Composable Core-sets for Determinant Maximization Problems via Spectral Spanners

Piotr Indyk
MIT

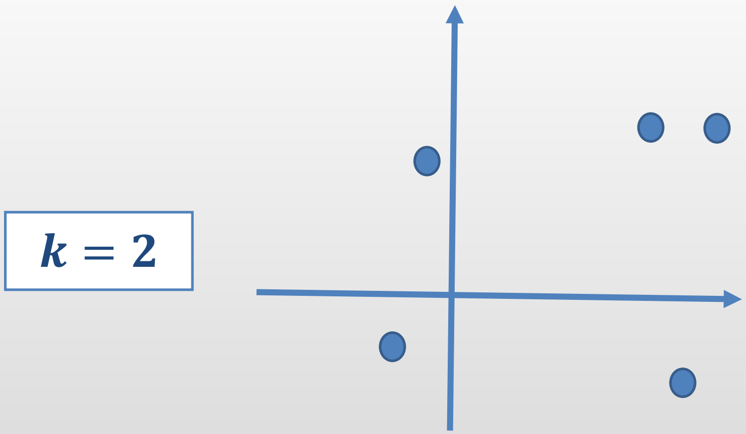
Sepideh Mahabadi
TTIC

Shayan Oveis Gharan
U. of Washington

Alireza Rezaei
U. of Washington

Volume (Determinant) Maximization Problem

Input: a set of n vectors $\mathrm{V} \in \mathbb{R}^d$ and a parameter $k \leq d$,



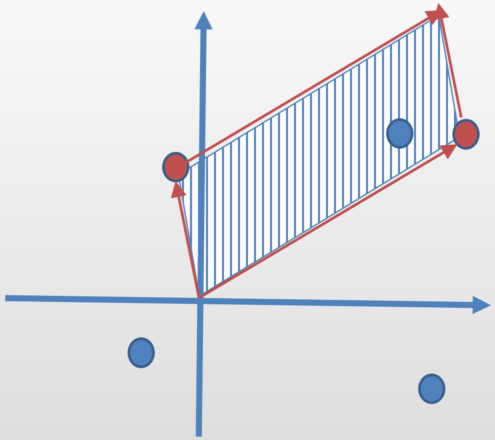
Volume (Determinant) Maximization Problem

Input: a set of n vectors $\mathrm{V} \in \mathbb{R}^d$ and a parameter $k \le d$,

Output: a subset $S \subset V$ of size k with the maximum volume

- Parallelepiped spanned by the points in S

$\boldsymbol{k = 2}$



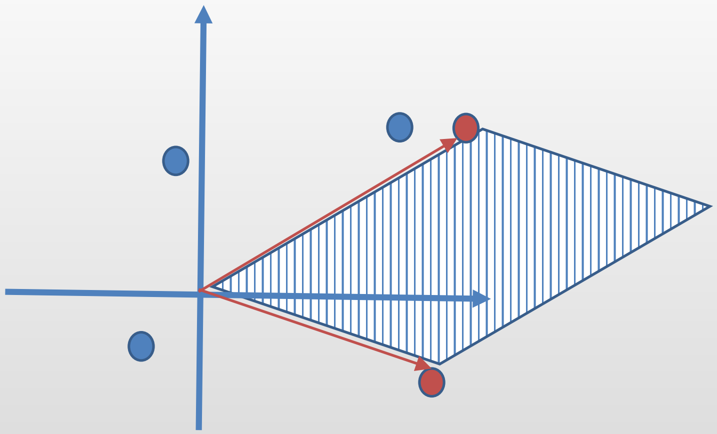
Volume (Determinant) Maximization Problem

Input: a set of n vectors $\mathrm{V} \in \mathbb{R}^d$ and a parameter $k \leq d$,

Output: a subset $S \subset V$ of size k with the maximum volume

- Parallelepiped spanned by the points in S

$k = 2$



Determinant (Volume) Maximization Problem

Input: a set of n vectors $\mathrm{V} \in \mathbb{R}^d$ and a parameter $k \leq d$,

Output: a subset $S \subset V$ of size k with the maximum volume

- Parallelepiped spanned by the points in S

$$\begin{bmatrix} v_1 \ v_2 \dots v_n \end{bmatrix}$$

$$\mathbb{V}$$

Equivalent Formulation:

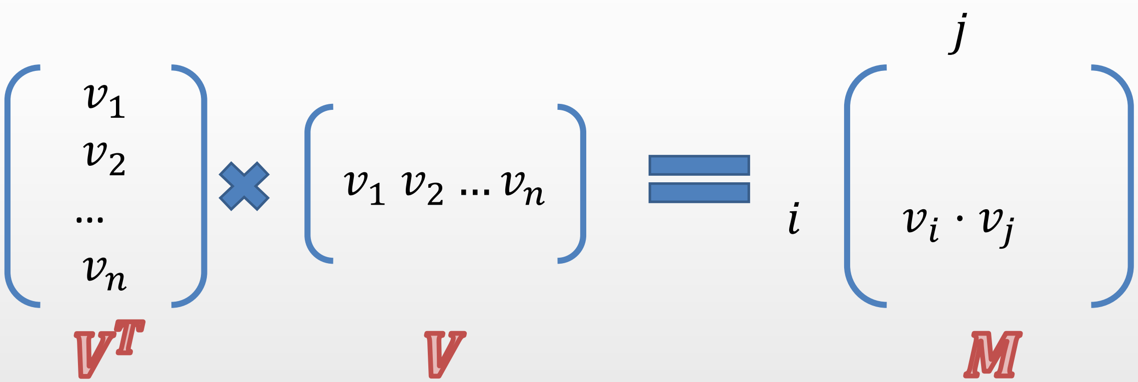
Reuse V to denote the matrix where its columns are the vectors in V

Determinant (Volume) Maximization Problem

Input: a set of n vectors $V \in \mathbb{R}^d$ and a parameter $k \leq d$,

Output: a subset $S \subset V$ of size k with the maximum volume

- Parallelepiped spanned by the points in S



Equivalent Formulation:

Reuse V to denote the matrix where its columns are the vectors in V

- Let M be the gram matrix $V^T V$

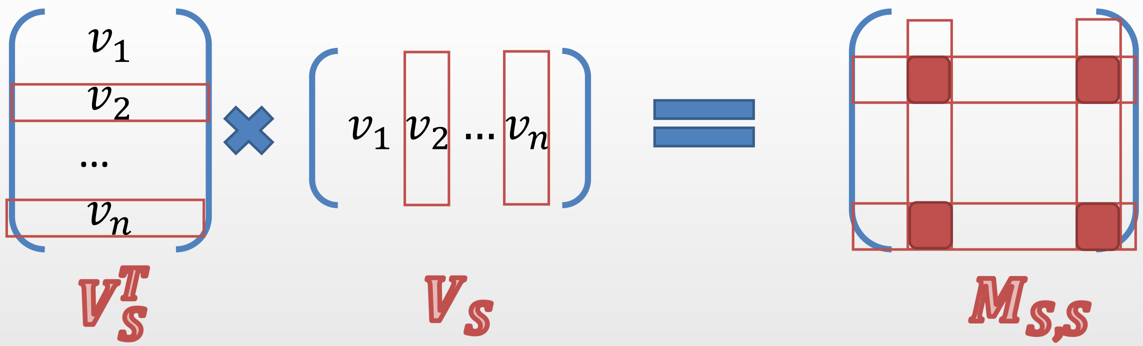
$M_{i,j} = v_i \cdot v_j$

Determinant (Volume) Maximization Problem

Input: a set of n vectors $\mathrm{V} \in \mathbb{R}^d$ and a parameter $k \leq d$,

Output: a subset $S \subset V$ of size k with the maximum volume

- Parallelepiped spanned by the points in S



Equivalent Formulation:

Reuse V to denote the matrix where its columns are the vectors in V

- Let M be the gram matrix $V^T V$
- Choose S such that $\det(M_{S,S})$ is maximized

$M_{i,j} = v_i \cdot v_j$

$\det(M_{S,S}) = Vol(S)^2$

What is known?

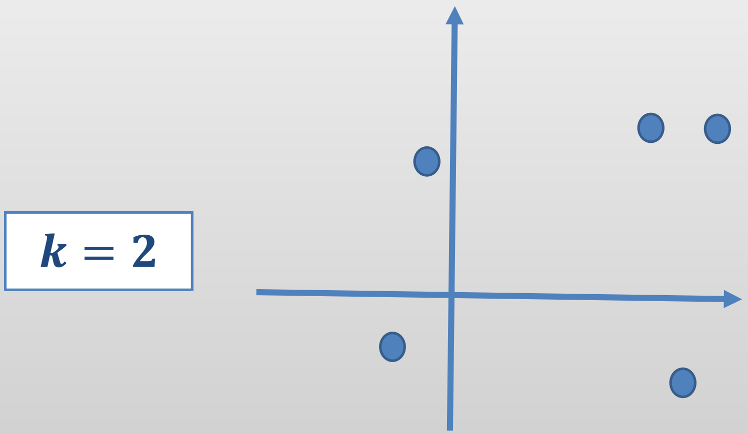
- Hard to approximate within a factor of 2^{ck} [CMI'13]

What is known?

- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^{k}-approximation [Nik'15]

What is known?

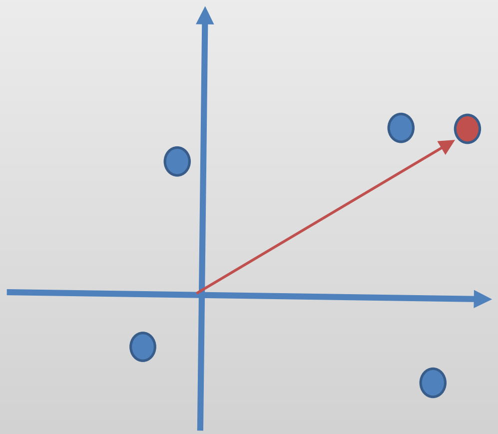
- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^k-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U



What is known?

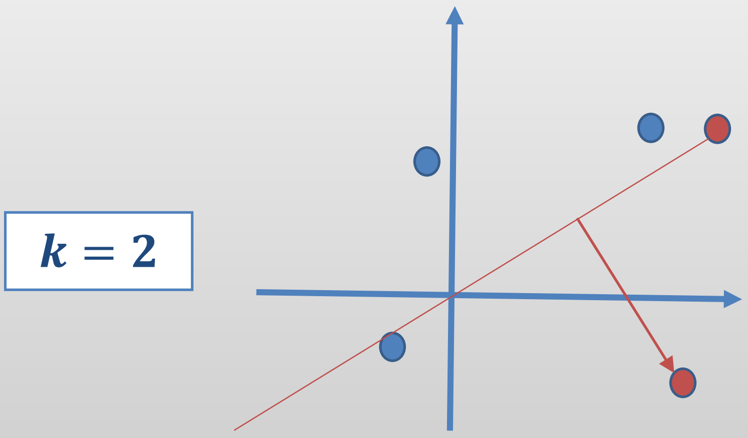
- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^k-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U

$\boldsymbol{k = 2}$



What is known?

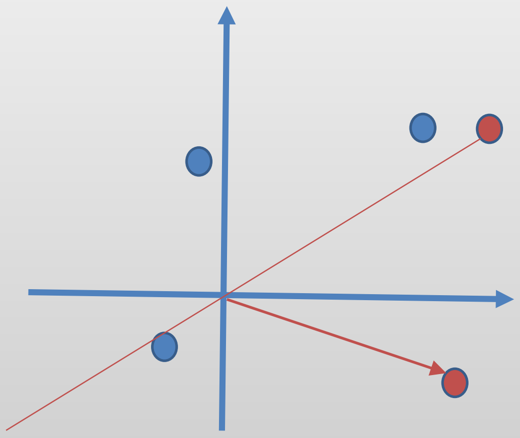
- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^{k}-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U



What is known?

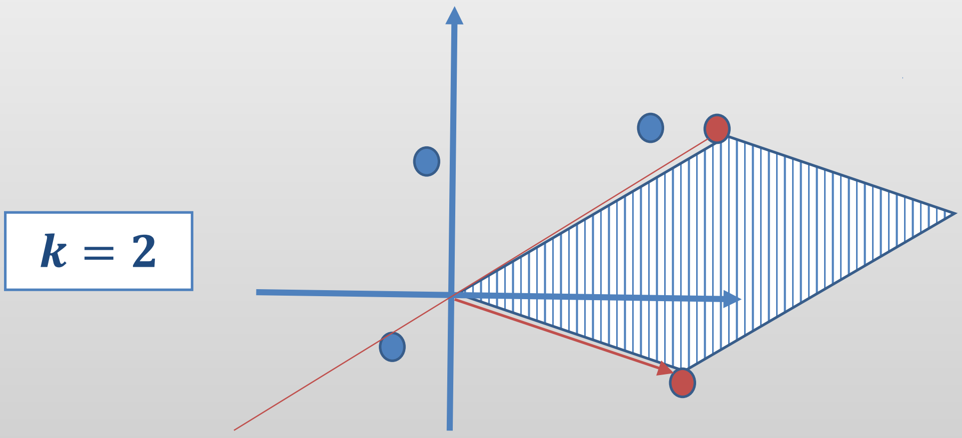
- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^k-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U

$k = 2$



What is known?

- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^k-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U

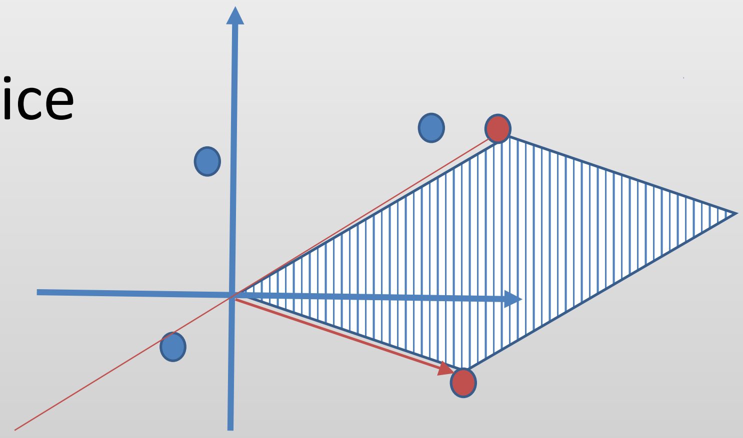


What is known?

- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^{k}-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U

- Greedy performs very well in practice

$\boldsymbol{k = 2}$



Determinantal Point Processes (DPP)

DPP: Very popular probabilistic model, where given a set of vectors V, **samples** any k-subset S with probability proportional to this determinant.

- Maximum a posteriori (MAP) decoding is determinant maximization
- Volume/determinant is a notion of **diversity**

Determinantal Point Processes (DPP)

DPP: Very popular probabilistic model, where given a set of vectors V, **samples** any k-subset S with probability proportional to this determinant.

- Maximum a posteriori (MAP) decoding is determinant maximization
- Volume/determinant is a notion of **diversity**

Applications

[MJK'17,GCGS'14] Video summarization

[KT+'12, CGGS'15,KT'11] Document summarization

[YFZ+'16] Tweet generation

[LCYO'16] Object detection

....

Determinantal Point Processes (DPP)

DPP: Very popular probabilistic model, where given a set of vectors V, **samples** any k-subset S with probability proportional to this determinant.

- Maximum a posteriori (MAP) decoding is determinant maximization
- Volume/determinant is a notion of **diversity**

Applications

[MJK'17,GCGS'14] Video summarization

[KT+'12, CGGS'15,KT'11] Document summarization

[YFZ+'16] Tweet generation

[LCYO'16] Object detection

....

- Most applications deal with **massive data**
- Lots of effort for solving the problem in massive data models of computation [MJK'17, WIB'14, PJG+'14, MKSK'13, MKBK'15, MZ'15, MZ'15, BENW'15]
- e.g. streaming, distributed, parallel

Determinantal Point Processes (DPP)

DPP: Very popular probabilistic model, where given a set of vectors V, **samples** any k-subset S with probability proportional to this determinant.

- Maximum a posteriori (MAP) decoding is determinant maximization
- Volume/determinant is a notion of **diversity**

Applications

[MJK'17,GCGS'14] Video summarization

[KT+'12, CGGS'15,KT'11] Document summarization

[YFZ+'16] Tweet generation

[LCYO'16] Object detection

....

- Most applications deal with **massive data**
- Lots of effort for solving the problem in massive data models of computation [MJK'17, WIB'14, PJG+'14, MKSK'13, MKBK'15, MZ'15, MZ'15, BENW'15]
- e.g. streaming, distributed, parallel

Composable Core-sets

Composable Core-sets

Core-sets [AHV’05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Solving the problem over $\boldsymbol{U}$ gives a good approximation of solving the problem over $\boldsymbol{V}$

Composable Core-sets

Core-sets [AHV'05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

Composable Core-sets

Core-sets [AHV'05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

Let $\boldsymbol{f}$ be an optimization function

–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

Composable Core-sets

Core-sets [AHV'05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

Let $\boldsymbol{f}$ be an optimization function

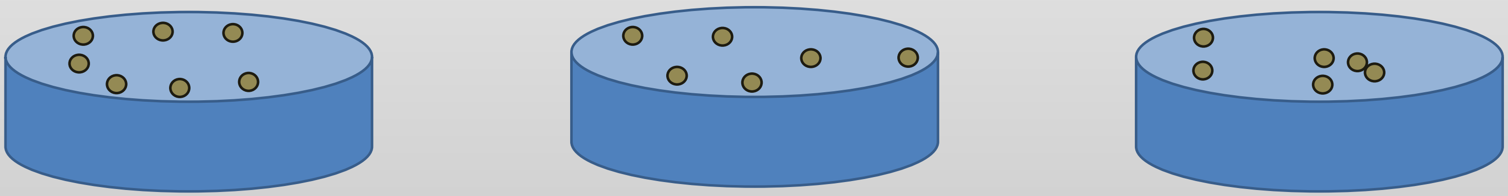
–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

i.e. for multiple data sets $\boldsymbol{V_1}, \cdots, \boldsymbol{V_m}$ and their coresets $\boldsymbol{U_1}, \cdots, \boldsymbol{U_m}$,

$\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$



Composable Core-sets

Core-sets [AHV'05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

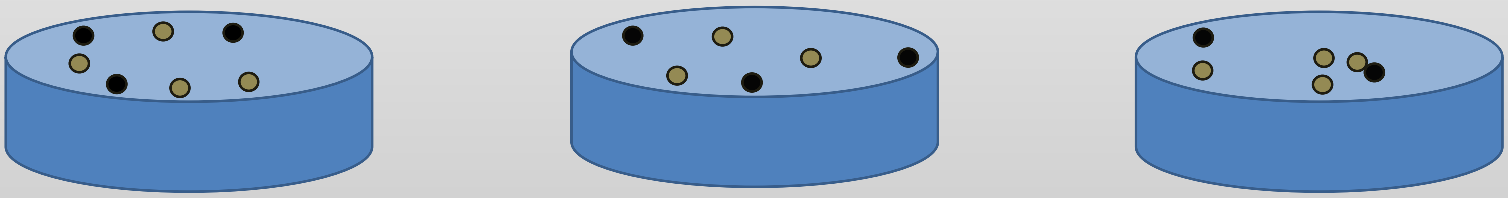
Let $\boldsymbol{f}$ be an optimization function

–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

i.e. for multiple data sets $\boldsymbol{V_1}, \cdots, \boldsymbol{V_m}$ and their coresets $\boldsymbol{U_1}, \cdots, \boldsymbol{U_m}$, $\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$



Composable Core-sets

Core-sets [AHV'05]**:** a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

Let $\boldsymbol{f}$ be an optimization function

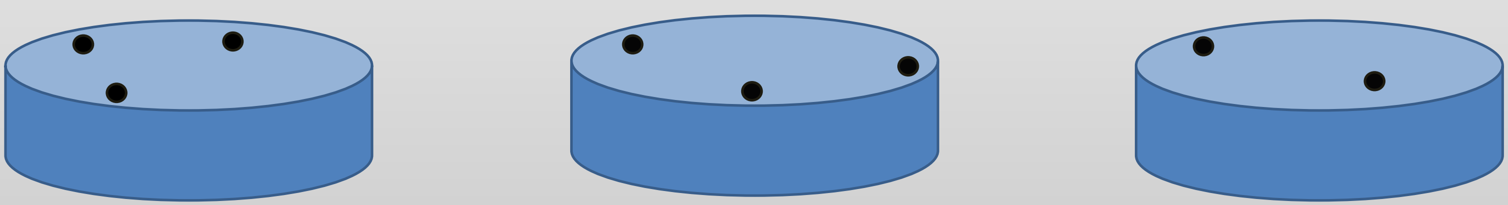
–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

i.e. for multiple data sets $\boldsymbol{V_1}, \cdots, \boldsymbol{V_m}$ and their coresets $\boldsymbol{U_1}, \cdots, \boldsymbol{U_m}$,

$\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$



Composable Core-sets

Core-sets [AHV'05]: a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

Let $\boldsymbol{f}$ be an optimization function

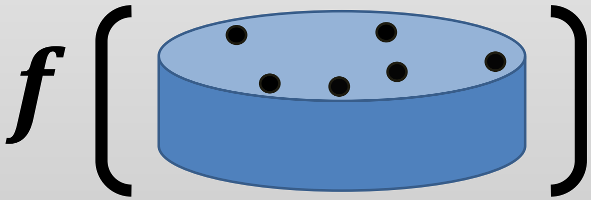
–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

i.e. for multiple data sets $\boldsymbol{V_1, \cdots, V_m}$ and their coresets $\boldsymbol{U_1, \cdots, U_m}$,

$\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$



Composable Core-sets

Core-sets [AHV'05]: a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAIMV'13 and IMMM'14]:

Let $\boldsymbol{f}$ be an optimization function

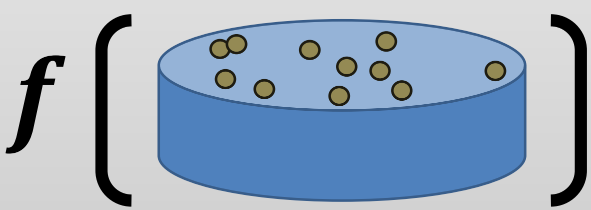
–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

i.e. for multiple data sets $\boldsymbol{V_1, \cdots, V_m}$ and their coresets $\boldsymbol{U_1, \cdots, U_m}$,

$\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$



Composable Core-sets

Core-sets [AHV'05]: a subset $\boldsymbol{U}$ of the data $\boldsymbol{V}$ that represents it well

Composable Core-sets [AAI**M**V'13 and I**M**MM'14]**:**

Let $\boldsymbol{f}$ be an optimization function

–E.g. $\boldsymbol{f}(\boldsymbol{V})$ is the solution to k determinant maximization

A subset $\boldsymbol{U} \subset \boldsymbol{V}$ is called composable coreset if

–The union of coresets is an α-approximate coreset for the union

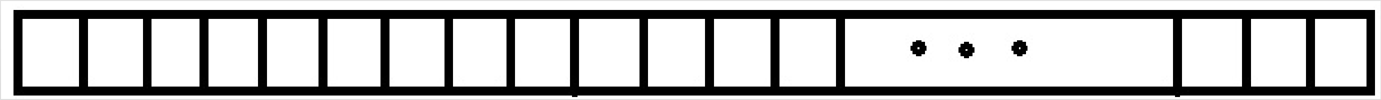
i.e. for multiple data sets $\boldsymbol{V_1}, \cdots, \boldsymbol{V_m}$ and their coresets $\boldsymbol{U_1}, \cdots, \boldsymbol{U_m}$,

$\boldsymbol{f}(\boldsymbol{U_1} \cup \cdots \cup \boldsymbol{U_m})$ approximates $\boldsymbol{f}(\boldsymbol{V_1} \cup \cdots \cup \boldsymbol{V_m})$ by a factor $\boldsymbol{\alpha}$

- ✓ Composable Core-sets have been studied for the **diversity Maximization** problems, for other notions of diversity: minimum pairwise distance, sum of pairwise distances, etc.
- ✓ Determinant maximization is a higher order notion of diversity

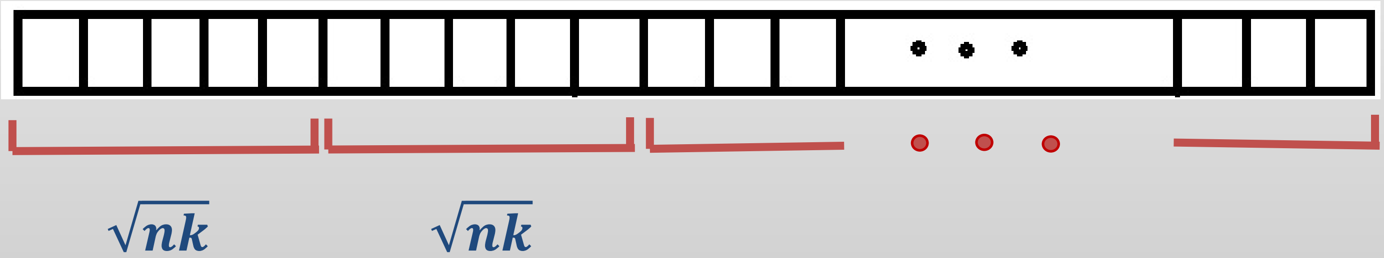
Applications: Streaming Computation

- **Streaming Computation:**
  - Processing sequence of n data elements "on the fly"
  - limited Storage



