

הזמנה

airou chshifa latvaar shni v'shliishi b'machlikot mdu'i ha'machshav, matematika, cimia v'fizika

יום רביעי 10.7.24 | 16:00-19:00

בבניין מדעי המחשב (503)

אוניברסיטת בר-אילן

בואו להכיר את מגוון מסלולי הלימוד
لتואר שני ושלישי (דוקטורט)
בפקולטה למדעים מדויקים



Metric Embeddings intro Trees

Arnold Filtser
Bar-Ilan University

July 02, 2024

Outline of the talk

- 1 Introduction
- 2 Stochastic embedding into trees
- 3 Ramsey type embeddings
- 4 Clan embedding
- 5 Group Steiner Tree
- 6 Conclusion

Metric space

A metric space is an ordered pair (X, d_X) , where X is a set and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function such that:

- ① **Identity:** $\forall x, y \in X, d_X(x, y) = 0 \iff x = y.$
- ② **Symmetry:** $\forall x, y \in X, d_X(x, y) = d_X(y, x).$
- ③ **Triangle-inequality:** $\forall x, y, z \in X, d_X(x, y) \leq d_X(x, z) + d_X(z, y).$

Metric space

A metric space is an ordered pair (X, d_X) , where X is a set and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function such that:

- ① **Identity:** $\forall x, y \in X, d_X(x, y) = 0 \iff x = y.$
- ② **Symmetry:** $\forall x, y \in X, d_X(x, y) = d_X(y, x).$
- ③ **Triangle-inequality:** $\forall x, y, z \in X, d_X(x, y) \leq d_X(x, z) + d_X(z, y).$

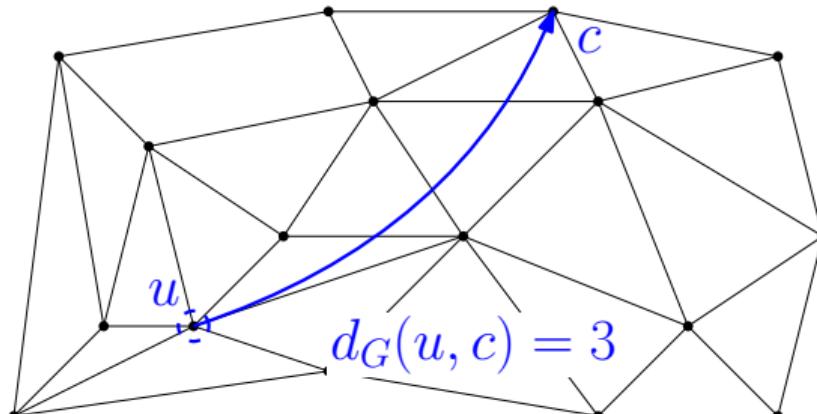
Examples:

- Euclidean space ℓ_2 in \mathbb{R}^d : $d_{\ell_2}(\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\|_2 = \sqrt{\sum_{i=1}^d (x_i - y_i)^2}.$
- Manhattan distance ℓ_1 in \mathbb{R}^d : $d_{\ell_1}(\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\|_1 = \sum_{i=1}^d |x_i - y_i|.$
- Edit distance: given two strings A, B how many edit operations
(insert, delete, substitute) are required to transform A to B ?
- Weighted graph $G = (V, E, w)$ with shortest path distance.

Metric space

A metric space is an ordered pair (X, d_X) , where X is a set and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function such that:

- ① **Identity:** $\forall x, y \in X, d_X(x, y) = 0 \iff x = y.$
- ② **Symmetry:** $\forall x, y \in X, d_X(x, y) = d_X(y, x).$
- ③ **Triangle-inequality:** $\forall x, y, z \in X, d_X(x, y) \leq d_X(x, z) + d_X(z, y).$
- Weighted graph $G = (V, E, w)$ with shortest path distance.



Metric space

A metric space is an ordered pair (X, d_X) , where X is a set and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function such that:

- ① **Identity:** $\forall x, y \in X, d_X(x, y) = 0 \iff x = y.$
- ② **Symmetry:** $\forall x, y \in X, d_X(x, y) = d_X(y, x).$
- ③ **Triangle-inequality:** $\forall x, y, z \in X, d_X(x, y) \leq d_X(x, z) + d_X(z, y).$

Many problems are defined w.r.t. metric spaces. Examples:

- k -center.
- Steiner tree.
- Metric TSP.

Metric space

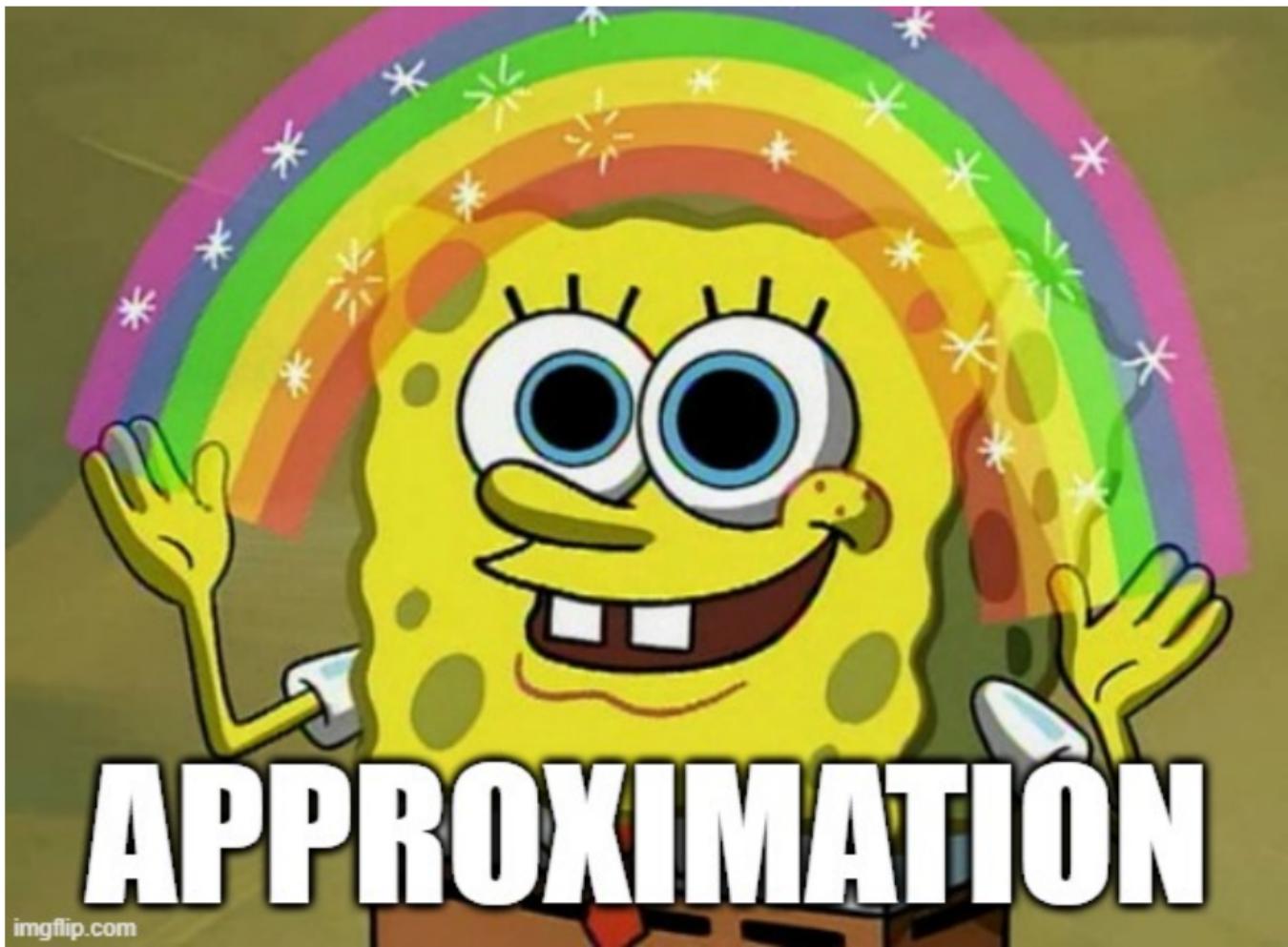
A metric space is an ordered pair (X, d_X) , where X is a set and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function such that:

- ① **Identity:** $\forall x, y \in X, d_X(x, y) = 0 \iff x = y.$
- ② **Symmetry:** $\forall x, y \in X, d_X(x, y) = d_X(y, x).$
- ③ **Triangle-inequality:** $\forall x, y, z \in X, d_X(x, y) \leq d_X(x, z) + d_X(z, y).$

Many problems are defined w.r.t. metric spaces. Examples:

- k -center.
- Steiner tree.
- Metric TSP.

Often these problems are NP-hard.

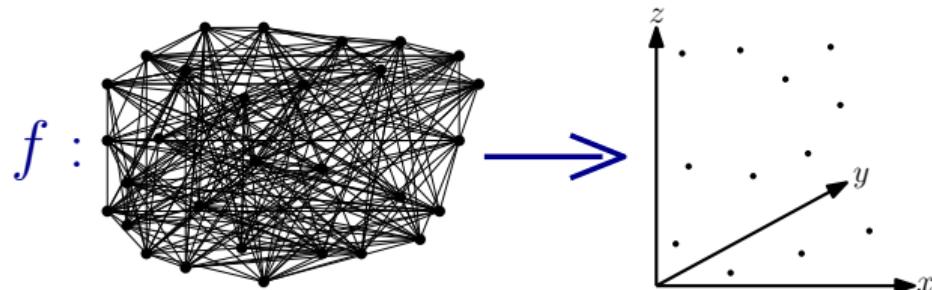


Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.

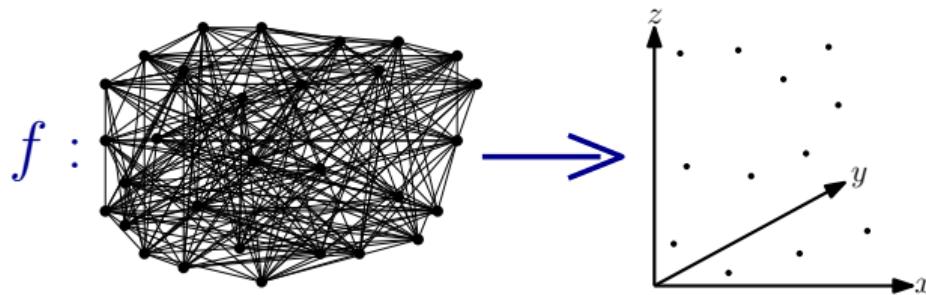


Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



Preserve (approximately) properties of the original space:

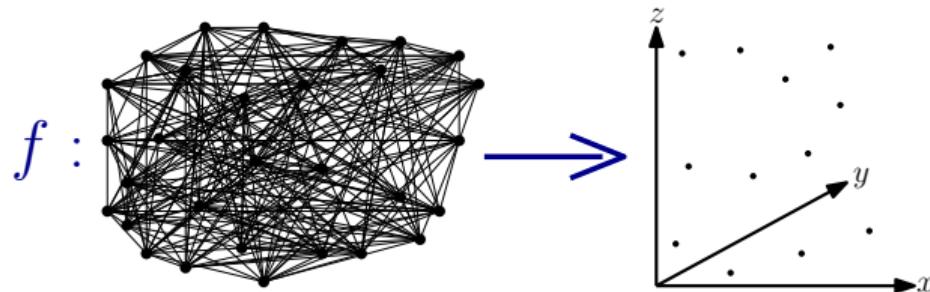
- Distances
- Cuts, Flows
- Commute time
- Effective resistance
- Clustering statistics.
- etc.

Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



f has **distortion** t if:

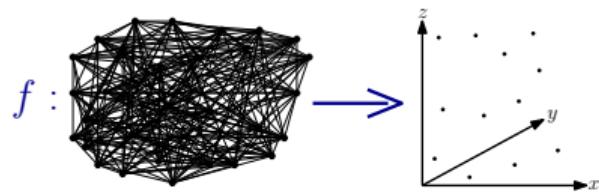
$$\forall x, y \in X, \quad d_X(x, y) \leq d_Y(f(x), f(y)) \leq t \cdot d_X(x, y).$$

Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



f has **distortion** t if:

$$\forall x, y \in X, d_X(x, y) \leq d_Y(f(x), f(y)) \leq t \cdot d_X(x, y) .$$

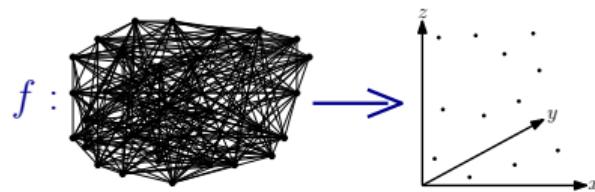
It is highly desirable that the target space Y will have **simple structure**.

Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



f has **distortion** t if:

$$\forall x, y \in X, d_X(x, y) \leq d_Y(f(x), f(y)) \leq t \cdot d_X(x, y) .$$

It is highly desirable that the target space Y will have **simple structure**.

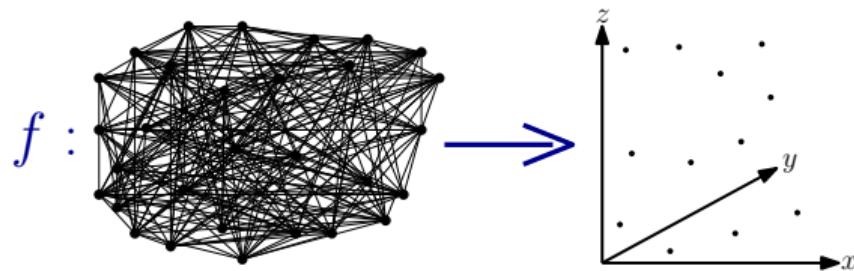
So that we could run efficient algorithms on it...

Metric Embeddings

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



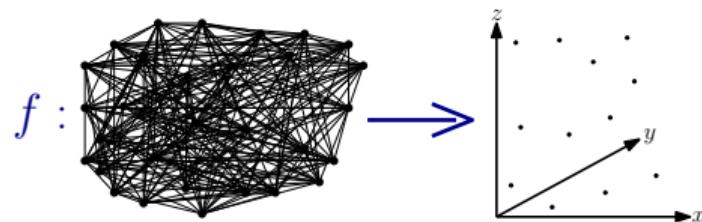
Theorem ([Bourgain 85])

Every n -point metric (X, d_X) is **embeddable** into Euclidean space $(\mathbb{R}^d, \|\cdot\|_2)$ with **distortion** $O(\log n)$.

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



Theorem ([Bourgain 85])

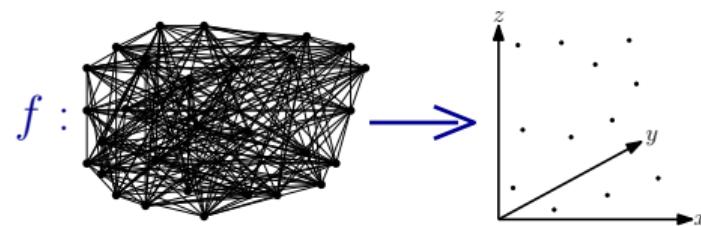
Every n -point metric (X, d_X) is **embeddable** into Euclidean space $(\mathbb{R}^d, \|\cdot\|_2)$ with **distortion** $O(\log n)$.

Theorem ([Linial, London, Rabinovich 95])

[Bou85] is tight.

Theorem ([Bourgain 85])

Every n -point metric (X, d_X) is **embeddable** into Euclidean space $(\mathbb{R}^d, \|\cdot\|_2)$ with **distortion** $O(\log n)$.

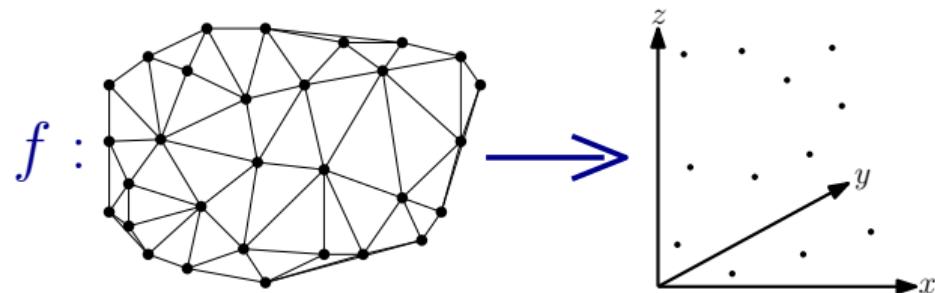


Applications:

- Approximation algorithms (e.g. **sparsest cut**, min graph bandwidth)
- Parallel computation (e.g. SSSP in MPC)
- Computational Biology (e.g. clustering and detecting protein seq.)
- etc.

Theorem ([Rao 99])

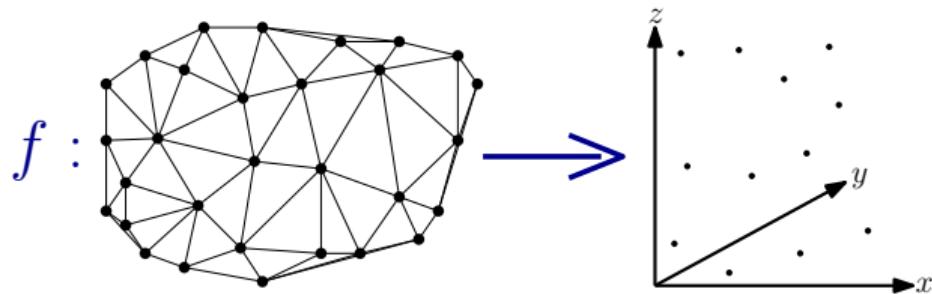
Every n -point **planar metric** (X, d_X) is **embeddable** into Euclidean space $(\mathbb{R}^d, \|\cdot\|_2)$ with **distortion** $O(\sqrt{\log n})$.



Planar metric- the shortest path metric of a planar graph.

Theorem ([Rao 99])

Every n -point **planar metric** (X, d_X) is **embeddable** into Euclidean space $(\mathbb{R}^d, \|\cdot\|_2)$ with **distortion** $O(\sqrt{\log n})$.



Planar metric- the shortest path metric of a planar graph.

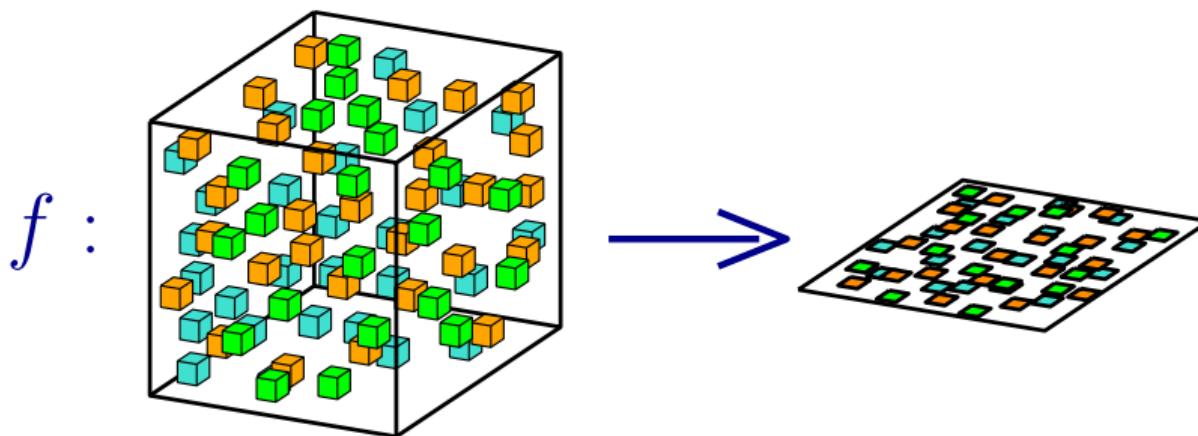
Theorem ([Newman, Rabinovich 03])

[Rao99] is tight.

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



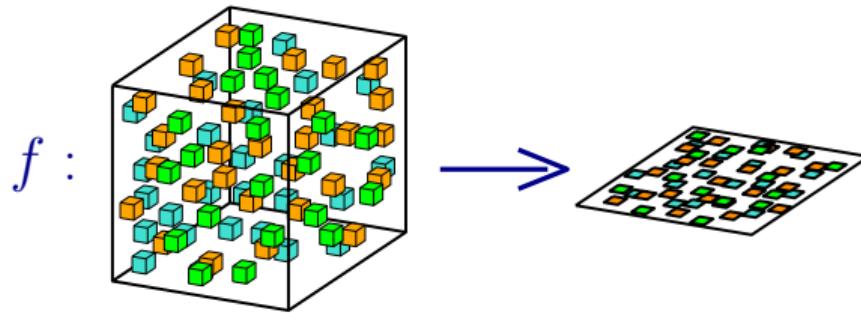
Theorem ([Johnson, Lindenstrauss 84], Dimension Reduction)

$X \subset (\mathbb{R}^d, \|\cdot\|_2)$ set of size n . Then X embeds into $O(\log n/\epsilon^2)$ dimensional Euclidean space with distortion $1 + \epsilon$.

Embedding

$(X, d_X), (Y, d_Y)$ metric spaces.

$f : (X, d_X) \rightarrow (Y, d_Y)$ is called an **embedding**.



Theorem ([Johnson, Lindenstrauss 84], Dimension Reduction)

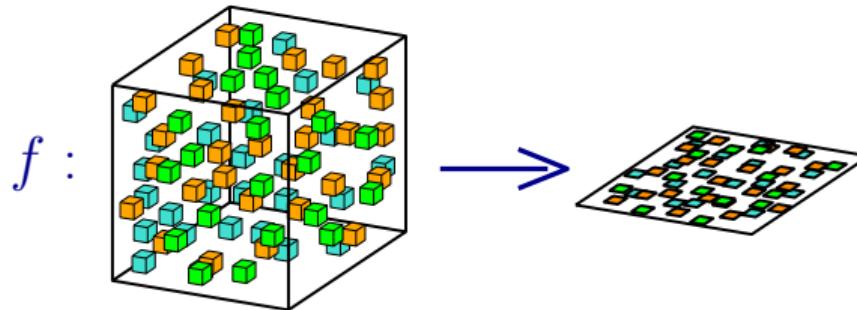
$X \subset (\mathbb{R}^d, \|\cdot\|_2)$ set of size n . Then X embeds into $O(\log n/\epsilon^2)$ dimensional Euclidean space with distortion $1 + \epsilon$.

Theorem ([Green Larsen, Nelson 17])

[JL84] is tight.

Theorem ([Johnson, Lindenstrauss 84], Dimension Reduction)

$X \subset (\mathbb{R}^d, \|\cdot\|_2)$ set of size n . Then X embeds into $O(\log n/\epsilon^2)$ dimensional Euclidean space with distortion $1 + \epsilon$.



Applications:

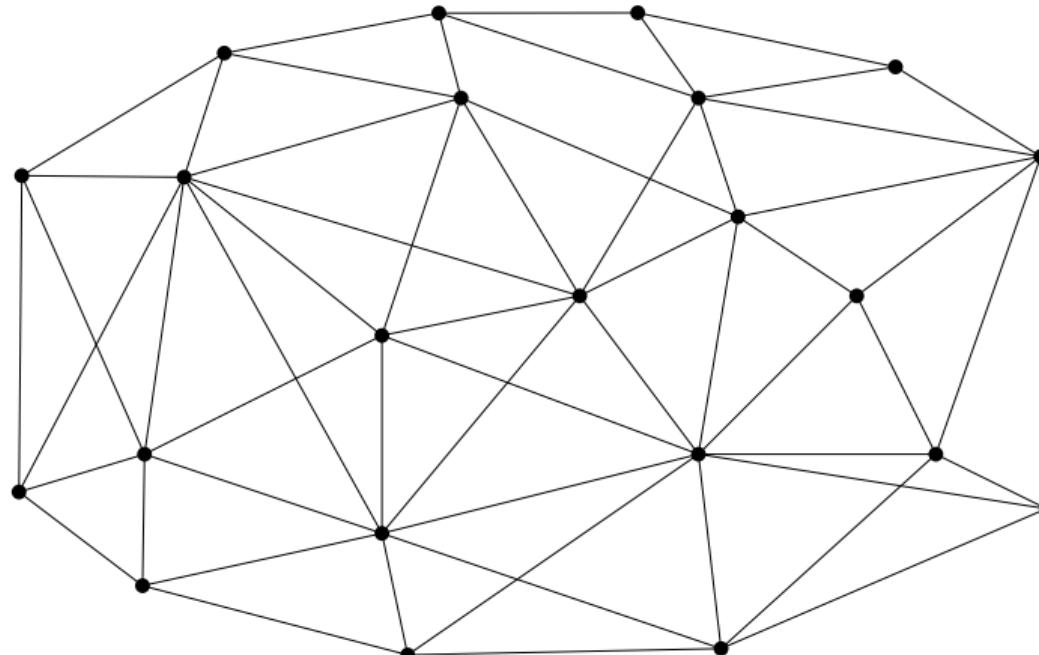
- Speeding up-computation
- Clustering
- Nearest Neighbor Search
- Machine Learning
- etc.

Graph Spanners

$G = (V, E, w)$ weighted graph, a **t -spanner** is a **subgraph** $H = (V, E_H)$

s.t.

$$\forall u, v \in V, \quad d_H(u, v) \leq t \cdot d_G(u, v)$$

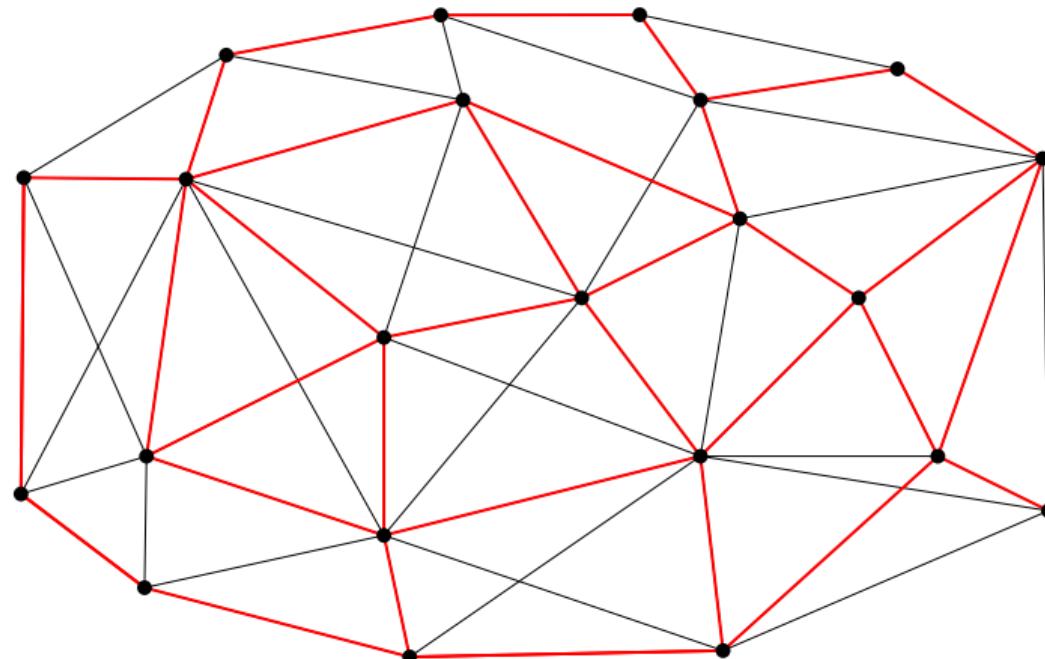


Graph Spanners

$G = (V, E, w)$ weighted graph, a **t -spanner** is a **subgraph** $H = (V, E_H)$

s.t.

$$\forall u, v \in V, \quad d_H(u, v) \leq t \cdot d_G(u, v)$$

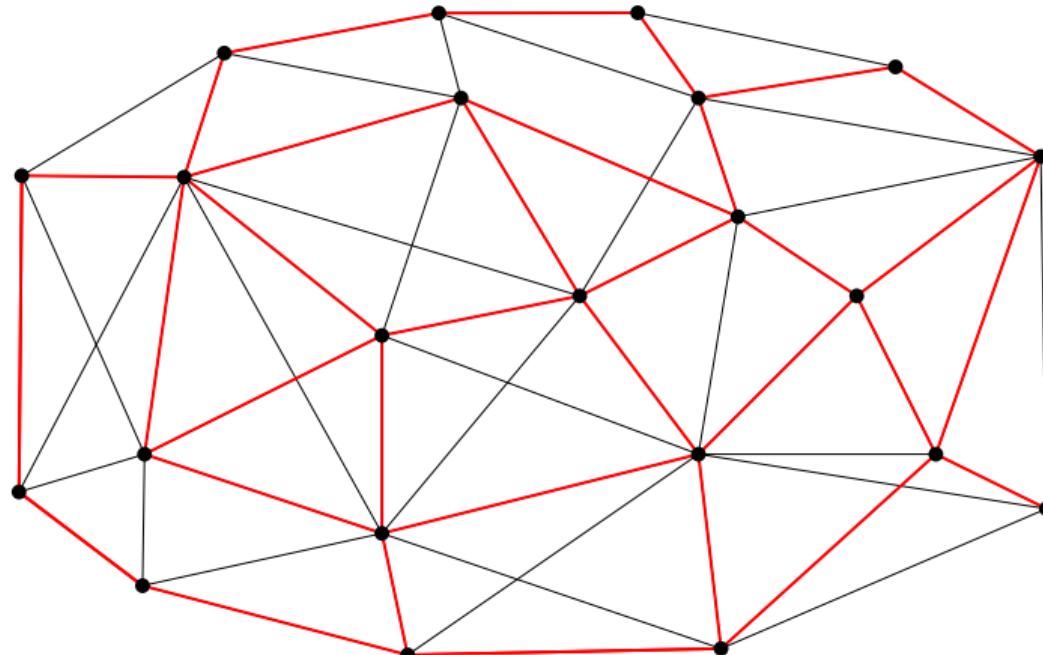


Graph Spanners

$G = (V, E, w)$ weighted graph, a **t -spanner** is a **subgraph** $H = (V, E_H)$

s.t.

$$\forall u, v \in V, \quad d_H(u, v) \leq t \cdot d_G(u, v)$$



Stretch

t

Sparsity

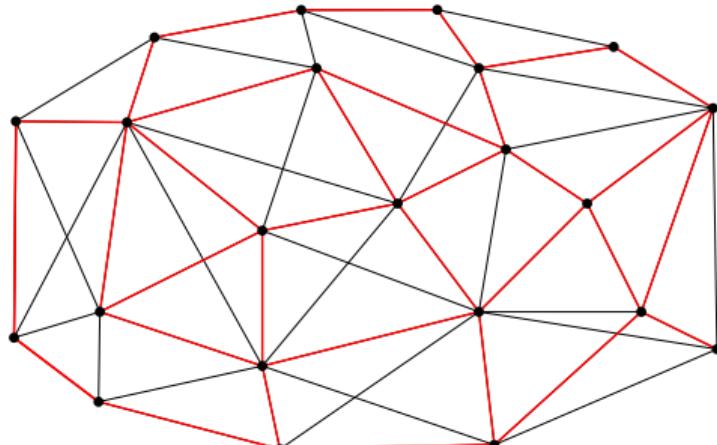
$|H|$

Graph Spanners

$G = (V, E, w)$ weighted graph, a **t -spanner** is a **subgraph** $H = (V, E_H)$

s.t.

$$\forall u, v \in V, \quad d_H(u, v) \leq t \cdot d_G(u, v)$$



Stretch

t

Sparsity

$|H|$

[Althofer, Das, Dobkin, Joseph, Soares 93]:

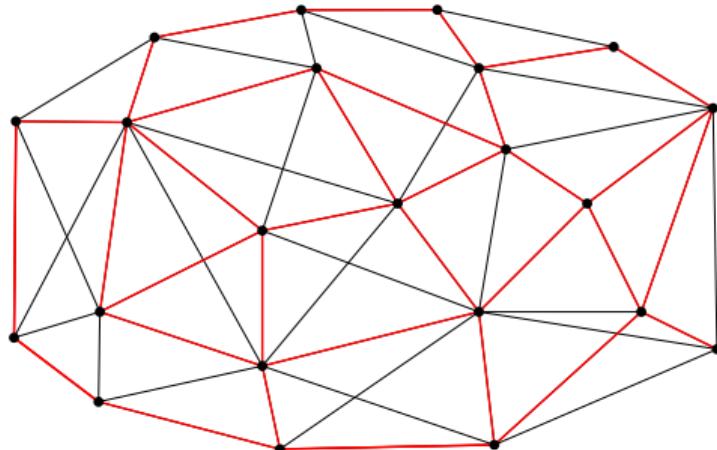
For $k \geq 1$, every graph admits $2k - 1$ spanner with $O(n^{1+\frac{1}{k}})$ edges.

Graph Spanners

$G = (V, E, w)$ weighted graph, a **t -spanner** is a **subgraph** $H = (V, E_H)$

s.t.

$$\forall u, v \in V, \quad d_H(u, v) \leq t \cdot d_G(u, v)$$



Stretch

t

Sparsity

$|H|$

[Althofer, Das, Dobkin, Joseph, Soares 93]:

For $k \geq 1$, every graph admits $2k - 1$ spanner with $O(n^{1+\frac{1}{k}})$ edges.

Tight. (assuming Erdős' girth conjecture).

Graph Spanners

[Althofer, Das, Dobkin, Joseph, Soares 93]:

For $k \geq 1$, every graph admits $2k - 1$ spanner with $O(n^{1+\frac{1}{k}})$ edges.

Tight. (assuming Erdős' girth conjecture).

Applications:

- Approximation Algorithms (e.g. PTAS for TSP)
- Distributed Computing
- Network Routing
- Computational Biology (e.g. measure genetic distance)
- etc.

Outline of the talk

- 1 Introduction
- 2 Stochastic embedding into trees
- 3 Ramsey type embeddings
- 4 Clan embedding
- 5 Group Steiner Tree
- 6 Conclusion

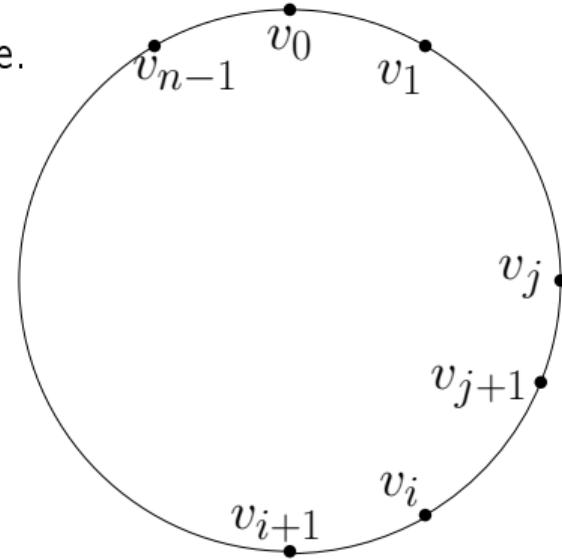
Embedding into Trees

Tree is very **simple** and **desirable** target space.

Embedding into Trees

Tree is very **simple** and **desirable** target space.

Embedding C_n **requires** distortion $n - 1$.

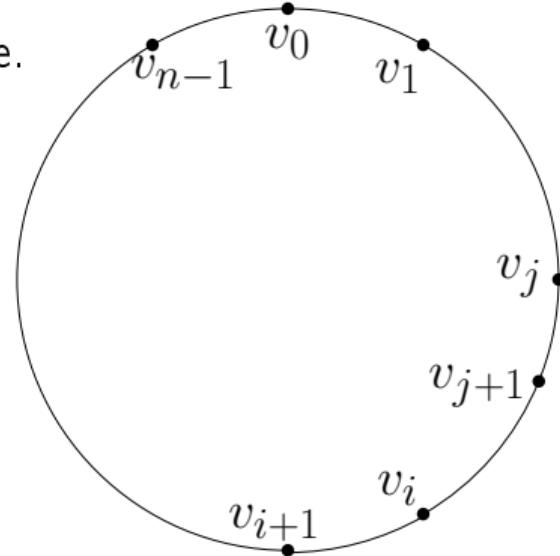


Embedding into Trees

Tree is very **simple** and **desirable** target space.

Embedding C_n **requires** distortion $n - 1$.

What if we delete a **random** edge \tilde{e} ?



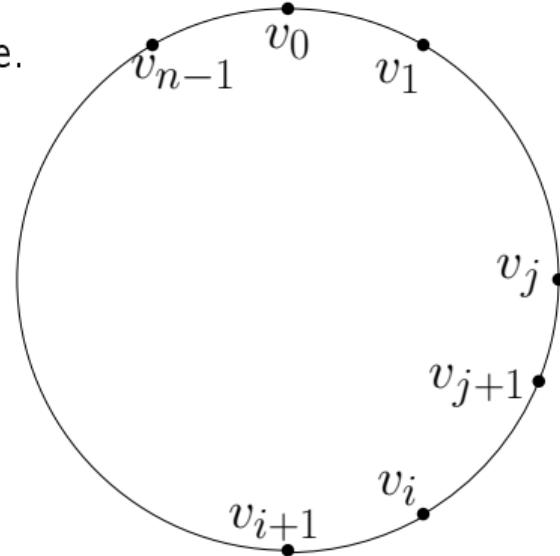
Embedding into Trees

Tree is very **simple** and **desirable** target space.

Embedding C_n **requires** distortion $n - 1$.

What if we delete a **random** edge \tilde{e} ?

$$\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})]$$

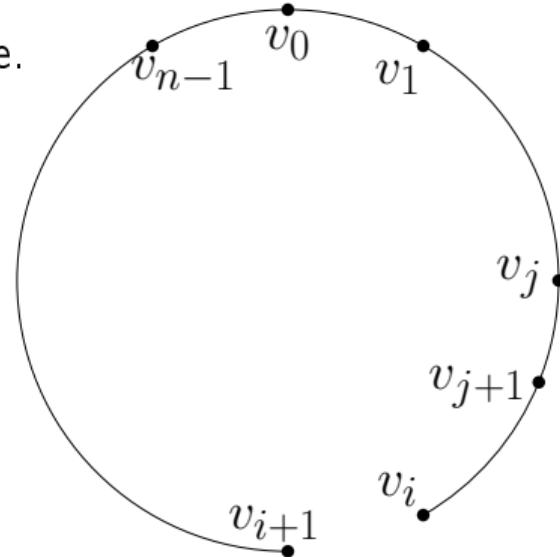


Embedding into Trees

Tree is very **simple** and **desirable** target space.

