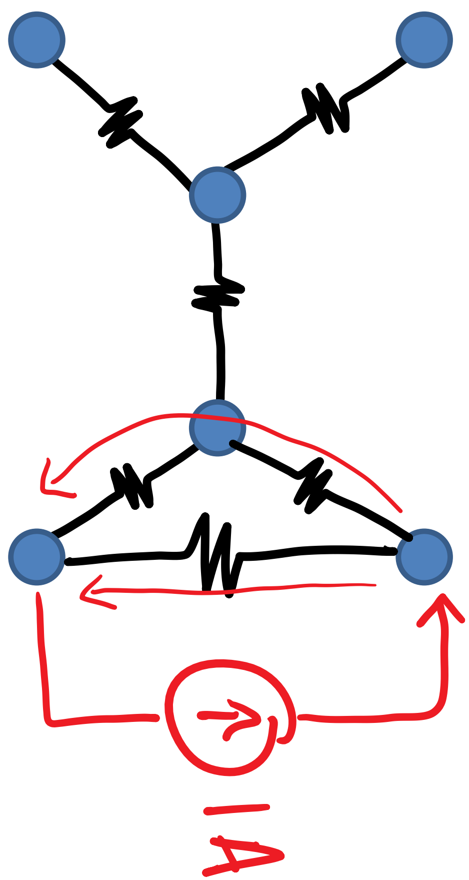
   choose edge $e = ij$ w.p. $p_e \propto ||Z(\delta_i - \delta_j)||^2$

   add e to H with weight $s_e = 1/p_e$

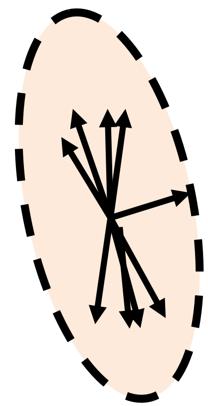
+improvements by [Koutis-Levin-Peng'12]

Two Useful Ways to view a Graph

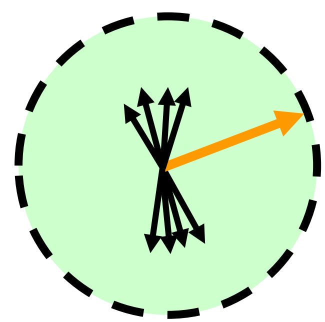
electrical network



bunch of vectors



$$L_G = \sum_e b_e b_e^T$$

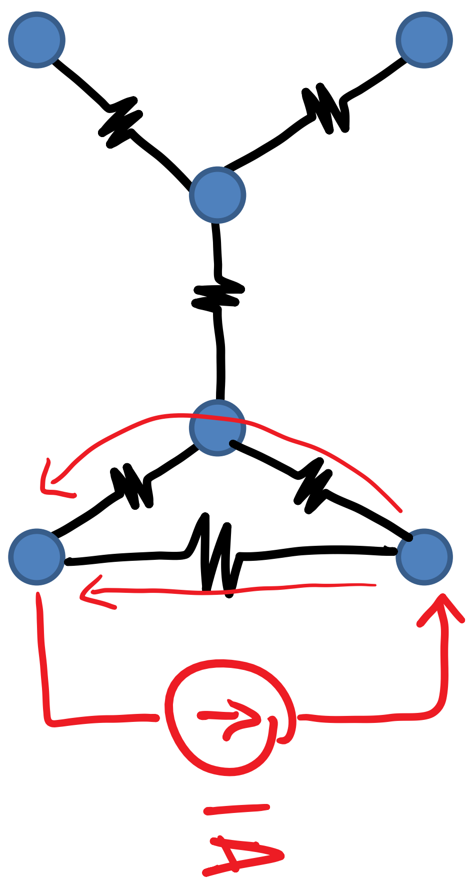


$$I = \sum_e v_e v_e^T$$

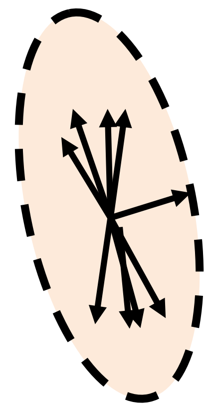
Tw … ph

$$Reff(e) = ||L_G^{-1/2} b_e||^2 = ||v_e||^2$$

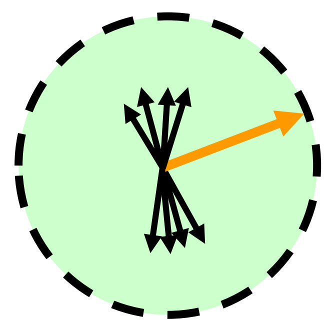
electrical network



bunch of vectors



$$L_G = \sum_e b_e b_e^T$$



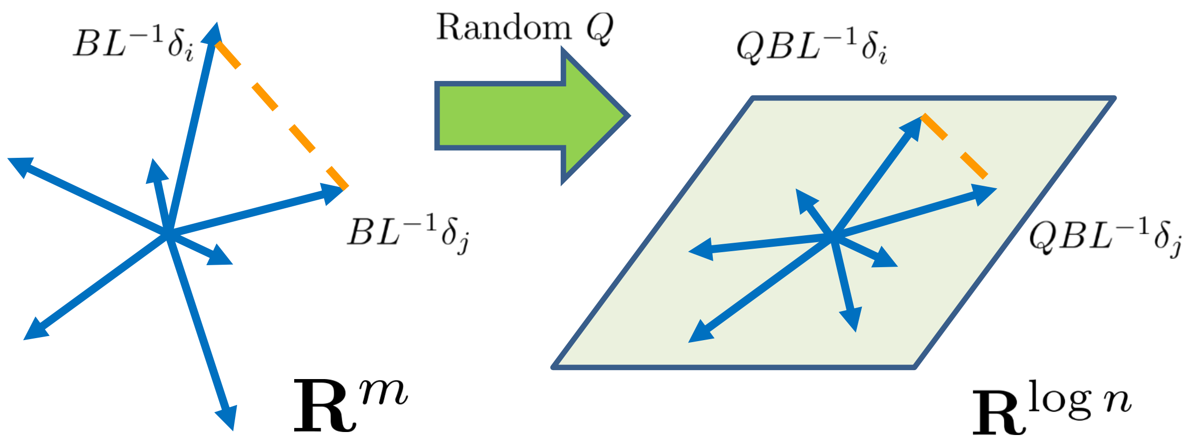
$$I = \sum_e v_e v_e^T$$

Two Useful Tools

Matrix Chernoff Bound

$$\mathbb{P}\left[\left\|\frac{1}{k}\sum_i X_i - I\right\| \geq \epsilon\right] \leq 2d \exp\left(-\frac{k\epsilon^2}{4M}\right)$$

Johnson-Lindenstrauss Lemma



Advantages over pure combinatorics

There is a global **rescaling** transformation:

$$L_G \approx L_H \quad \textbf{iff} \quad L_G^{-1/2} L_H L_G^{-1/2} \approx I$$

Powerful **random matrix** tools apply naturally:

1. Matrix Chernoff bound
2. Johnson-Lindenstrauss Lemma

Some Improvements

[Batson-Spielman-S'08]: $\frac{4n}{\epsilon^2}$ edges

[Koutis-Levin-Peng'12] $O\left(\frac{m \log^2 n}{\epsilon^2}\right)$ running time

[Koutis'14] parallel algorithm

[Kapralov, Lee, Musco x2, Sidford'14]

1-pass dynamic streaming algorithm

[Lee-Sun'17]: O(mlog(n)) time with O(n/eps^2) edges

Code

http://www.cs.cmu.edu/~jkoutis/SpectralAlgorithms.htm