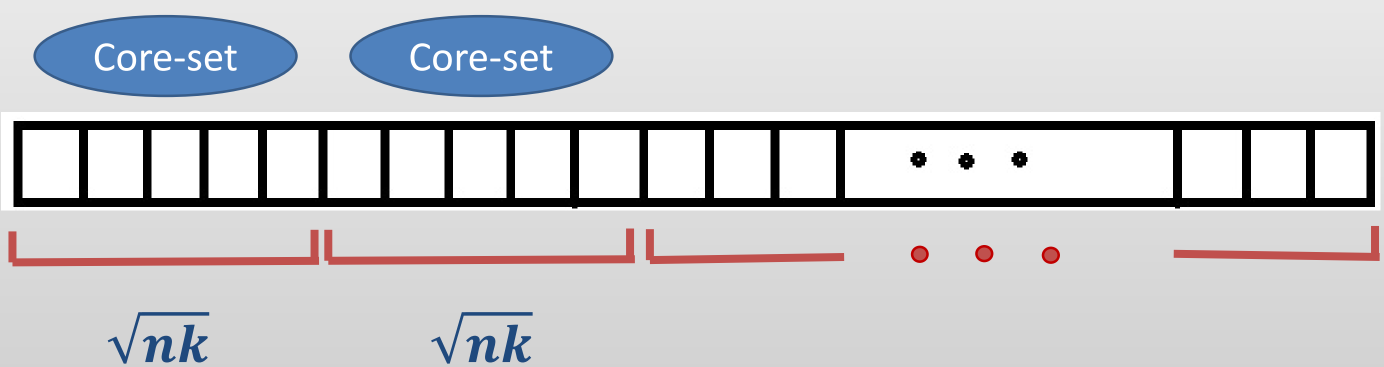
Applications: Streaming Computation

- **Streaming Computation:**
  - Processing sequence of n data elements "on the fly"
  - limited Storage
- **c-Composable Core-set of size k**
  - Chunks of size $\sqrt{nk}$, thus number of chunks = $\sqrt{n/k}$



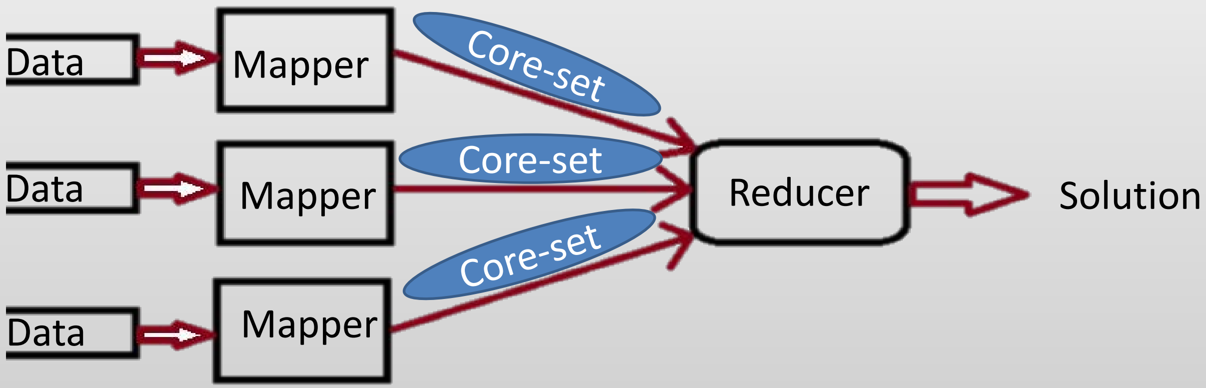
Applications: Streaming Computation

- **Streaming Computation:**
  - Processing sequence of n data elements "on the fly"
  - limited Storage
- **c-Composable Core-set of size k**
  - Chunks of size $\sqrt{nk}$, thus number of chunks = $\sqrt{n/k}$
  - Core-set for each chunk
  - Total Space: $k\sqrt{n/k} + \sqrt{nk} = O(\sqrt{nk})$
  - Approximation Factor: c



Applications: Distributed Computation

- Streaming Computation
- **Distributed System:**
  - Each machine holds a block of data.
  - A composable core-set is computed and sent to the server
- **Map-Reduce Model:**
  - One round of Map-Reduce
  - $\sqrt{n/k}$ mappers each getting $\sqrt{nk}$ points
  - Mapper computes a composable core-set of size k
  - Will be passed to a single reducer



Can we get a composable core-set of small size for the determinant maximization problem?

Results

Composable Core-sets for Determinant Maximization:

Algorithm:
There exists a polynomial time algorithm for computing an $\tilde{O}(k)^k$ -composable core-set of size $\tilde{O}(k)$ for the k-determinant maximization problem.

Lower bound:
Any composable core-set of size $k^{O(1)}$ for the k-determinant maximization problem must have an approximation factor of $\Omega(k)^{k(1-o(1))}$.

Results

Composable Core-sets for Determinant Maximization:

Algorithm:
There exists a polynomial time algorithm for computing an $\tilde{O}(k)^k$ -composable core-set of size $\tilde{O}(k)$ for the k-determinant maximization problem.

Lower bound:
Any composable core-set of size $k^{O(1)}$ for the k-determinant maximization problem must have an approximation factor of $\Omega(k)^{k(1-o(1))}$.

➢ Note the gap with the approximation factor of the best offline algorithm: e^k

Overall Picture

Spectral Spanners

→

Generalizing the notion of

Combinatorial Spanners

Overall Picture

Spectral Spanners

Generalizing the notion of Combinatorial Spanners

Composable Core-sets for many spectral optimization problems

Streaming algorithm

Distributed algorithm

Map-Reduce algorithm

Overall Picture

Spectral Spanners

Easy case $\boldsymbol{k = d}$

Spectral Spanners ($\boldsymbol{k = d}$)

- Spanners: sparsifying a graph while preserving distances between nodes.

- Spectral Spanners: sparsifying a point set while preserving distances to hyperplanes.
  - similar to core-sets for width [AHV'05]

Spectral Spanners ($\boldsymbol{k = d}$)

Given a set of points V

A subset $\boldsymbol{U \subseteq V}$ is a **α-spectral spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution μ_v over $\boldsymbol{U}$, s.t. for every direction $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[\langle x, u\rangle^2]$$

Spectral Spanners ($\boldsymbol{k = d}$)

Given a set of points V

A subset $\boldsymbol{U \subseteq V}$ is a **α-spectral spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution μ_v over $\boldsymbol{U}$, s.t. for every direction $\boldsymbol{x} \in \mathbb{R}^d$

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u\sim\mu_v}[\langle x, u\rangle^2]$$

$\boldsymbol{d(v, H_x)}$; $\boldsymbol{d(u, H_x)}$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of points in V to H_x is "preserved"(in expectation) over U

Spectral Spanners ($\boldsymbol{k = d}$)

Given a set of points V

A subset $\boldsymbol{U \subseteq V}$ is a **α-spectral spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution μ_v over $\boldsymbol{U}$, s.t. for every direction $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$

$$\langle x, v \rangle^2 \leq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[\langle x, u \rangle^2]$$

$\boldsymbol{d(v, H_x)}$

$\boldsymbol{d(u, H_x)}$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of points in V to H_x is "preserved"(in expectation) over U

equivalently,

$$vv^T \preceq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

Overall Picture

Spectral Spanners

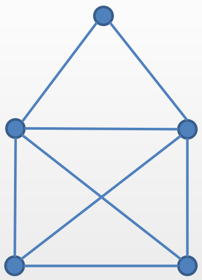
⇒

Generalizing the notion of

Combinatorial Spanners

Why we call them "spanners"?

- Let G be a graph with n vertices and m edges.

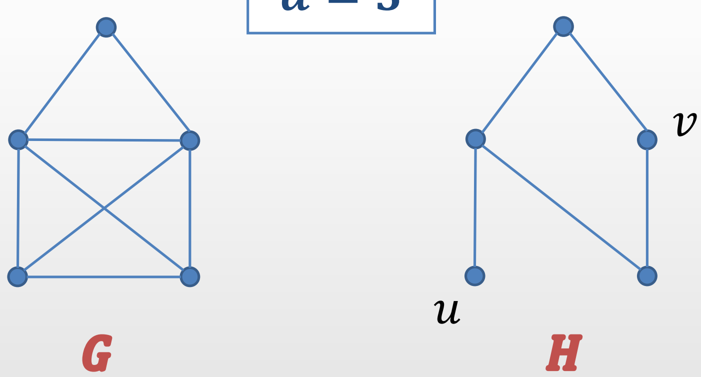


G

Why we call them "spanners"?

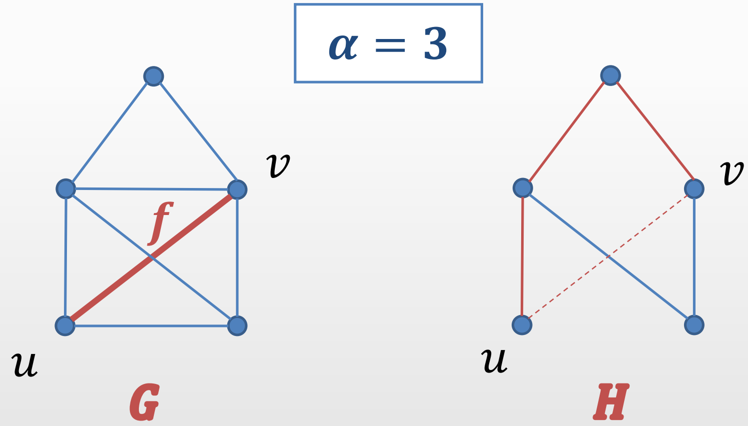
- Let G be a graph with n vertices and m edges.
- H is an α spanner if for every edge $f = (u, v) \in E_G$, $d_H(u, v) \leq \alpha$

$\boldsymbol{\alpha = 3}$



Why we call them "spanners"?

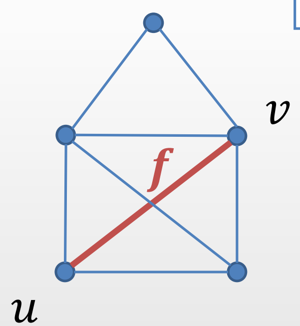
- Let G be a graph with n vertices and m edges.
- H is an α spanner if for every edge $f = (u, v) \in E_G$, $d_H(u, v) \leq \alpha$



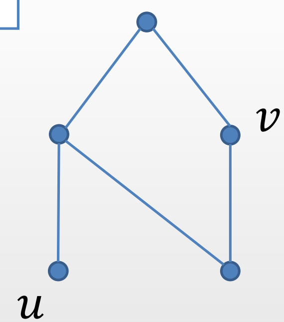
Why we call them "spanners"?

- Let G be a graph with n vertices and m edges.
- H is an α spanner if for every edge $f = (u, v) \in E_G$, $d_H(u, v) \leq \alpha$
- Let $V = \{b_f\}$ contain m vectors, $b_f = e_u - e_v$ is n-dimensional

$\alpha = 3$



G



H

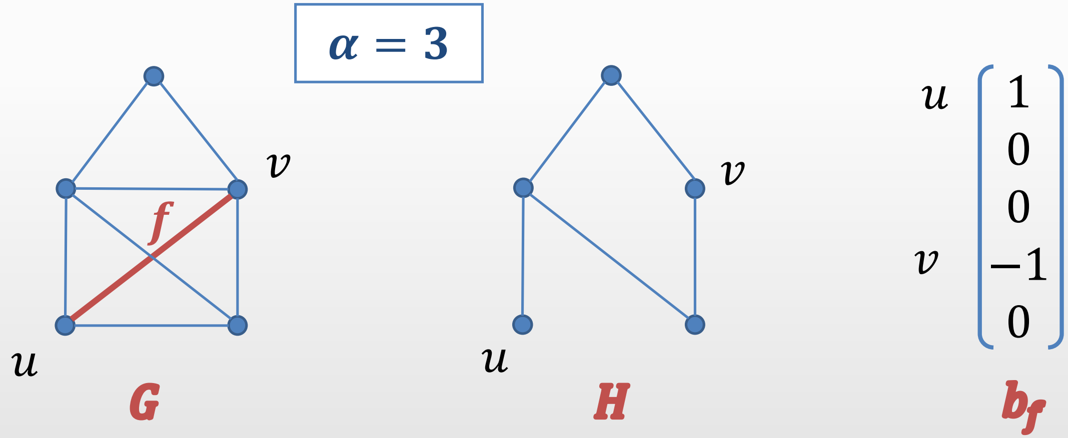
$$\begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}$$

$\boldsymbol{b_f}$

m vectors in n dimensions

Why we call them "spanners"?