Embedding C_n **requires** distortion $n - 1$.

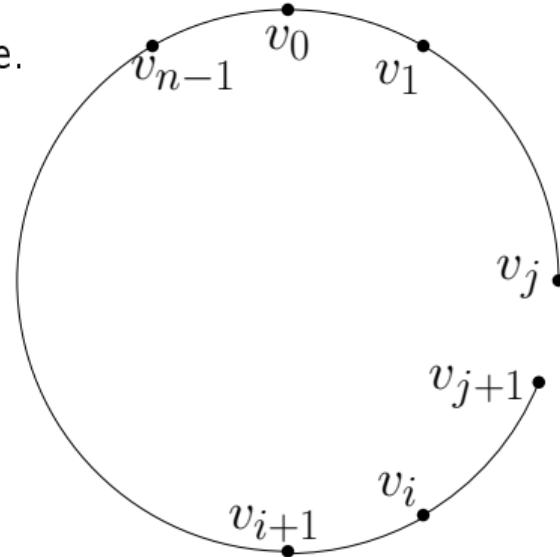
What if we delete a **random** edge \tilde{e} ?



$$\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})] = \Pr[\tilde{e} = \{v_i, v_{i+1}\}] \cdot (n - 1) + \Pr[\tilde{e} \neq \{v_i, v_{i+1}\}] \cdot 1$$

Embedding into Trees

Tree is very **simple** and **desirable** target space.



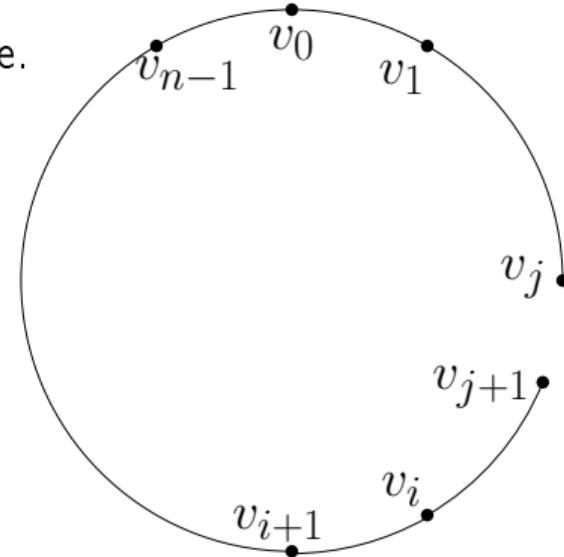
Embedding C_n **requires** distortion $n - 1$.

What if we delete a **random** edge \tilde{e} ?

$$\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})] = \Pr[\tilde{e} = \{v_i, v_{i+1}\}] \cdot (n - 1) + \Pr[\tilde{e} \neq \{v_i, v_{i+1}\}] \cdot 1$$

Embedding into Trees

Tree is very **simple** and **desirable** target space.



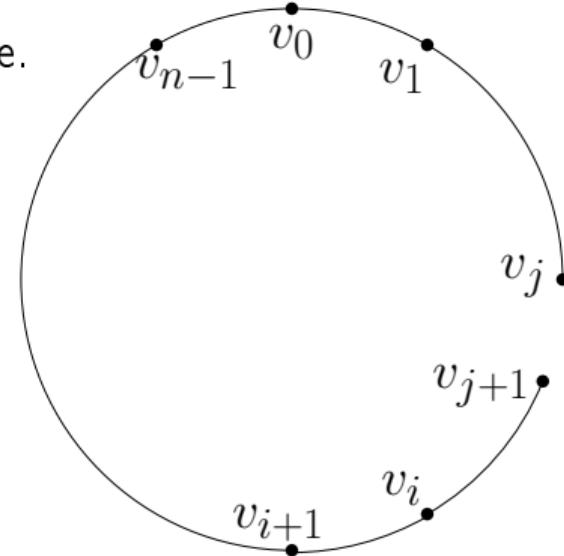
Embedding C_n **requires** distortion $n - 1$.

What if we delete a **random** edge \tilde{e} ?

$$\begin{aligned}\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})] &= \Pr[\tilde{e} = \{v_i, v_{i+1}\}] \cdot (n - 1) + \Pr[\tilde{e} \neq \{v_i, v_{i+1}\}] \cdot 1 \\ &= \frac{1}{n} \cdot (n - 1) + \frac{n - 1}{n} \cdot 1\end{aligned}$$

Embedding into Trees

Tree is very **simple** and **desirable** target space.



Embedding C_n **requires** distortion $n - 1$.

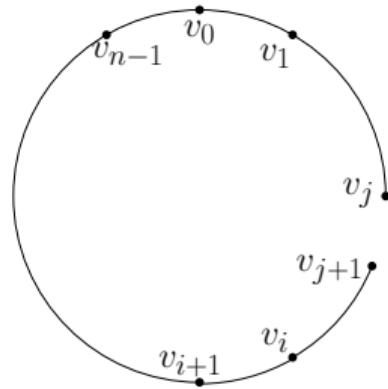
What if we delete a **random** edge \tilde{e} ?

$$\begin{aligned}\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})] &= \Pr[\tilde{e} = \{v_i, v_{i+1}\}] \cdot (n - 1) + \Pr[\tilde{e} \neq \{v_i, v_{i+1}\}] \cdot 1 \\ &= \frac{1}{n} \cdot (n - 1) + \frac{n - 1}{n} \cdot 1 = \frac{2(n - 1)}{n} < 2.\end{aligned}$$

Embedding into Trees

Embedding C_n requires distortion $n - 1$.

What if we delete a **random** edge \tilde{e} ?



$$\begin{aligned}\mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_{i+1})] &= \Pr[\tilde{e} = \{v_i, v_{i+1}\}] \cdot (n - 1) + \Pr[\tilde{e} \neq \{v_i, v_{i+1}\}] \cdot 1 \\ &= \frac{1}{n} \cdot (n - 1) + \frac{n - 1}{n} \cdot 1 = \frac{2(n - 1)}{n} < 2.\end{aligned}$$

By triangle inequality and linearity of expectation

$$\forall v_i, v_j, \quad \mathbb{E}_{T \sim \mathcal{D}}[d_T(v_i, v_j)] = \sum_{q=i}^{j-1} \mathbb{E}_{T \sim \mathcal{D}}[d_T(v_q, v_{q+1 \pmod n})] \leq 2 \cdot d_{C_n}(v_i, v_j).$$

Stochastic Embedding into Trees

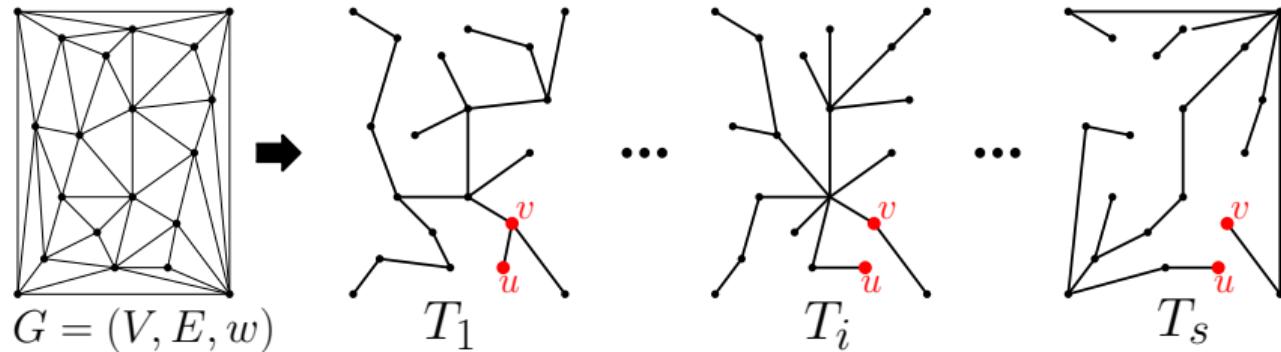
Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

*Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .*

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

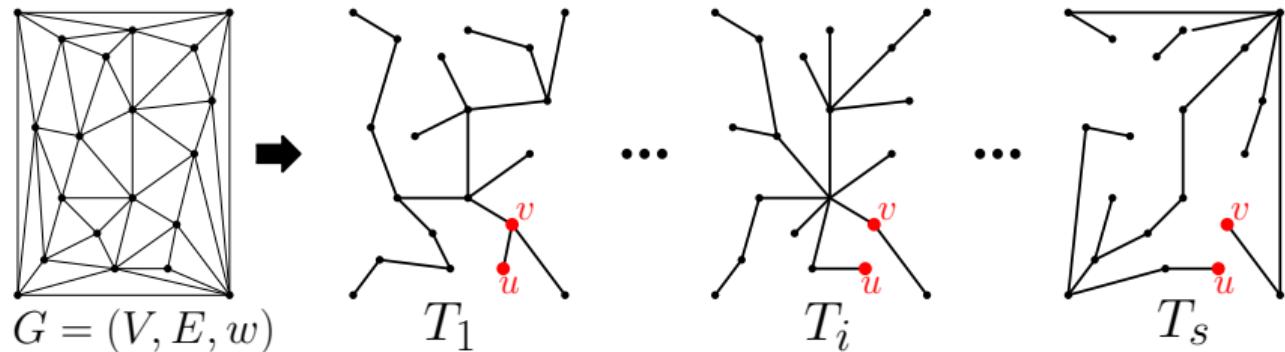
Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .



Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .

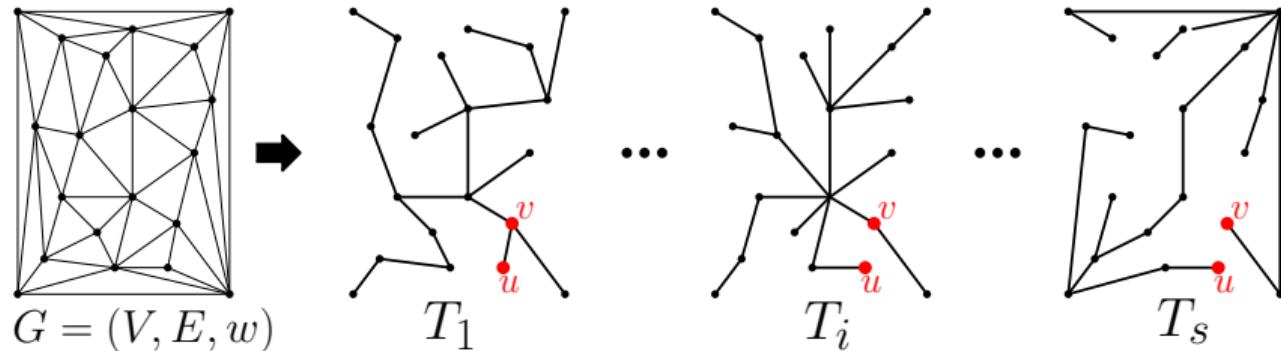


For every $u, v \in X$ and $T \in \text{supp}(\mathcal{D})$, $d_X(u, v) \leq d_T(f(u), f(v))$.

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion** $O(\log n)$.



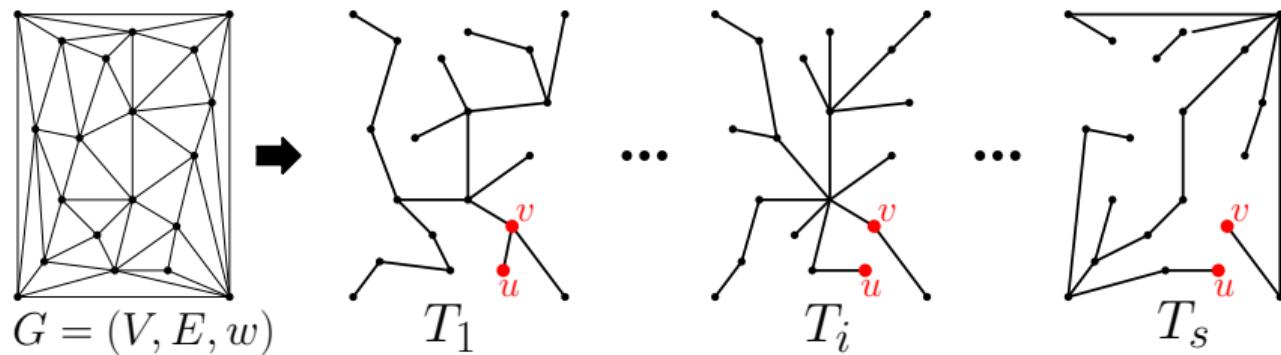
For every $u, v \in X$ and $T \in \text{supp}(\mathcal{D})$, $d_X(u, v) \leq d_T(f(u), f(v))$.

For every $u, v \in X$ $\mathbb{E}_{T \sim \mathcal{D}}[d_T(f(u), f(v))] \leq O(\log n) \cdot d_X(u, v)$.

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .



For every $u, v \in X$ and $T \in \text{supp}(\mathcal{D})$, $d_X(u, v) \leq d_T(f(u), f(v))$.

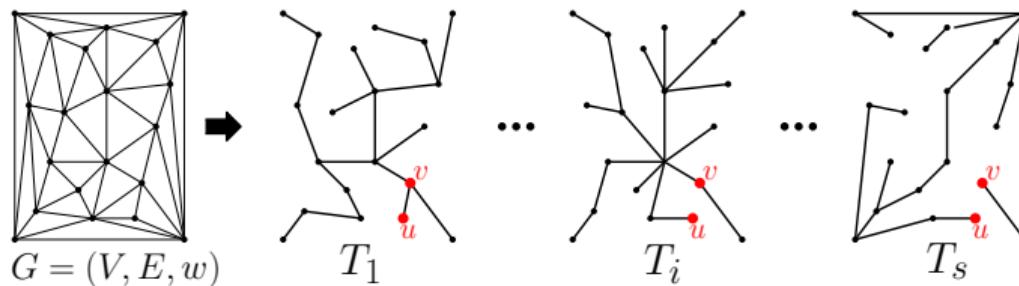
For every $u, v \in X$ $\mathbb{E}_{T \sim \mathcal{D}}[d_T(f(u), f(v))] \leq O(\log n) \cdot d_X(u, v)$.

[Bartal 96]: Tight!

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .



A useful hammer



Transforms arbitrary metric into a tree!

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

*Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .*

A useful hammer



Transforms arbitrary metric into a tree!

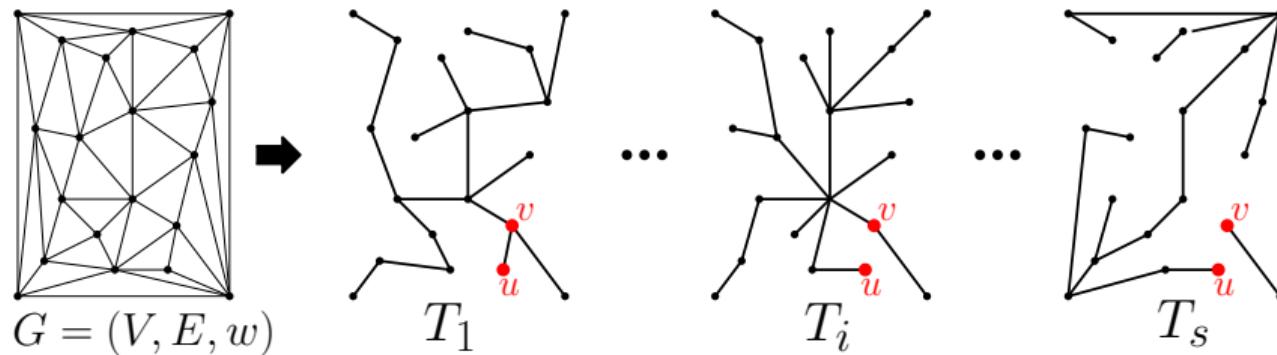
Applications:

- Approximation Algorithms.
- **Online Algorithms.**
- Distributed Computing.
- etc.

Stochastic Embedding into Trees

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion $O(\log n)$** .

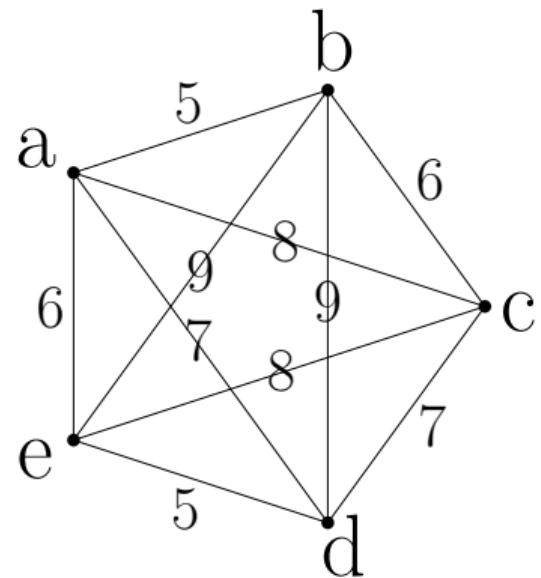


Compromises: Gurantee only in **expectation**



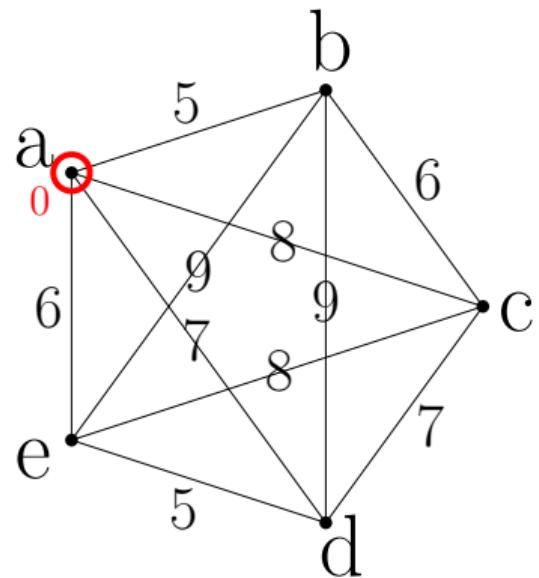
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.



Online problem - Metrical Task System (MTS)

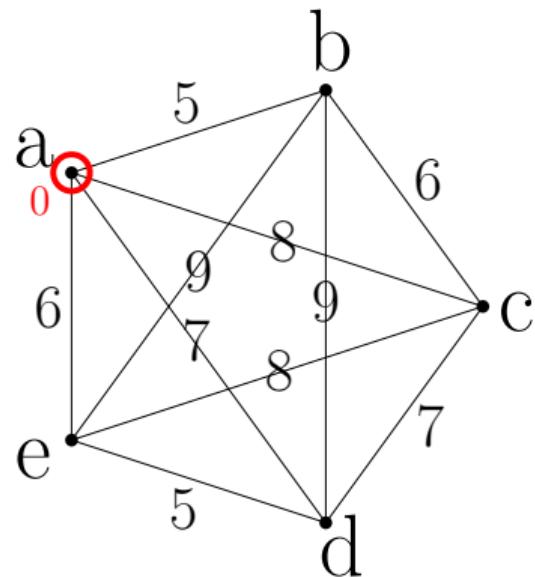
Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.



Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.



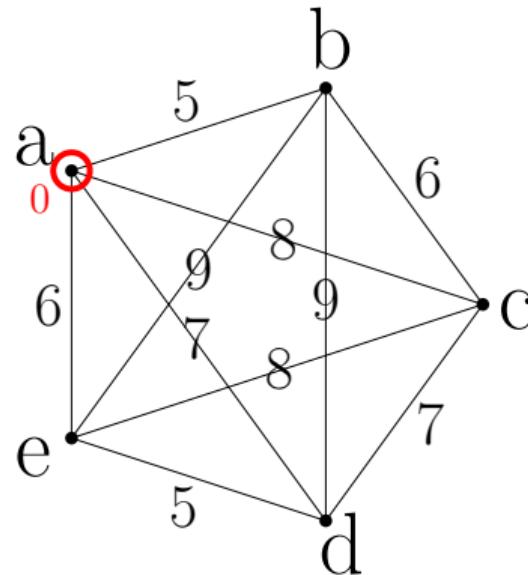
$$\begin{aligned}f_1(a) &= 3 \\f_1(b) &= 1 \\f_1(c) &= 2 \\f_1(d) &= 4 \\f_1(e) &= 1\end{aligned}$$

Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.



$$\begin{aligned}f_1(a) &= 3 \\f_1(b) &= 1 \\f_1(c) &= 2 \\f_1(d) &= 4 \\f_1(e) &= 1\end{aligned}$$

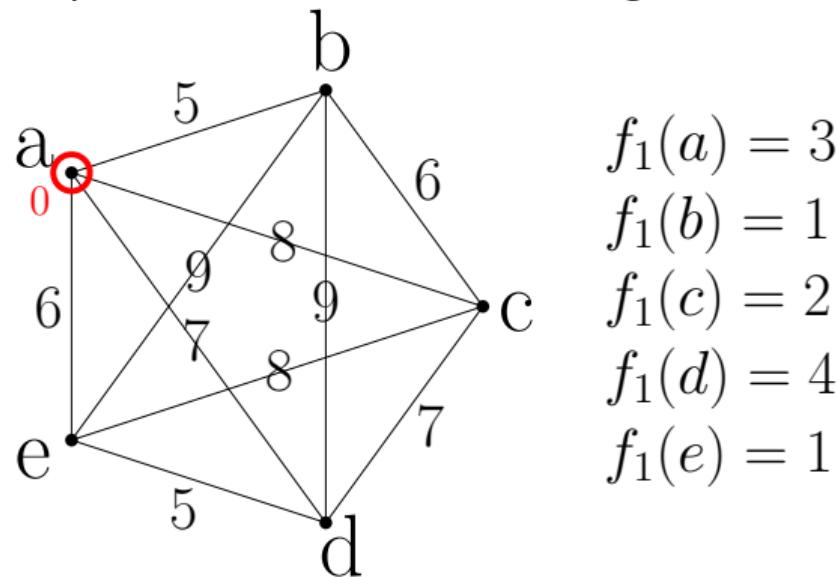
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



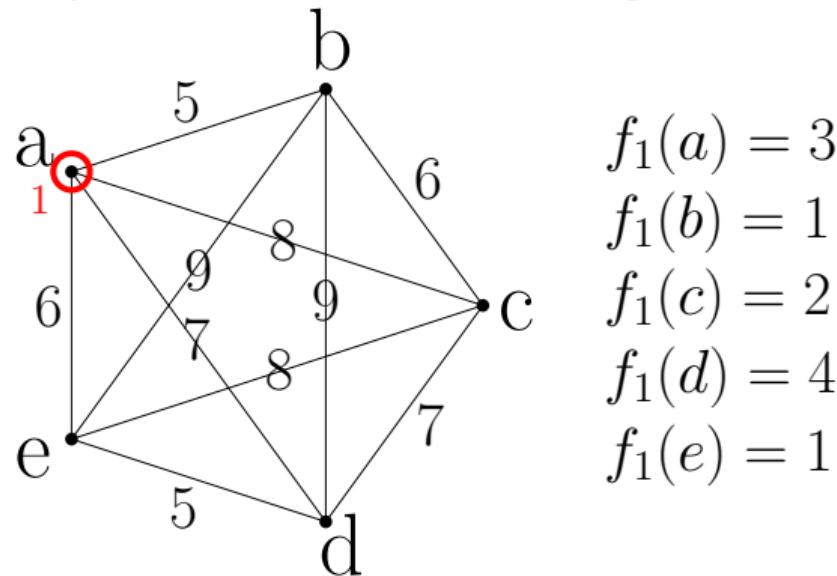
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



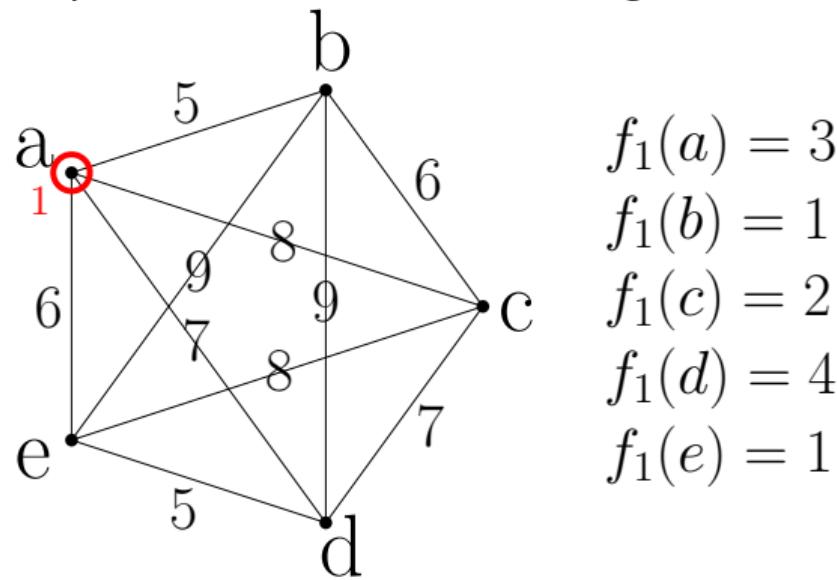
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



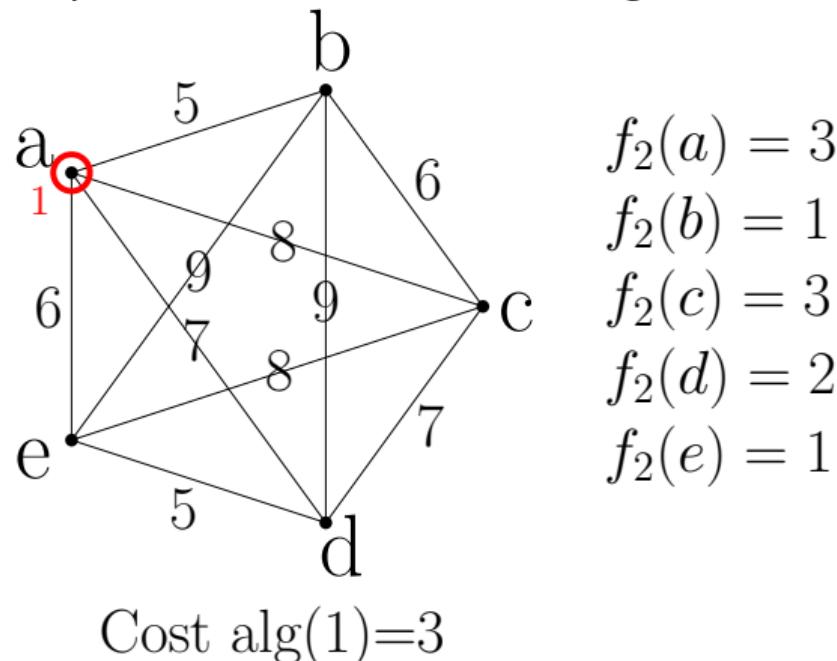
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



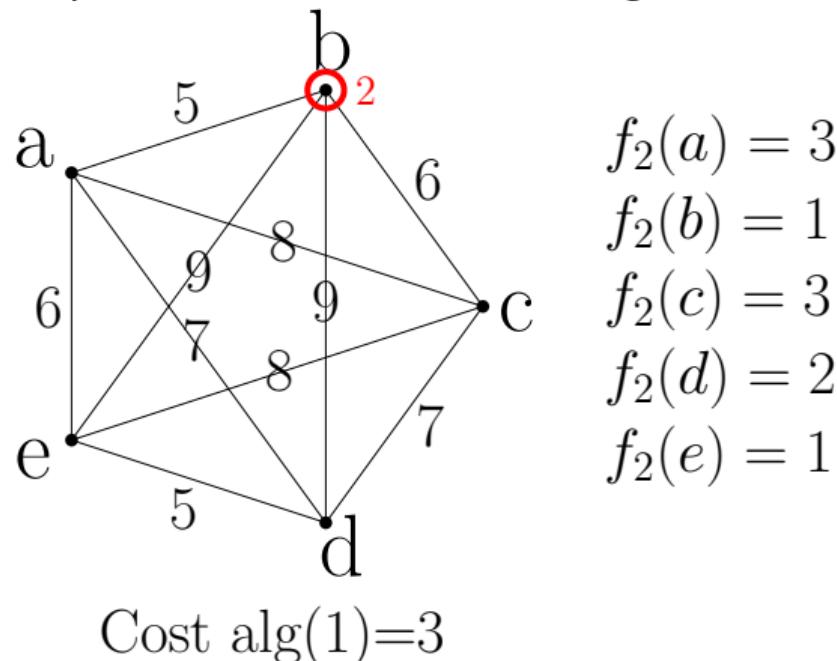
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



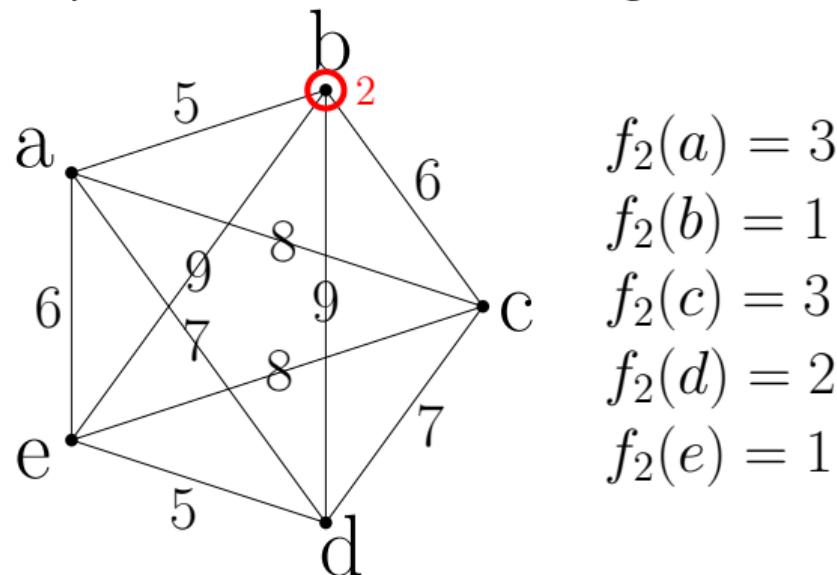
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

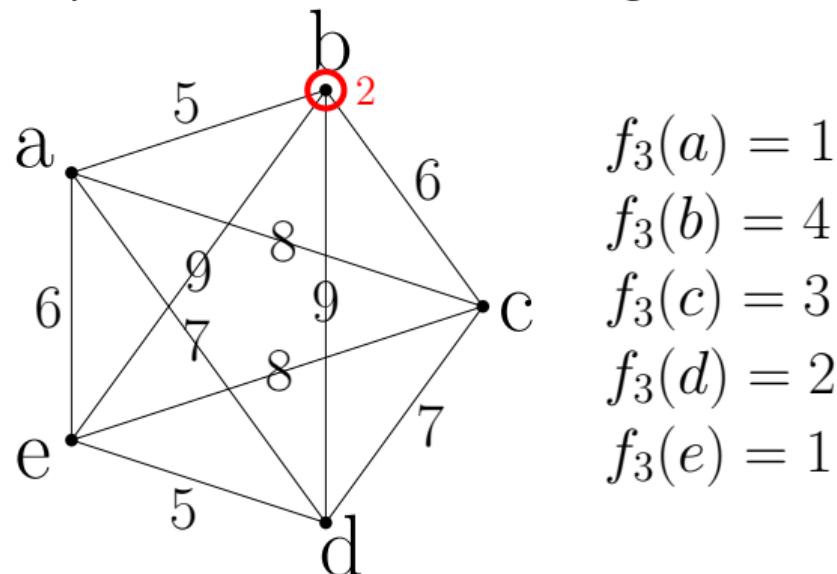
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

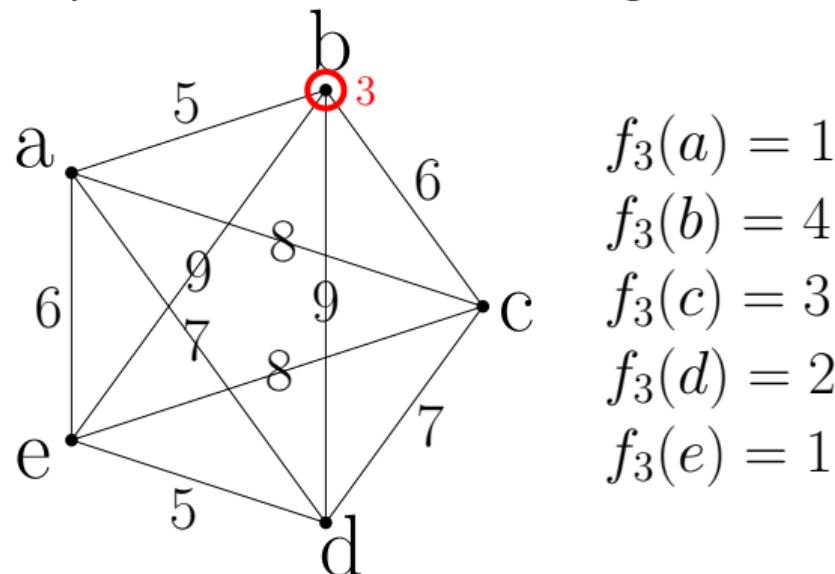
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

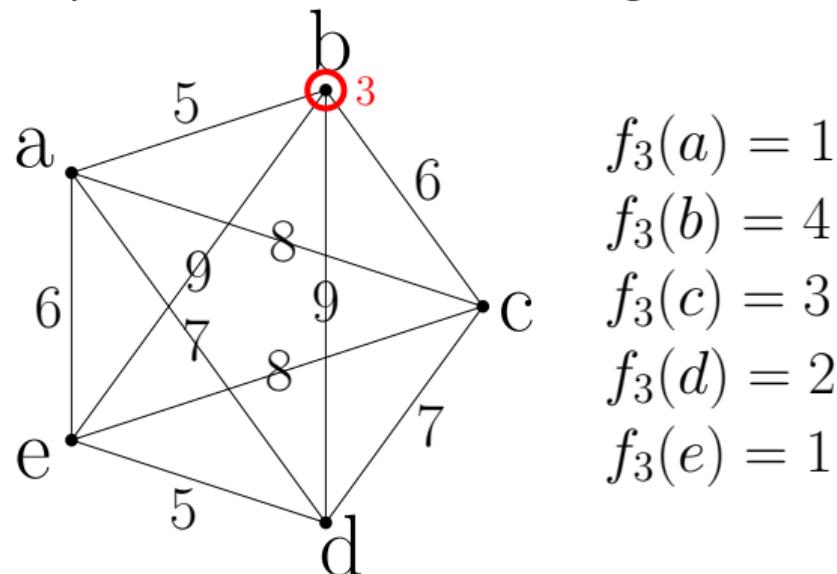
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(3) = 9 + 4 = 13$$

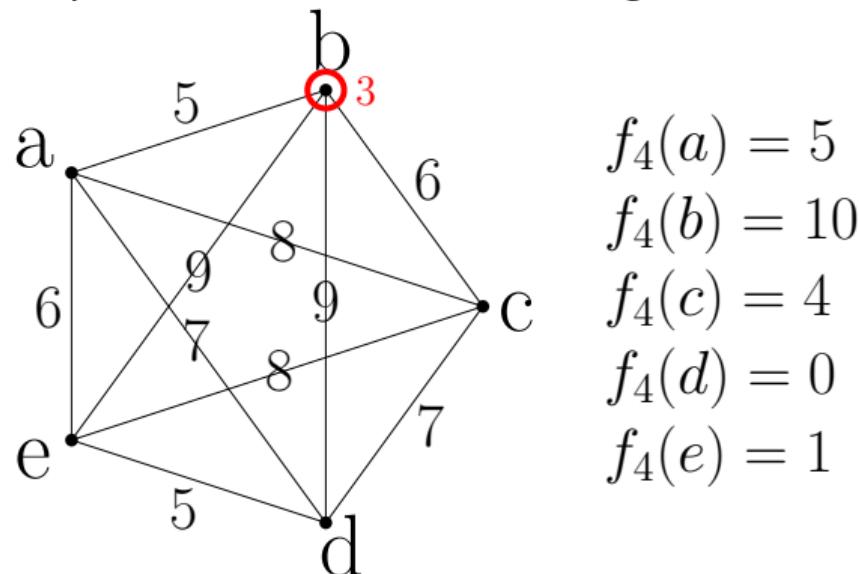
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(3) = 9 + 4 = 13$$

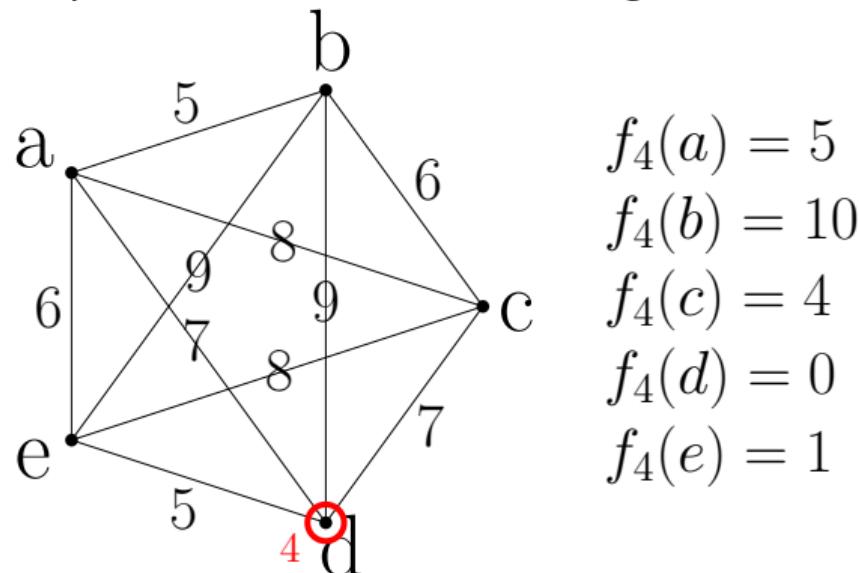
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(3) = 9 + 4 = 13$$

