The Geometry of Graphs and its Algorithmic Applications

Katia Papakonstantinopoulou

Department of Informatics and Telecommunications
University of Athens

2007/06/25

The geometry of graphs

- Study of graphs from a geometric perspective.
- Geometric models: topological, adjacency, metric.
- General idea in metric models:
  Reduce problems from 'hard' to 'easy' metric spaces.
  How? Using *embeddings*.

Metric and Normed spaces

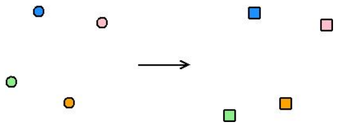
- Metric space: a set of points and a distance function.
- Normed space: a set of points and a *norm* (distance$(x, y) = \|x - y\|$).
- Norm of $x \in \mathbb{R}^d$ is $\|x\|$ where
  - $\|x\| \geq 0$
  - $\|\lambda x\| = |\lambda| \; \|x\|, \quad \forall \lambda \in \mathbb{R}$
  - $\|x + y\| \leq \|x\| + \|y\|, \quad \forall y \in \mathbb{R}^d$.
- l_p norm: $\|x\|_p = \|(x_1, \ldots, x_d)\|_p = \sqrt[p]{\sum_{i=1}^{d} |x_i|^p}$
- l_p^d space: $\mathbb{R}^d$ equipped with l_p norm

Embeddings

Embedding: a mapping $f : P_A \longrightarrow P_B$

- P_A: points in the (original) metric space, with distance function $D(\cdot, \cdot)$
- P_B: points in the (host) normed space l_s^d
- $\forall\ p, q \in P_A$, and a certain parameter c (*distortion*):

$$\frac{1}{c} \cdot D(p,q) \leq \| f(p) - f(q) \|_s \leq D(p,q)$$



Why embeddings?

- Reductions from ‘hard’ to ‘easy’ spaces $\rightsquigarrow$ ‘Good’ embeddings minimize
  - the dimension of the host space,
  - the distortion (isometric, near-isometric).
- Widely applicable
- Many tools available (combinatorics, functional analysis)

A Toy Example

Example (Diameter in l_1^d)

Given a set P of n points in l_1^d
find the diameter of P ($\max_{p,q\in P} \| p - q \|_1$)

- Solution in $O(dn^2)$ time.
- Can we reduce the dependence on n?
- Yes. We can:
  - embed l_1^d into $l_\infty^{d'}$, where $d' = 2^d$
  - solve the problem in $l_\infty^{d'}$

  in $O(dd'n)$ time.

A Toy Example (cont.)

Solution via embedding:

- $f(p) = < s_0 \cdot p, s_1 \cdot p, \ldots, s_{2^d-1} \cdot p >$,
  where s_i is the ith vector in $\{-1, 1\}^d$.
  Then $\| f(p) - f(q) \|_\infty = \| f(p-q) \|_\infty = \max_s |s \cdot (p-q)| = \max_s |\sum_{i=1}^d s_i \cdot (p-q)_i| = |\sum_{i=1}^d \mathrm{sgn}((p-q)_i)(p-q)_i| = \sum_{i=1}^d |(p-q)_i| = \|p-q\|_1 \Rightarrow c = 1$.
- $\max_{p,q \in P} \| f(p) - f(q) \|_\infty = \max_{p,q \in P} \max_{i=1,\ldots,d'} |f(p)_i - f(q)_i| = \max_{i=1,\ldots,d'} (\max_{p,q \in P} |f(p)_i - f(q)_i|) = \max_{i=1,\ldots,d'} (\max_{p \in P} f(p)_i - \max_{q \in P} f(q)_i)$.

Diameter found in: $O(d'n)$ in $l_\infty^{d'} \rightsquigarrow O(dd'n)$ in l_1^d.

Outline of the remainder of the talk

1. Low-distortion low-dimensional embeddings
2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions
3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Notation

$c_2(X)$ is the least distortion with which a metric space (X, d) may be embedded in l_2 (of any dimension).

- $(X, d) \xhookrightarrow{\geq c_2(X)} l_2$,
- $c_2(X) = O(\log n)$ for an n-point metric space.

Outline

1. Low-distortion low-dimensional embeddings

2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

‘Good’ embeddings

Theorem (Johnson-Lindenstrauss, 1984)

For any n-point set in a Euclidean space (X, d)*, and any* $\varepsilon > 0$,

$$(X, d) \xhookrightarrow{\leq 1+\varepsilon} l_2^{O(\frac{\log n}{\varepsilon^2})},$$

in random polynomial time.

‘Good’ embeddings - Thm 1

Theorem (Bourgain, 1985 - Linial, London, Rabinovich, 1994)

For any n-point metric space (X, d)*,*

$$(X, d) \xhookrightarrow{O(\log n)} l_2^{O(\log n)}.$$

Proof

It is an immediate corollary of theorems:

- $(X, d) \xrightarrow{O(c_2(X))} l_p^{O(\log n)}$, for any $1 \leq p \leq 2$,
- $(X, d) \xrightarrow{O(\log n)} l_p^{O(\log^2 n)}$, for any $p > 2$

in random polynomial time (will be proved later), and

- the J-L theorem.