- Let G be a graph with n vertices and m edges.
- H is an α spanner if for every edge $f = (u, v) \in E_G$, $d_H(u, v) \leq \alpha$
- Let $V = \{b_f\}$ contain m vectors, $b_f = e_u - e_v$ is n-dimensional

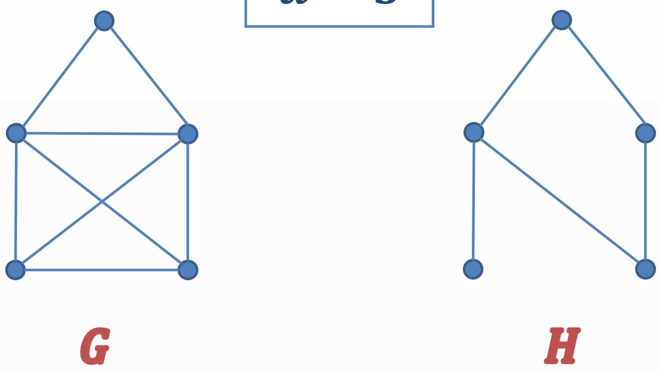


m vectors in n dimensions

Observation: If H is **an α-combinatorial spanner** of G, then $U = \{b_f\}_{f \in E_H}$ is an **α^2-spectral spanner** of $V = \{b_f\}_{f \in E_G}$

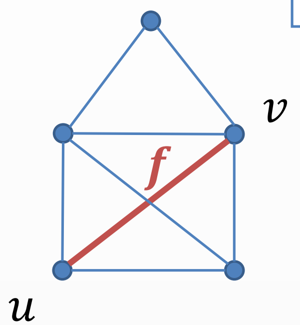
Proof for H being a spectral spanner

$\alpha = 3$

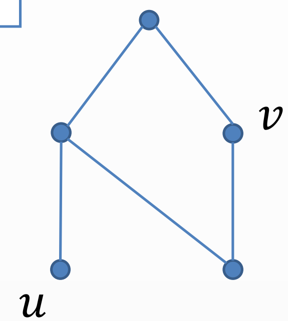


Proof for H being a spectral spanner

$\boldsymbol{\alpha = 3}$



G



H

$$\begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}$$

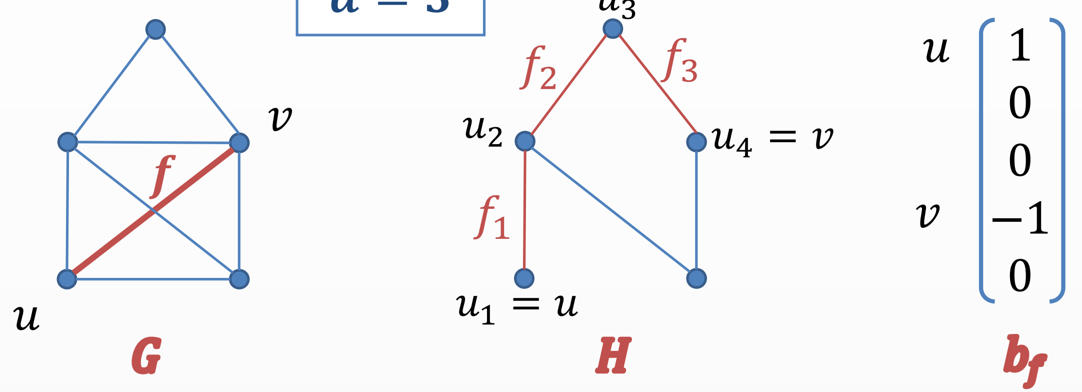
$\boldsymbol{b_f}$

Take b_f
$f = (u, v)$

Proof for H being a spectral spanner

Define μ_f

$\alpha = 3$



Take b_f
$f = (u, v)$

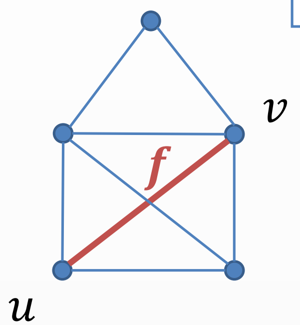
H is a spanner

There exists $f_1, \ldots, f_\alpha$ s.t.

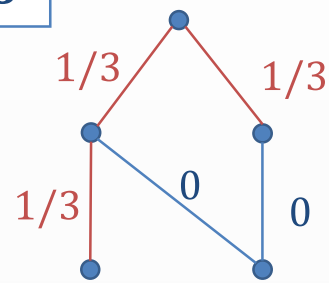
- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

Proof for H being a spectral spanner

$\alpha = 3$



G



H

$$\begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}$$

$\boldsymbol{b_f}$

Take b_f
$f = (u, v)$

H is a spanner

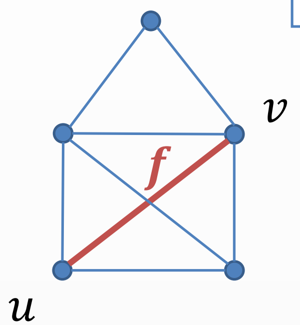
There exists $f_1, \dots, f_\alpha$ s.t.

- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

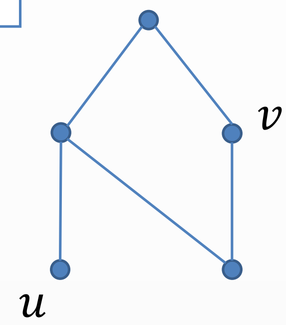
$$\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$$

Goal: $\langle x, b_f\rangle^2 \le \alpha^2 \cdot \mathbb{E}_{e\sim\mu_f}[\langle x, b_e\rangle^2]$

$\alpha = 3$



G



H

$$\begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

b_f $\quad$ x

Take b_f
$f = (u, v)$

→ H is a spanner →

There exists $f_1, \dots, f_\alpha$ s.t.

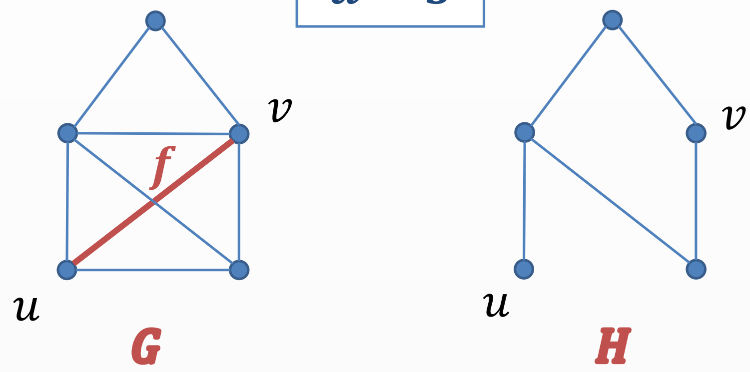
- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

→

$$\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$$

Goal: $\langle x, b_f \rangle^2 \leq \alpha^2 \cdot \mathbb{E}_{e \sim \mu_f}[\langle x, b_e \rangle^2]$

$\alpha = 3$



$$b_f = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

Take b_f
$f = (u, v)$

→ H is a spanner →

There exists $f_1, \ldots, f_\alpha$ s.t.
- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

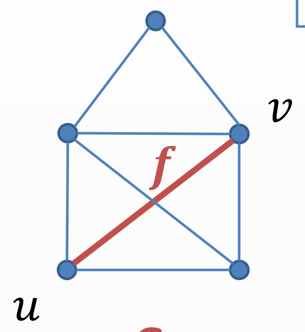
→

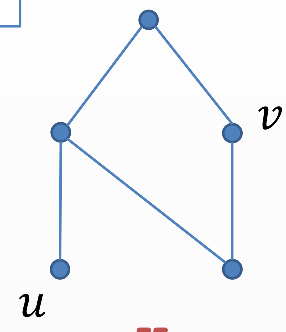
$$\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$$

$$Z \coloneqq \langle x, b_f \rangle = (x_u - x_v)$$
$$Z_i \coloneqq \langle x, b_{f_i} \rangle = (x_{u_i} - x_{u_{i+1}})$$

Goal: $Z^2 \le \alpha^2 \cdot \sum_i \frac{1}{\alpha} {Z_i}^2$

$\alpha = 3$





$$b_f = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

Take b_f
$f = (u, v)$

→ H is a spanner →

There exists $f_1, \ldots, f_\alpha$ s.t.

- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

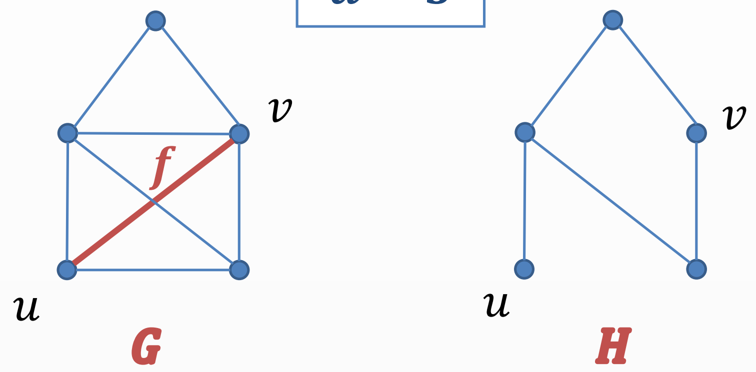
→

$$\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$$

$$Z := \langle x, b_f \rangle = (x_u - x_v)$$
$$Z_i := \langle x, b_{f_i} \rangle = (x_{u_i} - x_{u_{i+1}})$$

Goal: $Z^2 \le \alpha^2 \cdot \sum_i \frac{1}{\alpha} {Z_i}^2$

$\alpha = 3$



$$u \quad v \quad \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

b_f $\qquad$ x

Take b_f
$f = (u, v)$

H is a spanner

There exists $f_1, \dots, f_\alpha$ s.t.

- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

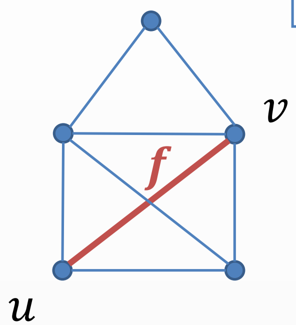
$$\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$$

$$Z = \sum_{i=1}^{\alpha} Z_i$$

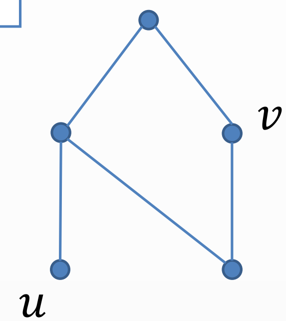
$$Z \coloneqq \langle x, b_f \rangle = (x_u - x_v)$$
$$Z_i \coloneqq \langle x, b_{f_i} \rangle = (x_{u_i} - x_{u_{i+1}})$$

Goal: $Z^2 \leq \alpha^2 \cdot \sum_i \frac{1}{\alpha} Z_i^{\,2}$

$\alpha = 3$



G



H

$$b_f = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

Take b_f
$f = (u, v)$

→ H is a spanner →

There exists $f_1, \dots, f_\alpha$ s.t.

- $f_i = (u_i, u_{i+1})$
- $u_1 = u$
- $u_{\alpha+1} = v$

→ $\mu_f(e) = \begin{cases} (1/\alpha), & e = f_i \\ 0, & o.w. \end{cases}$

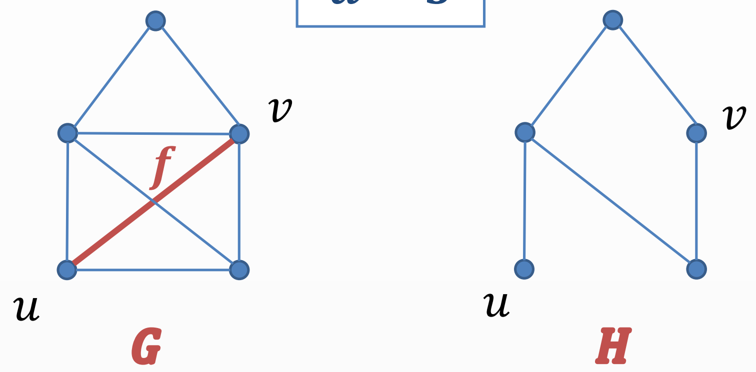
$Z := \langle x, b_f \rangle = (x_u - x_v)$
$Z_i := \langle x, b_{f_i} \rangle = (x_{u_i} - x_{u_{i+1}})$

→ $Z = \sum_{i=1}^{\alpha} Z_i$ → Cauchy-Schwarz →

$$Z^2 \leq \alpha \sum_{i=1}^{\alpha} Z_i^2 \leq \alpha^2 \sum_{i=1}^{\alpha} \frac{1}{\alpha} \cdot Z_i^2$$