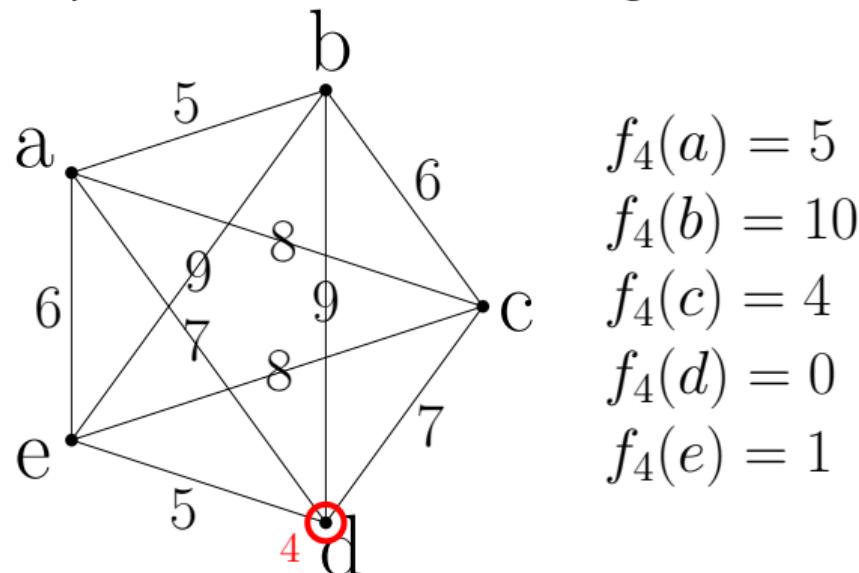
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

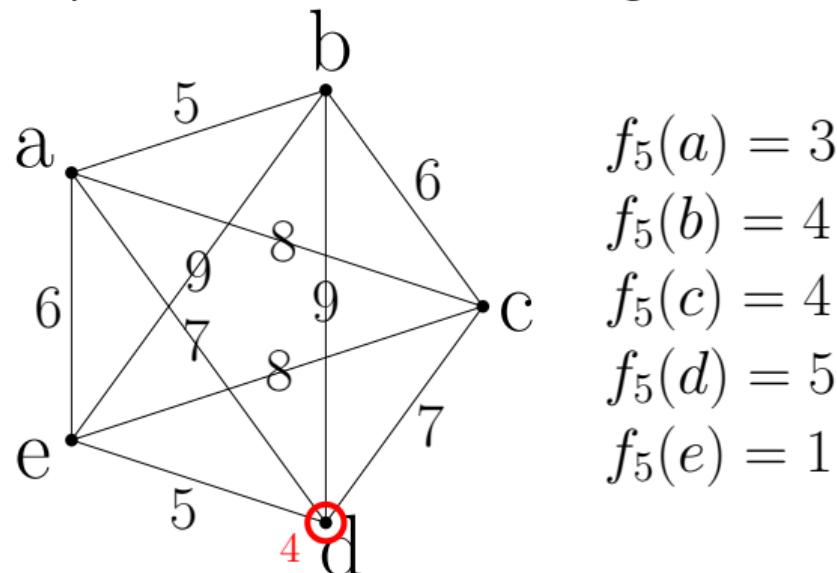
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

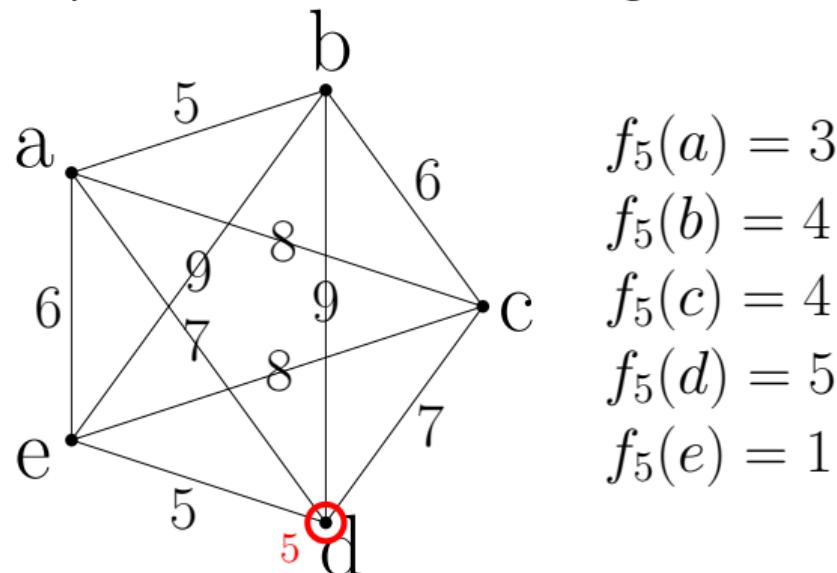
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

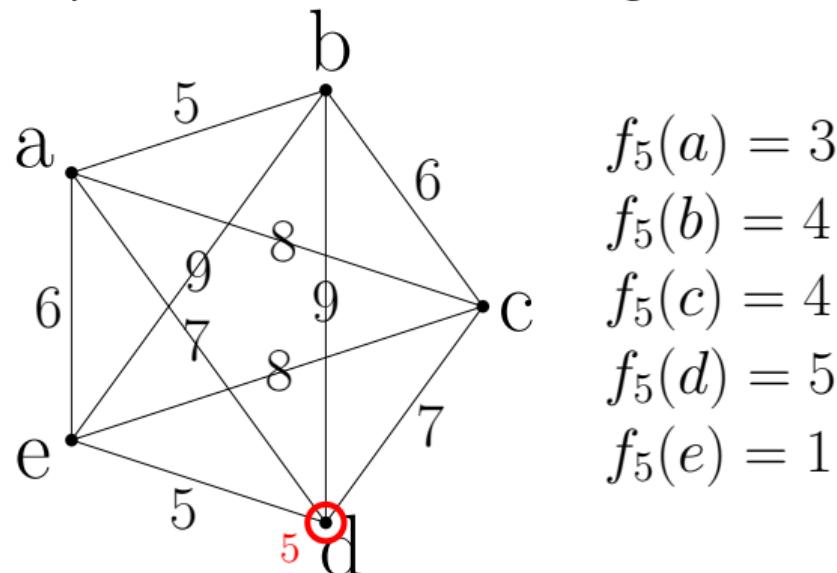
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(5) = 22 + 5 = 27$$

Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

What about Opt?

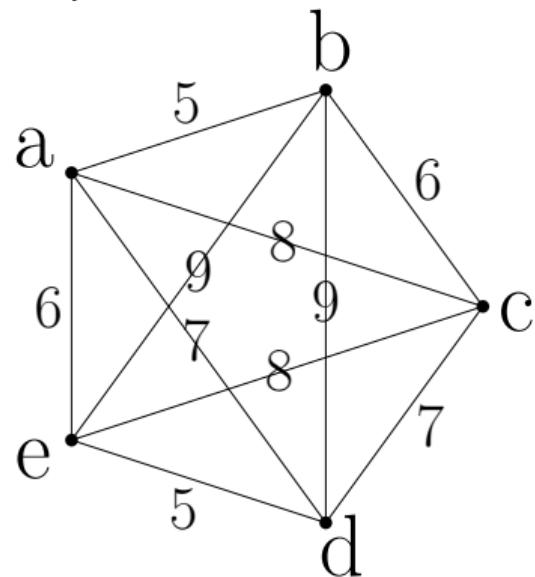
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



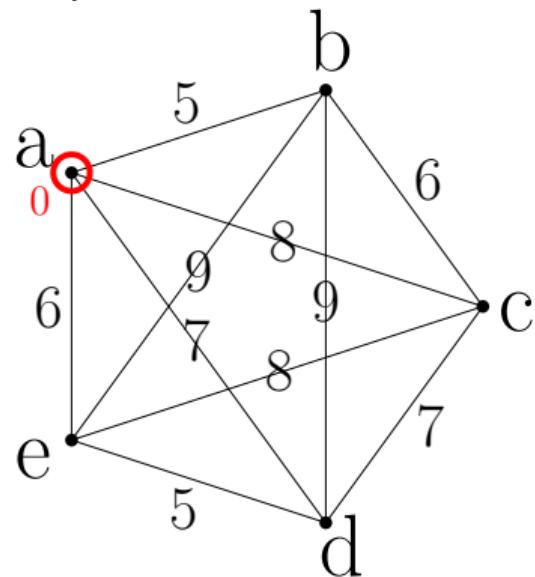
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



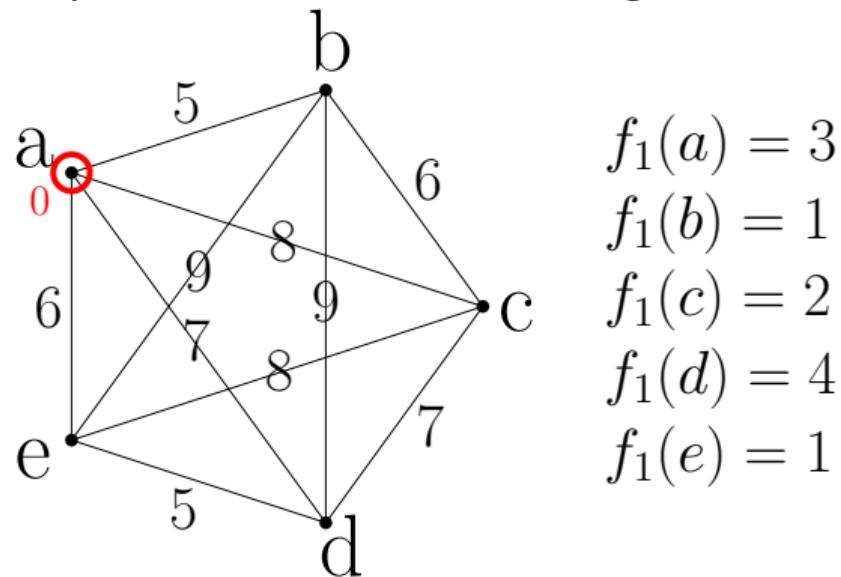
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



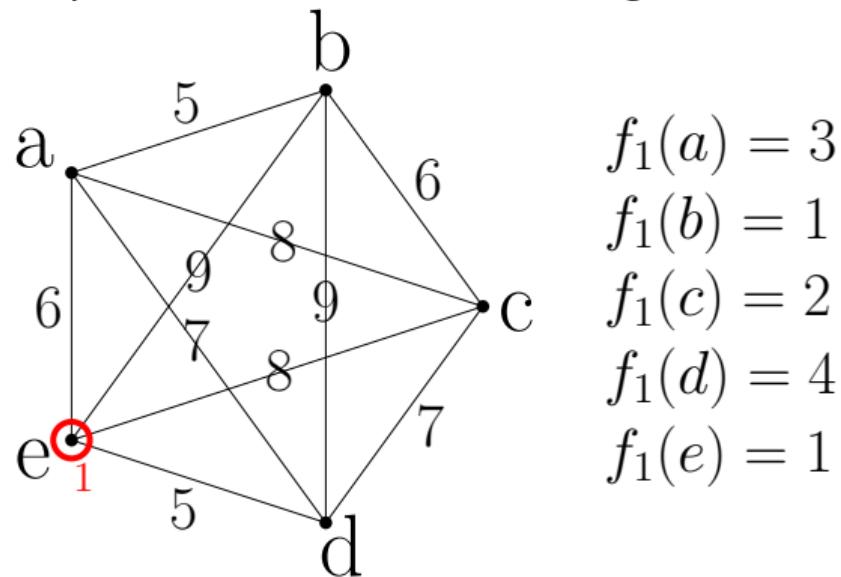
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



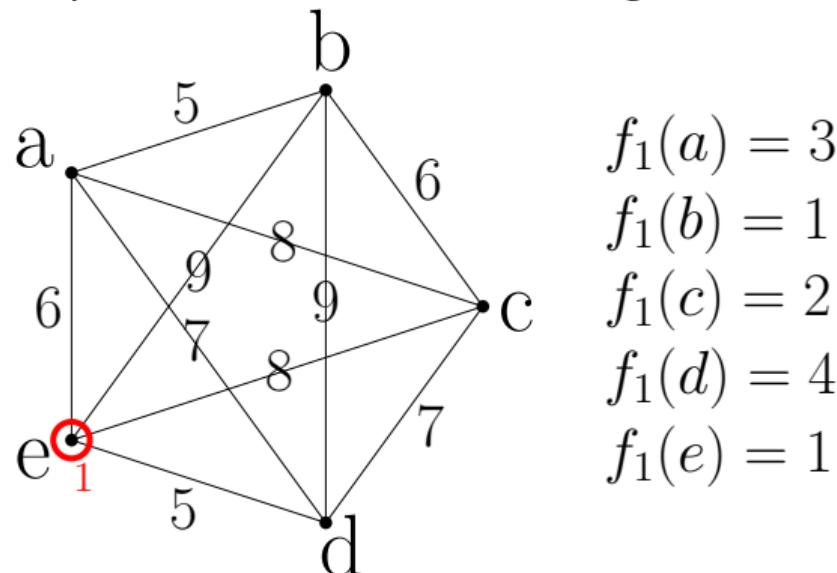
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(1) = 6 + 1 = 7$$

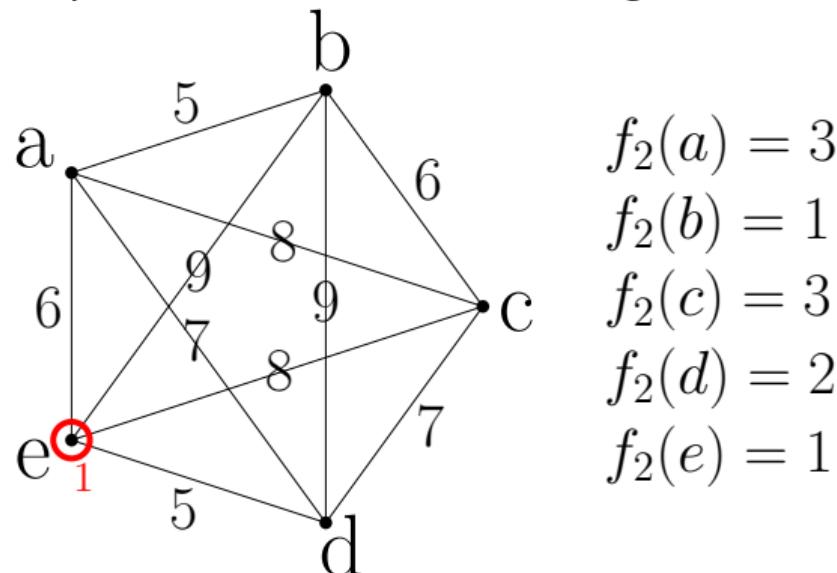
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(1) = 6 + 1 = 7$$

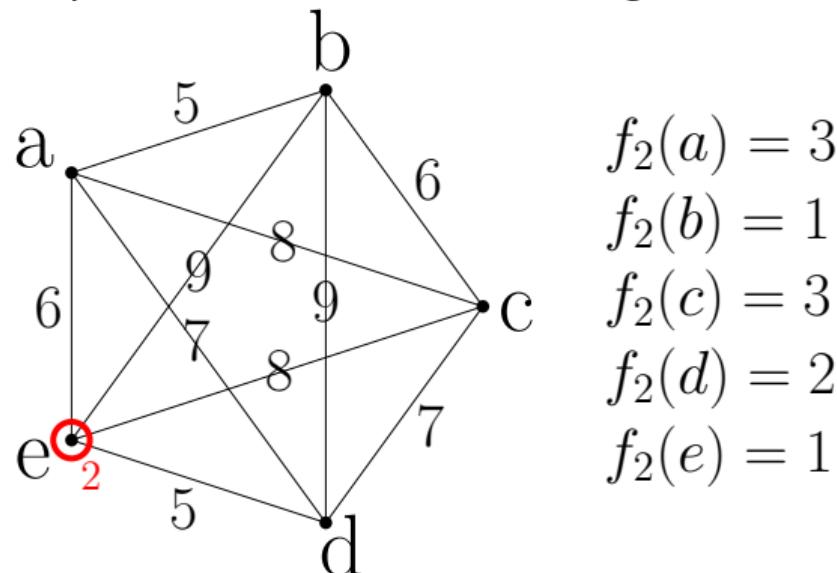
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(1) = 6 + 1 = 7$$

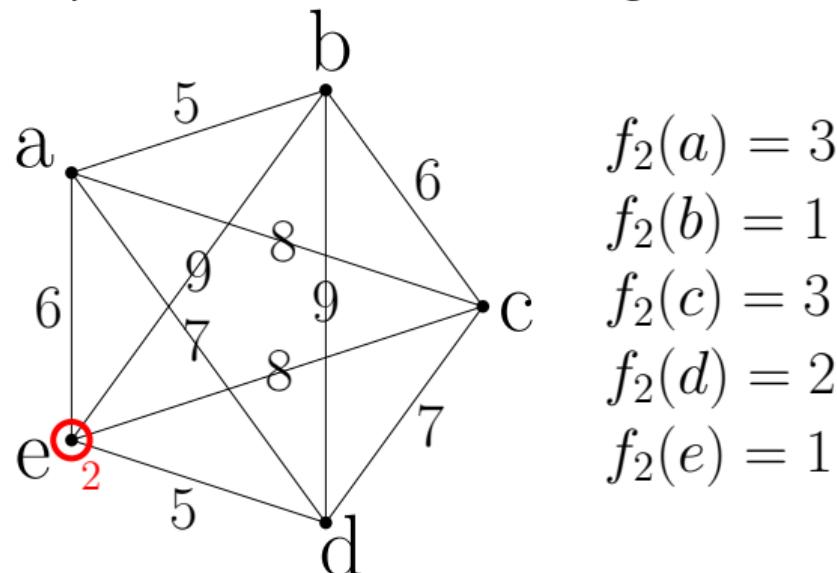
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(2) = 7 + 1 = 8$$

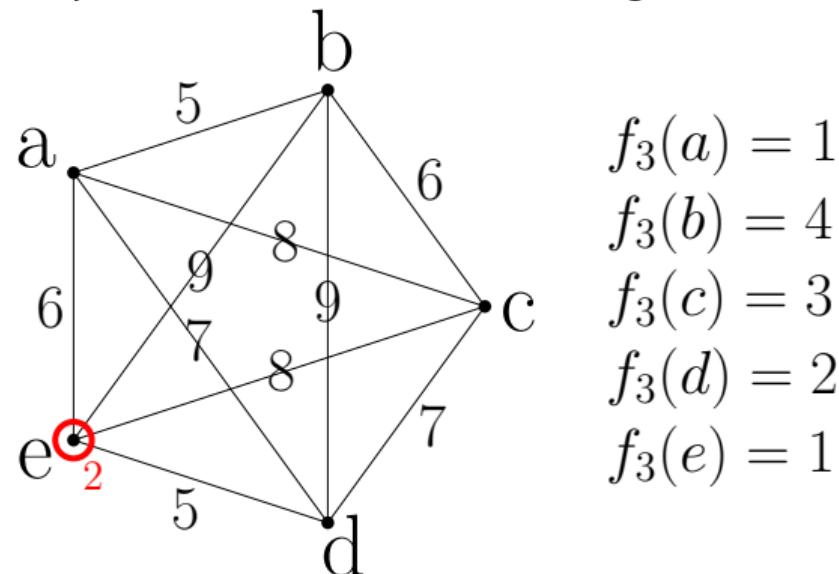
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(2) = 7 + 1 = 8$$

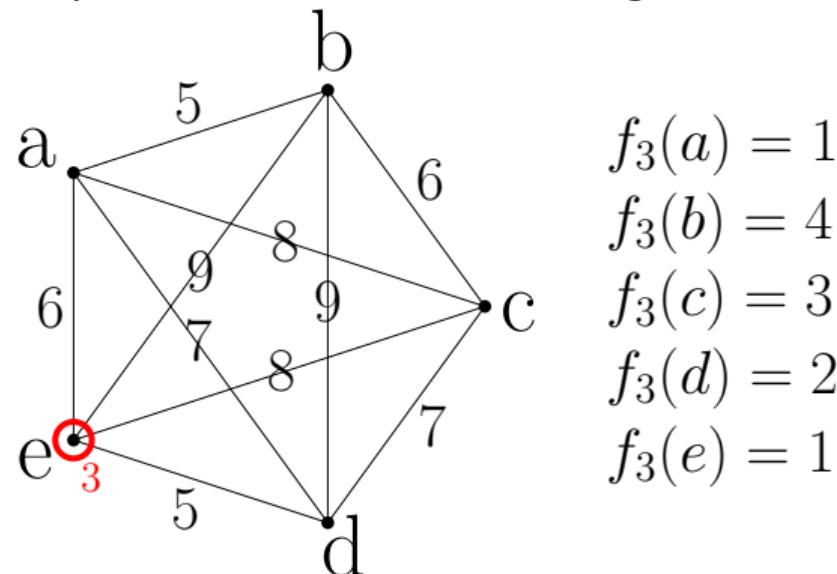
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(2) = 7 + 1 = 8$$

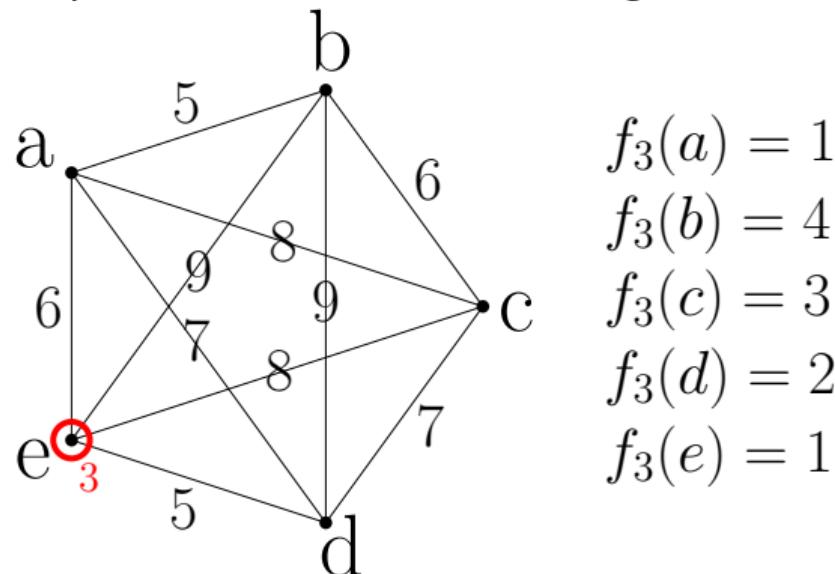
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(3) = 8 + 1 = 9$$

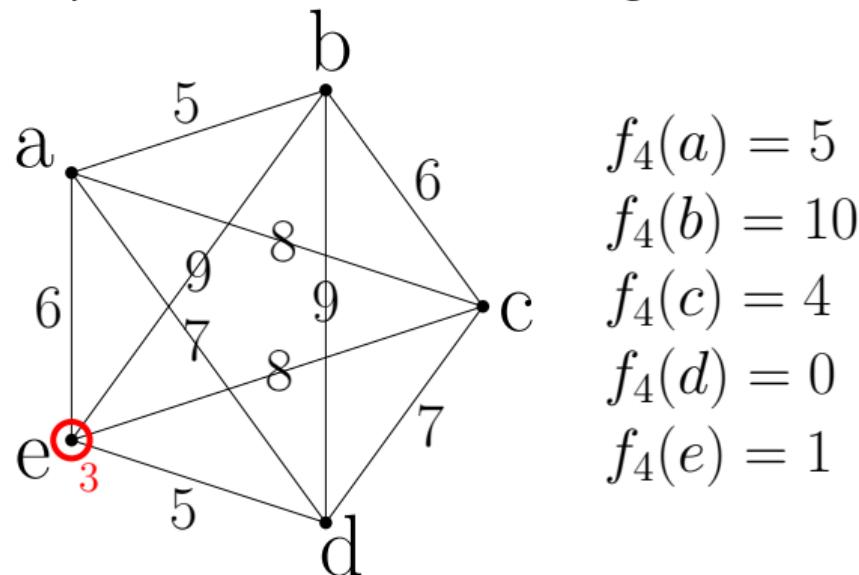
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(3)=8 + 1 = 9$$

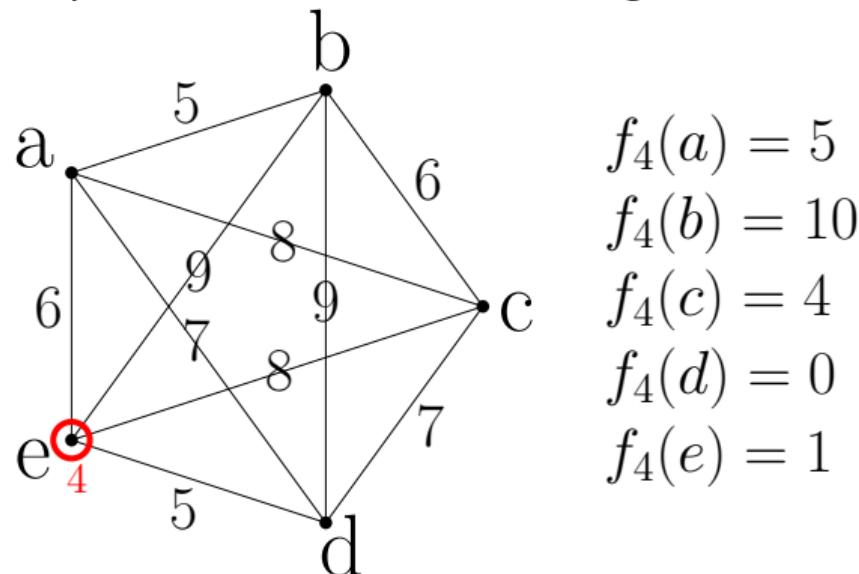
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\begin{aligned}f_4(a) &= 5 \\f_4(b) &= 10 \\f_4(c) &= 4 \\f_4(d) &= 0 \\f_4(e) &= 1\end{aligned}$$

$$\text{Cost opt}(3) = 8 + 1 = 9$$

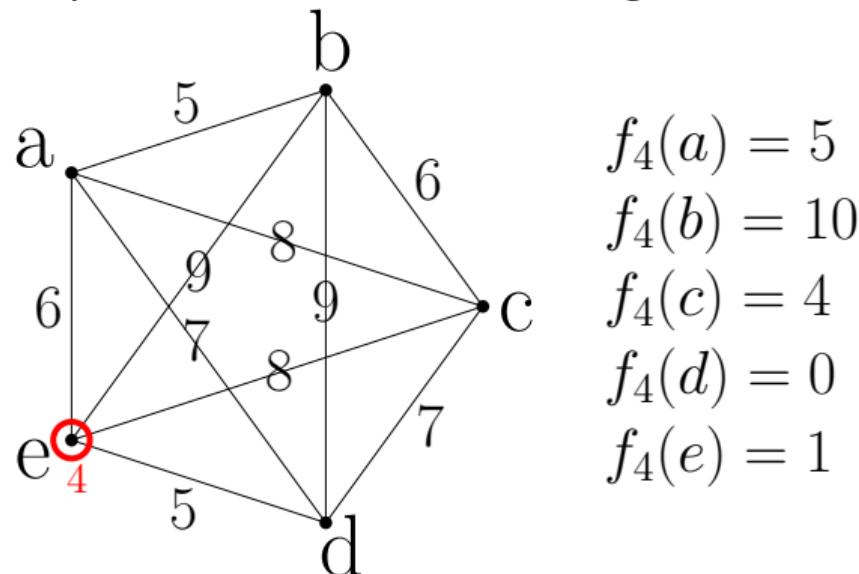
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(4) = 9 + 1 = 10$$

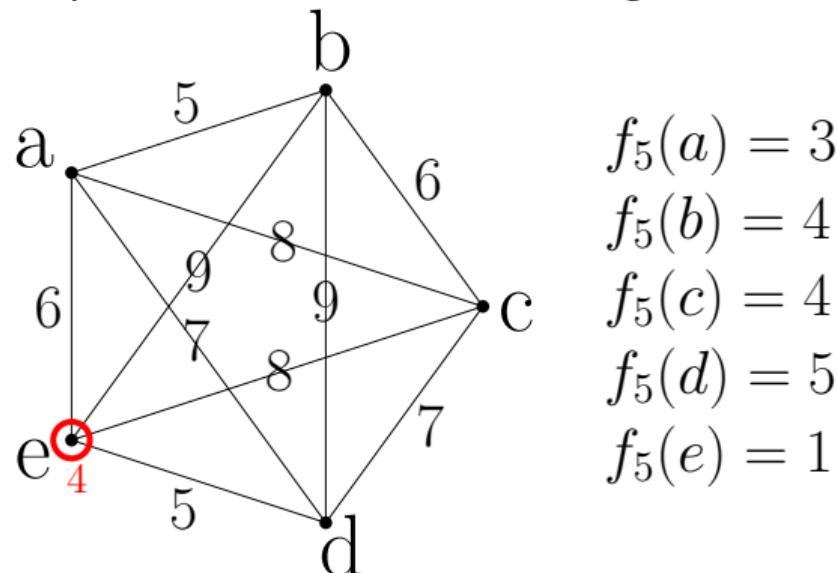
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(4) = 9 + 1 = 10$$

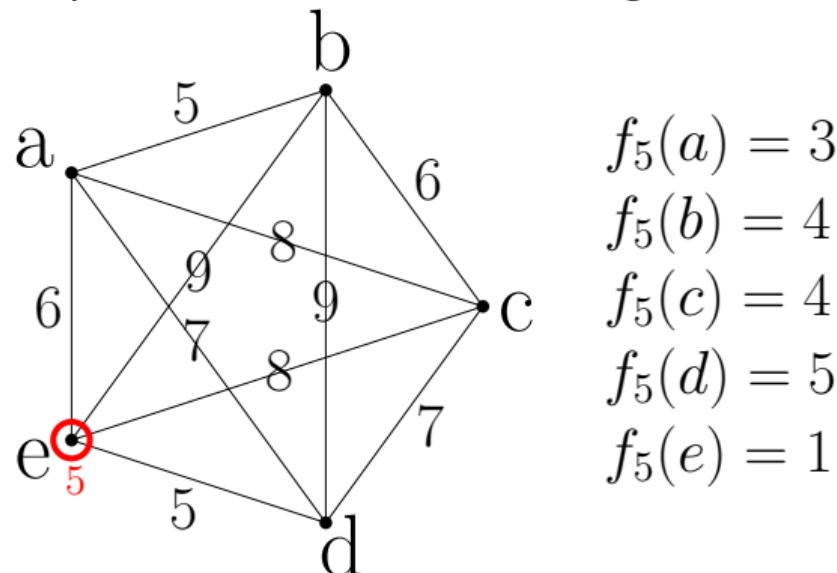
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(4) = 9 + 1 = 10$$

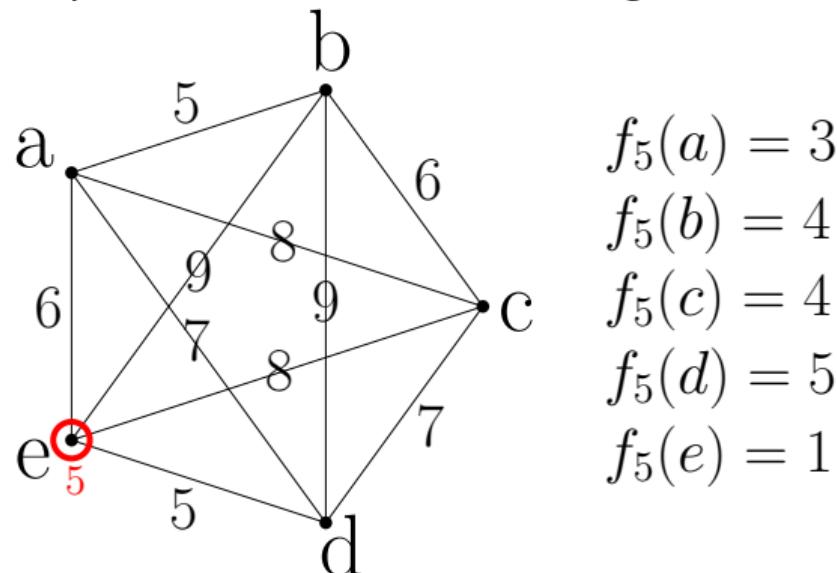
Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost opt}(5) = 10 + 1 = 11$$

Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\text{Alg}(I)}{\text{opt}(I)}.$$

Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\text{Alg}(I)}{\text{opt}(I)}.$$

Here $\frac{\text{Alg}(I)}{\text{opt}(I)} = \frac{27}{11}$.

Online problem - Metrical Task System (MTS)

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\text{Alg}(I)}{\text{opt}(I)}$$

$$\text{Here } \frac{\text{Alg}(I)}{\text{opt}(I)} = \frac{27}{11}.$$

Competitive ratio against oblivious adversary is

$$\max_{\text{input } I} \frac{\mathbb{E}[\text{Alg}(I)]}{\text{opt}(I)}.$$

Online problem - Metrical Task System (MTS)

Approach: embed into a tree, and then make all the decisions based on the tree.

Online problem - Metrical Task System (MTS)

Approach: embed into a tree, and then make all the decisions based on the tree.

Theorem ([Fiat, Mendel 2000])

Given an n point tree T , there is an online algorithm for MTS with competitive ratio $O(\log n \cdot \log \log n)$ against oblivious adversary.*

* Actually on an HST, which is a special kind of tree. [FRT04] is into HST's.

Online problem - Metrical Task System (MTS)

Approach: embed into a tree, and then make all the decisions based on the tree.

Theorem ([Fiat, Mendel 2000])

Given an n point tree T , there is an online algorithm for MTS with competitive ratio $O(\log n \cdot \log \log n)$ against oblivious adversary.*

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].

Online problem - Metrical Task System (MTS)

Approach: embed into a tree, and then make all the decisions based on the tree.

Theorem ([Fiat, Mendel 2000])

Given an n point tree T , there is an online algorithm for MTS with competitive ratio $O(\log n \cdot \log \log n)$ against oblivious adversary.*

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].

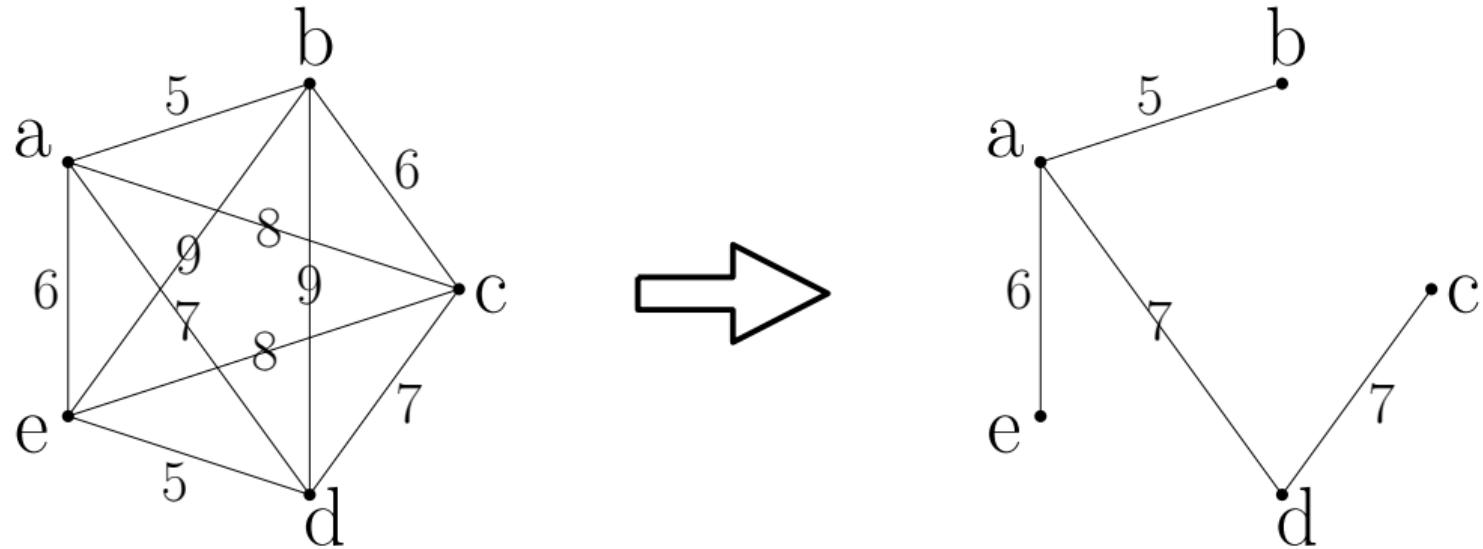
2. Run [FM00] on T with the same cost functions.

Make the same decisions as [FM00].