Goal: $Z^2 \leq \alpha^2 \cdot \sum_i \frac{1}{\alpha} Z_i^{\;2}$

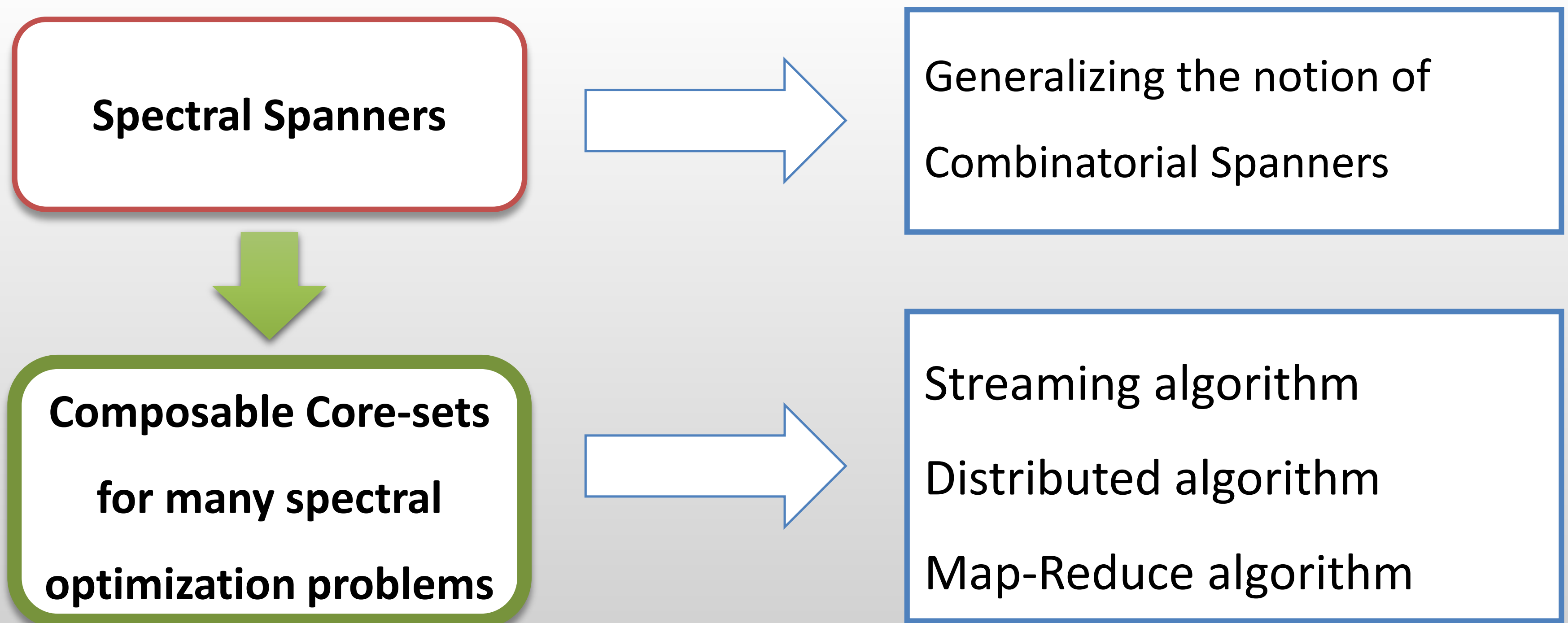
$\alpha = 3$



$$b_f = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} \begin{matrix} u \\ \\ \\ v \\ \\ \end{matrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

Observation: If H is **an α-combinatorial spanner** of G, then $U = \{b_f\}_{f \in E_H}$ is an **α^2-spectral spanner** of $V = \{b_f\}_{f \in E_G}$

Overall Picture



Spectral spanners are good core-sets

Spectral spanners are good core-sets

- Let V be the point set
- Let U be the spectral spanner of V as its core-set
- Let OPT be a subset of k points in V whose det is maximized.
- **Goal:** U contains a “good” solution for det. Maximization

Spectral spanners are good core-sets

- Let V be the point set
- Let U be the spectral spanner of V as its core-set
- Let OPT be a subset of k points in V whose det is maximized.
- **Goal:** U contains a “good” solution for det. Maximization

$OPT \subset V$: the optimal solution over V

Spectral spanners are good core-sets

- Let V be the point set
- Let U be the spectral spanner of V as its core-set
- Let OPT be a subset of k points in V whose det is maximized.
- **Goal:** U contains a "good" solution for det. Maximization

$OPT \subset V$: the optimal (integral) solution over V → A: a fractional solution over U

By spanner property, replace every $v \in OPT$ with a distribution on U
App. factor: α^d

Spectral spanners are good core-sets

- Let V be the point set
- Let U be the spectral spanner of V as its core-set
- Let OPT be a subset of k points in V whose det is maximized.
- **Goal:** U contains a "good" solution for det. Maximization

$OPT \subset V$: the optimal (integral) solution over V → A: a fractional solution over U → T: an integral solution over U

By spanner property, replace every $v \in OPT$ with a distribution on U
App. factor: α^d

[Nik'15]: using randomized rounding
App. factor: e^d

Spectral spanners are good core-sets

- Let V be the point set
- Let U be the spectral spanner of V as its core-set
- Let OPT be a subset of k points in V whose det is maximized.
- **Goal:** U contains a "good" solution for det. Maximization

$OPT \subset V$: the optimal (integral) solution over V → A: a fractional solution over U → T: an integral solution over U

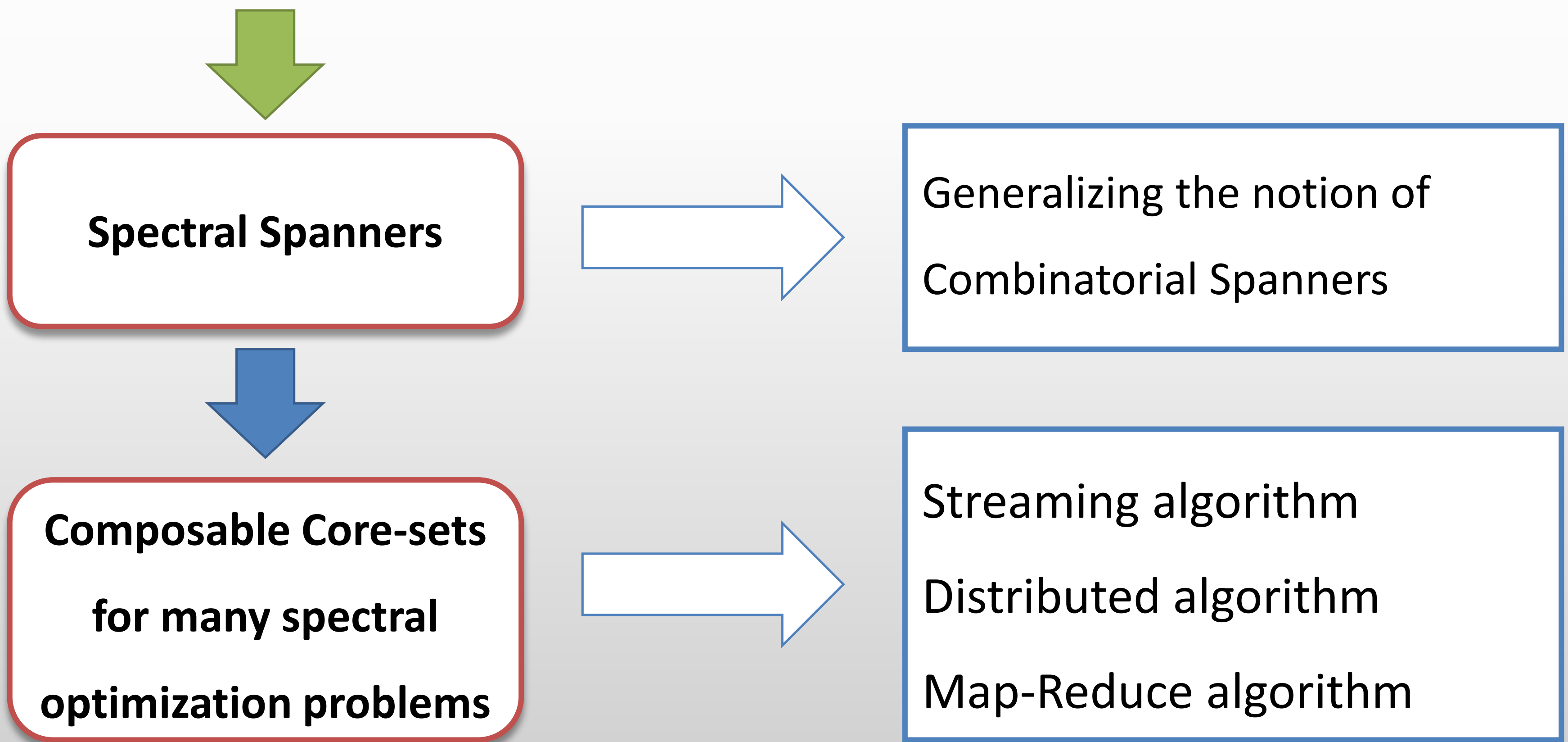
By spanner property, replace every $v \in OPT$ with a distribution on U
App. factor: α^d

[Nik'15]: using randomized rounding
App. factor: e^d

$$\det(OPT) \leq \alpha^d \det(A) \leq (\alpha e)^d \det(T)$$

Will show: spectral spanner with $\alpha = \tilde{O}(d)$ and thus the app. factor is $\tilde{O}(d)^d$.

Overall Picture



Algorithm for finding spanners

In graph: for any edge (u, v), where $dist(u, v) > \alpha$ put the edge in the spanner.

Algorithm for finding spanners

In graph: for any edge (u, v), where $dist(u, v) > \alpha$ put the edge in the spanner.

Spectral variant of the algorithm: Find a vector v such that for any distribution μ_v over U, the spanner condition is not satisfied, i.e.,

$$vv^T \precneqq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

add it to the spanner U.

Algorithm for finding spanners

In graph: for any edge (u, v), where $dist(u, v) > \alpha$ put the edge in the spanner.

Spectral variant of the algorithm: Find a vector v such that for any distribution μ_v over U, the spanner condition is not satisfied, i.e.,

$$vv^T \precneqq \alpha \cdot \mathbb{E}_{u\sim\mu_v}[uu^T]$$

add it to the spanner U.

- We can implement in polynomial time
- We cannot bound the size of the spanner.

Algorithm for finding spanners

In graph: for any edge (u, v), where $dist(u, v) > \alpha$ put the edge in the spanner.

Spectral variant of the algorithm: Find a vector v such that for any distribution μ_v over U, the spanner condition is not satisfied, i.e.,

$$vv^T \lnsim \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

add it to the spanner U.

- We can implement in polynomial time
- We cannot bound the size of the spanner. → Take a detour

Overall Picture

Weak Spectral Spanners → **Geometric Interpretation:** Preserves maximum distance to any k-dimensional hyperplane.

Spectral Spanners → Generalizing the notion of Combinatorial Spanners

↓

Composable Core-sets for many spectral optimization problems →

- Streaming algorithm
- Distributed algorithm
- Map-Reduce algorithm

Weak Spanners

- Change the order of quantifiers in the spanner:
- **Strong Spanners**: For **every** $\boldsymbol{v}$, there **exists** $\boldsymbol{\mu_v}$, s.t. for **every** $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$,

$$\langle x, v \rangle^2 \leq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[\langle x, u \rangle^2]$$

Weak Spanners

- Change the order of quantifiers in the spanner:
- **Strong Spanners**: For **every $\boldsymbol{v}$**, there **exists $\boldsymbol{\mu_v}$**, s.t. for **every $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$**,

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u\sim\mu_v}[\langle x, u\rangle^2]$$

- **Weak Spanners**: For every v and **every $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u\sim\mu_{v,x}}[\langle x, u\rangle^2]$$

Weak Spanners

- Change the order of quantifiers in the spanner:
- **Strong Spanners**: For **every v**, there **exists μ_v**, s.t. for **every $x \in \mathbb{R}^d$**,

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[\langle x, u\rangle^2]$$

- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$** , there exists $\mu_{v,x}$, s.t.

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u \sim \mu_{v,x}}[\langle x, u\rangle^2] \iff \langle x, v\rangle^2 \leq \alpha \cdot \max_u \langle x, u\rangle^2$$

Weak Spanners

- Change the order of quantifiers in the spanner:
- **Strong Spanners**: For **every $\boldsymbol{v}$**, there **exists $\boldsymbol{\mu_v}$**, s.t. for **every $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$**,

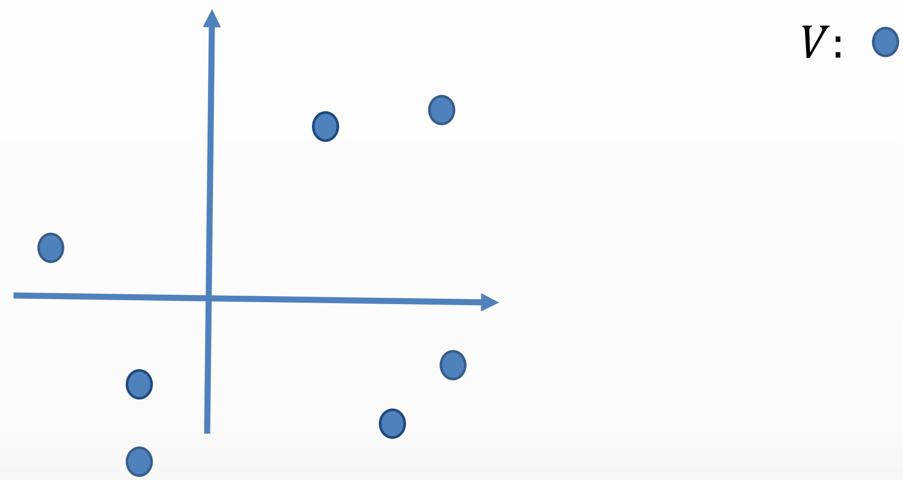
$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u\sim\mu_v}[\langle x, u\rangle^2]$$

- **Weak Spanners**: For every v and **every $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v\rangle^2 \leq \alpha \cdot \mathbb{E}_{u\sim\mu_{v,x}}[\langle x, u\rangle^2] \iff \langle x, v\rangle^2 \leq \alpha \cdot \max_u \langle x, u\rangle^2$$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Weak Spanners

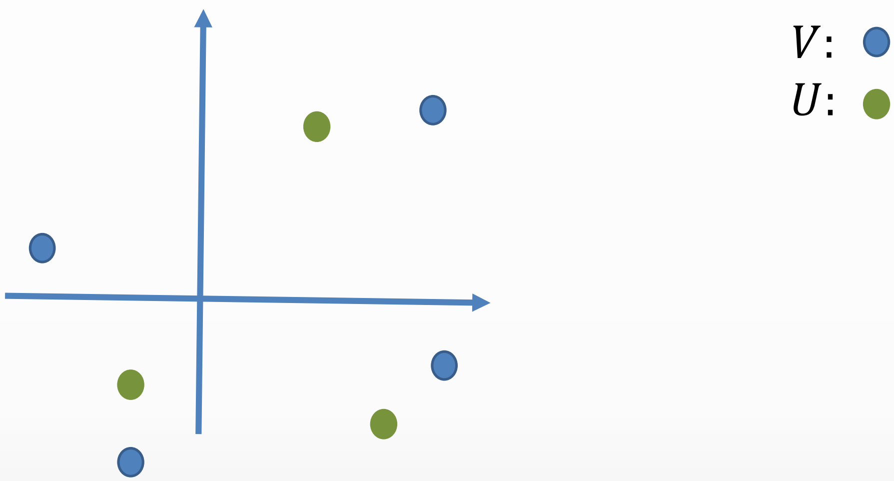


- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v \rangle^2 \leq \alpha \cdot \max_u \langle x, u \rangle^2$$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Weak Spanners

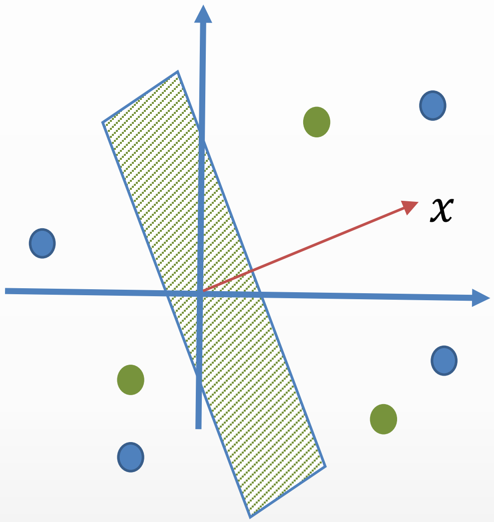


- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v\rangle^2 \leq \alpha \cdot \max_u \langle x, u\rangle^2$$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Weak Spanners



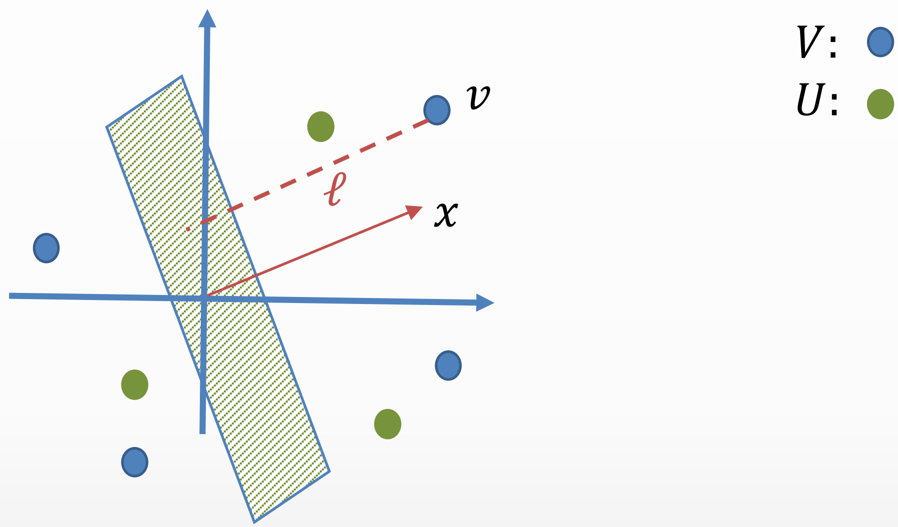
V: (blue dot)
U: (green dot)

- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v \rangle^2 \leq \alpha \cdot \max_u \langle x, u \rangle^2$$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Weak Spanners

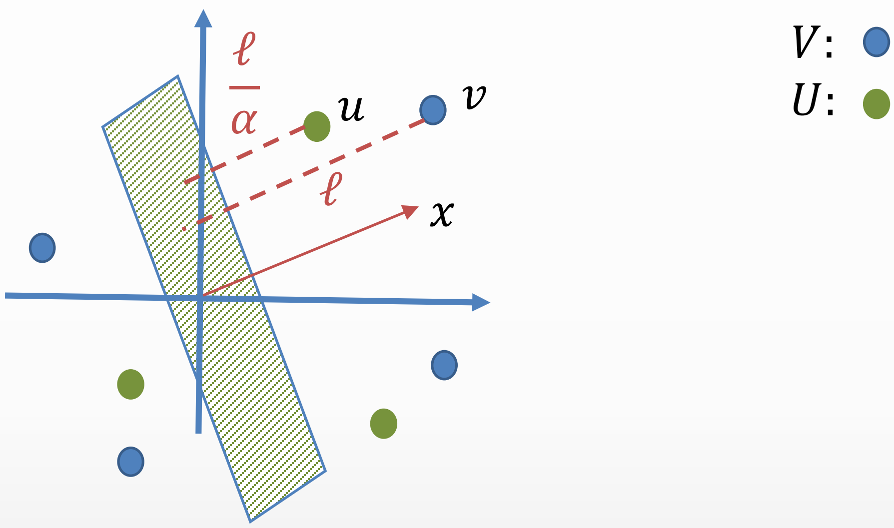


- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v\rangle^2 \leq \alpha \cdot \max_u \langle x, u\rangle^2$$

Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Weak Spanners



- **Weak Spanners**: For every v and **every $x \in \mathbb{R}^d$**, there exists $\boldsymbol{\mu_{v,x}}$, s.t.

$$\langle x, v\rangle^2 \le \alpha \cdot \max_u \langle x, u\rangle^2$$

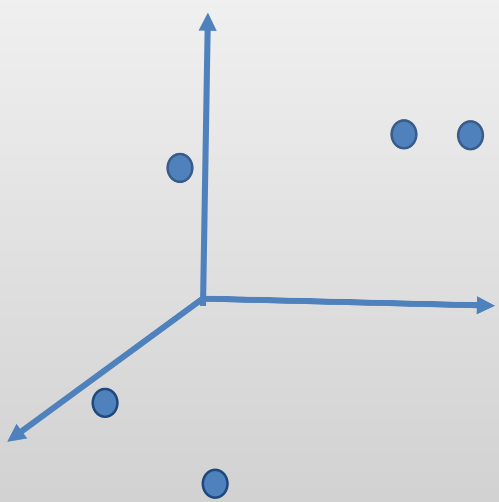
Geometric interpretation: For any hyperplane $\boldsymbol{H_x}$ (perpendicular to x), the maximum distance of the points in V to H_x is **"preserved"** over U

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\operatorname{argmax}_v \langle x, v\rangle^2}$ to the spanner U.

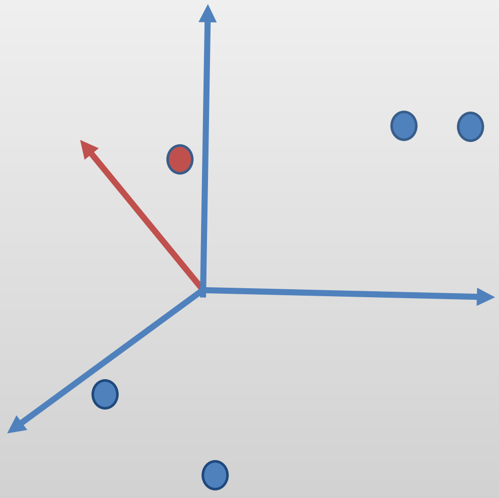


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\arg\max}_v \langle x, v\rangle^2$ to the spanner U.

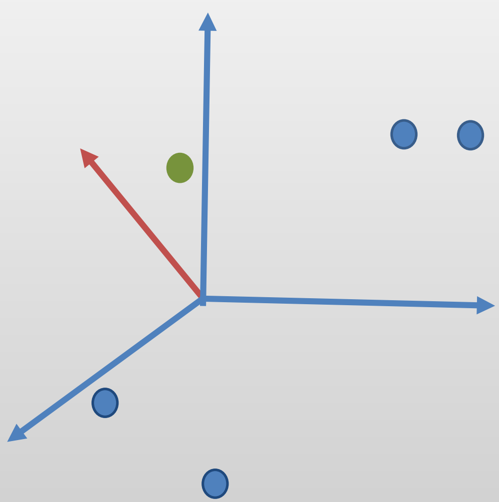


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\underset{v}{\mathbf{argmax}}\langle x, v\rangle^2$ to the spanner U.

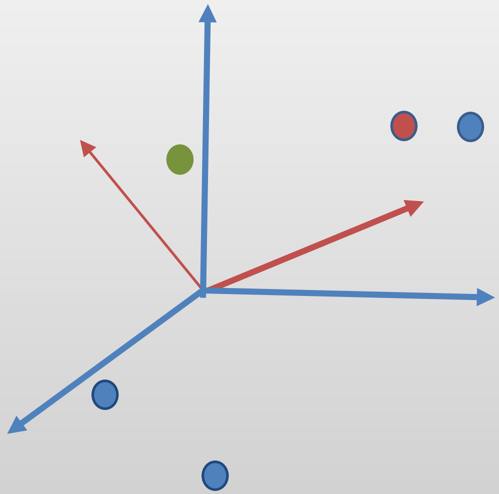


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v \rangle^2 > \alpha \cdot \max_u \langle x, u \rangle^2$$

add $\mathbf{\arg\max_v \langle x, v \rangle^2}$ to the spanner U.

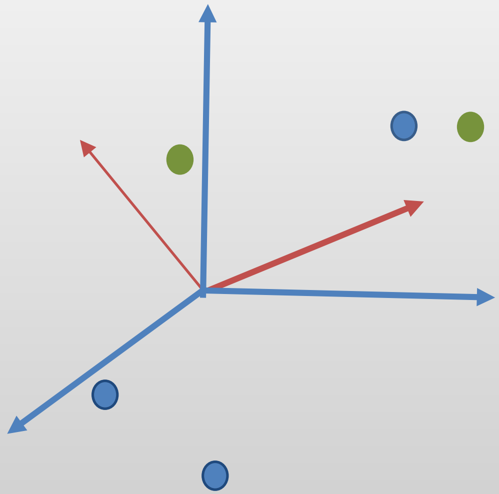


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\operatorname{argmax}_v \langle x, v\rangle^2$ to the spanner U.

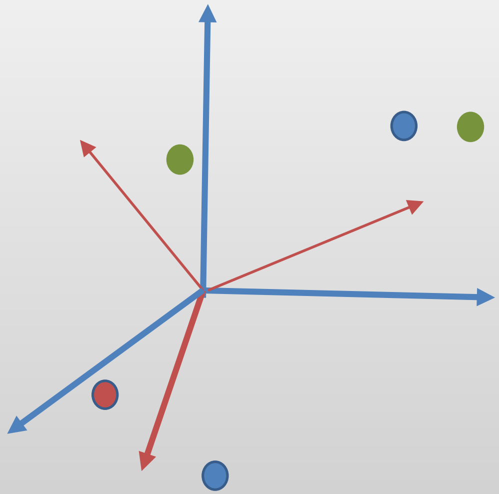


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\operatorname{argmax}_v \langle x, v\rangle^2$ to the spanner U.

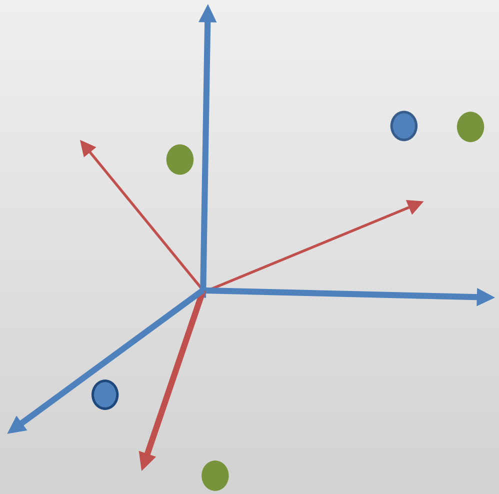


Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\underset{v}{argmax}\langle x, v\rangle^2}$ to the spanner U.



Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\arg\max_v \langle x, v\rangle^2}$ to the spanner U.