Online problem - Metrical Task System (MTS)

Algorithm:

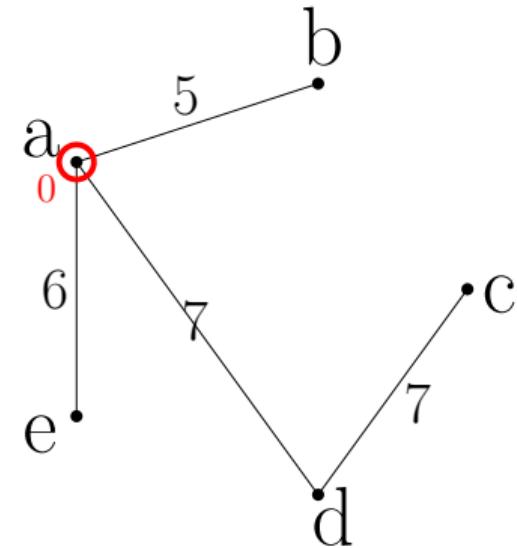
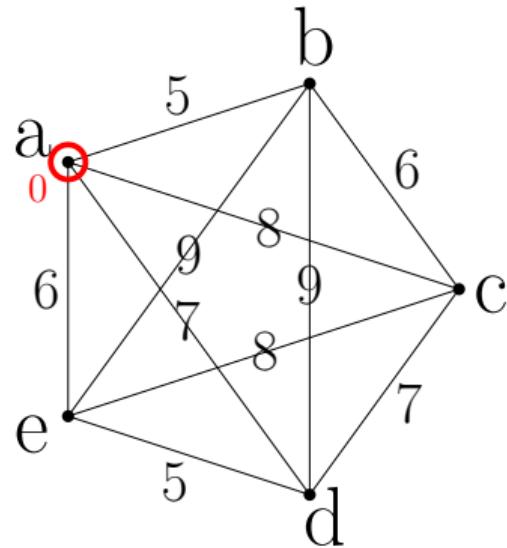
1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



Online problem - Metrical Task System (MTS)

Algorithm:

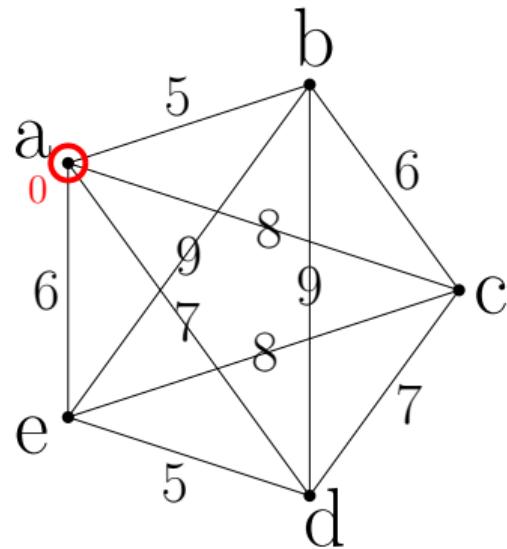
1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



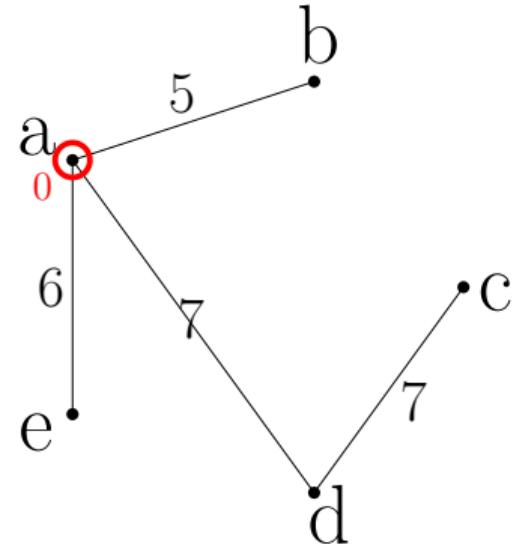
Online problem - Metrical Task System (MTS)

Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



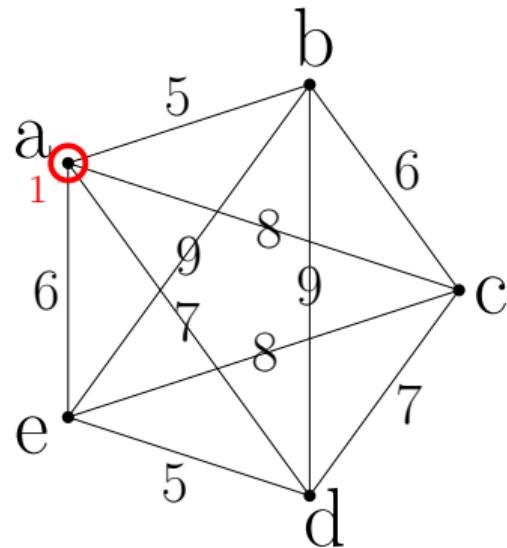
$$\begin{aligned}f_1(a) &= 3 \\f_1(b) &= 1 \\f_1(c) &= 2 \\f_1(d) &= 4 \\f_1(e) &= 1\end{aligned}$$



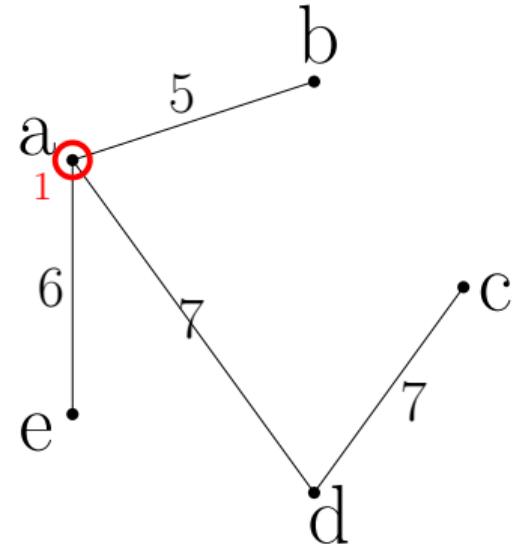
Online problem - Metrical Task System (MTS)

Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].

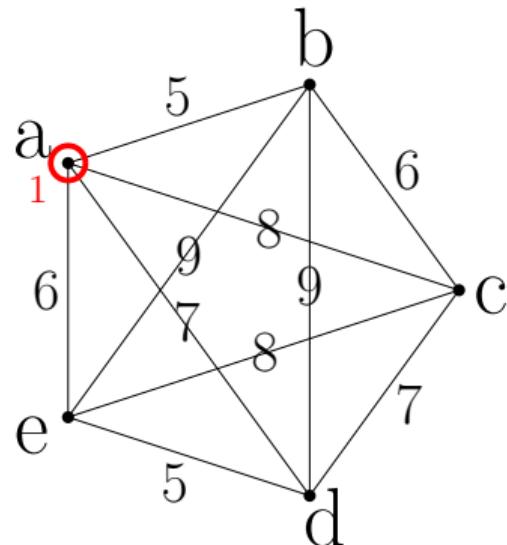


$$\begin{aligned}f_1(a) &= 3 \\f_1(b) &= 1 \\f_1(c) &= 2 \\f_1(d) &= 4 \\f_1(e) &= 1\end{aligned}$$



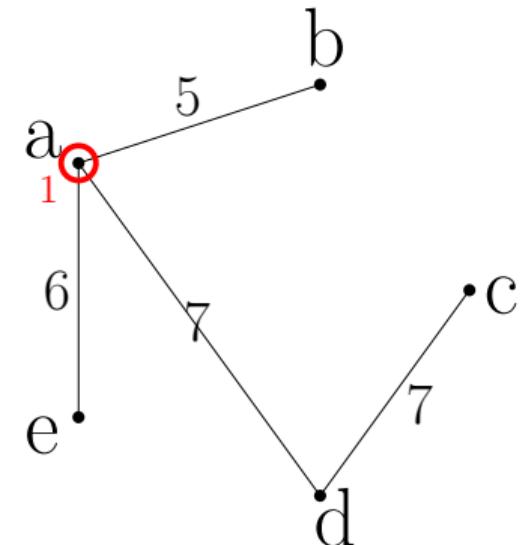
Online problem - Metrical Task System (MTS)

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



Cost alg(1)=3

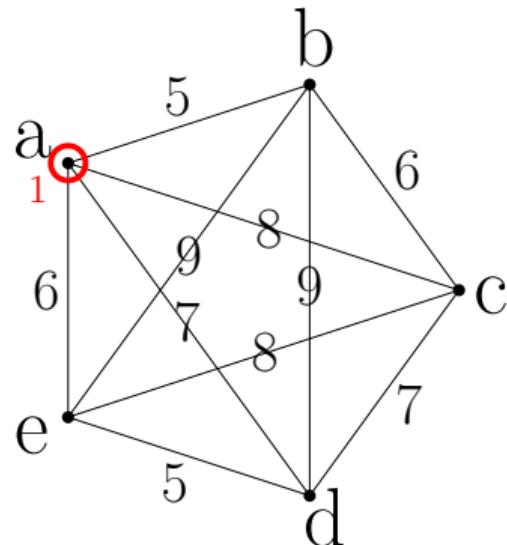
$$\begin{aligned}f_1(a) &= 3 \\f_1(b) &= 1 \\f_1(c) &= 2 \\f_1(d) &= 4 \\f_1(e) &= 1\end{aligned}$$



Cost alg $_T$ (1)=3

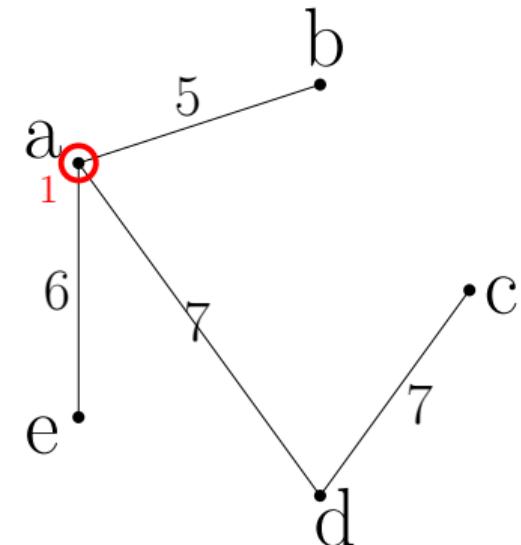
Online problem - Metrical Task System (MTS)

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



Cost alg(1)=3

$$\begin{aligned}f_2(a) &= 3 \\f_2(b) &= 1 \\f_2(c) &= 3 \\f_2(d) &= 2 \\f_2(e) &= 1\end{aligned}$$

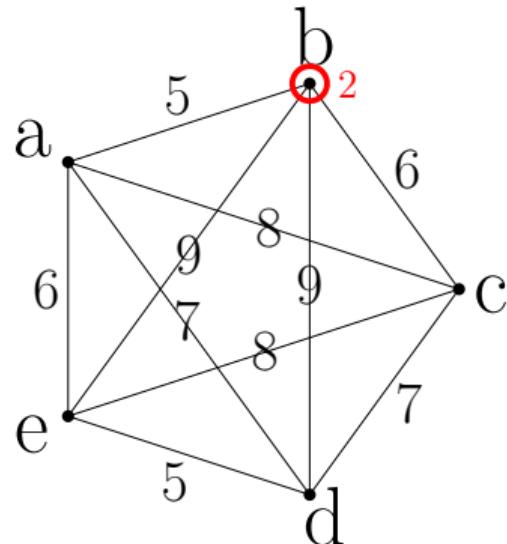


Cost alg $_T$ (1)=3

Online problem - Metrical Task System (MTS)

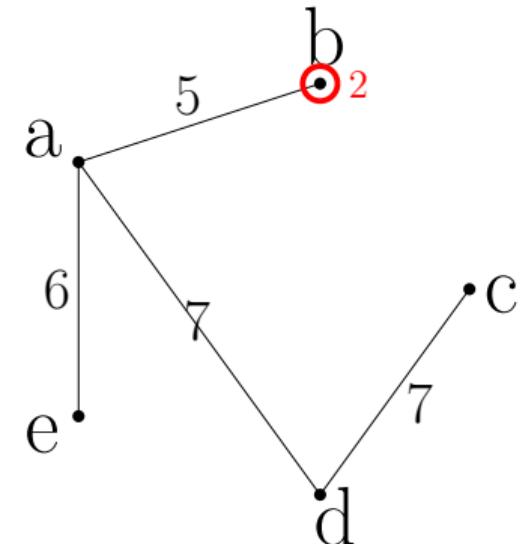
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



Cost alg(1)=3

$$\begin{aligned}f_2(a) &= 3 \\f_2(b) &= 1 \\f_2(c) &= 3 \\f_2(d) &= 2 \\f_2(e) &= 1\end{aligned}$$

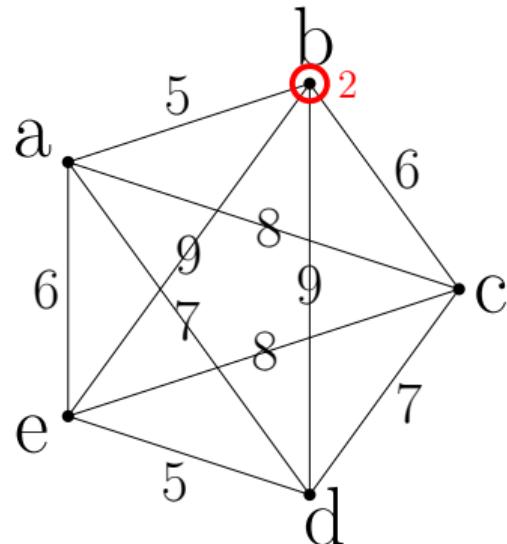


Cost alg $_T$ (1)=3

Online problem - Metrical Task System (MTS)

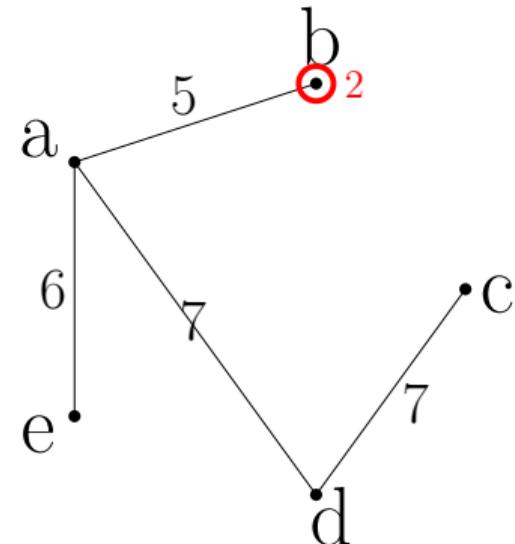
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\begin{aligned}f_2(a) &= 3 \\f_2(b) &= 1 \\f_2(c) &= 3 \\f_2(d) &= 2 \\f_2(e) &= 1\end{aligned}$$

$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

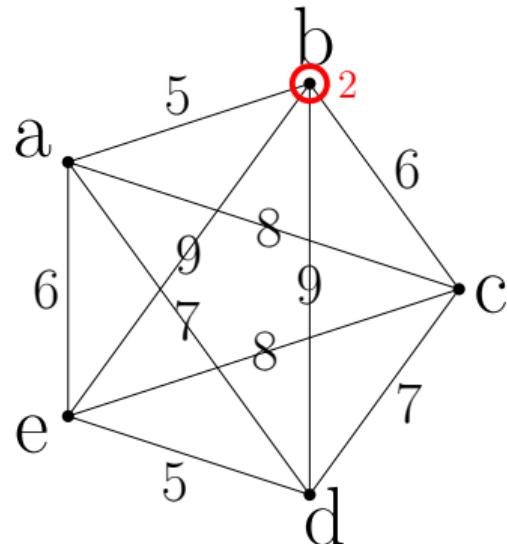


$$\text{Cost alg}_T(2) = 3 + 5 + 1 = 9$$

Online problem - Metrical Task System (MTS)

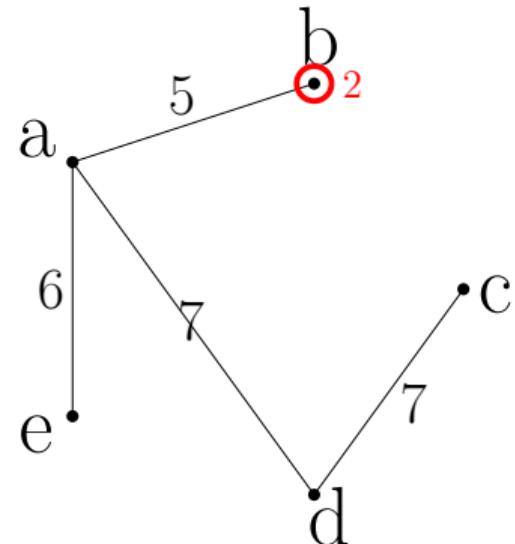
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

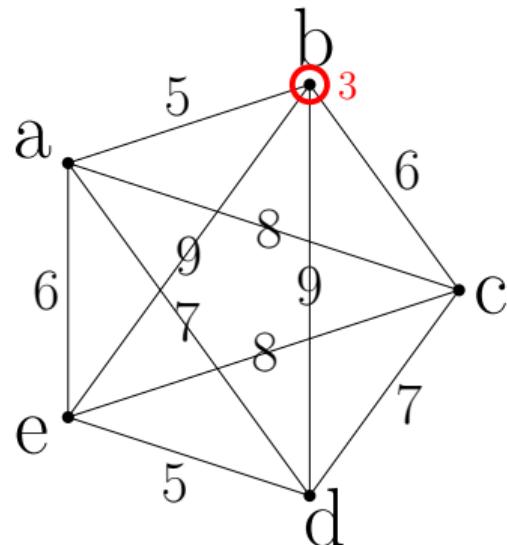
$$\begin{aligned}f_3(a) &= 1 \\f_3(b) &= 4 \\f_3(c) &= 3 \\f_3(d) &= 2 \\f_3(e) &= 1\end{aligned}$$



$$\text{Cost alg}_T(2) = 3 + 5 + 1 = 9$$

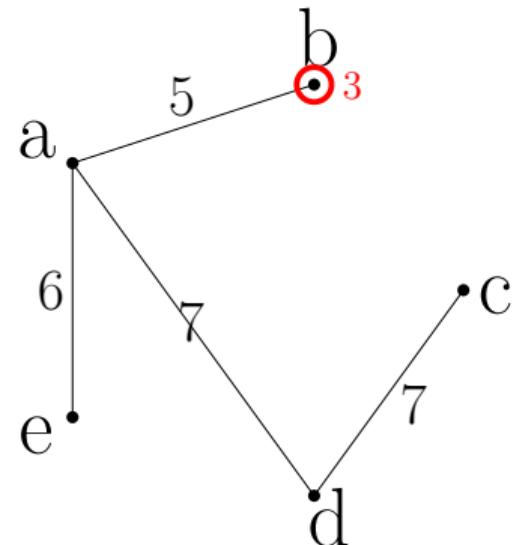
Online problem - Metrical Task System (MTS)

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\begin{aligned}f_3(a) &= 1 \\f_3(b) &= 4 \\f_3(c) &= 3 \\f_3(d) &= 2 \\f_3(e) &= 1\end{aligned}$$

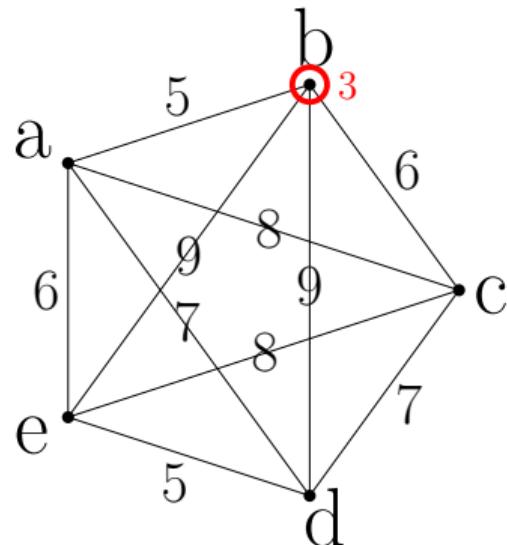
$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$



$$\text{Cost alg}_T(2) = 3 + 5 + 1 = 9$$

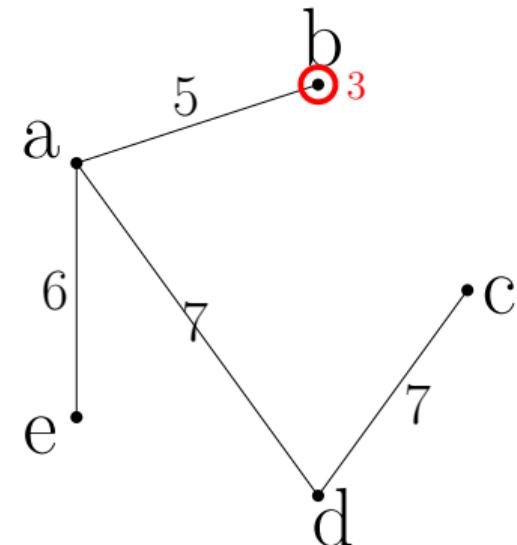
Online problem - Metrical Task System (MTS)

- Algorithm:**
1. Sample a tree T over (X, d_X) using [FRT04].
 2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(3) = 9 + 4 = 13$$

$$\begin{aligned}f_3(a) &= 1 \\f_3(b) &= 4 \\f_3(c) &= 3 \\f_3(d) &= 2 \\f_3(e) &= 1\end{aligned}$$

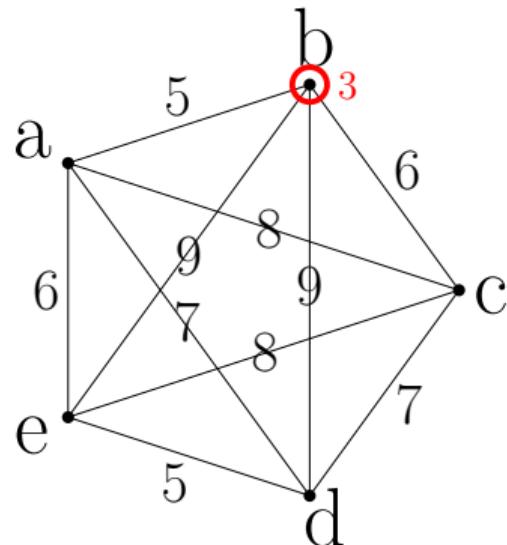


$$\text{Cost alg}_T(3) = 9 + 4 = 13$$

Online problem - Metrical Task System (MTS)

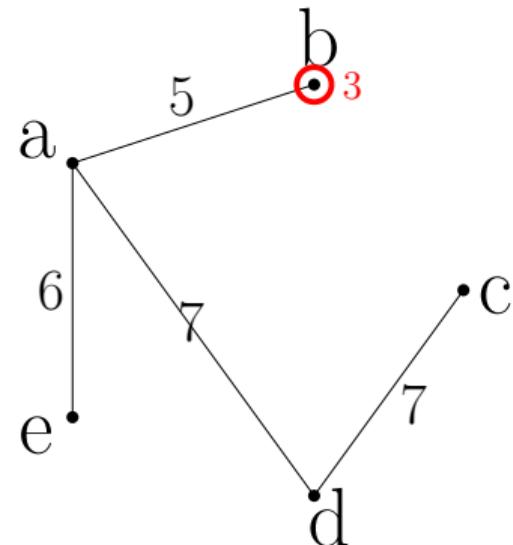
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(3)=9 + 4 = 13$$

$$\begin{aligned}f_4(a) &= 5 \\f_4(b) &= 10 \\f_4(c) &= 4 \\f_4(d) &= 0 \\f_4(e) &= 1\end{aligned}$$

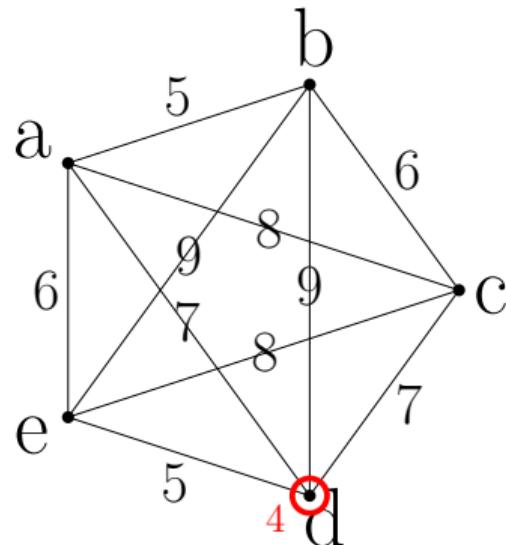


$$\text{Cost alg}_T(3)=9 + 4 = 13$$

Online problem - Metrical Task System (MTS)

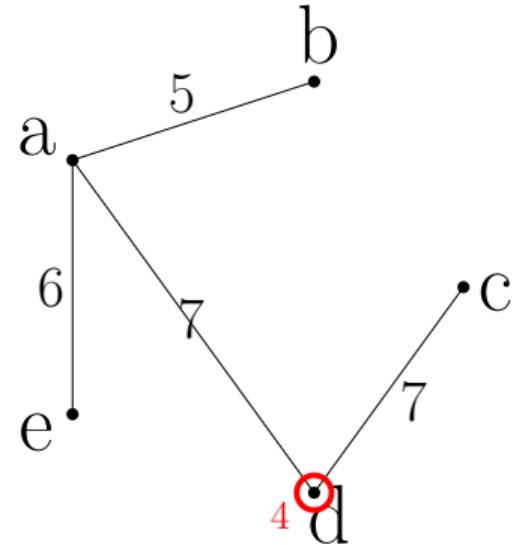
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(3)=9+4=13$$

$$\begin{aligned}f_4(a) &= 5 \\f_4(b) &= 10 \\f_4(c) &= 4 \\f_4(d) &= 0 \\f_4(e) &= 1\end{aligned}$$

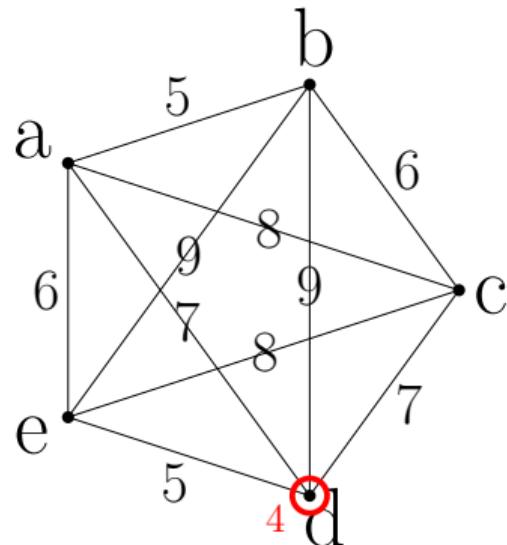


$$\text{Cost alg}_T(3)=9+4=13$$

Online problem - Metrical Task System (MTS)

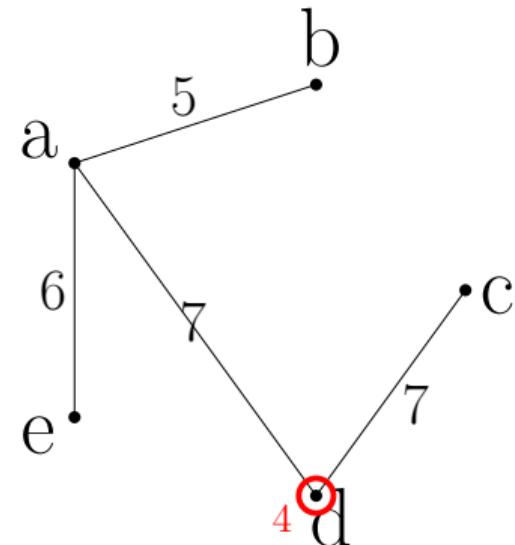
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\begin{aligned}f_4(a) &= 5 \\f_4(b) &= 10 \\f_4(c) &= 4 \\f_4(d) &= 0 \\f_4(e) &= 1\end{aligned}$$

$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

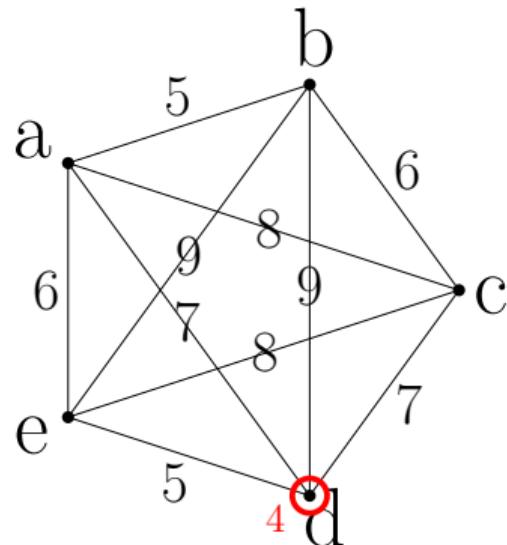


$$\text{Cost alg}_T(4) = 13 + 12 + 0 = 25$$

Online problem - Metrical Task System (MTS)

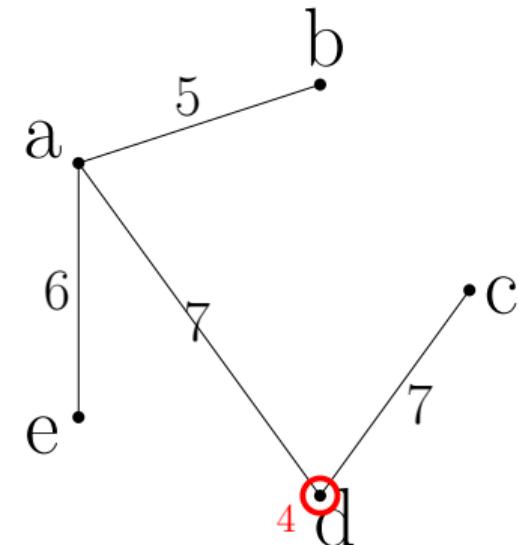
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

$$\begin{aligned}f_5(a) &= 3 \\f_5(b) &= 4 \\f_5(c) &= 4 \\f_5(d) &= 5 \\f_5(e) &= 1\end{aligned}$$

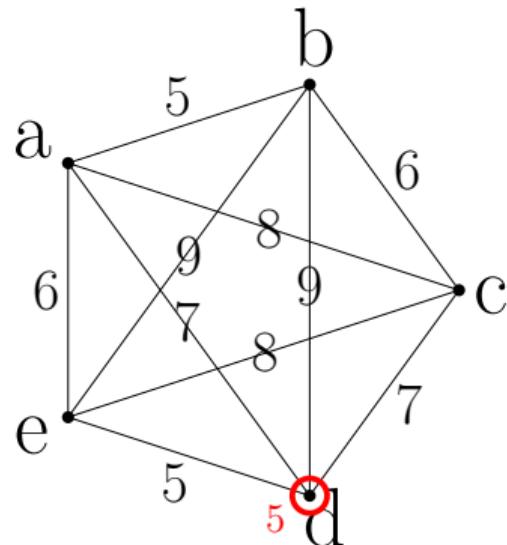


$$\text{Cost alg}_T(4) = 13 + 12 + 0 = 25$$

Online problem - Metrical Task System (MTS)

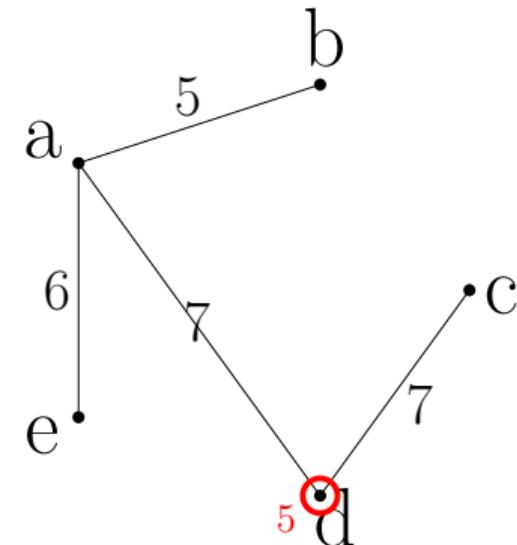
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(4) = 13 + 9 + 0 = 22$$

$$\begin{aligned}f_5(a) &= 3 \\f_5(b) &= 4 \\f_5(c) &= 4 \\f_5(d) &= 5 \\f_5(e) &= 1\end{aligned}$$

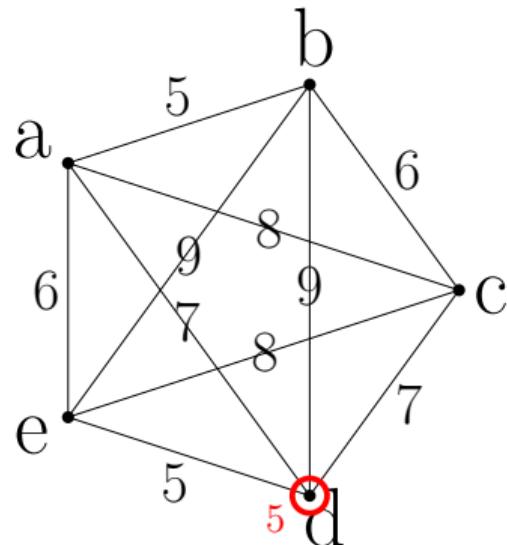


$$\text{Cost alg}_T(4) = 13 + 12 + 0 = 25$$

Online problem - Metrical Task System (MTS)

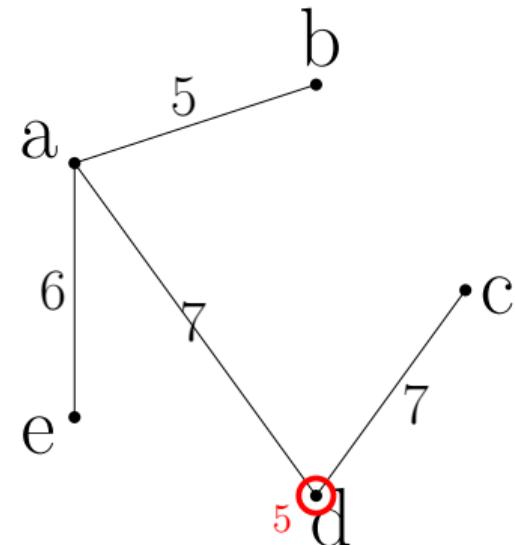
Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].



$$\text{Cost alg}(5) = 22 + 5 = 27$$

$$\begin{aligned}f_5(a) &= 3 \\f_5(b) &= 4 \\f_5(c) &= 4 \\f_5(d) &= 5 \\f_5(e) &= 1\end{aligned}$$



$$\text{Cost alg}_T(5) = 25 + 5 = 30$$

Online problem - Metrical Task System (MTS)

Algorithm:

1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].

Analysis. Let x_1, x_2, \dots, x_k be the decisions of opt. Thus
 $\text{opt} = \sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_X(x_{i-1}, x_i).$

Online problem - Metrical Task System (MTS)

Algorithm: 1. Sample a tree T over (X, d_X) using [FRT04].
2. Run [FM00] on T with the same cost functions.
 Make the same decisions as [FM00].

Analysis. Let x_1, x_2, \dots, x_k be the decisions of opt. Thus
 $\text{opt} = \sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_X(x_{i-1}, x_i).$

We've sampled a tree T , x_1, x_2, \dots, x_k is also a valid decisions for T . Hence
 $\text{opt}_T \leq \sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i).$

Online problem - Metrical Task System (MTS)

- Algorithm:**
1. Sample a tree T over (X, d_X) using [FRT04].
 2. Run [FM00] on T with the same cost functions.
Make the same decisions as [FM00].

Analysis. Let x_1, x_2, \dots, x_k be the decisions of opt. Thus
 $\text{opt} = \sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_X(x_{i-1}, x_i).$

We've sampled a tree T , x_1, x_2, \dots, x_k is also a valid decisions for T . Hence
 $\text{opt}_T \leq \sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i).$

[FM00] is $O(\log n \cdot \log \log n)$ -competitive on T . Hence it choose points y_1, \dots, y_k such that $\mathbb{E}[\text{alg}_T] = \mathbb{E}[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i)] \leq O(\log n \cdot \log \log n) \cdot \text{opt}_T.$

We made the same decisions, so overall:

$$\mathbb{E} [\text{alg}] = \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right]$$

We made the same decisions, so overall:

$$\begin{aligned}\mathbb{E} [\text{alg}] &= \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right] \\ &\leq \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i) \right]\end{aligned}$$

We made the same decisions, so overall:

$$\begin{aligned}\mathbb{E} [\text{alg}] &= \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right] \\ &\leq \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i) \right] \\ &\leq O(\log n \cdot \log \log n) \cdot \mathbb{E} \left[\sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i) \right]\end{aligned}$$

We made the same decisions, so overall:

$$\begin{aligned}\mathbb{E} [\text{alg}] &= \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right] \\ &\leq \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i) \right] \\ &\leq O(\log n \cdot \log \log n) \cdot \mathbb{E} \left[\sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i) \right] \\ &= O(\log n \cdot \log \log n) \cdot \left(\sum_{i=1}^k f_i(x_i) + O(\log n) \cdot \sum_{i=1}^k d_T(x_{i-1}, x_i) \right)\end{aligned}$$

We made the same decisions, so overall:

$$\begin{aligned}\mathbb{E} [\text{alg}] &= \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right] \\ &\leq \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i) \right] \\ &\leq O(\log n \cdot \log \log n) \cdot \mathbb{E} \left[\sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i) \right] \\ &= O(\log n \cdot \log \log n) \cdot \left(\sum_{i=1}^k f_i(x_i) + O(\log n) \cdot \sum_{i=1}^k d_T(x_{i-1}, x_i) \right) \\ &\leq O(\log^2 n \cdot \log \log n) \cdot \text{Opt} .\end{aligned}$$

We made the same decisions, so overall:

$$\begin{aligned}\mathbb{E} [\text{alg}] &= \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_X(y_{i-1}, y_i) \right] \\ &\leq \mathbb{E} \left[\sum_{i=1}^k f_i(y_i) + \sum_{i=1}^k d_T(y_{i-1}, y_i) \right] \\ &\leq O(\log n \cdot \log \log n) \cdot \mathbb{E} \left[\sum_{i=1}^k f_i(x_i) + \sum_{i=1}^k d_T(x_{i-1}, x_i) \right] \\ &= O(\log n \cdot \log \log n) \cdot \left(\sum_{i=1}^k f_i(x_i) + O(\log n) \cdot \sum_{i=1}^k d_T(x_{i-1}, x_i) \right) \\ &\leq O(\log^2 n \cdot \log \log n) \cdot \text{Opt} .\end{aligned}$$

Theorem

MTS has an $O(\log^2 n \cdot \log \log n)$ competitive algorithm against oblivious adversary.

Outline of the talk

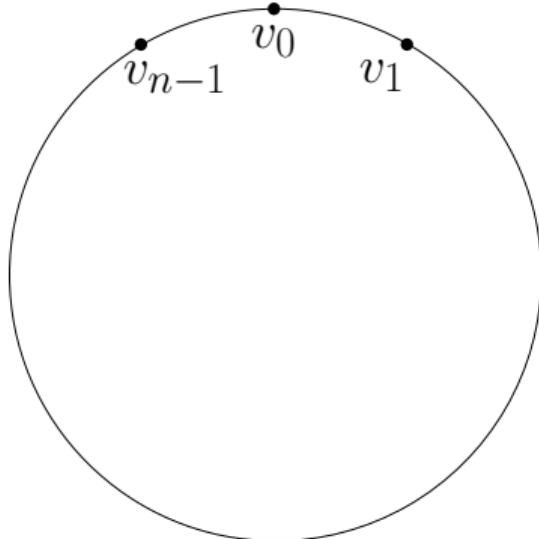
- 1 Introduction
- 2 Stochastic embedding into trees
- 3 Ramsey type embeddings
- 4 Clan embedding
- 5 Group Steiner Tree
- 6 Conclusion

Ramsey type Embeddings

Ramsey type theorem: Every **big** enough object, contains a **structured subset**.