- **We should bound the size of *U***

Algorithm for finding spanners

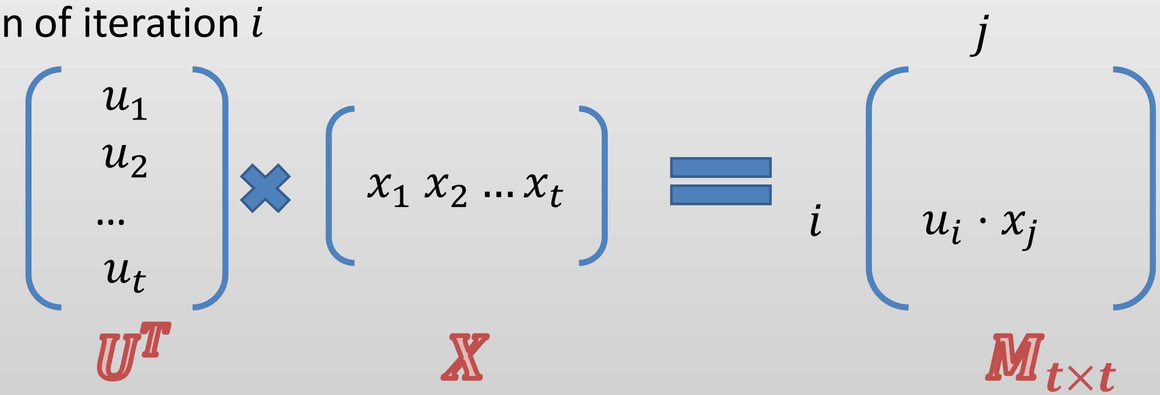
Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\underset{v}{\mathbf{argmax}}\langle x, v\rangle^2$ to the spanner U.

➢ **We should bound the size of U**

- Let $t = |U|$ be the number of iterations
- Let u_i be the vector picked at iteration i in the spanner
- Let x_i be the bad direction of iteration i



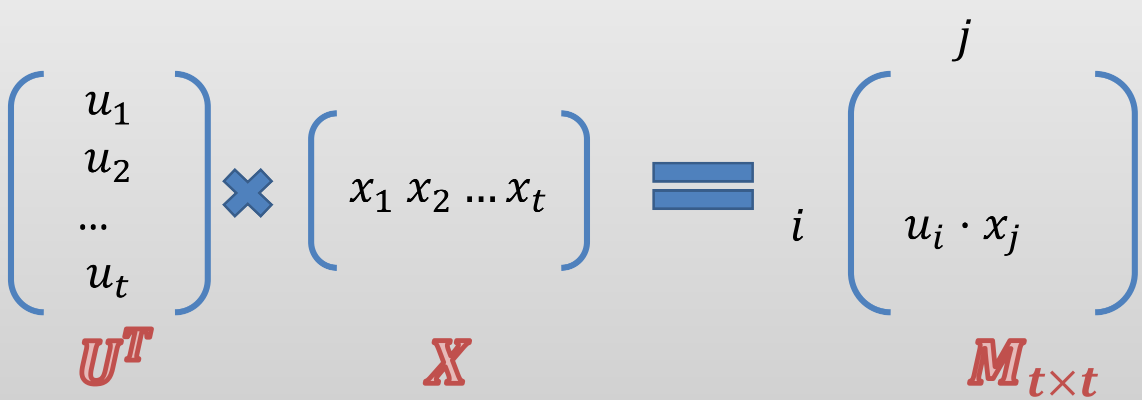
Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v \rangle^2 > \alpha \cdot \max_u \langle x, u \rangle^2$$

add $\mathbf{\underset{v}{argmax}} \langle x, v \rangle^2$ to the spanner U.

- **Fact:** $rank(M) \leq d$ as M is a product of a $t \times d$ and a $d \times t$ matrix
- **Goal:** Show $rank(M) \geq C \cdot t$
- **Implies:** $t \leq d/C$



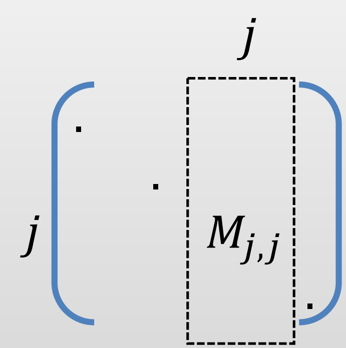
Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{argmax}_v \langle x, v\rangle^2$ to the spanner U.

Take the matrix $M_{t\times t}$ where $M_{i,j} = \langle u_i, x_j\rangle$



$M_{t\times t}$

➢**Goal:** Show $rank(M) \geq C \cdot t$

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

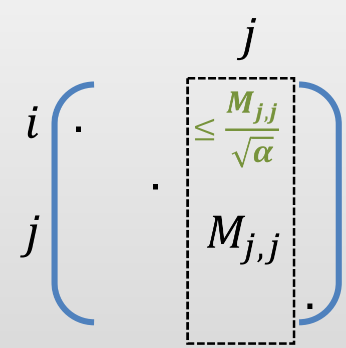
$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{argmax}_{v} \langle x, v\rangle^2$ to the spanner U.

$\langle u_j, x_j\rangle \geq \sqrt{\alpha}\langle u_i, x_j\rangle$

Take the matrix $M_{t\times t}$ where $M_{i,j} = \langle u_i, x_j\rangle$

- For any $i < j$, we have $M_{i,j} \leq \left(\frac{1}{\sqrt{\alpha}}\right) M_{j,j}$



$M_{t\times t}$

➢**Goal:** Show $rank(M) \geq C \cdot t$

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

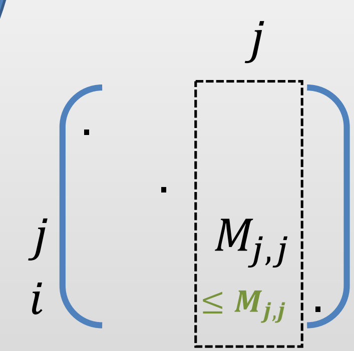
add $\mathbf{argmax}_v \langle x, v\rangle^2$ to the spanner U.

$\langle u_j, x_j\rangle \geq \sqrt{\alpha}\langle u_i, x_j\rangle$

$\langle u_j, x_j\rangle \geq \langle u_i, x_j\rangle$

Take the matrix $M_{t\times t}$ where $M_{i,j} = \langle u_i, x_j\rangle$

- For any $i < j$, we have $M_{i,j} \leq \left(\frac{1}{\sqrt{\alpha}}\right) M_{j,j}$
- For any $i > j$, we have $M_{i,j} \leq M_{j,j}$



$M_{t\times t}$

➢**Goal:** Show $rank(M) \geq C \cdot t$

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\operatorname{argmax}_v \langle x, v\rangle^2}$ to the spanner U.

- **Divide each column $\boldsymbol{j}$ of $\boldsymbol{M}$ by $\boldsymbol{M_{j,j}}$**

- **Goal:** Show $rank(M) \geq C \cdot t$

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

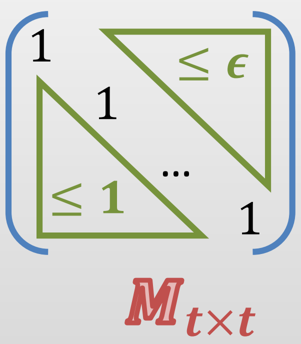
$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\underset{v}{\mathbf{argmax}}\langle x, v\rangle^2$ to the spanner U.

➢ **Divide each column j of M by $M_{j,j}$**

Take the matrix $M_{t\times t}$ where $M_{i,j} = \langle x_i, v_j\rangle / \langle x_j, v_j\rangle$

- All 1 on the diagonal
- For any $i < j$, we have $M_{i,j} \le \left(\frac{1}{\sqrt{\alpha}}\right) := \epsilon$
- For any $i > j$, we have $M_{i,j} \le 1$



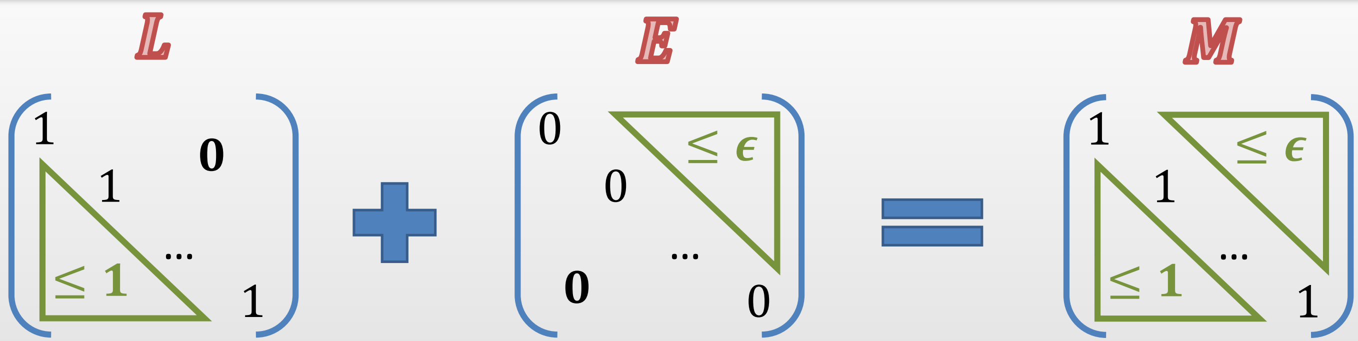
➢ **Goal:** Show $rank(M) \ge C \cdot t$

Algorithm for finding spanners

Spectral variant of the algorithm: Find a vector v and a direction x s.t.

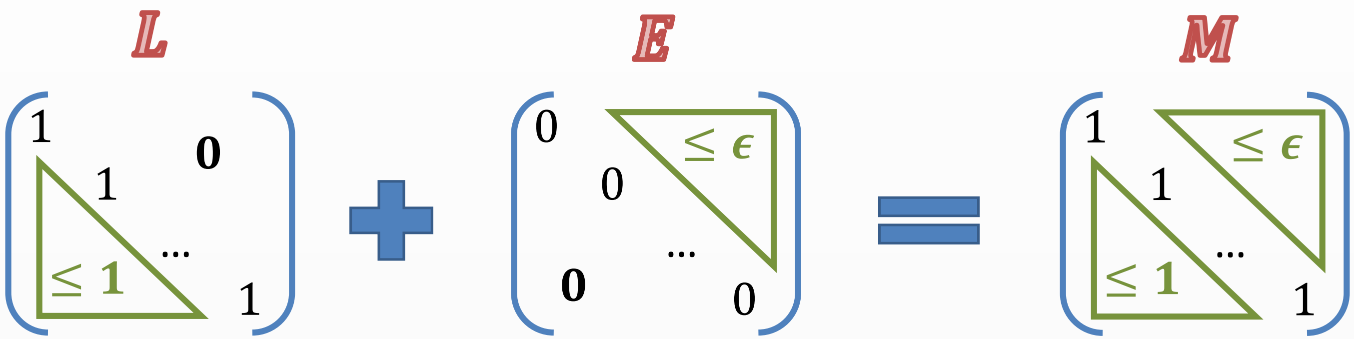
$$\langle x, v\rangle^2 > \alpha \cdot \max_u \langle x, u\rangle^2$$

add $\mathbf{\underset{v}{argmax}\langle x, v\rangle^2}$ to the spanner U.



➢**Goal:** Show $rank(M) \geq C \cdot t$

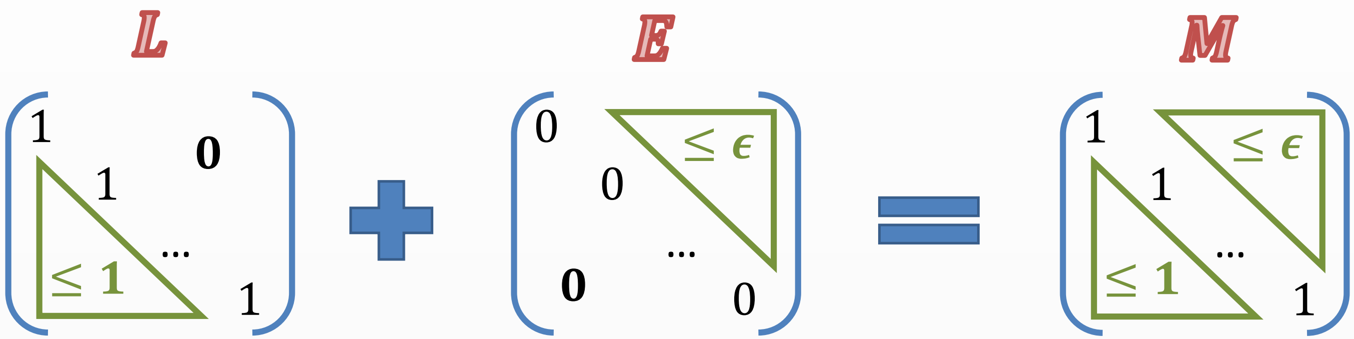
Algorithm for finding spanners



- ***L*** is a diagonally dominant lower triangular
- $rank(L) = t$

- ***E*** has small entries

Algorithm for finding spanners



- $\boldsymbol{L}$ is a diagonally dominant lower triangular
- $rank(L) = t$

- $\boldsymbol{E}$ has small entries

- **Theorem: Adding $\boldsymbol{E}$ does not decrease the rank "a lot"**

➢**Goal:** Show $rank(M) \geq C \cdot t$

Overall Picture

Weak Spectral Spanners

Spectral Spanners

Composable Core-sets for many spectral optimization problems

Geometric Interpretation:

Preserves maximum distance to any k-dimensional hyperplane.