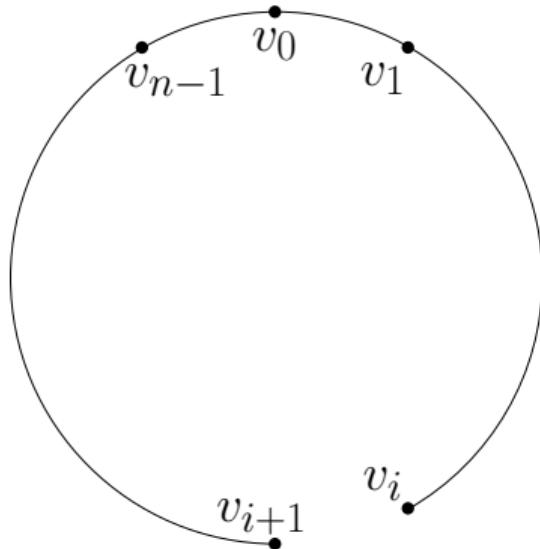
Ramsey type Embeddings

Ramsey type theorem: Every **big** enough object, contains a **structured subset**.



Ramsey type Embeddings

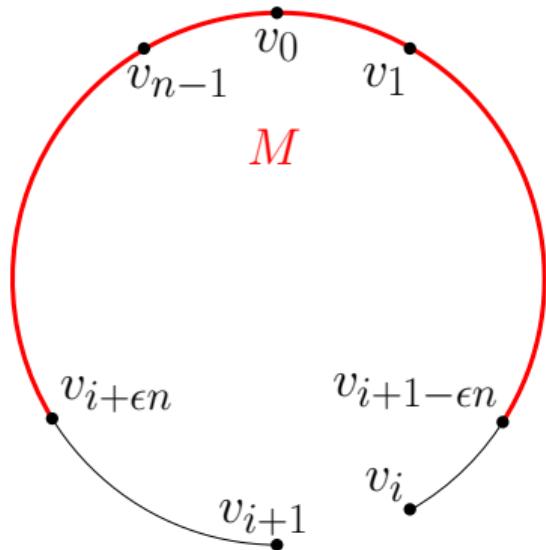
Ramsey type theorem: Every **big** enough object, contains a **structured subset**.



Suppose we delete $\{v_i, v_{i+1}\}$.

Ramsey type Embeddings

Ramsey type theorem: Every **big** enough object, contains a **structured subset**.

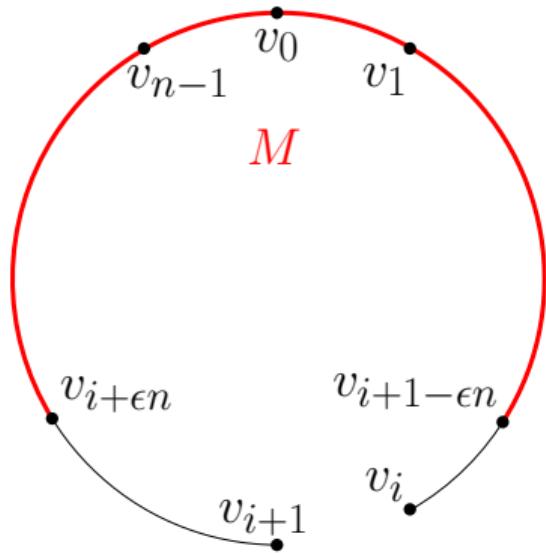


Suppose we delete $\{v_i, v_{i+1}\}$.

Set $M = \{v_{i+1-\epsilon n}, v_{i+2-\epsilon n}, \dots, v_{i+\epsilon n}\}$.

Ramsey type Embeddings

Ramsey type theorem: Every **big** enough object, contains a **structured subset**.



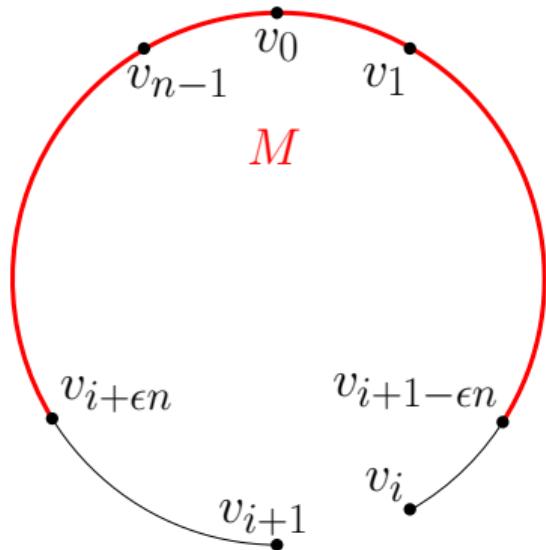
Suppose we delete $\{v_i, v_{i+1}\}$.

Set $M = \{v_{i+1-\epsilon n}, v_{i+2-\epsilon n}, \dots, v_{i+\epsilon n}\}$.

For every $x, y \in M$, $\frac{d_T(x,y)}{d_{C_n}(x,y)} < \frac{1}{2\epsilon}$.

Ramsey type Embeddings

Ramsey type theorem: Every **big** enough object, contains a **structured subset**.



Suppose we delete $\{v_i, v_{i+1}\}$.

Set $M = \{v_{i+1-\epsilon n}, v_{i+2-\epsilon n}, \dots, v_{i+\epsilon n}\}$.

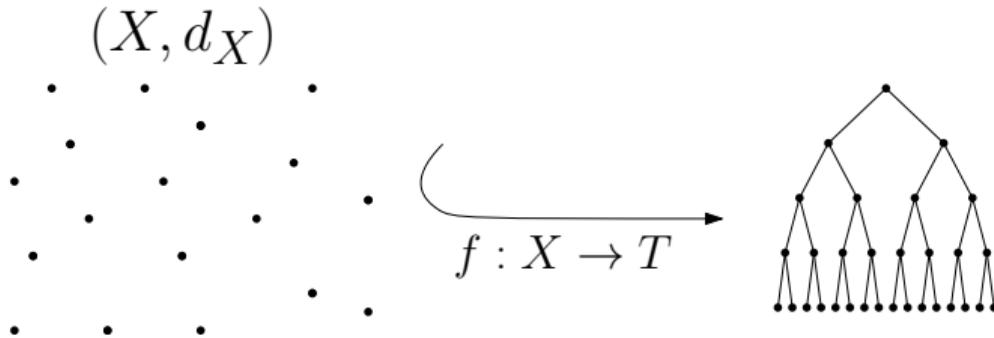
For every $x, y \in M$, $\frac{d_T(x,y)}{d_{C_n}(x,y)} < \frac{1}{2\epsilon}$.

Choose i u.a.r., then $\Pr[v \in M] = 1 - 2\epsilon$.

Ramsey type Embeddings

Fix $k > 1$, what is the largest subset $M \subset X$,

s.t. (M, d_X) embeds into a tree with distortion k ?

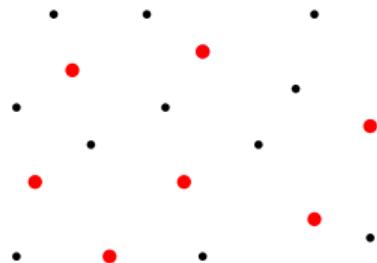


Ramsey type Embeddings

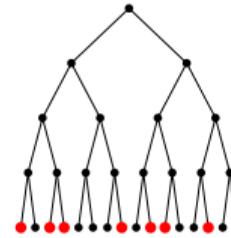
Fix $k > 1$, what is the largest subset $\textcolor{red}{M} \subset X$,

s.t. $(\textcolor{red}{M}, d_X)$ embeds into a tree with distortion k ?

$\textcolor{red}{M}$ (X, d_X)



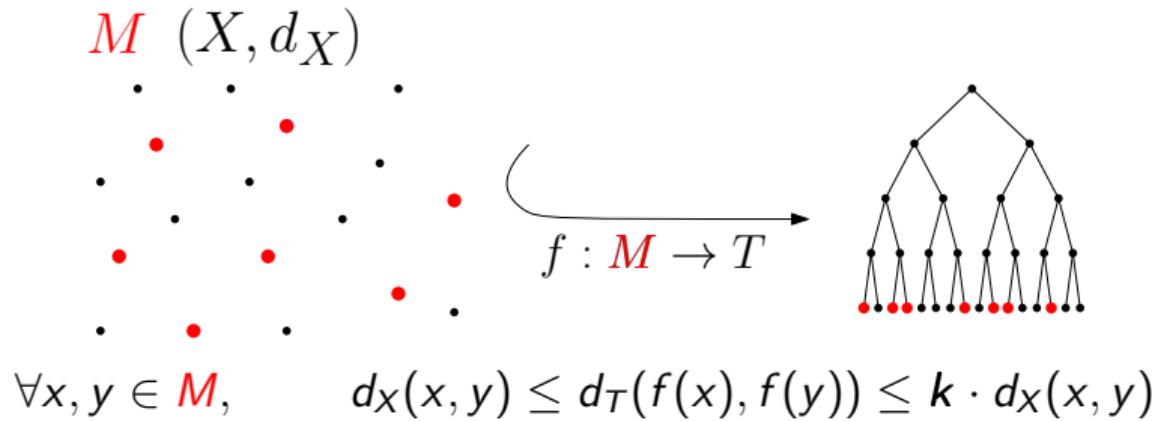
$f : \textcolor{red}{M} \rightarrow T$



$$\forall x, y \in \textcolor{red}{M}, \quad d_X(x, y) \leq d_T(f(x), f(y)) \leq k \cdot d_X(x, y)$$

Ramsey type Embeddings

Fix $k > 1$, what is the largest subset $M \subset X$,
s.t. (M, d_X) embeds into a tree with distortion k ?



Theorem ([Mendel, Naor 07], following [BFM86, BLMN05])

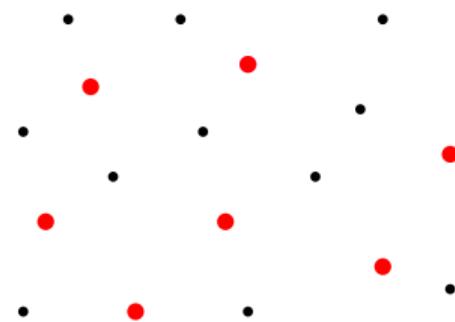
∀ n -point metric space and $k \geq 1$, ∃ subset M of size $n^{1-1/k}$
that embeds into a tree with distortion $O(k)$.

Ramsey type Embeddings

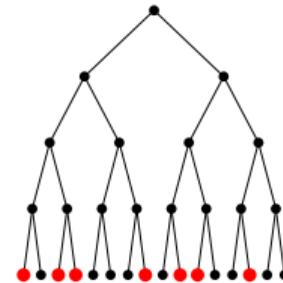
Theorem ([Mendel, Naor 07], following [BFM86, BLMN05])

$\forall n$ -point metric space and $k \geq 1$, \exists subset M of size $n^{1-1/k}$
that embeds into a tree with distortion $O(k)$.

$$M \quad (X, d_X)$$



$$f : M \rightarrow T$$

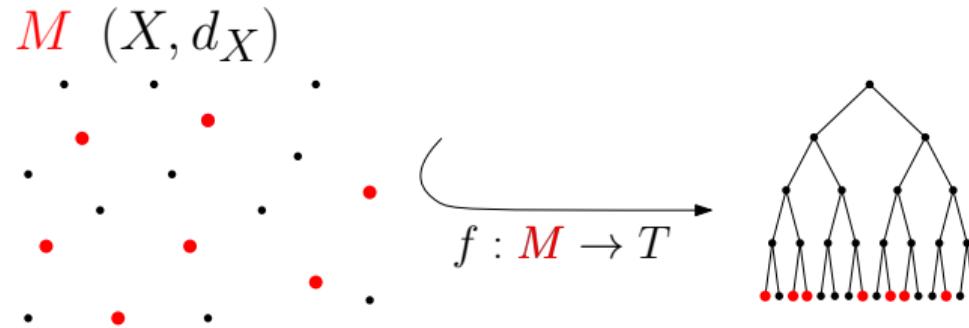


Asymptotically tight.

Ramsey type Embeddings

Theorem ([Mendel, Naor 07], following [BFM86, BLMN05])

$\forall n$ -point metric space and $k \geq 1$, \exists subset M of size $n^{1-1/k}$
that embeds into a tree with distortion $O(k)$.



Asymptotically tight.

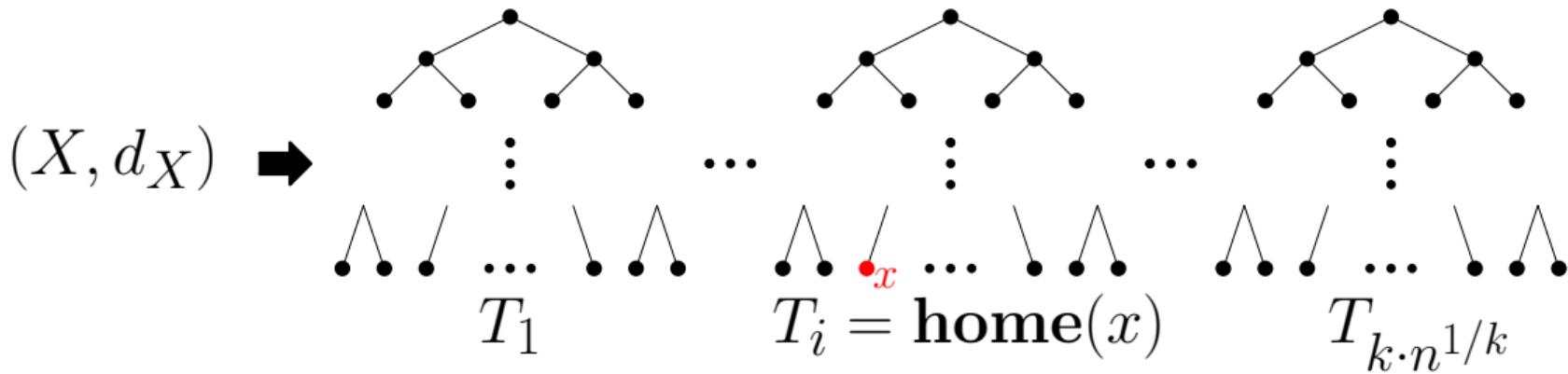
[Naor, Tao 12]: distortion $2e \cdot k$.

Ramsey type Embeddings

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(y) \leq O(k) \cdot d_X(x, y)$$



Ramsey type Embeddings

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(x, y) \leq O(k) \cdot d_X(x, y)$$

Applications:

- **Distance oracle**
- Compact routing scheme
- Online algorithms
- Approximate ranking
- etc.

Ramsey type Embeddings

Theorem ([Mendel, Naor 07], following [BFM86, BLMN05])

\forall *n-point metric space and $k \geq 1$, \exists subset M of size $n^{1-1/k}$ that embeds into a tree with distortion $O(k)$.*

Compromises: only partial guarantees



Distance Oracle

A **succinct** data structure that **approximately** answers distance queries.

Distance Oracle

A **succinct** data structure that **approximately** answers distance queries.



Distance Oracle

A **succinct** data structure that **approximately** answers distance queries.



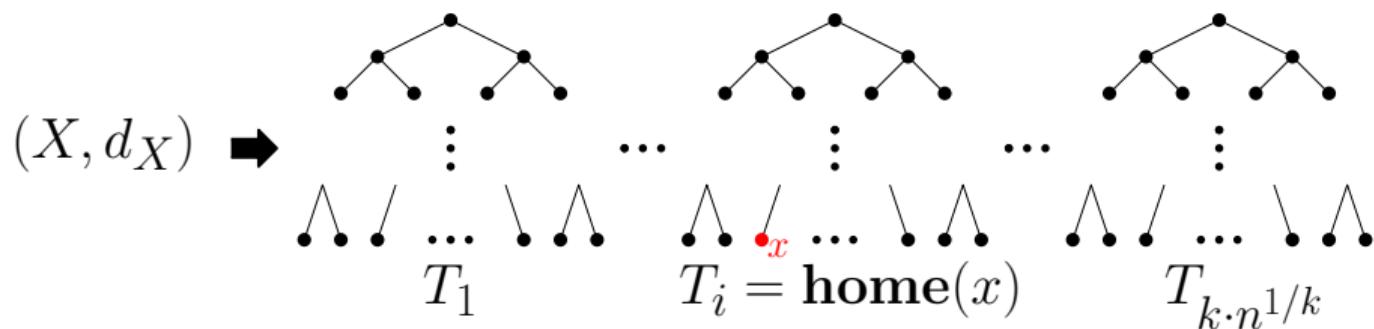
The properties of interest are size, distortion and query time.

Distance Oracles

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(y) \leq O(k) \cdot d_X(x, y)$$



Distance Oracles

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(x, y) \leq O(k) \cdot d_X(x, y)$$



Distance Oracles

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(x, y) \leq O(k) \cdot d_X(x, y)$$

Theorem (Tree Distance Oracle [Harel, Tarjan 84], [Bender, Farach-Colton 00])

For every tree metric, there is an exact distance oracle of linear size and constant query time.

Distance Oracles

Corollary

For every n -point metric space and $k \geq 1$, there is a set \mathcal{T} of $k \cdot n^{\frac{1}{k}}$ trees and a mapping $\text{home} : X \rightarrow \mathcal{T}$, such that for every $x, y \in X$,

$$d_{\text{home}(x)}(x, y) \leq O(k) \cdot d_X(x, y)$$

Theorem (Tree Distance Oracle [Harel, Tarjan 84], [Bender, Farach-Colton 00])

For every tree metric, there is an exact distance oracle of linear size and constant query time.

Theorem (Ramsey based Deterministic Distance Oracle)

For any n -point metric space, there is a distance oracle with :

Distortion	Size	Query time
$O(k)$	$O(k \cdot n^{1+1/k})$	$O(1)$

Outline of the talk

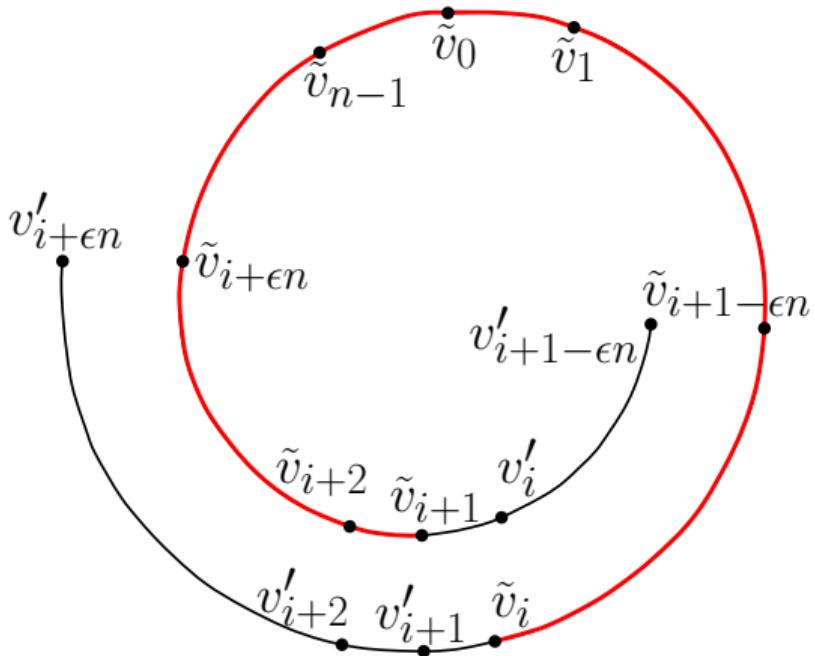
- 1 Introduction
- 2 Stochastic embedding into trees
- 3 Ramsey type embeddings
- 4 Clan embedding
- 5 Group Steiner Tree
- 6 Conclusion

Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!

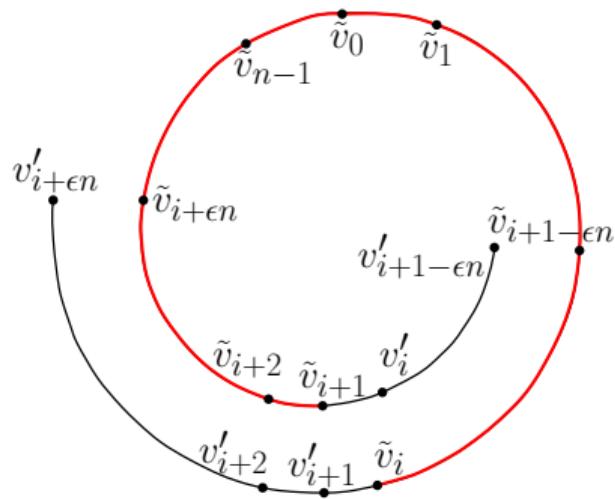
Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!



Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!

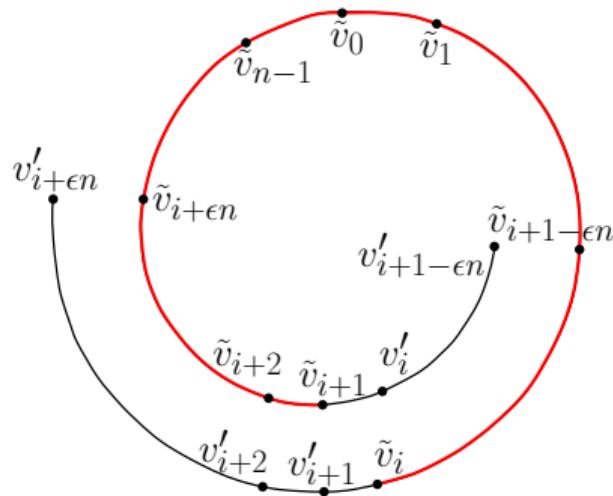


One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!



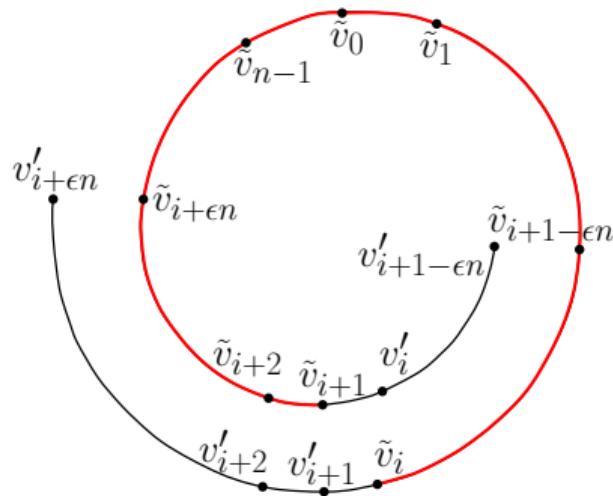
One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

$f(x)$ is the **clan** of x . Each $x' \in f(x)$ is a **copy** of x .

Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!



One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

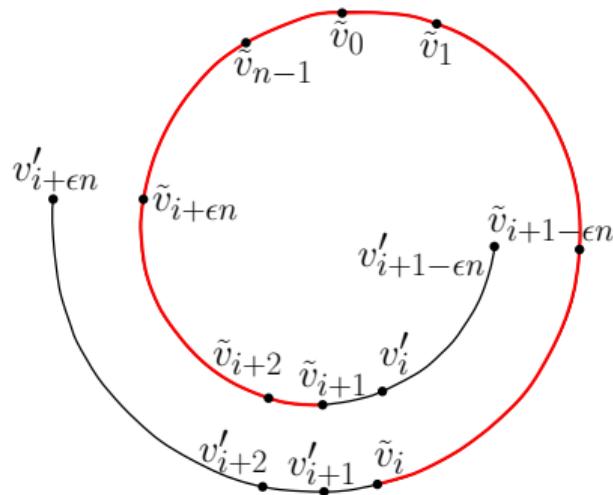
- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

$f(x)$ is the **clan** of x . Each $x' \in f(x)$ is a **copy** of x .

Dominating: $d_X(x, y) \leq \min_{x' \in f(x), y' \in f(y)} d_Y(x', y')$

Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!



One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

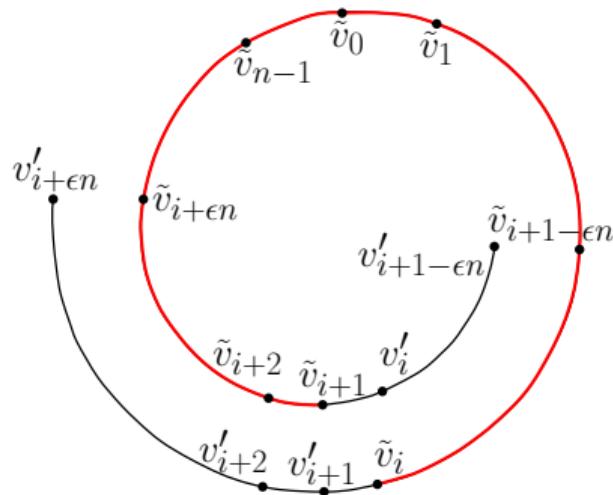
$f(x)$ is the **clan** of x . Each $x' \in f(x)$ is a **copy** of x .

Dominating: $d_X(x, y) \leq \min_{x' \in f(x), y' \in f(y)} d_Y(x', y')$

Each clan $f(x)$ has a **chief**: $\chi(x) \in f(x)$

Clan Embedding

Idea: **duplicate** vertices to meet all guarantees!



One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

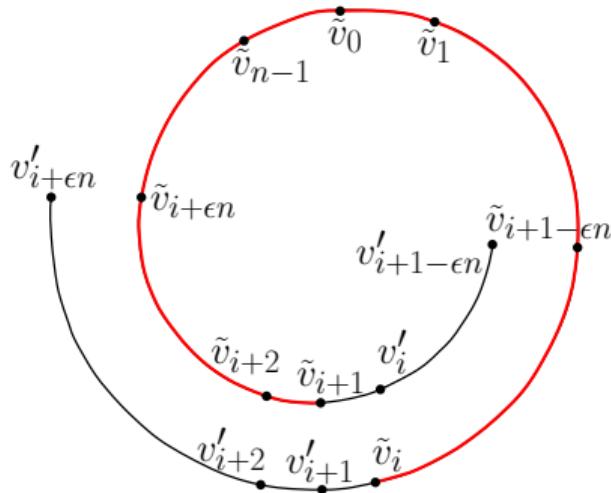
$f(x)$ is the **clan** of x . Each $x' \in f(x)$ is a **copy** of x .

Dominating: $d_X(x, y) \leq \min_{x' \in f(x), y' \in f(y)} d_Y(x', y')$

Each clan $f(x)$ has a **chief**: $\chi(x) \in f(x)$

$$f : C_n \rightarrow 2^{P_{2^n}}, \quad f(v_i) \subseteq \{\tilde{v}_i, v_i'\}, \quad \chi(v_i) = \tilde{v}_i$$

Clan Embedding



One-to-many embedding from (X, d_X) to (Y, d_Y) : A map $f : X \rightarrow 2^Y$ where:

- 1) $\forall x, f(x) \neq \emptyset$.
- 2) $\forall x, y, f(x) \cap f(y) = \emptyset$.

$f(x)$ is the **clan** of x . Each $x' \in f(x)$ is a **copy** of x .

Dominating: $d_X(x, y) \leq \min_{x' \in f(x), y' \in f(y)} d_Y(x', y')$

Each clan $f(x)$ has a **chief**: $\chi(x) \in f(x)$

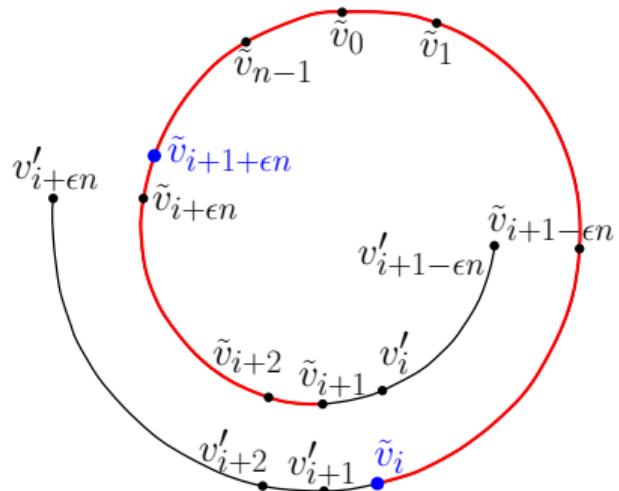
$$f : C_n \rightarrow 2^{P_{2^n}}, \quad f(v_i) \subseteq \{\tilde{v}_i, v_i'\}, \quad \chi(v_i) = \tilde{v}_i$$

Clan embedding is a pair (f, χ) , where f is **dominating** one-to-many embedding.
 (f, χ) has **distortion** t , if $\forall x, y \in X, \min_{y' \in f(y)} d_Y(\chi(x), y') \leq t \cdot d_X(x, y)$

Clan Embedding

Clan embedding is a pair (f, χ) , where f is **dominating** one-to-many embedding.

(f, χ) has **distortion** t , if $\forall x, y \in X$,

$$\min_{y' \in f(y)} d_Y(\chi(x), y') \leq t \cdot d_X(x, y)$$


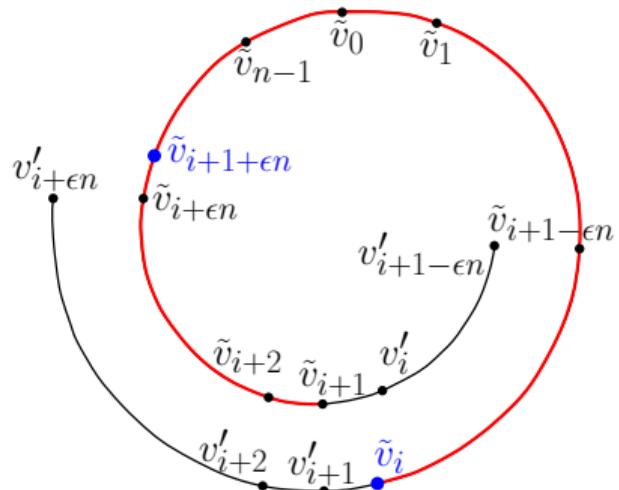
$$f : C_n \rightarrow 2^{P_{2n}}, \quad f(v_i) \subseteq \{\tilde{v}_i, v'_i\}, \quad \chi(v_i) = \tilde{v}_i$$

$$\begin{aligned} & \min_{y' \in f(v_{i+1+\epsilon n})} d_{P_{2n}}(\chi(v_i), y') \leq (1 - \epsilon)n \\ & < \frac{1}{\epsilon} \cdot d_{C_n}(v_i, v_{i+\epsilon n+1}) \end{aligned}$$

Clan Embedding

Clan embedding is a pair (f, χ) , where f is **dominating** one-to-many embedding.

(f, χ) has **distortion** t , if $\forall x, y \in X$,

$$\min_{y' \in f(y)} d_Y(\chi(x), y') \leq t \cdot d_X(x, y)$$


$$f : C_n \rightarrow 2^{P_{2n}}, \quad f(v_i) \subseteq \{\tilde{v}_i, v'_i\}, \quad \chi(v_i) = \tilde{v}_i$$

$$\begin{aligned} \min_{y' \in f(v_{i+1+\epsilon n})} d_{P_{2n}}(\chi(v_i), y') &\leq (1 - \epsilon)n \\ &< \frac{1}{\epsilon} \cdot d_{C_n}(v_i, v_{i+\epsilon n+1}) \end{aligned}$$

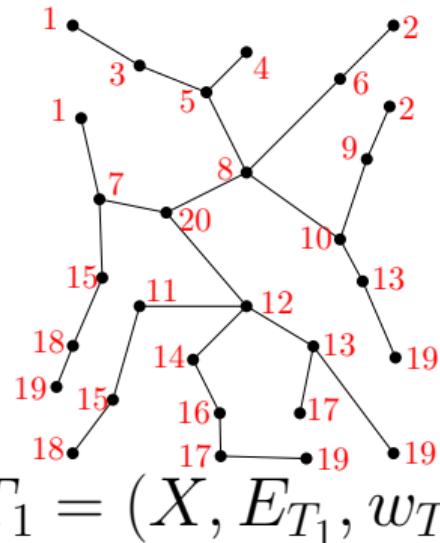
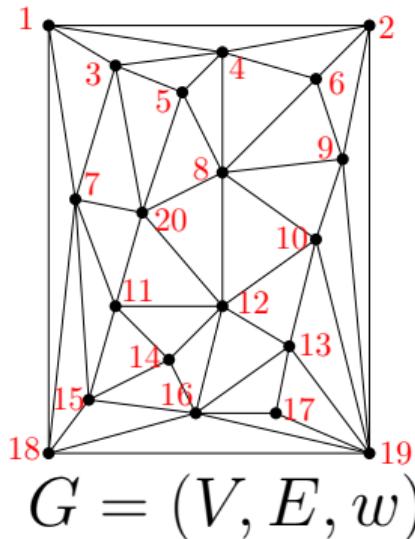
Choose i u.a.r., then $\mathbb{E}[|f(v_i)|] = 1 + 2\epsilon$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

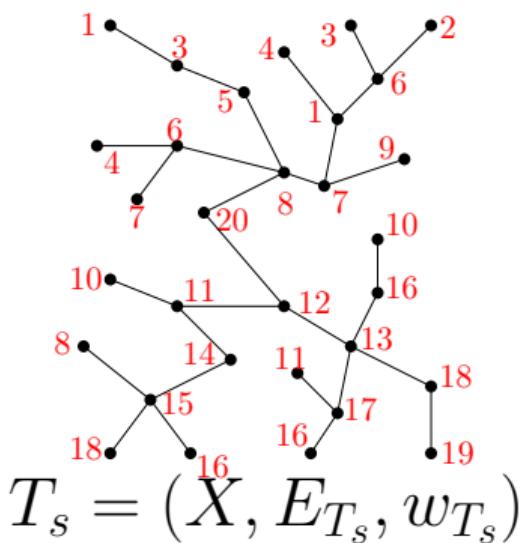
(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(\frac{\log n}{\epsilon})$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.



...

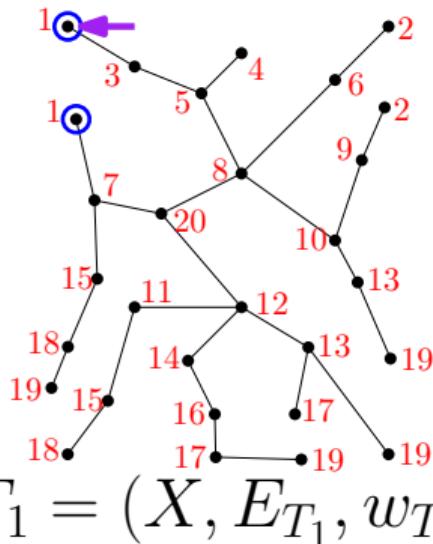
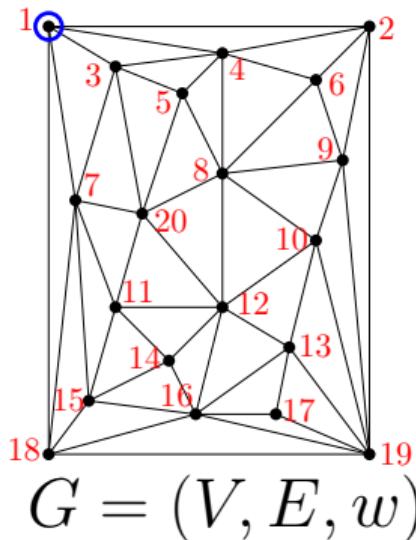


Theorem (Clan embedding into trees, [Filtser, Le 21])

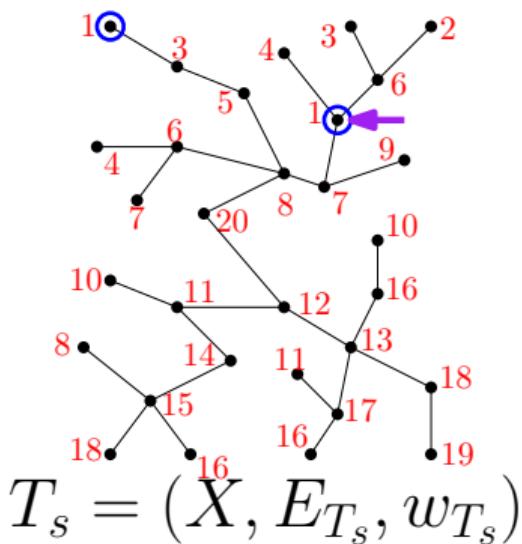
(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(\frac{\log n}{\epsilon})$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.



...