Generalizing the notion of Combinatorial Spanners

Streaming algorithm

Distributed algorithm

Map-Reduce algorithm

From weak spanners to strong spanners

Theorem: any weak spectral spanner is in fact a strong spectral spanner

From weak spanners to strong spanners

Theorem: any weak spectral spanner is in fact a strong spectral spanner

- We need to show existence of μ_v → write an SDP for each v
- Instead consider the dual of SDP
- Use hyperplane separating theorem to show such a distribution exists.

Overall Picture

Weak Spectral Spanners

Geometric Interpretation:

Preserves maximum distance to any k-dimensional hyperplane.

Spectral Spanners

Generalizing the notion of Combinatorial Spanners

Composable Core-sets for many spectral optimization problems

Streaming algorithm

Distributed algorithm

Map-Reduce algorithm

Generalizing to $\boldsymbol{k < d}$

Spectral Spanners ($\boldsymbol{k = d}$)

A set $\boldsymbol{U} \subseteq \boldsymbol{V}$ is a $\boldsymbol{\alpha}$**-spectral spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution $\boldsymbol{\mu_v}$ over $\boldsymbol{U}$, s.t.

$$vv^T \preceq \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

Spectral Spanners ($\boldsymbol{k \leq d}$)

A set $\boldsymbol{U} \subseteq \boldsymbol{V}$ is a **$\boldsymbol{\alpha}$-spectral $\boldsymbol{k}$-spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution $\boldsymbol{\mu_v}$ over $\boldsymbol{U}$, s.t.

$$vv^T \preceq_k \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

Spectral Spanners ($\boldsymbol{k \leq d}$)

A set $\boldsymbol{U} \subseteq \boldsymbol{V}$ is a **$\boldsymbol{\alpha}$-spectral $\boldsymbol{k}$-spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution $\boldsymbol{\mu_v}$ over $\boldsymbol{U}$, s.t.

$$vv^T \preceq_k \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

Definition: For two matrices $A_{d \times d}$ and $B_{d \times d}$

$$\boldsymbol{A \preceq_k B}$$

- Iff sum of the bottom $d - k + 1$ eigen values of $\boldsymbol{B - A}$ is non-negative

- $A \preceq_d B$ iff $A \preceq B$
- $A \preceq_1 B$ iff $tr(A) \leq tr(B)$

Spectral Spanners ($\boldsymbol{k \leq d}$)

A set $\boldsymbol{U} \subseteq \boldsymbol{V}$ is a **$\boldsymbol{\alpha}$-spectral $\boldsymbol{k}$-spanner** of V if for every $\boldsymbol{v} \in V$, there exists a probability distribution $\boldsymbol{\mu_v}$ over $\boldsymbol{U}$, s.t.

$$vv^T \preceq_k \alpha \cdot \mathbb{E}_{u \sim \mu_v}[uu^T]$$

Definition: For two matrices $A_{d \times d}$ and $B_{d \times d}$

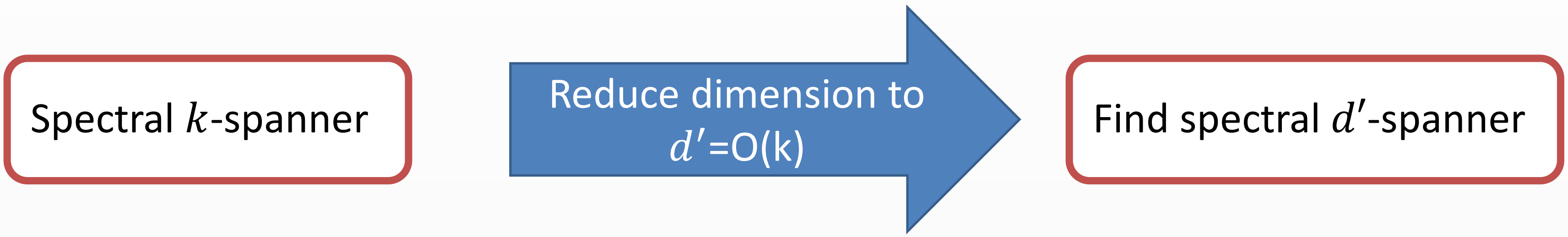
$$\boldsymbol{A \preceq_k B}$$

- ➢ Iff sum of the bottom $d - k + 1$ eigen values of $\boldsymbol{B - A}$ is non-negative

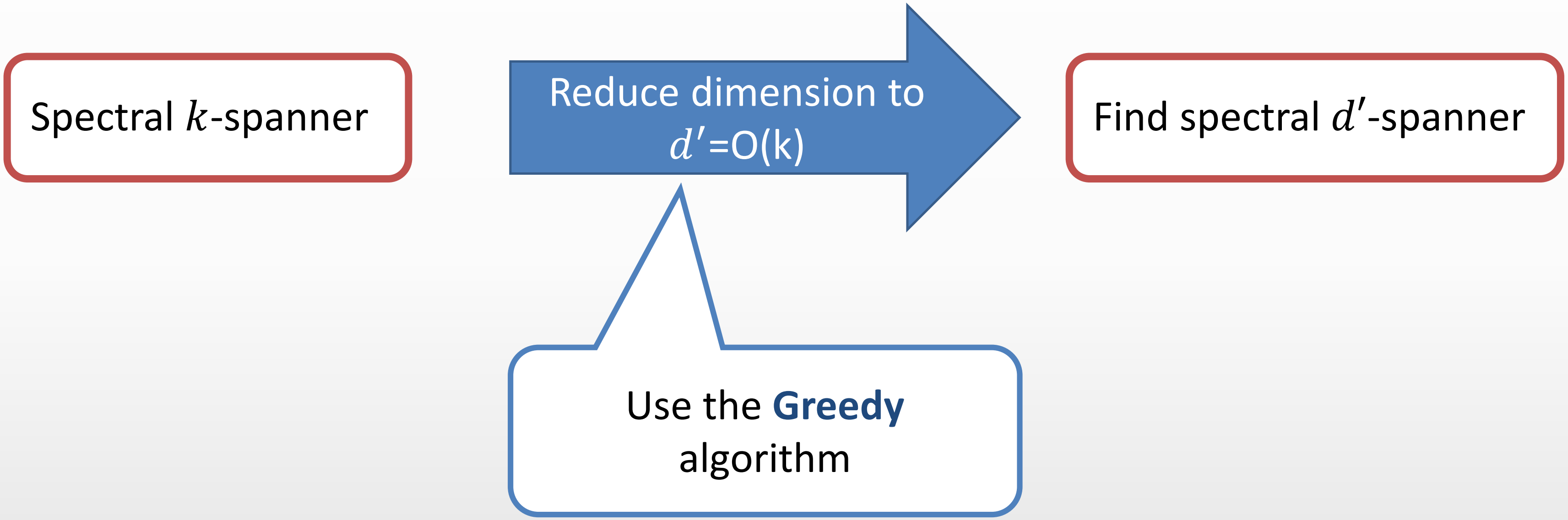
- $A \preceq_d B$ iff $A \preceq B$
- $A \preceq_1 B$ iff $tr(A) \leq tr(B)$

✓ Preserve distances to k-subspaces.

How to find spectral k-spanner?



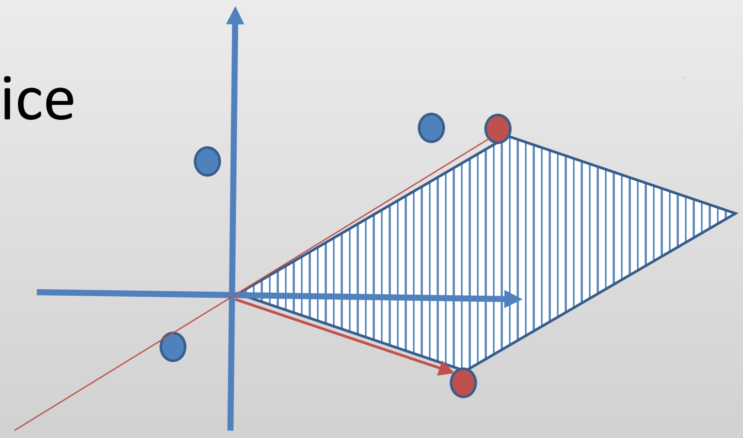
How to find spectral k-spanner?



What is known?

- Hard to approximate within a factor of 2^{ck} [CMI'13]
- Best algorithm: e^{k}-approximation [Nik'15]
- **Greedy** is a popular algorithm: achieves approximation factor $k!$
  - $U \leftarrow \emptyset$
  - For k iterations,
    - Add to U the farthest point from the subspace spanned by U
- Greedy performs very well in practice

$\boldsymbol{k = 2}$



Reduction to full dimensional case

Input: a set of points V and a parameter $k < d$

Output: k-spanner of V

- Run The **Greedy** algorithm on V for $2k$ iterations $\rightarrow \boldsymbol{U_1}$
- Let $\boldsymbol{S}$ be the subspace spanned by U_1
- **Project** V on S
- (Now $\boldsymbol{d' = O(k)}$)
- Find d'-spanner on projected $V \rightarrow \boldsymbol{U_2}$
- **Return** $\boldsymbol{U_1 \cup U_2}$

Summary

- Defined the notion of α-Spectral k-Spanner

Summary

- Defined the notion of α-Spectral k-Spanner
- Its connection to graph spanners and its geometric interpretation

Summary

- Defined the notion of α-Spectral k-Spanner
- Its connection to graph spanners and its geometric interpretation
- Spanner Result:
  - Showed there exists an $\tilde{O}(k)$-spectral k-spanner of size $\tilde{O}(k)$.
  - There exists a set of size $e^{\Omega(k^{\epsilon})}$ such that any $k^{1-\epsilon}$-Spanner must contain all vectors

Summary

- Defined the notion of α-Spectral k-Spanner
- Its connection to graph spanners and its geometric interpretation
- Spanner Result:
  - Showed there exists an $\tilde{O}(k)$-spectral k-spanner of size $\tilde{O}(k)$.
  - There exists a set of size $e^{\Omega(k^{\epsilon})}$ such that any $k^{1-\epsilon}$-Spanner must contain all vectors
- Composable Core-set Result:
  - There exists an $\tilde{\boldsymbol{O}}(\boldsymbol{k})^{\boldsymbol{k}}$ -composable core-set of size $\tilde{\boldsymbol{O}}(\boldsymbol{k})$ for the determinant maximization problem.
  - Any composable core-set of size $\boldsymbol{k}^{\boldsymbol{\beta}}$ for the determinant maximization must have an approximation factor of $\left(\frac{\boldsymbol{k}}{\boldsymbol{\beta}}\right)^{\boldsymbol{k}(\mathbf{1}-\boldsymbol{o}(\mathbf{1}))}$

Summary

- Defined the notion of α-Spectral k-Spanner
- Its connection to graph spanners and its geometric interpretation
- Spanner Result:
  - Showed there exists an $\tilde{O}(k)$-spectral k-spanner of size $\tilde{O}(k)$.
  - There exists a set of size $e^{\Omega(k^{\epsilon})}$ such that any $k^{1-\epsilon}$-Spanner must contain all vectors
- Composable Core-set Result:
  - There exists an $\tilde{\boldsymbol{O}}(\boldsymbol{k})^{\boldsymbol{k}}$ -composable core-set of size $\tilde{\boldsymbol{O}}(\boldsymbol{k})$ for the determinant maximization problem.
  - Any composable core-set of size $\boldsymbol{k}^{\boldsymbol{\beta}}$ for the determinant maximization must have an approximation factor of $\left(\frac{\boldsymbol{k}}{\boldsymbol{\beta}}\right)^{\boldsymbol{k}(\mathbf{1}-\boldsymbol{o}(\mathbf{1}))}$
- Similar results for other spectral optimization problems.

Comparison in practice

- Greedy algorithm
  - Widely used in Practice
  - We showed it achieves $O(C^{k^2})$
- Local Search algorithm
  - Performs better than Greedy but runs ~4 times slower.
  - Achieves $O\left(k^{2k}\right)$
- This algorithm
  - Achieves $\tilde{O}\left(k^k\right)$
  - Performs worse than Local Search and runs slower.

Open Problems

- Other applications of Spectral Spanners?
- Analogue of k-spanners for graphs?
- Composable Core-sets for DPP sampling?

Open Problems

- Other applications of Spectral Spanners?
- Analogue of k-spanners for graphs?
- Composable Core-sets for DPP sampling?

THANK YOU!