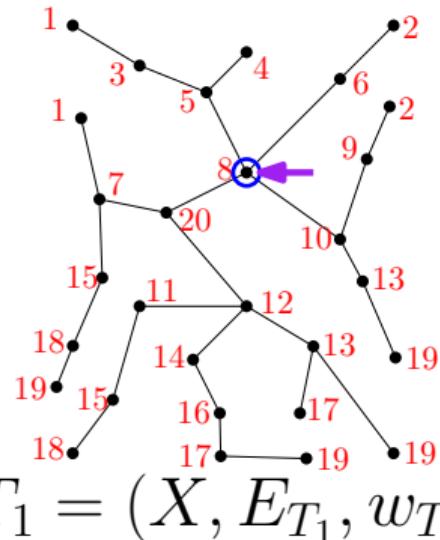
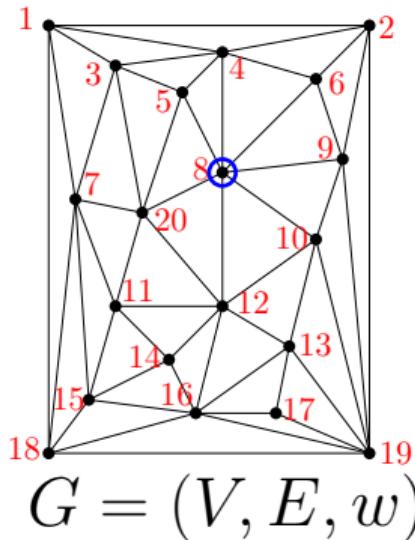


Theorem (Clan embedding into trees, [Filtser, Le 21])

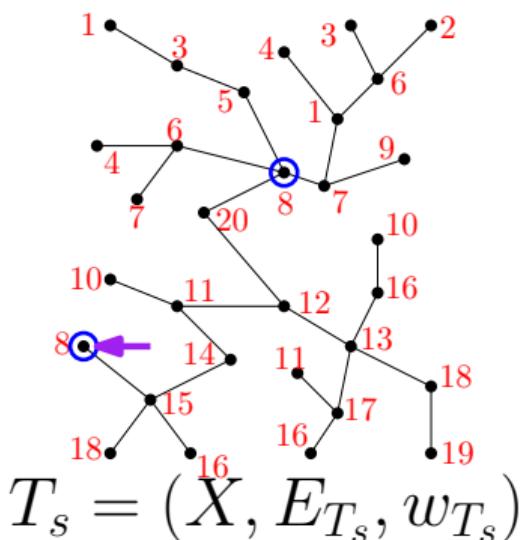
(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(\frac{\log n}{\epsilon})$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.



...

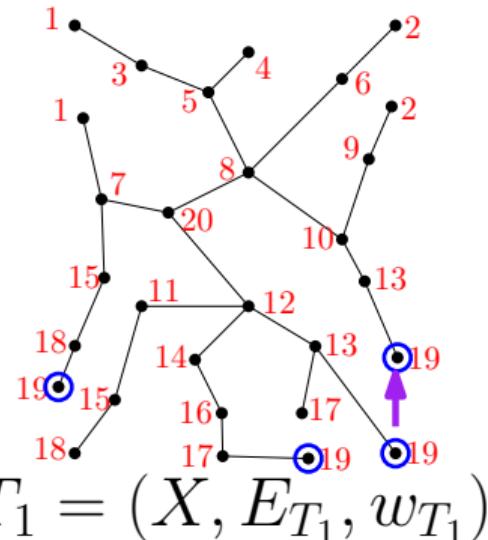
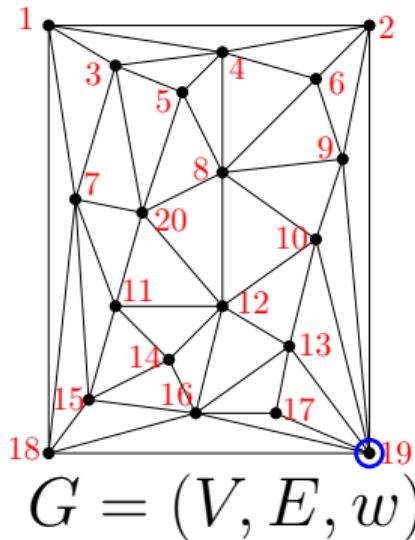


Theorem (Clan embedding into trees, [Filtser, Le 21])

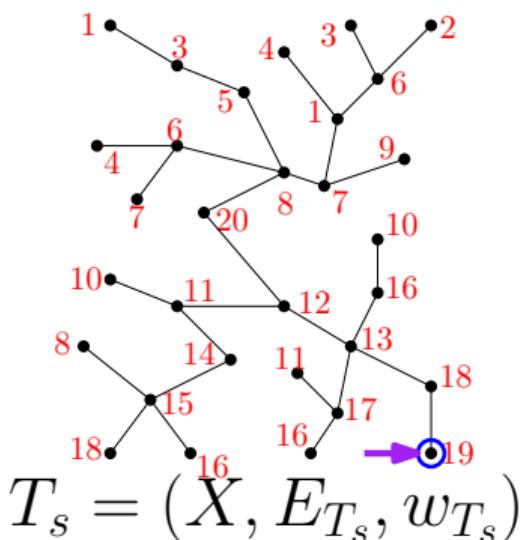
(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(\frac{\log n}{\epsilon})$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.



...



Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O\left(\frac{\log n}{\epsilon}\right)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall \epsilon \in (0, 1)$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O\left(\frac{\log n}{\epsilon}\right)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq 1 + \epsilon$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

(both) Tight!

Clan Embedding

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Compromises: Not a real classic embedding



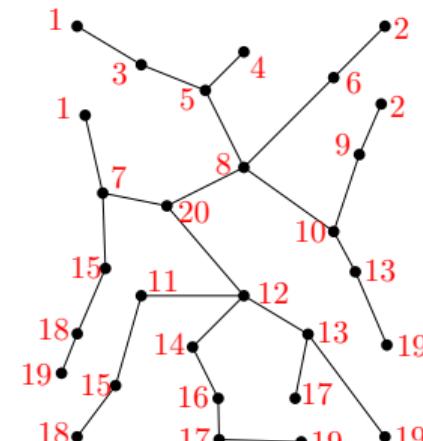
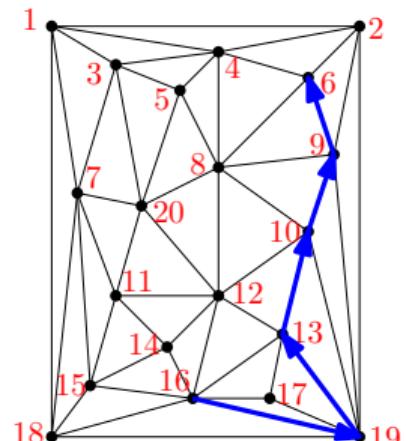
Path Distortion

Originally appeared in [Bartal, Mendel 04] in the context of multi-embeddings.

Definition

One-to-many embedding $f : X \rightarrow 2^Y$ has *path-distortion* t if for every sequence (x_0, x_1, \dots, x_m) in X there is a sequence y_0, \dots, y_m where $y_i \in f(x_i)$, and

$$\sum\nolimits_{i=0}^{m-1} d_Y(y_i, y_{i+1}) \leq t \cdot \sum\nolimits_{i=1}^{m-1} d_X(x_i, x_{i+1}).$$



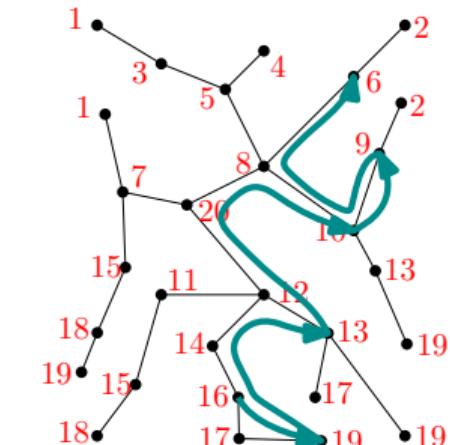
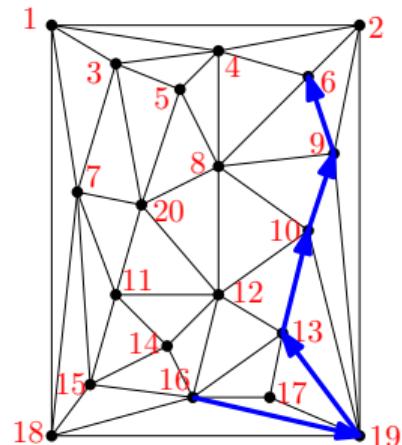
Path Distortion

Originally appeared in [Bartal, Mendel 04] in the context of multi-embeddings.

Definition

One-to-many embedding $f : X \rightarrow 2^Y$ has *path-distortion* t if for every sequence (x_0, x_1, \dots, x_m) in X there is a sequence y_0, \dots, y_m where $y_i \in f(x_i)$, and

$$\sum_{i=0}^{m-1} d_Y(y_i, y_{i+1}) \leq t \cdot \sum_{i=1}^{m-1} d_X(x_i, x_{i+1}).$$



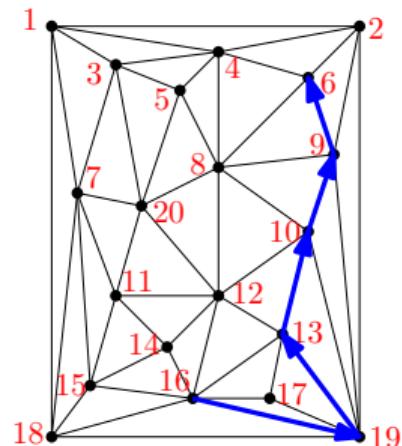
Path Distortion

Originally appeared in [Bartal, Mendel 04] in the context of multi-embeddings.

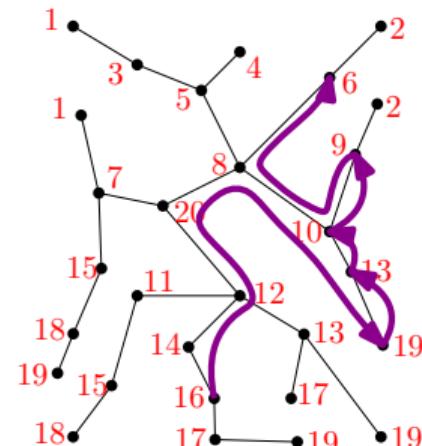
Definition

One-to-many embedding $f : X \rightarrow 2^Y$ has *path-distortion* t if for every sequence (x_0, x_1, \dots, x_m) in X there is a sequence y_0, \dots, y_m where $y_i \in f(x_i)$, and

$$\sum_{i=0}^{m-1} d_Y(y_i, y_{i+1}) \leq t \cdot \sum_{i=1}^{m-1} d_X(x_i, x_{i+1}).$$



$$G = (V, E, w)$$

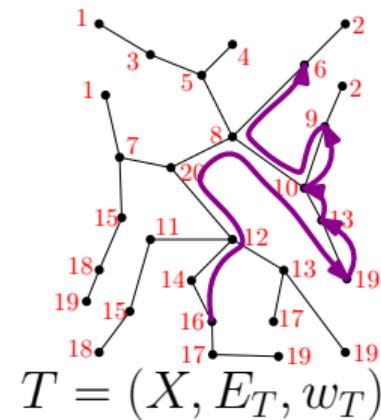
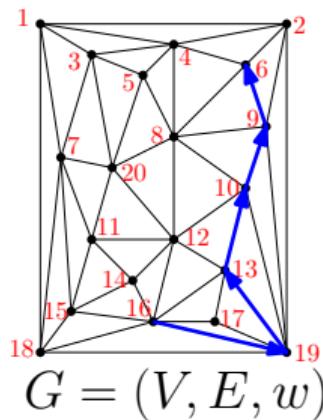


$$T = (X, E_T, w_T)$$

Definition

One-to-many embedding $f : X \rightarrow 2^Y$ has *path-distortion* t if for every sequence (x_0, x_1, \dots, x_m) in X there is a sequence y_0, \dots, y_m where $y_i \in f(x_i)$, and

$$\sum_{i=0}^{m-1} d_Y(y_i, y_{i+1}) \leq t \cdot \sum_{i=1}^{m-1} d_X(x_i, x_{i+1}).$$



Theorem (Clan embedding into trees, [Filtser, Le 21])

Our clan embeddings with distortion $O(k)$ have path distortion $O(k \cdot \log n)$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

Our clan embeddings with distortion $O(k)$ have path distortion $O(k \cdot \log n)$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Theorem (Clan embedding into trees, [Filtser, Le 21])

Our clan embeddings with distortion $O(k)$ have path distortion $O(k \cdot \log n)$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

That is there is a one-to-many embedding with a total of $O(n^{1+\frac{1}{k}})$ copies
and path distortion $O(k \cdot \log n)$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

Our clan embeddings with distortion $O(k)$ have path distortion $O(k \cdot \log n)$.

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

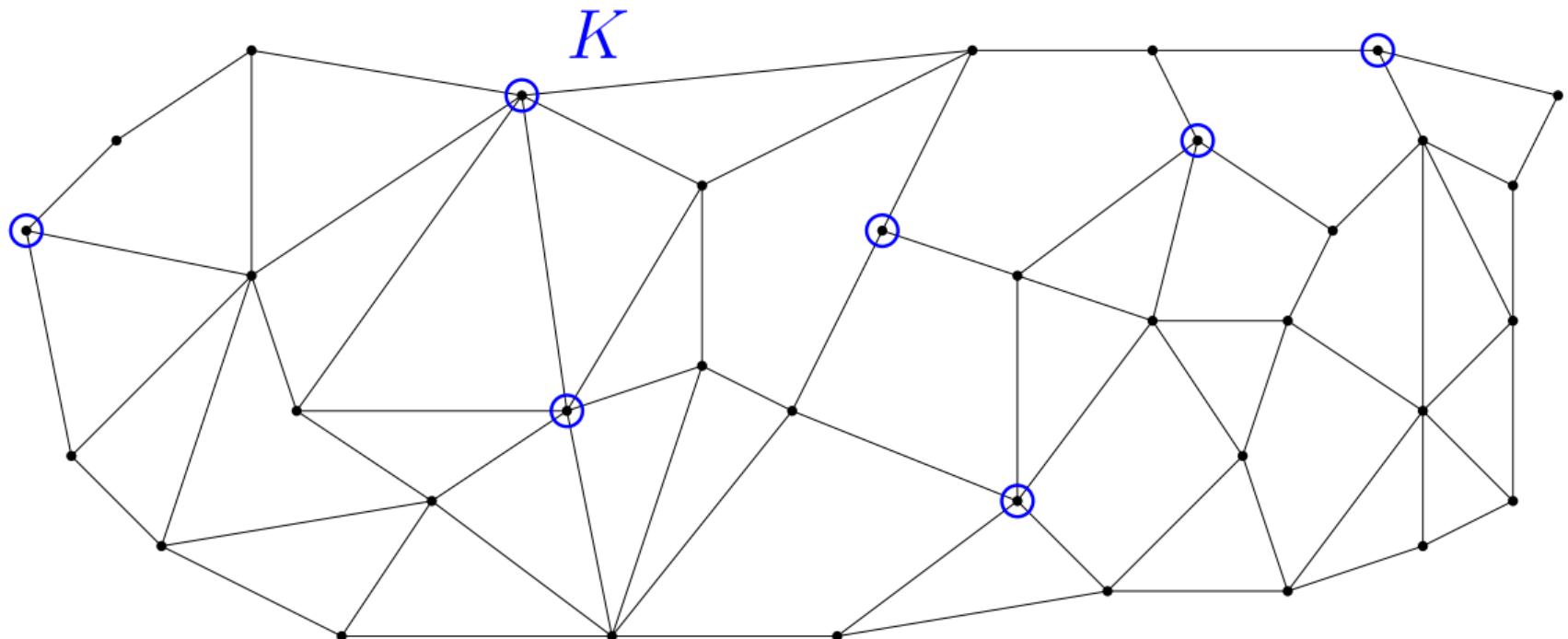
- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

That is there is a one-to-many embedding with a total of $O(n^{1+\frac{1}{k}})$ copies
and path distortion $O(k \cdot \log n)$.

Or a total of $O(n^{1+\frac{1}{2}})$ copies and path distortion $O(\log n)$.

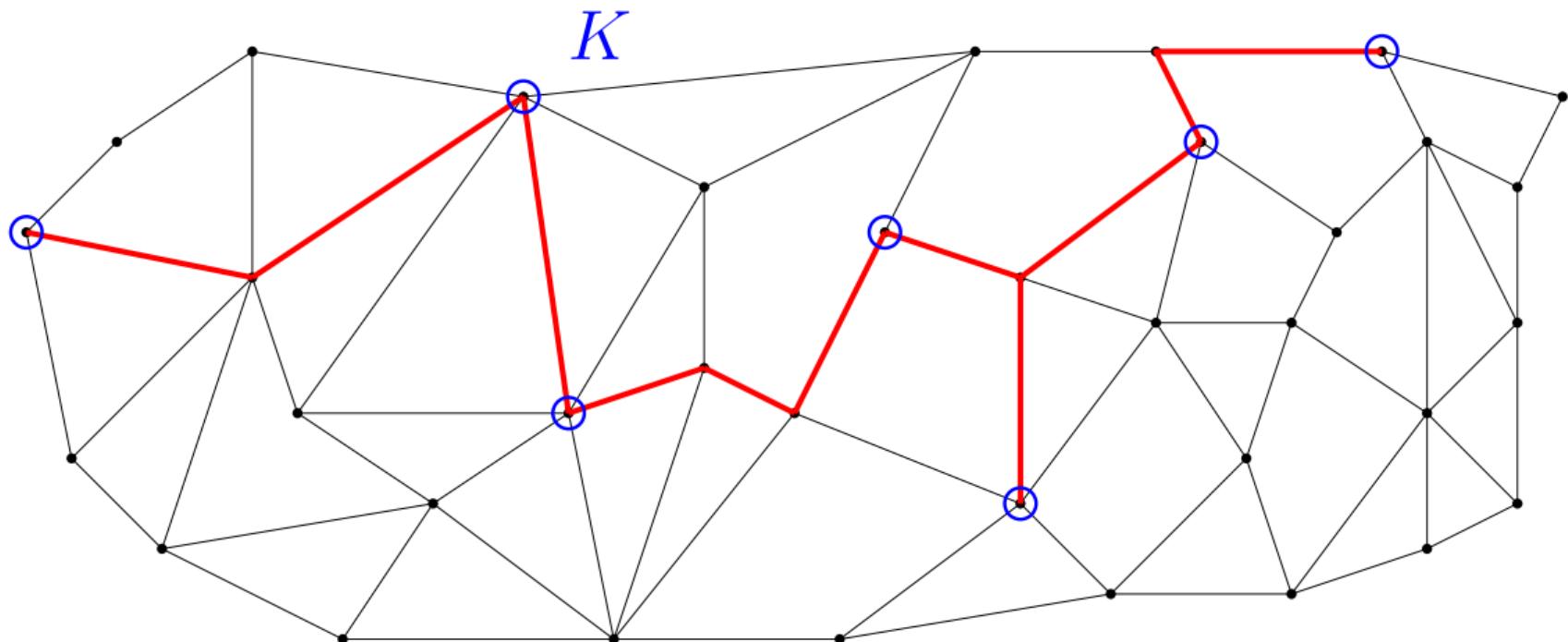
Steiner Tree

Given set of terminals K , find minimum weight tree T spanning K



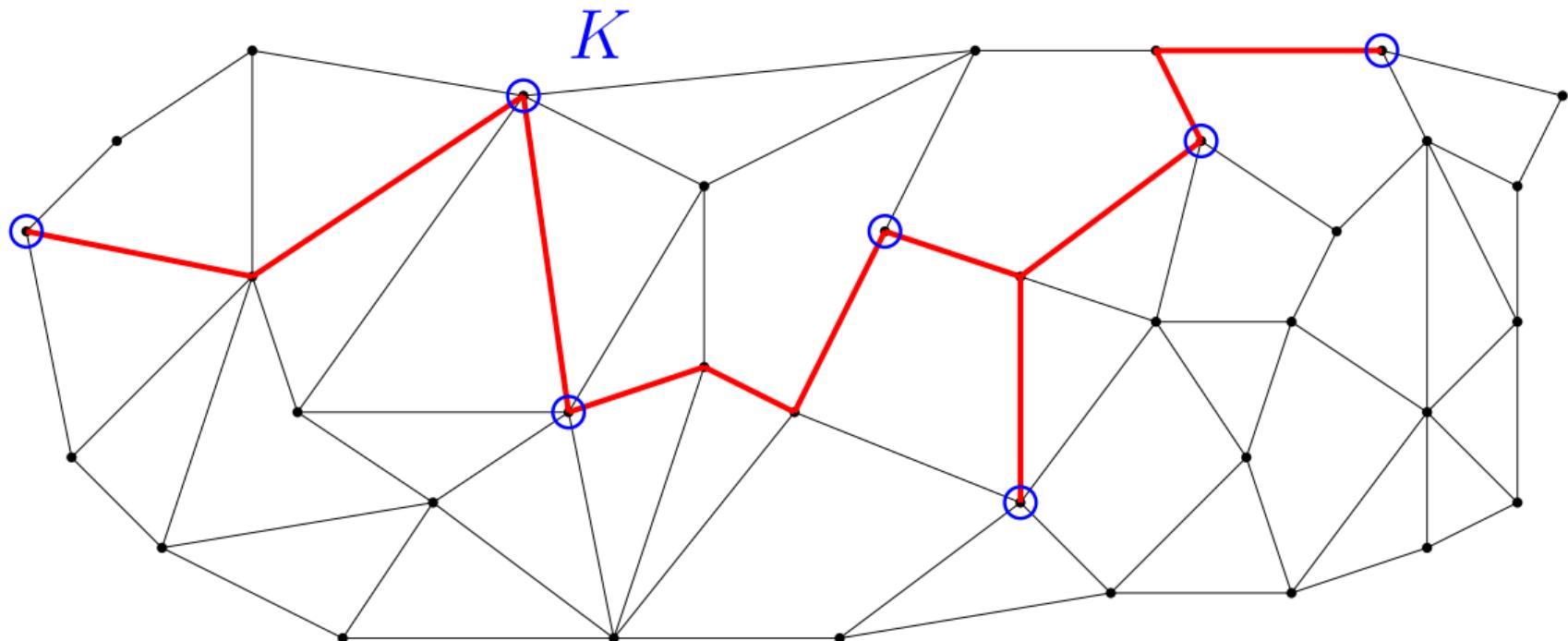
Steiner Tree

Given set of terminals K , find minimum weight tree T spanning K



Steiner Tree

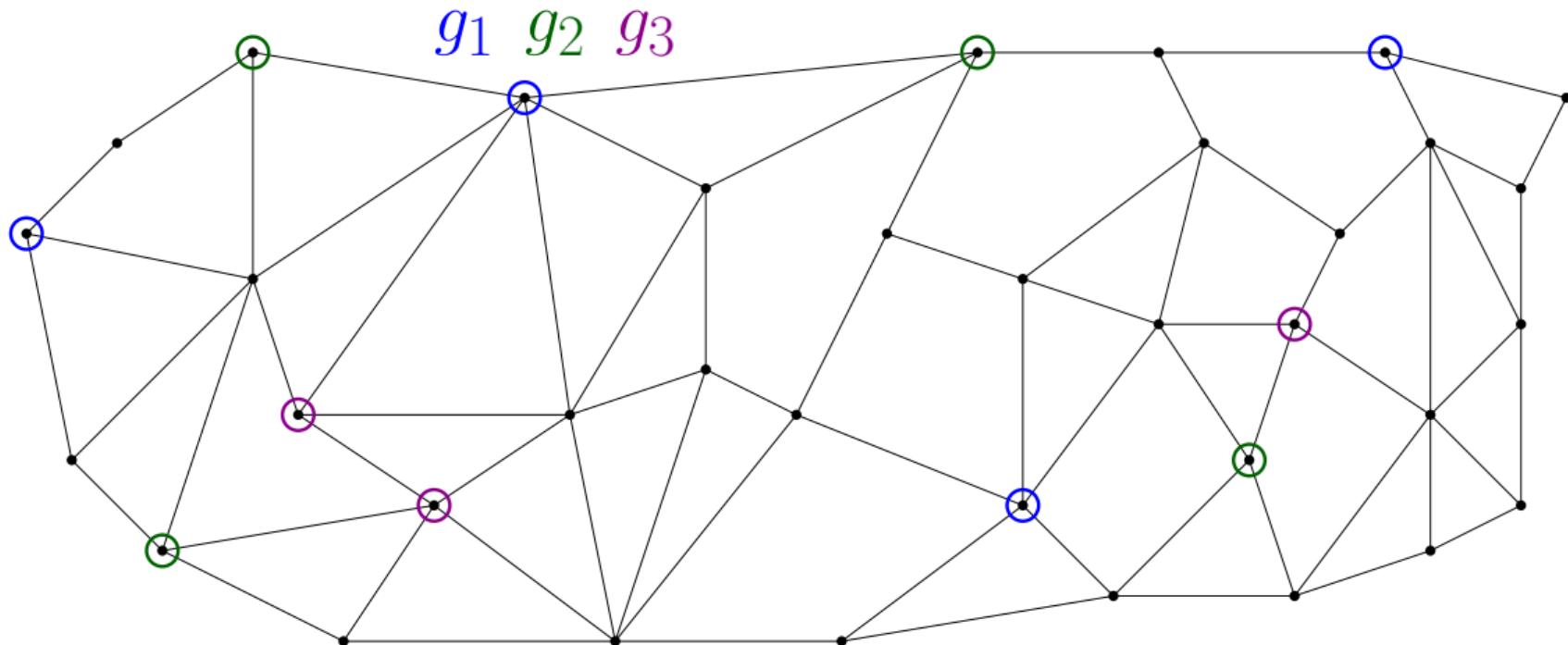
Given set of terminals K , find minimum weight tree T spanning K



In class we saw a 2-approximation algorithm for the Steiner tree problem.

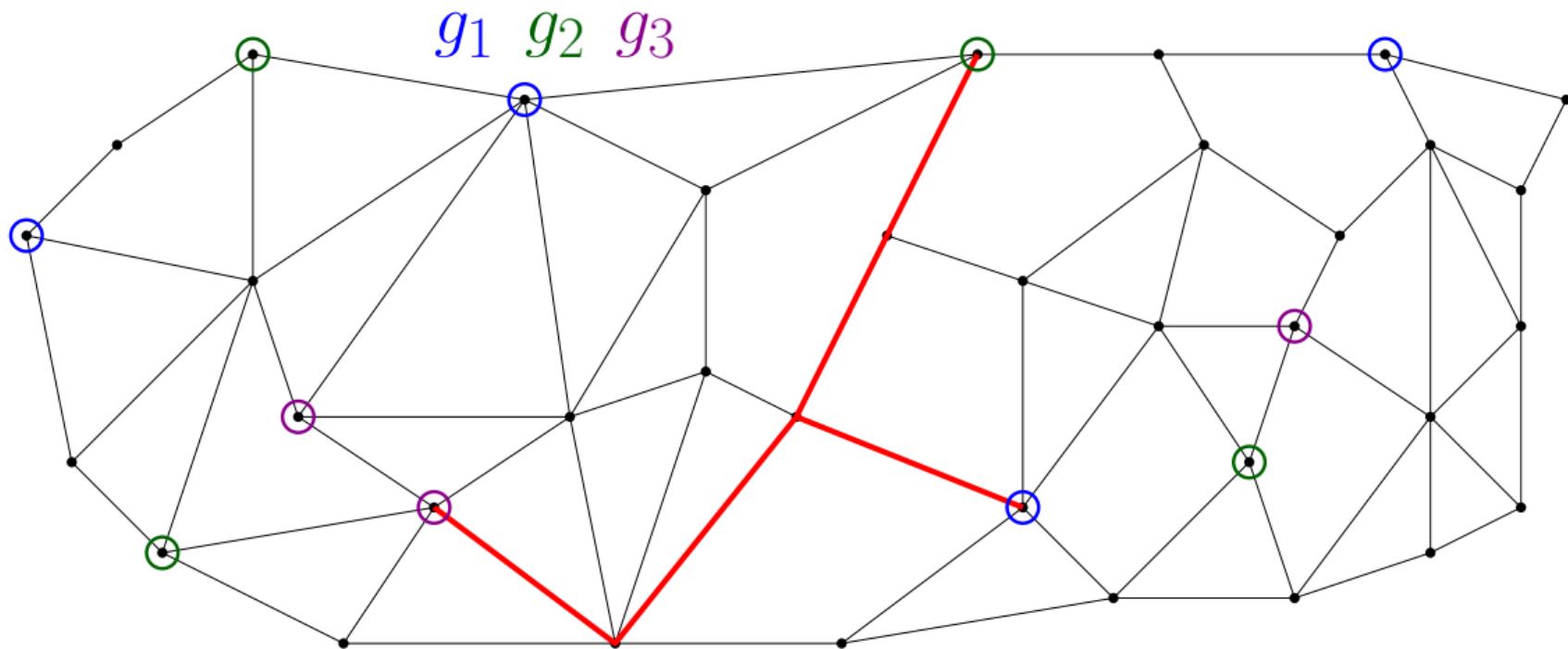
Group Steiner Tree

Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i ;



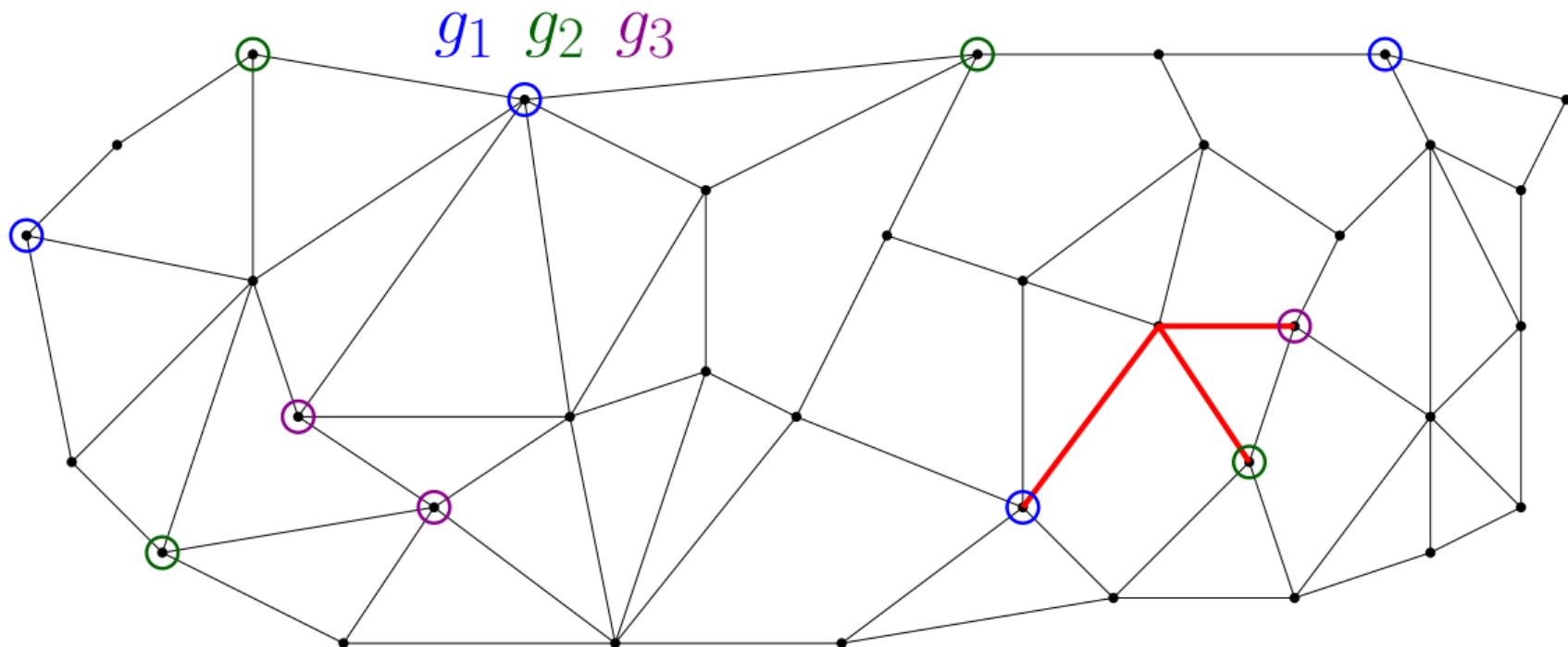
Group Steiner Tree

Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i ;



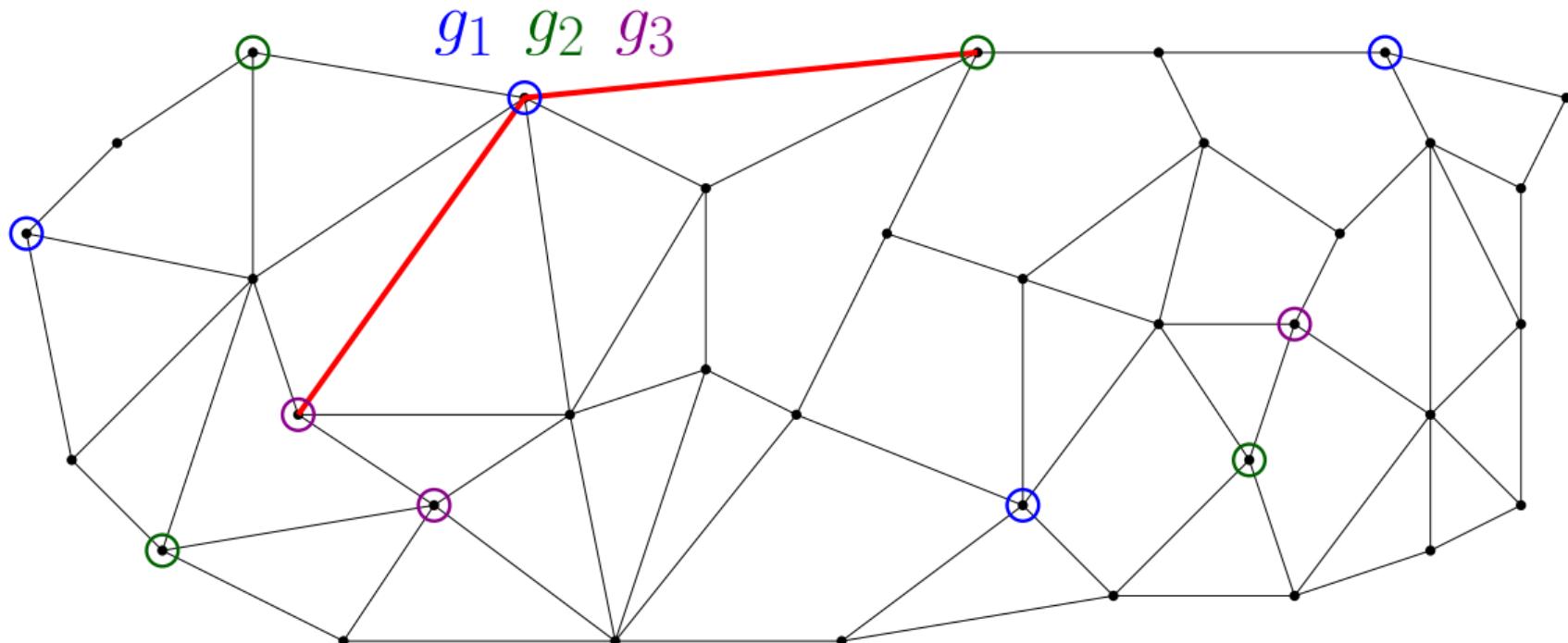
Group Steiner Tree

Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .



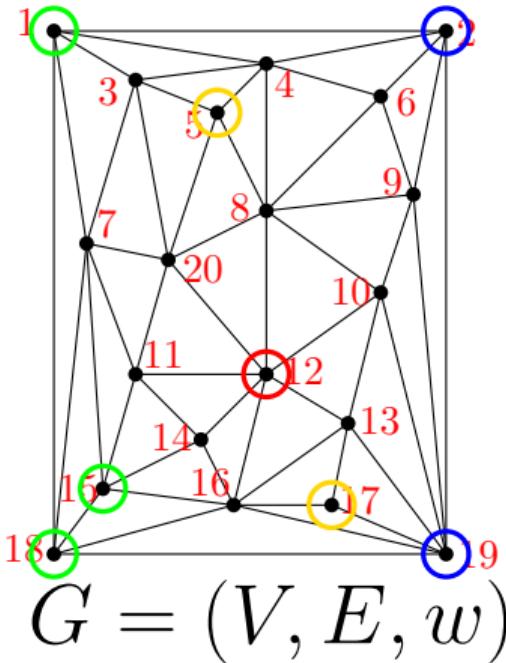
Group Steiner Tree

Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

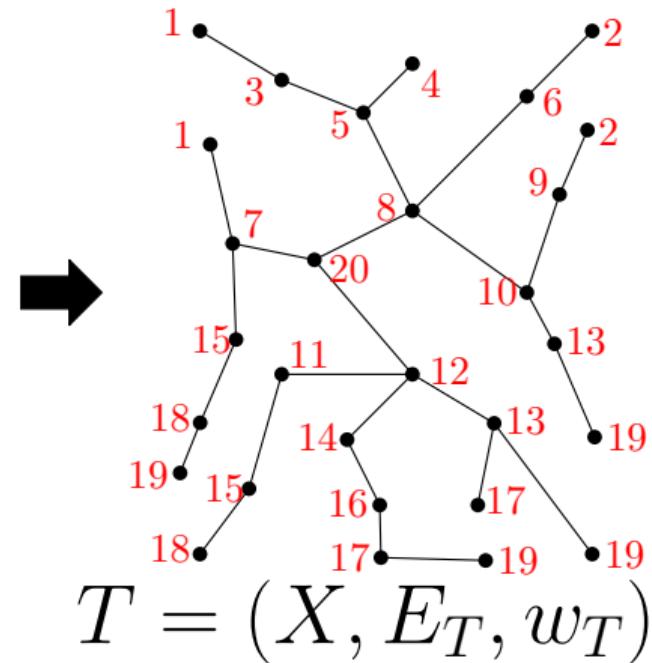


Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

Clan embedding f with path distortion $O(\log n)$.



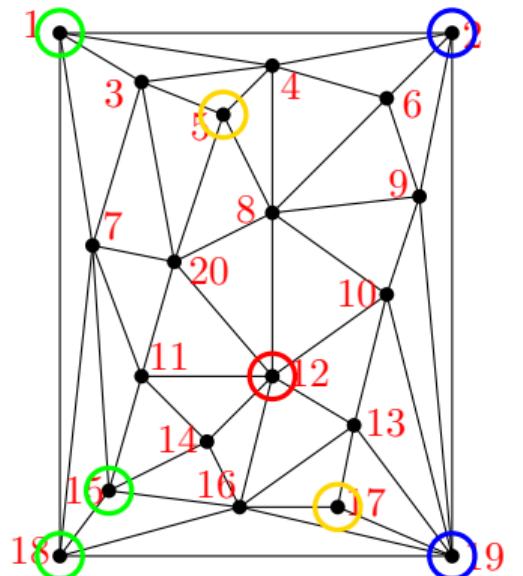
$g_1 \ g_2 \ g_3 \ g_4$



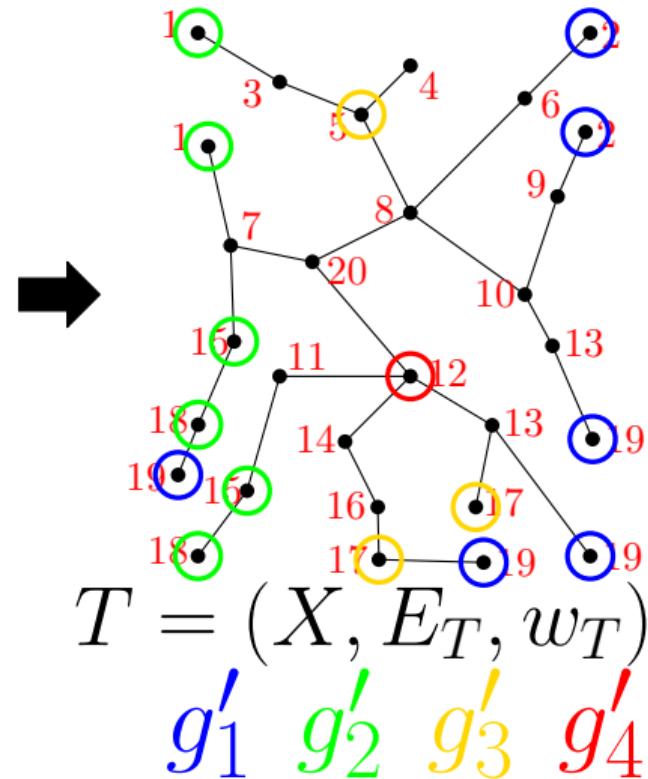
Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

Clan embedding f with path distortion $O(\log n)$.

$$g'_i = f(g_i)$$



$$g'_1 \ g'_2 \ g'_3 \ g'_4$$

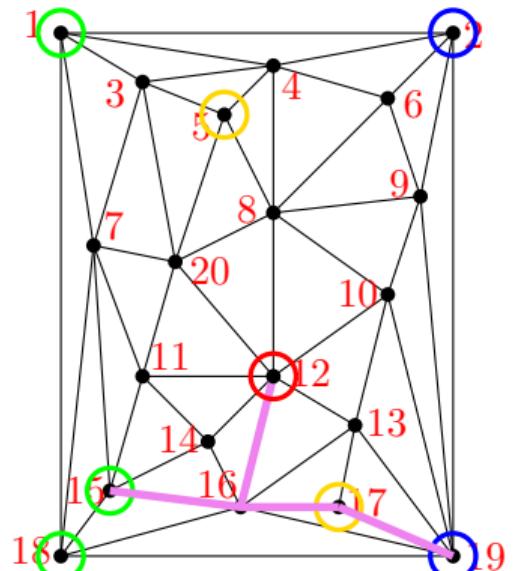


Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

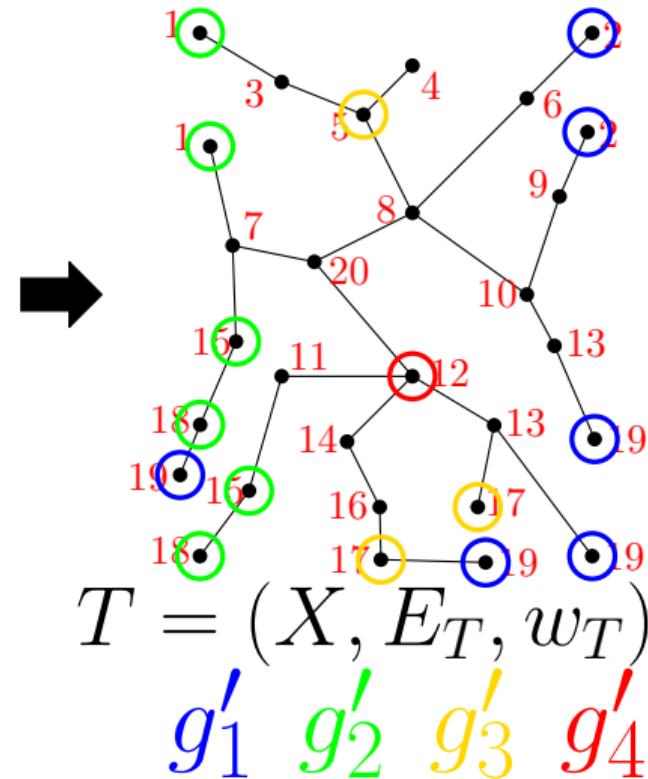
Clan embedding f with path distortion $O(\log n)$.

$$g'_i = f(g_i)$$

S^* optimal solution.



$$g_1 \ g_2 \ g_3 \ g_4$$



$$g'_1 \ g'_2 \ g'_3 \ g'_4$$

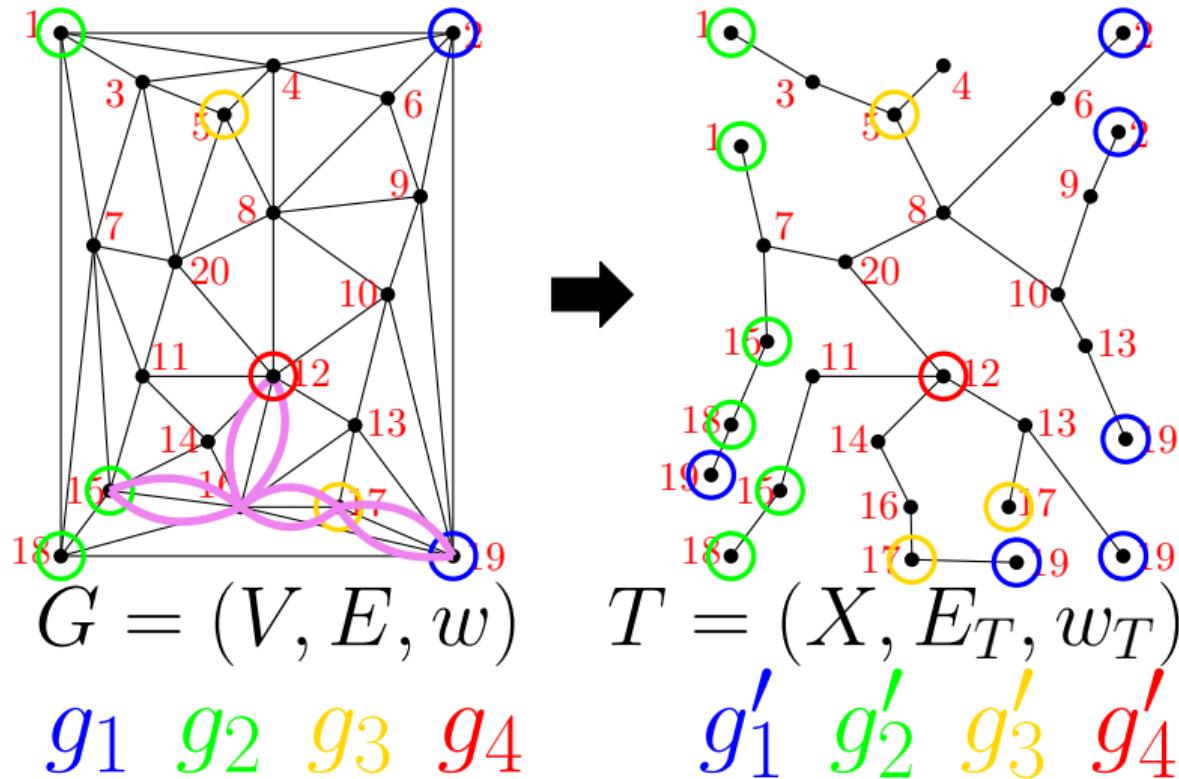
Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

Clan embedding f with path distortion $O(\log n)$.

$$g'_i = f(g_i)$$

S^* optimal solution.

Double each edge: $2S^*$



Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

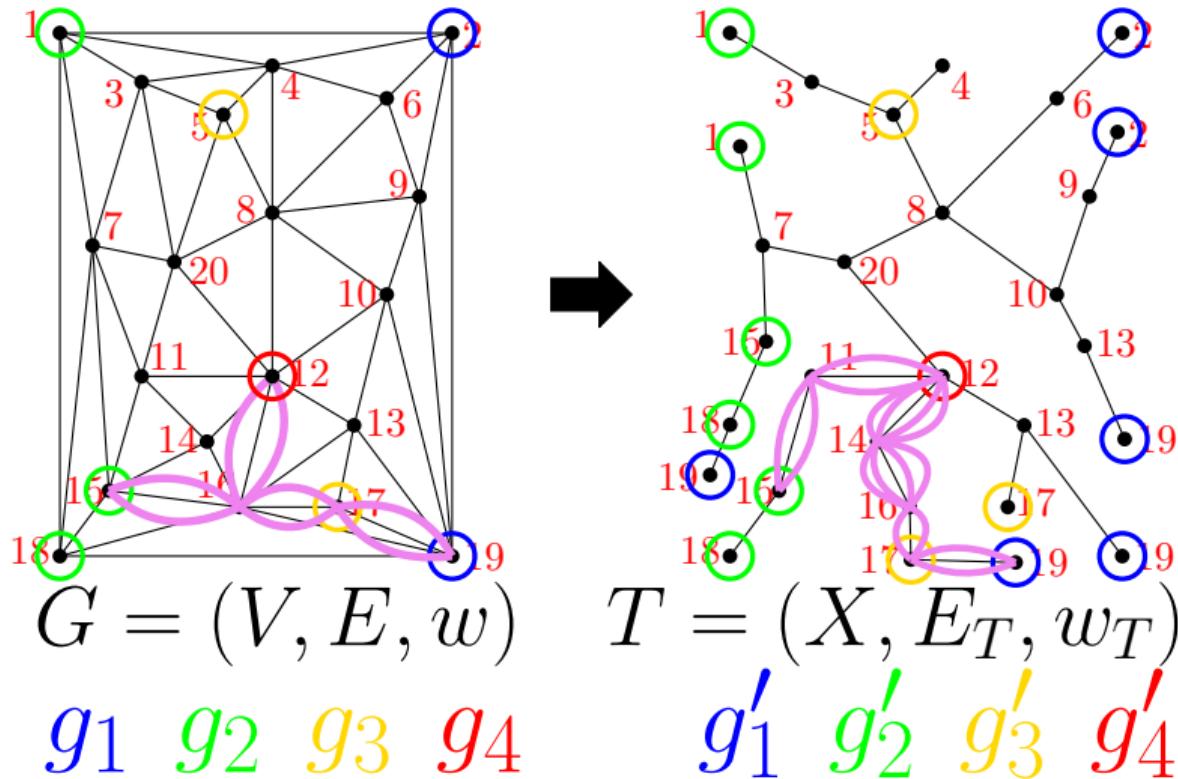
Clan embedding f with path distortion $O(\log n)$.

$$g'_i = f(g_i)$$

S^* optimal solution.

Double each edge: $2S^*$

Guaranteed path: S_T^*
(valid solution),
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$



Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

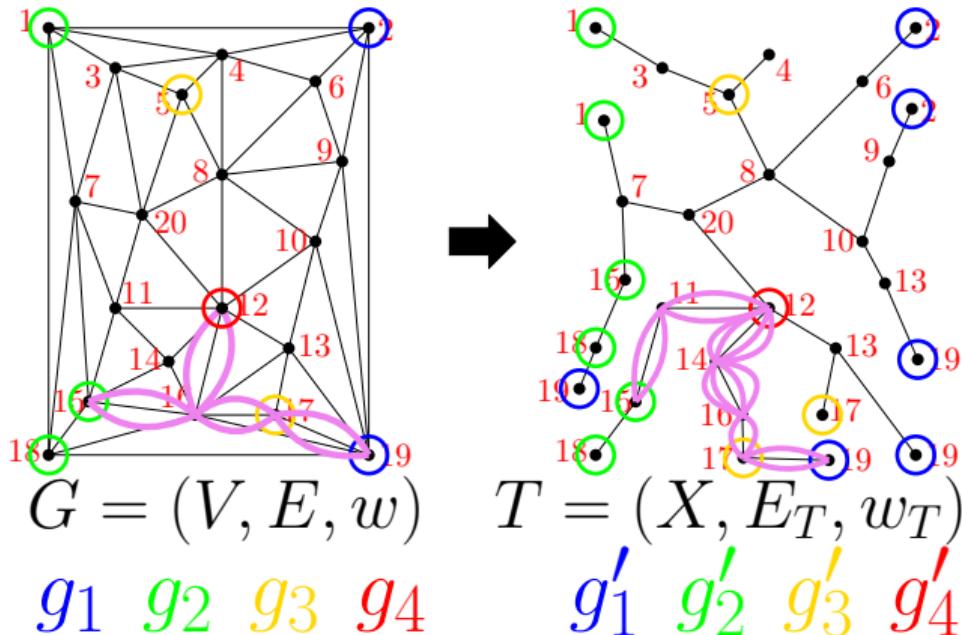
Clan embedding f with path distortion $O(\log n)$.

$$g'_i = f(g_i)$$

S^* optimal solution.

Double each edge: $2S^*$

Guaranteed path: S_T^* ,
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$



Theorem ([Garg, Konjevod, Ravi 00])

$O(\log n \cdot \log k)$ -approximation algorithm for the GST problem on trees.

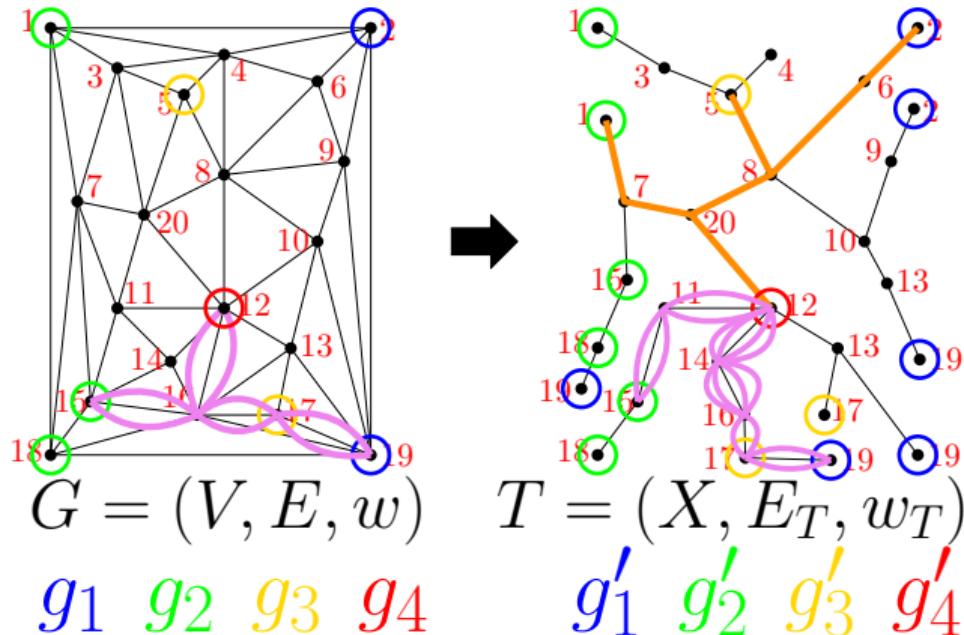
Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

S^* optimal solution.

Double each edge: $2S^*$

Guaranteed path: S_T^* ,
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$

\tilde{S}_T solution, $w(\tilde{S}_T) \leq$
 $O(\log n \cdot \log k) \cdot w(S_T^*) \leq$
 $O(\log^2 n \cdot \log k) \cdot w(S^*)$



Theorem ([Garg, Konjevod, Ravi 00])

$O(\log n \cdot \log k)$ -approximation algorithm for the GST problem on trees.

Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

S^* optimal solution.

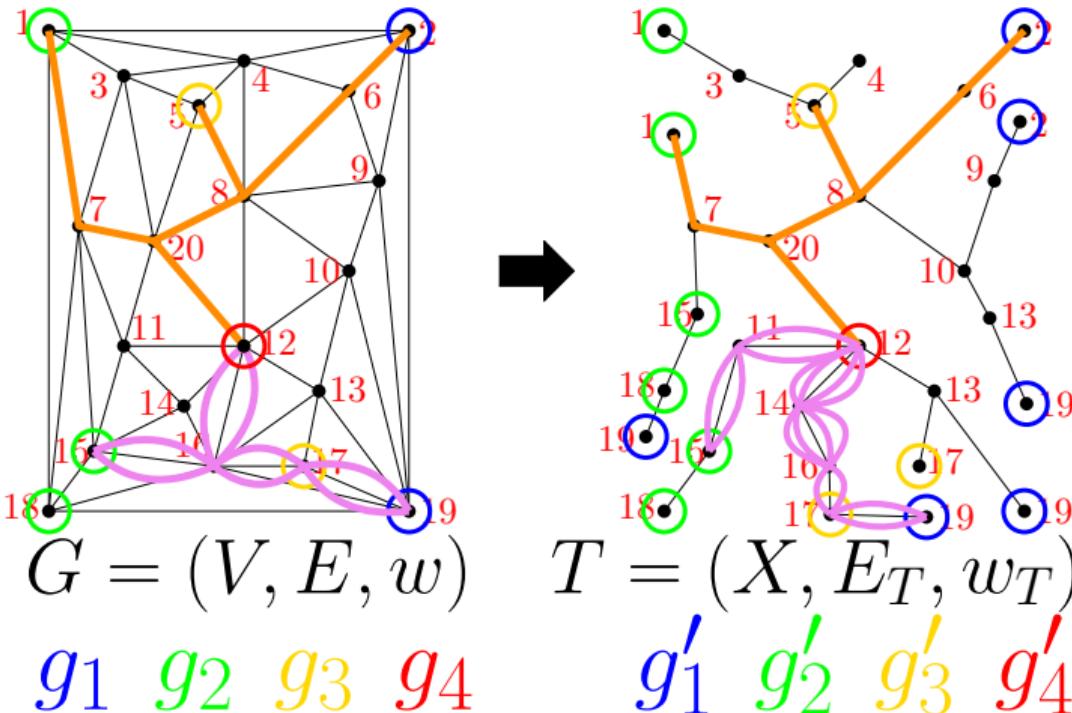
Double each edge: $2S^*$

Guaranteed path: S_T^* ,
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$

\tilde{S}_T solution, $w(\tilde{S}_T) \leq$
 $O(\log n \cdot \log k) \cdot w(S_T^*) \leq$
 $O(\log^2 n \cdot \log k) \cdot w(S^*)$

Return: $\tilde{S} = f^{-1}(\tilde{S}_T)$

$\tilde{S} = \bigcup_{\{v', u'\} \in \tilde{S}_T} P_{v', u'}^T$ is a union of shortest paths in G



Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

S^* optimal solution.

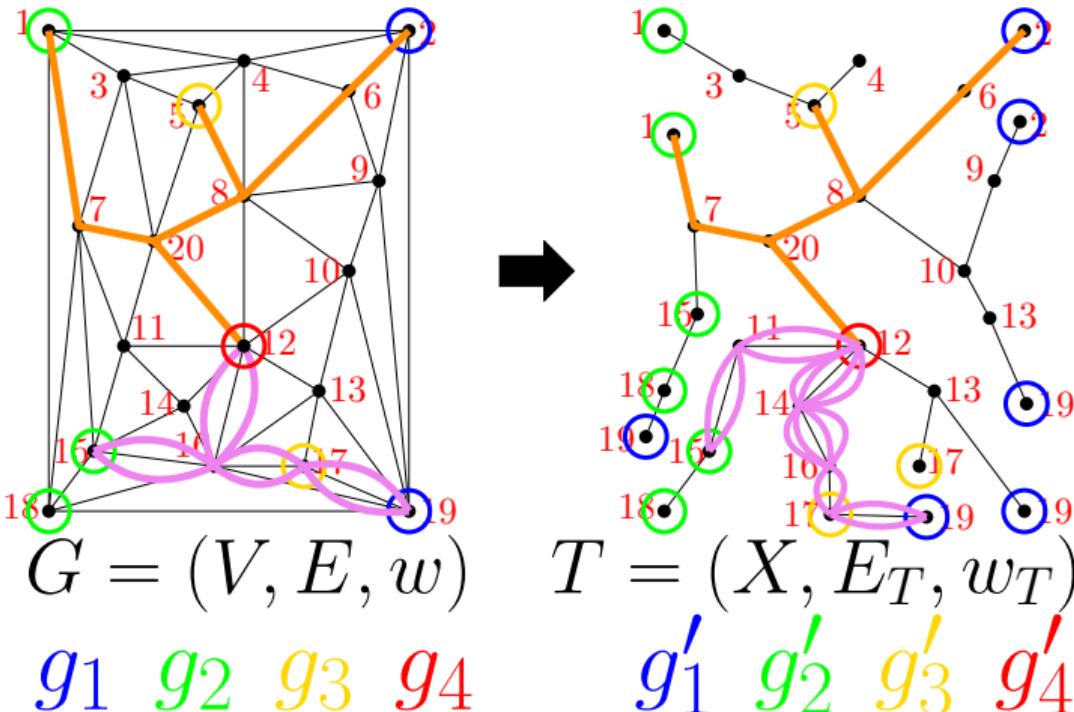
Double each edge: $2S^*$

Guaranteed path: S_T^* ,
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$

\tilde{S}_T solution, $w(\tilde{S}_T) \leq O(\log n \cdot \log k) \cdot w(S_T^*) \leq O(\log^2 n \cdot \log k) \cdot w(S^*)$

Return: $\tilde{S} = f^{-1}(\tilde{S}_T)$

$\tilde{S} = \bigcup_{\{v', u'\} \in \tilde{S}_T} P_{v', u'}^T$ is a union of shortest paths in G
 $w(\tilde{S}) \leq w(\tilde{S}_T) \leq O(\log^2 n \cdot \log k) \cdot w(S^*)$.



Given subsets $g_1, g_2, \dots, g_k \subseteq V$, find minimum weight tree T spanning at least one vertex from each g_i .

S^* optimal solution.

Double each edge: $2S^*$

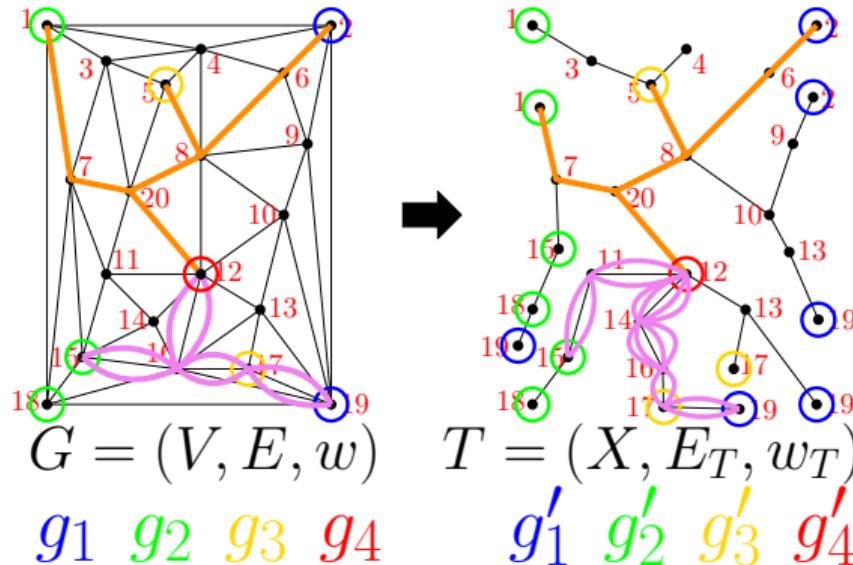
Guaranteed path: S_T^* ,
 $w(S_T^*) \leq O(\log n) \cdot w(S^*)$

\tilde{S}_T solution, $w(\tilde{S}_T) \leq O(\log n \cdot \log k) \cdot w(S_T^*) \leq O(\log^2 n \cdot \log k) \cdot w(S^*)$

Return: $\tilde{S} = f^{-1}(\tilde{S}_T)$

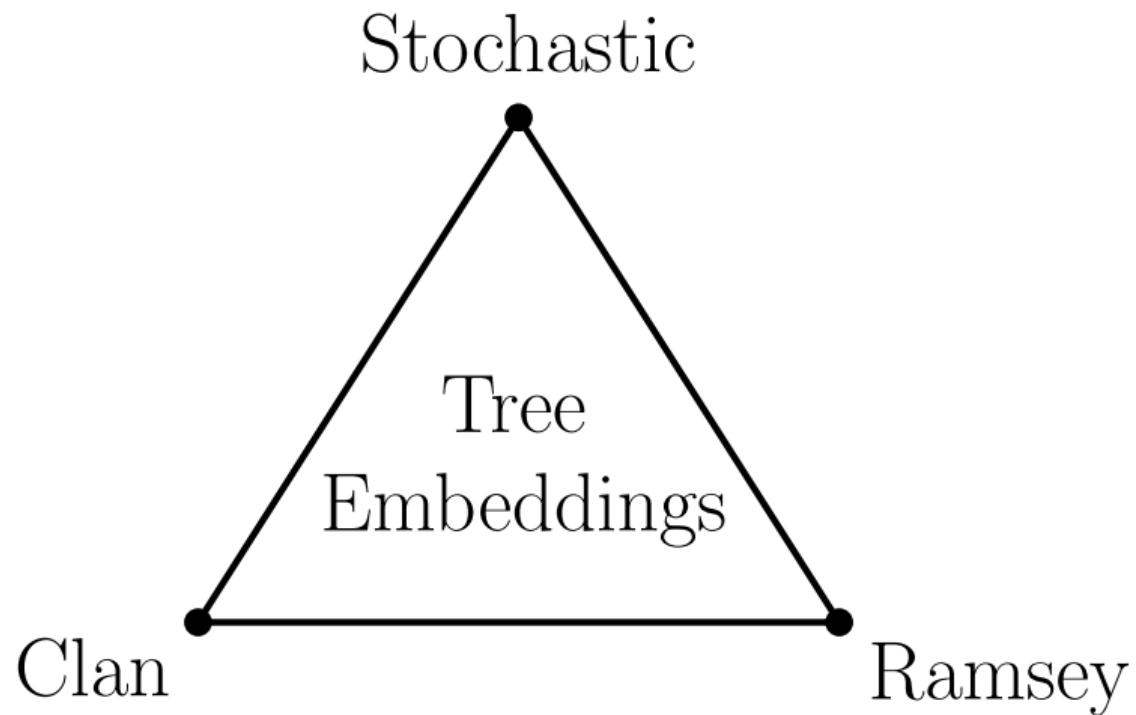
$$w(\tilde{S}) \leq w(\tilde{S}_T) \leq O(\log^2 n \cdot \log k) \cdot w(S^*).$$

We got an $O(\log^2 n \cdot \log k)$ **approximation**.

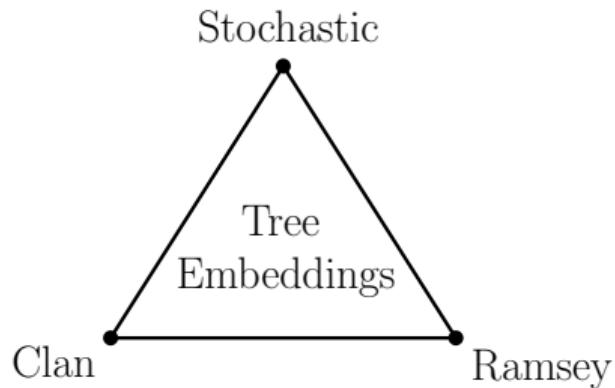


Outline of the talk

- 1 Introduction
- 2 Stochastic embedding into trees
- 3 Ramsey type embeddings
- 4 Clan embedding
- 5 Group Steiner Tree
- 6 Conclusion



Trinity



	Worst case guarantee	Satisfy all vertices	one-to-one embedding
Stochastic	✗	✓	✓
Ramsey	✓	✗	✓
Clan	✓	✓	✗

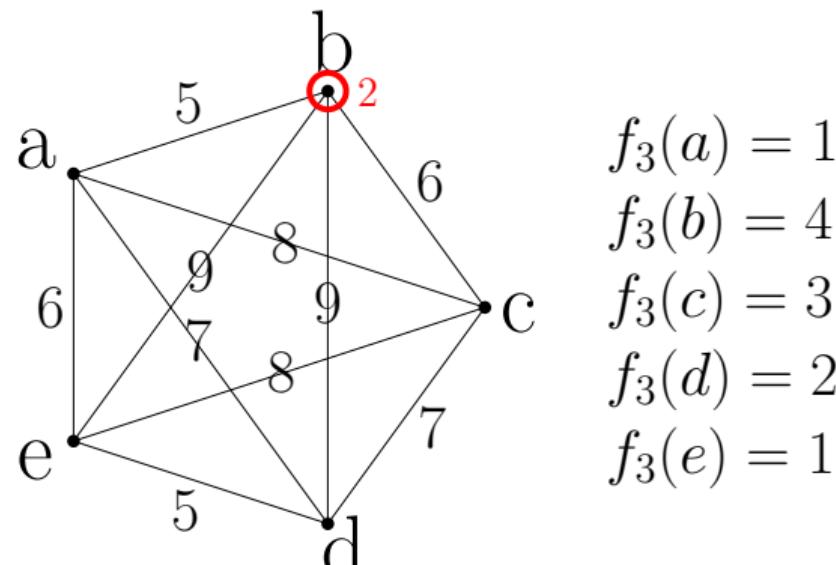
Online problem - Metrical Task System

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



$$\text{Cost alg}(2) = 3 + 5 + 1 = 9$$

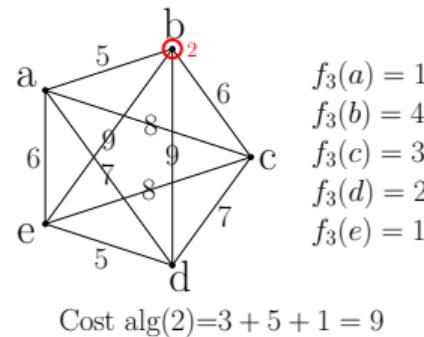
Online problem - Metrical Task System

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.



We showed competitive ratio against oblivious adversary:

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\mathbb{E}[\text{Alg}(I)]}{\text{opt}(I)} = O(\log^2 n \cdot \log \log n).$$

Online problem - Metrical Task System

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

We showed competitive ratio against oblivious adversary:

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\mathbb{E}[\text{Alg}(I)]}{\text{opt}(I)} = O(\log^2 n \cdot \log \log n).$$

Theorem ([Fakcharoenphol, Rao, Talwar 04], improving [Bartal 96+98])

Every n -point metric space (X, d) embeds into **distribution \mathcal{D}** over **dominating trees** with **expected distortion** $O(\log n)$.

Theorem ([Fiat, Mendel 2000])

Given an n point tree* T , there is an online algorithm for MTS with competitive ratio $O(\log n \cdot \log \log n)$ against oblivious adversary.

Online problem - Metrical Task System

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

We showed competitive ratio against oblivious adversary:

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\mathbb{E}[\text{Alg}(I)]}{\text{opt}(I)} = O(\log^2 n \cdot \log \log n).$$

Suppose that you are given a **deterministic** k -competitive algorithm for MTS on trees.

Online problem - Metrical Task System

Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Goal: Minimize the competitive ratio between our algorithm to opt.

We showed competitive ratio against oblivious adversary:

$$\text{Competitive ratio} = \max_{\text{input } I} \frac{\mathbb{E}[\text{Alg}(I)]}{\text{opt}(I)} = O(\log^2 n \cdot \log \log n).$$

Suppose that you are given a **deterministic** k -competitive algorithm for MTS on trees.

Problem: Propose a **deterministic** algorithm for **MTS** for a general n -point metric.

What is the competitive ratio?

Link to quiz:



Can also be found in my homepage:

arnold.filtser.com

Or just google Arnold Filtser.

Link to slides:



MTS Input: Metric space (X, d_X) . Initial configuration $x_0 \in X$.

At step i , a task arrives with cost function $f_i : X \rightarrow \mathbb{R}_{\geq 0}$.

Output: Point $x_i \in X$ s.t. the task performed at x_i at cost $d_X(x_{i-1}, x_i) + f_i(x_i)$.

Assumption: there is a **deterministic** k -competitive algorithm for MTS on trees.

Problem: Propose a **deterministic** algorithm for **MTS** for a general n -point metric.
What is the competitive ratio?

Clan Embeddings construction

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Clan Embeddings construction

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f,\chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Lemma

(X, d_X) n point metric, there is **dominating clan embedding** into a tree s.t.:

- (f, χ) has distortion $O(k)$.
- $\sum_{x \in X} |f(x)| \leq O(n^{1+\frac{1}{k}})$

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$

Replace cardinality with weights

Lemma

(X, d_X) n point metric, **and weights** $\mu : X \rightarrow \mathbb{R}_{\geq 1}$

there is dominating clan embedding into a tree s.t.:

- (f, χ) has distortion $O(k)$.
- $\sum_{x \in X} |f(x)| \cdot \mu(x) \leq O(\mu(X)^{1+\frac{1}{k}})$

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$



Replace cardinality with weights

mini-max theorem

Lemma

(X, d_X) n point metric, **and weights** $\mu : X \rightarrow \mathbb{R}_{\geq 1}$

there is **dominating clan embedding** into a tree s.t.:

- (f, χ) has distortion $O(k)$.
- $\sum_{x \in X} |f(x)| \cdot \mu(x) \leq O(\mu(X)^{1+\frac{1}{k}})$

Theorem (Clan embedding into trees, [Filtser, Le 21])

(X, d_X) n point metric space, $\forall k \in \mathbb{N}$, there is

distribution \mathcal{D} over dominating clan embeddings into trees such that:

- $\forall (f, \chi) \in \text{supp}(\mathcal{D})$ has distortion $O(k)$.
- $\forall x \in X, \mathbb{E}_{(f, \chi) \sim \mathcal{D}}[|f(x)|] \leq O(n^{\frac{1}{k}})$



Replace cardinality with weights

mini-max theorem

Lemma

(X, d_X) n point metric, there is **dominating clan embedding** into a tree s.t.:

- (f, χ) has distortion $O(k)$.
- $\sum_{x \in X} |f(x)| \leq O(n^{1+\frac{1}{k}})$

Definition (Ultrametric)

Ultrametric (X, d) is a metric space satisfying the **strong** triangle inequality:

$$\forall x, y, z \in X, \quad d(x, z) \leq \max \{d(x, y), d(y, z)\} .$$

Definition (Ultrametric)

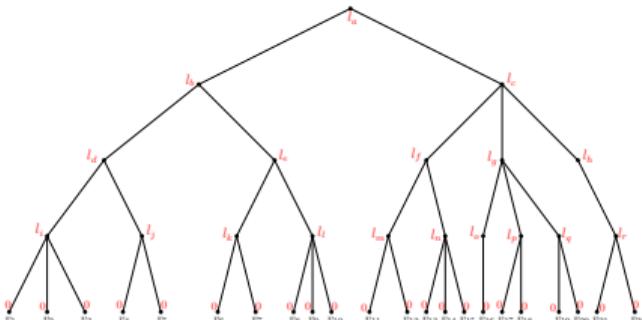
Ultrametric (X, d) is a metric space satisfying the **strong** triangle inequality:

$$\forall x, y, z \in X, \quad d(x, z) \leq \max \{d(x, y), d(y, z)\} .$$

Definition (Hierarchical well-separated tree (HST))

(X, d_X) is a HST if X is mapped (by ϕ) to **leaves** of a rooted tree T where:

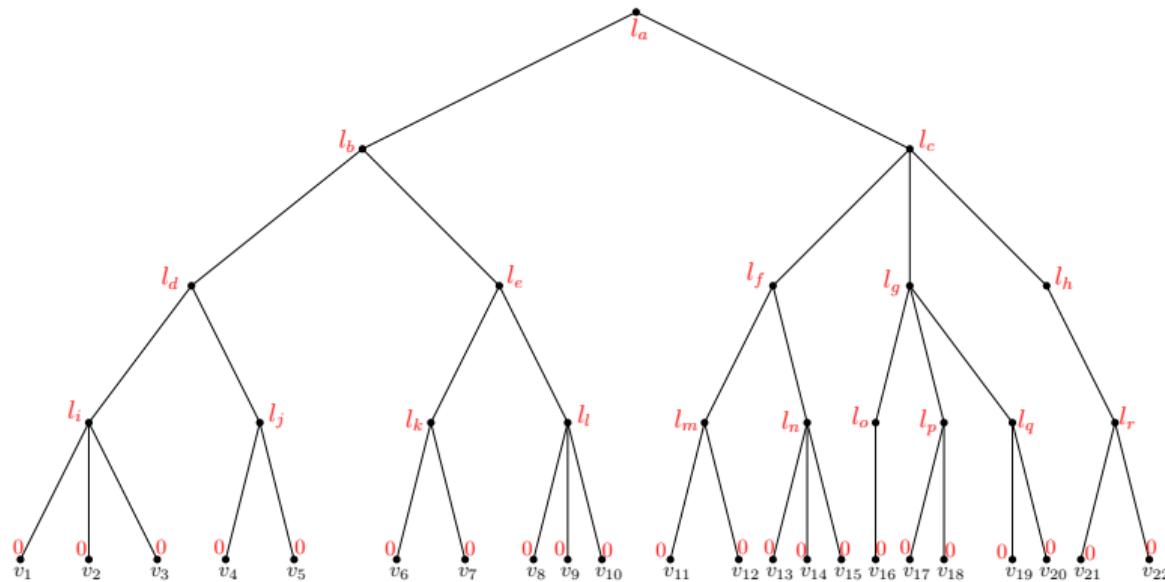
- The nodes of T associated with **monotone** labels l_v .
- $d_X(x, y) = \Gamma_{\text{lca}(\varphi(x), \varphi(y))}$.



Definition (Hierarchical well-separated tree (HST))

(X, d_X) is a HST if X is mapped (by ϕ) to leaves of a rooted tree T where:

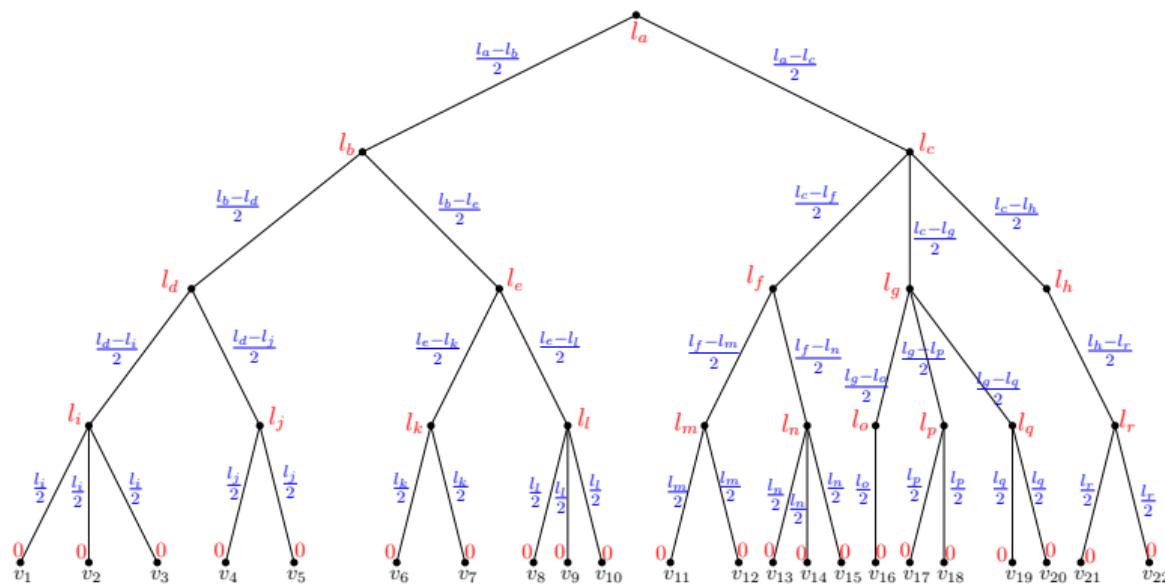
- The nodes of T associated with **monotone** labels l_v .
- $d_X(x, y) = \Gamma_{\text{lca}(\varphi(x), \varphi(y))}$.



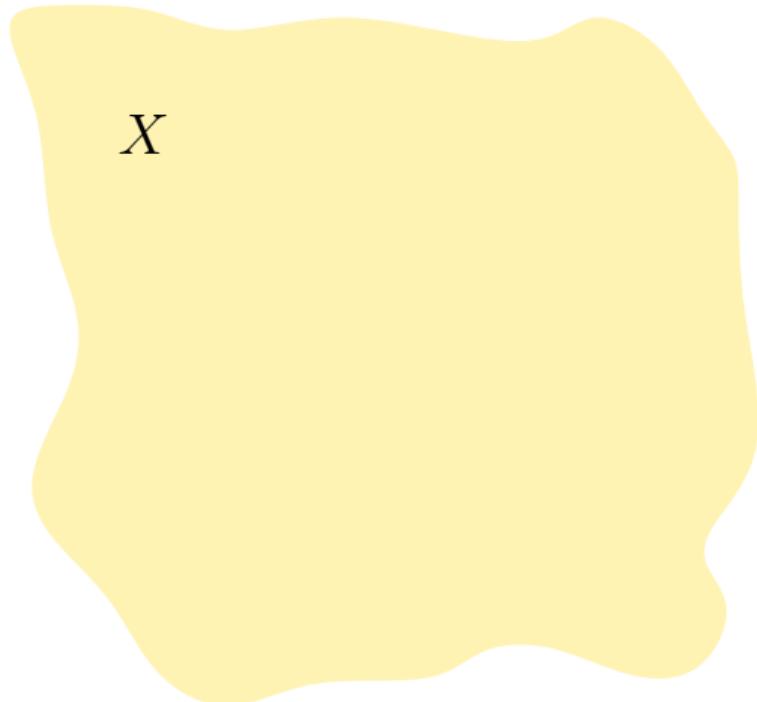
Definition (Hierarchical well-separated tree (HST))

(X, d_X) is a HST if X is mapped (by ϕ) to leaves of a rooted tree T where:

- The nodes of T associated with **monotone** labels l_v .
 - $d_X(x, y) = \Gamma_{\text{lca}(\varphi(x), \varphi(y))}$.



Construction

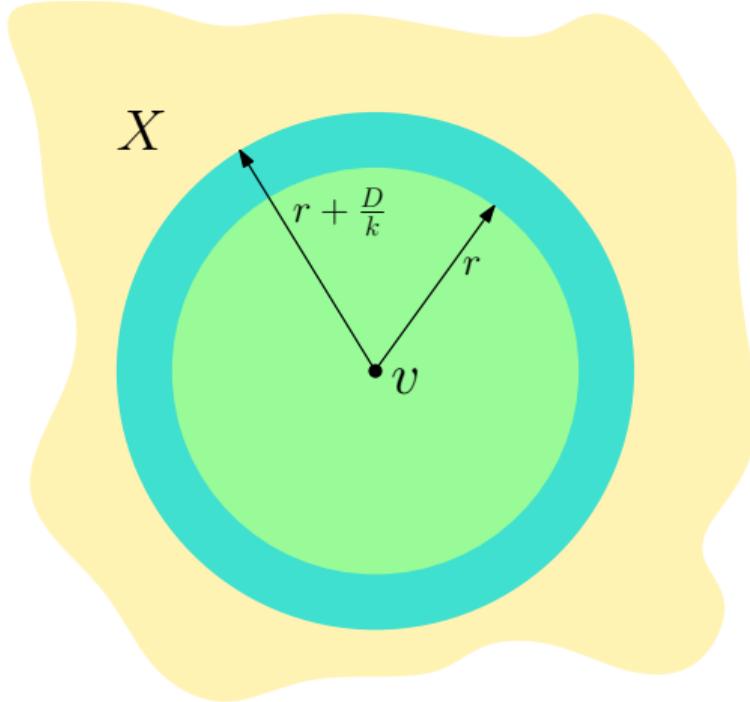


X

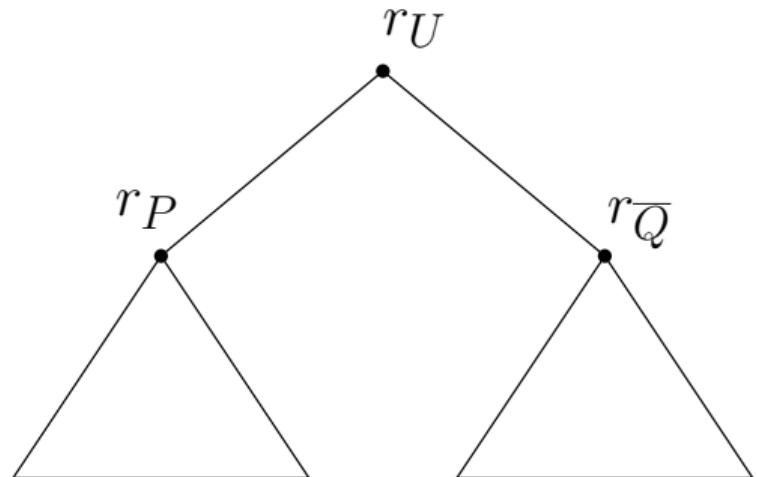
$$\ell(r_U) = \text{diam}(X) = D$$

r_U
•

Construction



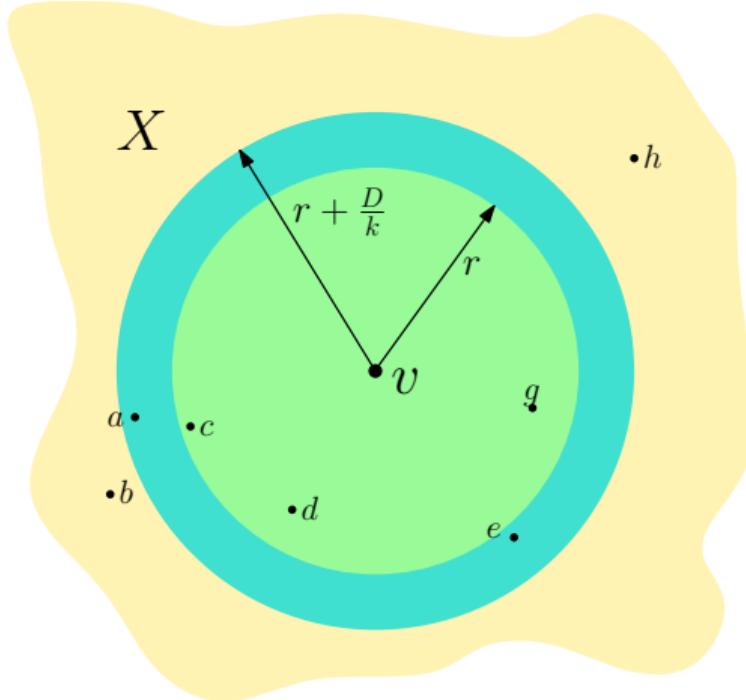
$$\ell(r_U) = \text{diam}(X) = D$$



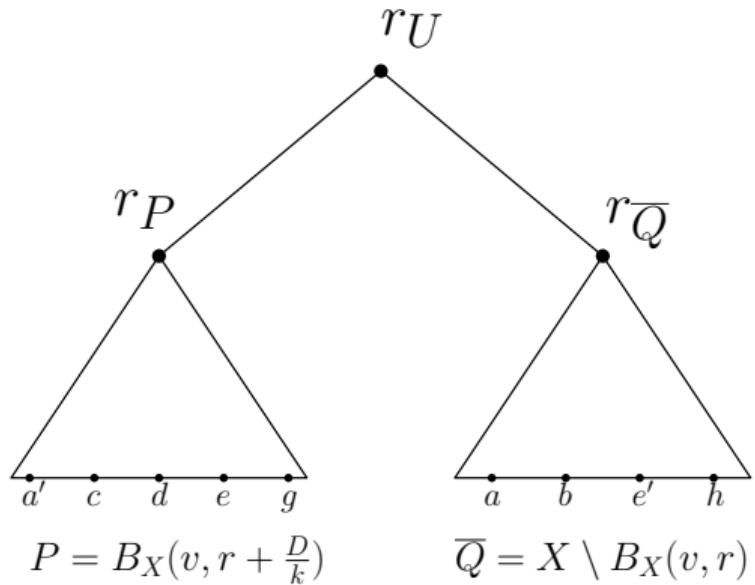
$$P = B_X(v, r + \frac{D}{k})$$

$$\overline{Q} = X \setminus B_X(v, r)$$

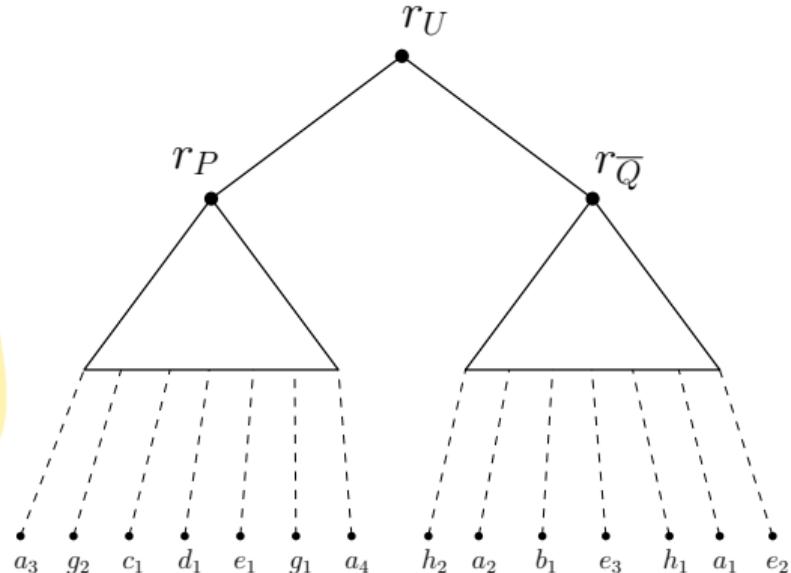
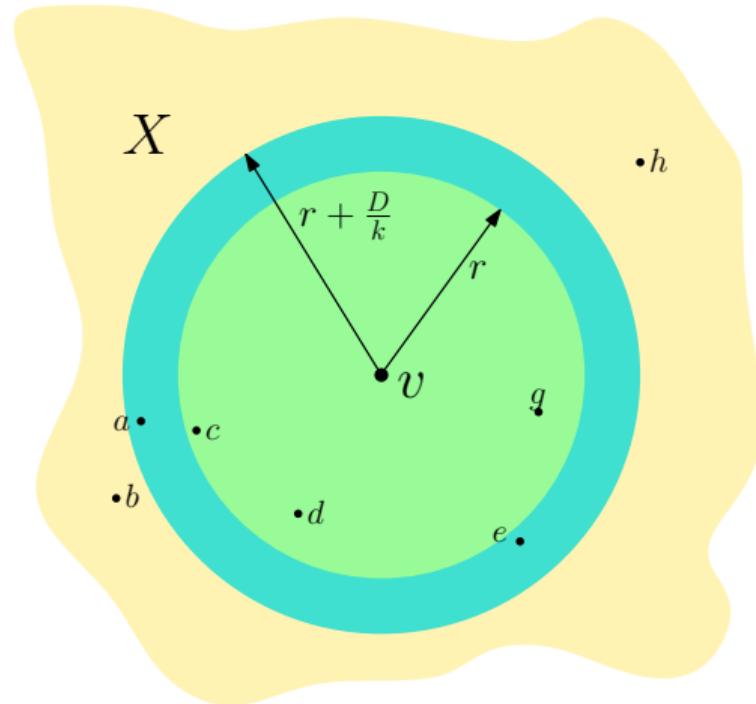
Construction



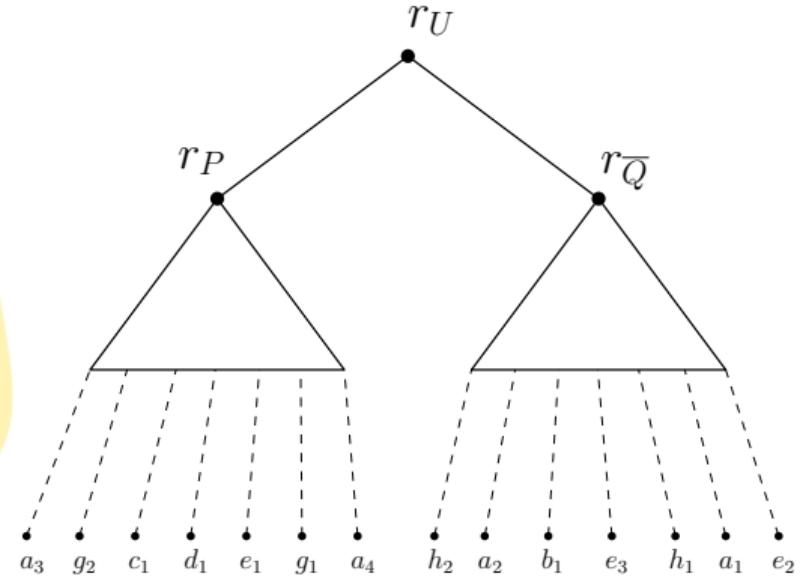
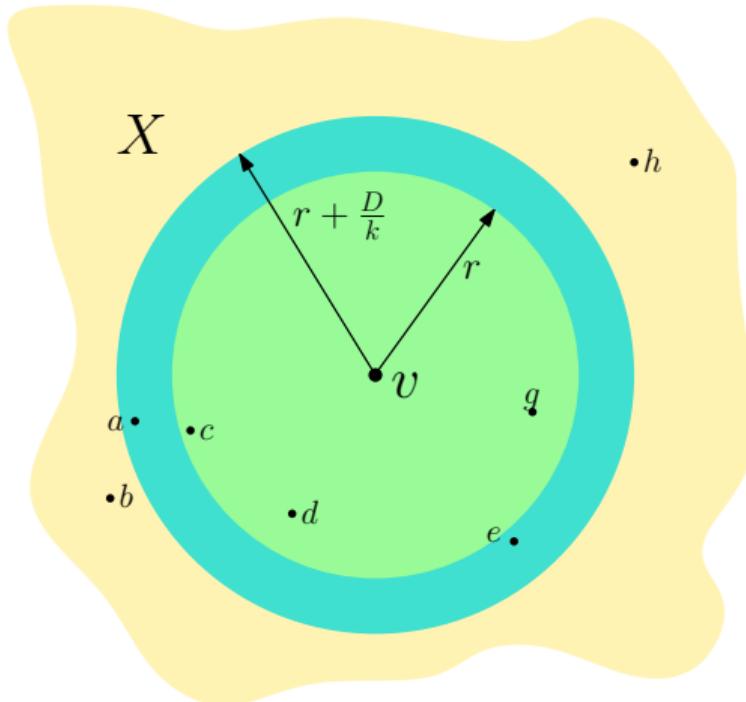
$$\ell(r_U) = \text{diam}(X) = D$$



Construction

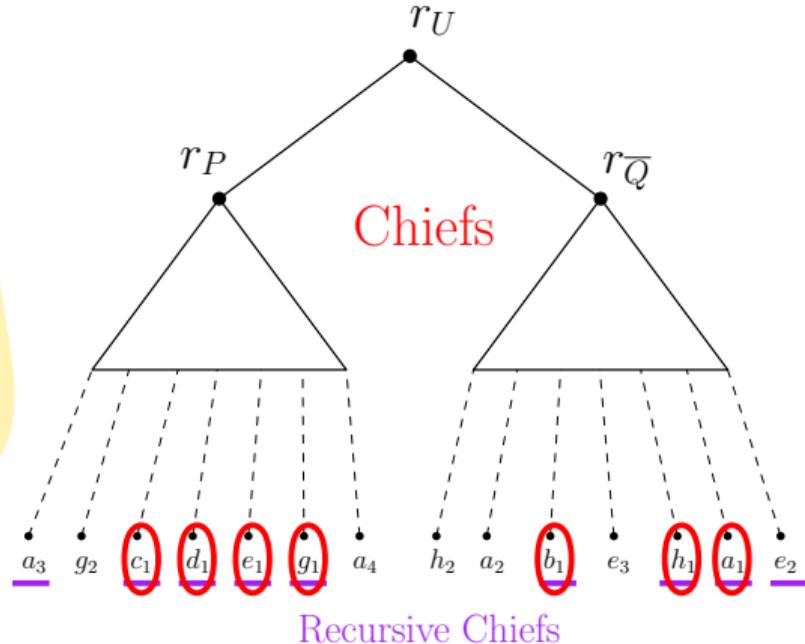
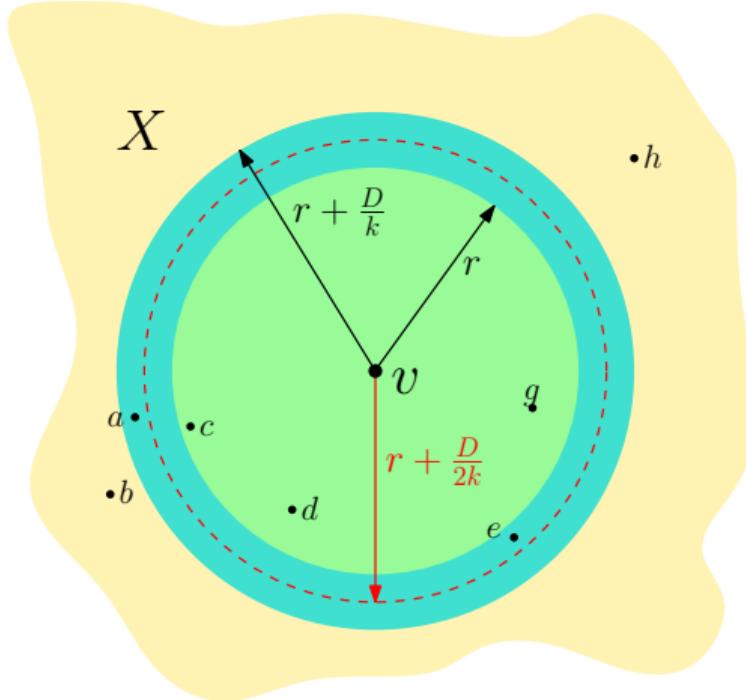


Construction

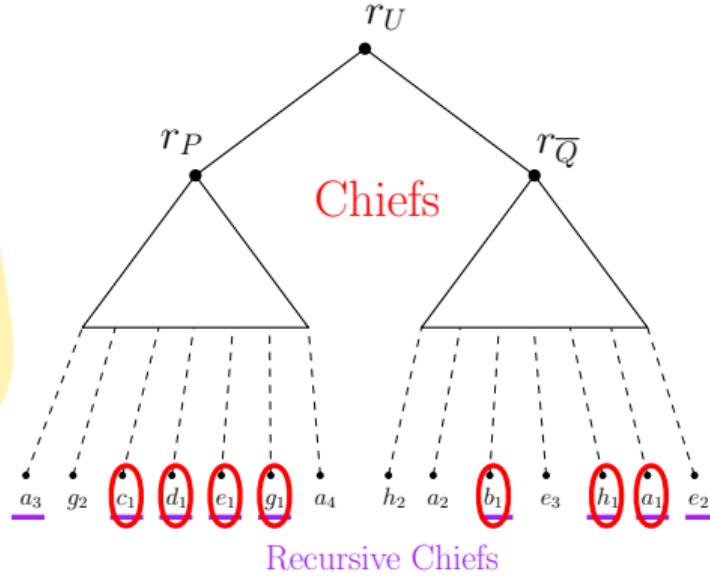
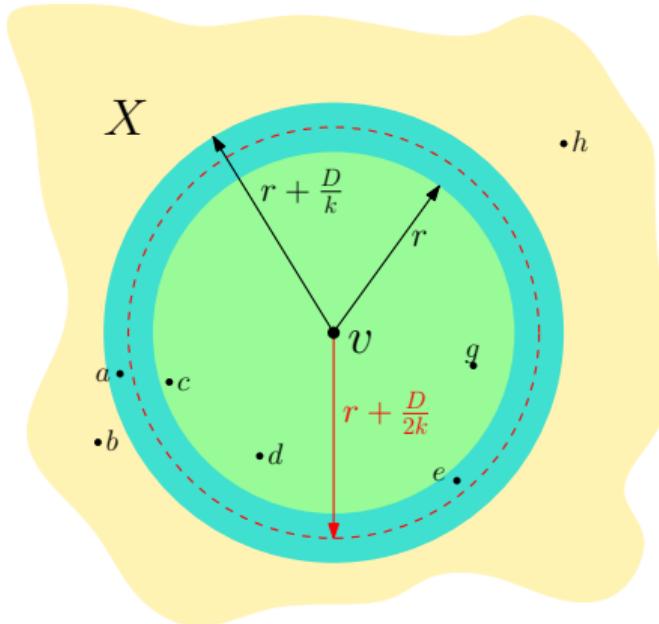


Recursive Chiefs

Construction

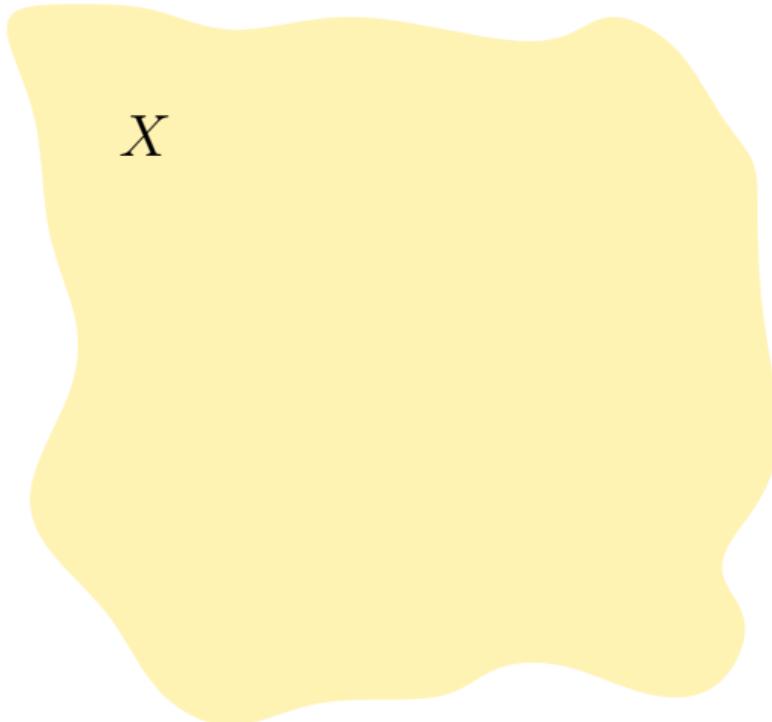


Construction - distortion bound



$$\min_{c' \in f(c)} d_U(c', \chi(a)) = D \leq 2k \cdot d_X(c, a) .$$

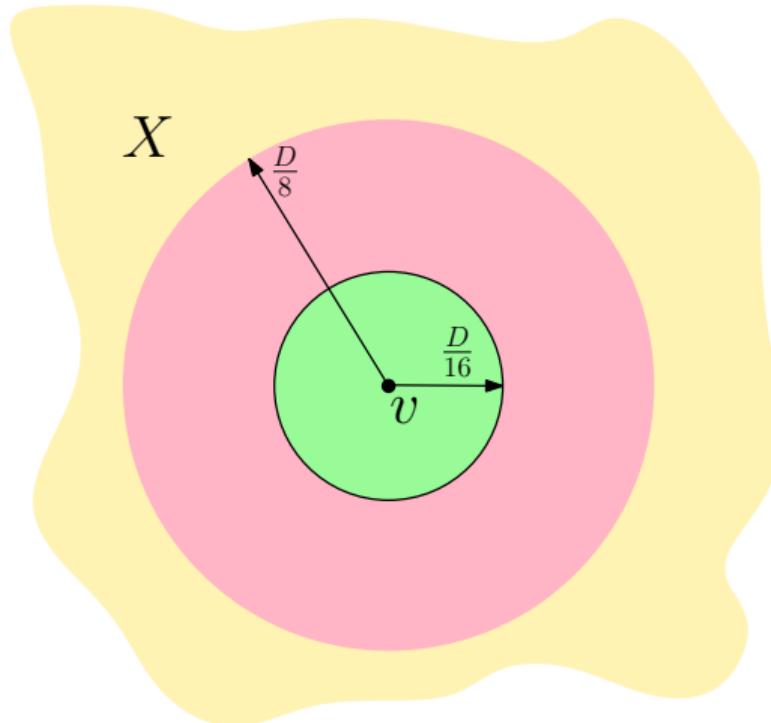
Construction - cardinality bound



X

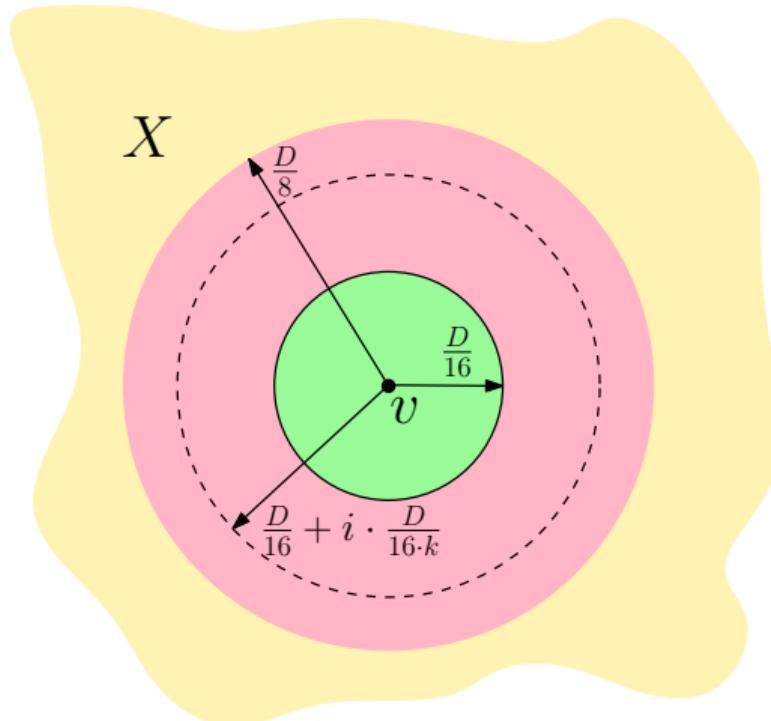
$$D = \text{diam}(X)$$

Construction - cardinality bound



$$D = \text{diam}(X)$$

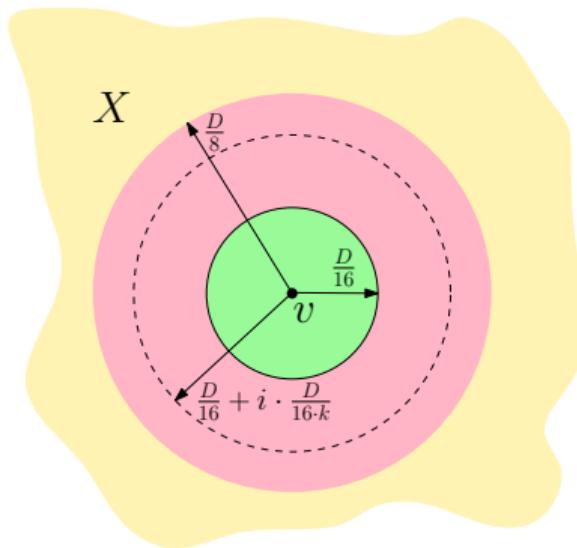
Construction - cardinality bound



$$D = \text{diam}(X)$$

$$A_i = B_X(v, \frac{D}{16} + i \cdot \frac{D}{16 \cdot k})$$
$$i \in \{0, \dots, k-1\}$$

Construction - cardinality bound

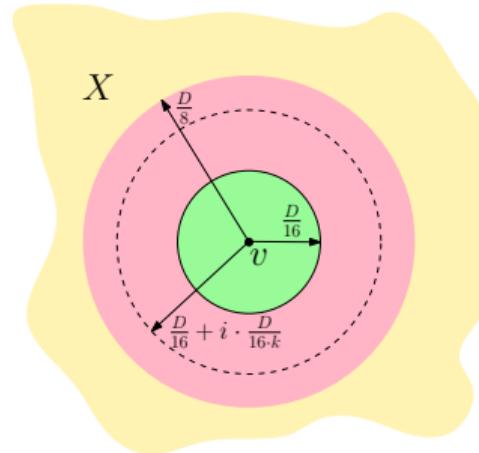


$$D = \text{diam}(X)$$

$$A_i = B_X(v, \frac{D}{16} + i \cdot \frac{D}{16 \cdot k})$$
$$i \in \{0, \dots, k-1\}$$

There is some i s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{|A_k|}{|A_0|}\right)^{\frac{1}{k}}$.

Construction - cardinality bound



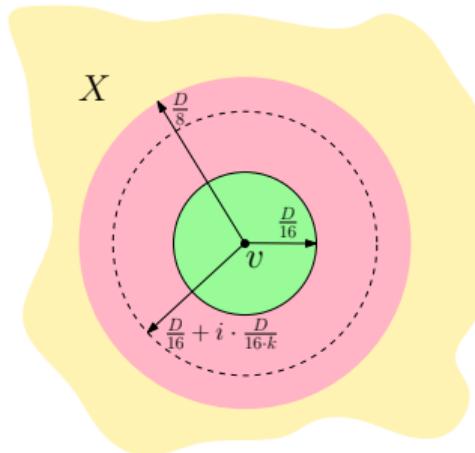
$$D = \text{diam}(X)$$

$$A_i = B_X(v, \frac{D}{16} + i \cdot \frac{D}{16 \cdot k})$$
$$i \in \{0, \dots, k-1\}$$

There is some i s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{|A_k|}{|A_0|}\right)^{\frac{1}{k}}$. Otherwise

$$|A_k| > |A_{k-1}| \cdot \left(\frac{|A_k|}{|A_0|}\right)^{\frac{1}{k}} > |A_{k-2}| \cdot \left(\frac{|A_k|}{|A_0|}\right)^{\frac{2}{k}} > \dots > |A_0| \cdot \left(\frac{|A_k|}{|A_0|}\right)^{\frac{k}{k}} = |A_k|$$

Construction - cardinality bound



$$D = \text{diam}(X)$$

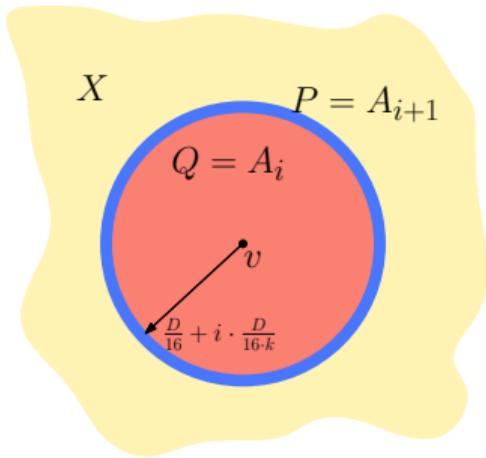
$$A_i = B_X(v, \frac{D}{16} + i \cdot \frac{D}{16 \cdot k})$$

$$i \in \{0, \dots, k-1\}$$

There is some i s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{|A_k|}{|A_0|}\right)^{\frac{1}{k}}$.

Claim

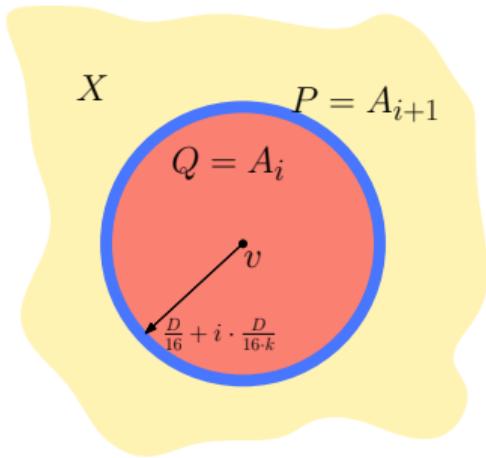
There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.



Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.*

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

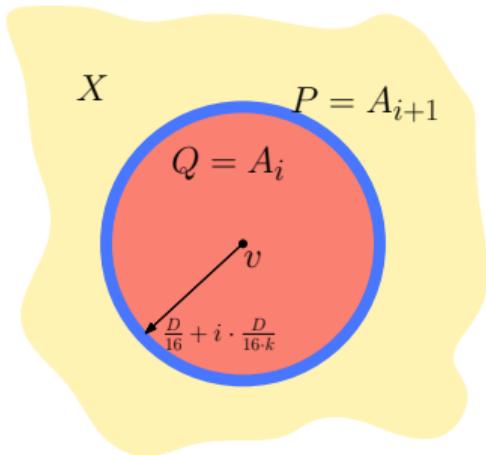


Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.*

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$.



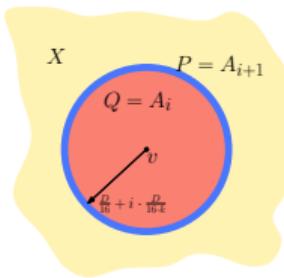
Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.*

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$.

Recurse on P and \overline{Q} .



Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})} \right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|} \right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \overline{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

Here $|f(X)| = \sum_{x \in X} |f(x)|$.

Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \overline{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

$$|f(X)| = |f(P)| + |f(\overline{Q})|$$

Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \overline{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

$$\begin{aligned}|f(X)| &= |f(P)| + |f(\overline{Q})| \\&\leq |P| \cdot \mu^*(P)^{\frac{1}{k}} + |\overline{Q}| \cdot \mu^*(\overline{Q})^{\frac{1}{k}}\end{aligned}$$

Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\bar{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \bar{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

$$\begin{aligned}|f(X)| &= |f(P)| + |f(\bar{Q})| \\&\leq |P| \cdot \mu^*(P)^{\frac{1}{k}} + |\bar{Q}| \cdot \mu^*(\bar{Q})^{\frac{1}{k}} \\&\leq |Q| \cdot \mu^*(X)^{\frac{1}{k}} + |\bar{Q}| \cdot \mu^*(X)^{\frac{1}{k}}\end{aligned}$$

Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \overline{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

$$\begin{aligned}|f(X)| &= |f(P)| + |f(\overline{Q})| \\&\leq |P| \cdot \mu^*(P)^{\frac{1}{k}} + |\overline{Q}| \cdot \mu^*(\overline{Q})^{\frac{1}{k}} \\&\leq |Q| \cdot \mu^*(X)^{\frac{1}{k}} + |\overline{Q}| \cdot \mu^*(X)^{\frac{1}{k}} \\&= |X| \cdot \mu^*(X)^{\frac{1}{k}}.\end{aligned}$$

Claim

There is $v \in X$, and i , s.t. $\frac{|A_{i+1}|}{|A_i|} \leq \left(\frac{\mu^*(X)}{\mu^*(A_{i+1})}\right)^{1/k} = \left(\frac{\max_{x \in X} |B_X(x, \frac{\text{diam}(X)}{4})|}{\max_{x \in A_{i+1}} |B_{A_{i+1}}(x, \frac{\text{diam}(A_{i+1})}{4})|}\right)^{1/k}$.

Set $P = A_{i+1}$, $Q = A_i$, and $\overline{Q} = X \setminus A_i$.

By the claim: $|P| \cdot \mu^*(P)^{\frac{1}{k}} \leq |Q| \cdot \mu^*(X)^{\frac{1}{k}}$. Recurse on P and \overline{Q} .

We argue by induction: $|f(X)| \leq |X| \cdot \mu^*(X)^{\frac{1}{k}} \leq |X|^{1+\frac{1}{k}}$.

$$\begin{aligned}|f(X)| &= |f(P)| + |f(\overline{Q})| \\&\leq |P| \cdot \mu^*(P)^{\frac{1}{k}} + |\overline{Q}| \cdot \mu^*(\overline{Q})^{\frac{1}{k}} \\&\leq |Q| \cdot \mu^*(X)^{\frac{1}{k}} + |\overline{Q}| \cdot \mu^*(X)^{\frac{1}{k}} \\&= |X| \cdot \mu^*(X)^{\frac{1}{k}}.\end{aligned}$$