‘Good’ embeddings - Thm 2

Theorem (Embedding in l_2)

For any metric space (X, d),

- $(X, d) \xhookrightarrow{<c_2(X)+\varepsilon} l_2$, *for every* $\varepsilon > 0$, *in polynomial time,*
- $(X, d) \xhookrightarrow{O(c_2(X))} l_2^{O(\log n)}$, *in random polynomial time.*

Proof

- Consider a matrix M. Let its rows be the images of the points of X under a distortion-c embedding in some Euclidean space.
- Let $A = MM^t$. A is positive semidefinite, and

$$\frac{1}{c^2}\, d_{i,j}^2 \leq a_{i,i} + a_{j,j} - 2a_{i,j} \leq d_{i,j}^2, \quad \forall i \neq j.$$

- Now, the ellipsoid algorithm gives us an ε-approximation of c in polynomial time.
- The dimension is reduced to $O(\log n)$ by applying J-L theorem.

'Good' embeddings - Thm 3

Theorem (Embedding in $l_p, 1 \leq p \leq 2$, rp-time)

For any metric space (X, d) and for any $1 \leq p \leq 2$,

$$(X, d) \xhookrightarrow{O(c_2(X))} l_p^{O(\log n)},$$

in random polynomial time.

Proof

- $(X, d) \xhookrightarrow{O(c_2(X))} l_2^{O(\log n)}$ in random polynomial time (previous theorem).
- For any m, and any $1 \leq p \leq 2$,

$$l_2^m \xhookrightarrow{O(1)} l_p^{2m}$$

  in random polynomial time (known).
  - (In fact, it is enough to map l_2^m isometrically into a random m-dimensional subspace of l_p^{2m}.)
- Any n points in l_2 space $\xhookrightarrow{1} l_1^{O(n^2)}$.
  In particular, for any finite metric space X, $c_1(X) \leq c_2(X)$.

'Good' embeddings - Thm 4

Theorem (Embedding in $l_p, 1 \leq p \leq 2$, p-time)

For any metric space (X, d) and for any $1 \leq p \leq 2$,

$$(X, d) \xhookrightarrow{O(c_2(X))} l_p^{O(n^2)},$$

in polynomial time.

Sketch of the Proof

- We find an optimal embedding into Euclidean space (of dimension $\leq n$):

$$(X, d) \xhookrightarrow{<c_2(X)+\varepsilon} l_2, \quad \forall \varepsilon > 0, \quad \text{in polynomial time}$$

  (proved earlier).
- We can embed l_2^m to $l_p^{O(m^2)}$, for any $1 \leq p \leq 2$, in polynomial time (see details in paper LLR95).
- Thereby, we map the n-dimensional Euclidean space we found to $l_p^{O(n^2)}$, for any $1 \leq p \leq 2$:

$$l_2^n \xhookrightarrow{O(1)} l_p^{O(n^2)}$$

  (details in paper LLR95).

‘Good’ embeddings - Thm 5

Theorem (Embedding in l_p, $p > 2$, rp-time)

For any metric space (X, d) and for any $p > 2$,

$$(X, d) \xhookrightarrow{O(\log n)} l_p^{O(\log^2 n)},$$

in random polynomial time.

Sketch of the Proof

- For each cardinality $k < n$ which is a power of 2, randomly pick $O(\log n)$ sets $A \subseteq V(G)$ of cardinality k.
- Find an embedding: $(X, d) \hookrightarrow l_1^{\log^2 n}$.
  - Map every vertex x to the vector $(d(x, A))$ (where $d(x, A) = \min\{d(x, y) | y \in A\}$), with one coordinate for each A selected.
- This mapping has almost surely an $O(\log n)$ distortion (see details in paper LLR95).
- For every $p \geq 1$, a proper normalization of this embedding satisfies the same statement with respect to the l_p norm (details in paper LLR95).

‘Good’ embeddings - Thm 6

Theorem (Embedding of expanders into l_p)

For any n-vertex constant-degree expander (Y, d) and for any $1 \leq p \leq 2$,

$$(Y, d) \xhookrightarrow{\Omega(\log n)} l_p$$

(of any dimension).

Sketch of the Proof

- The max-flow min-cut gap is $\Omega(\log n)$ for the all-pairs, unit-demand flow problem on a constant-degree expander, where all capacities are one (known).
- This gap implies that every embedding of the expander's metric in l_1 (of any dimension) has distortion $\Omega(\log n)$ (see details in paper LLR95).
- This conclusion holds also for embeddings into l_p for $1 \leq p \leq 2$, because in this range, every finite dimensional l_p space can be embedded in l_1 with a constant distortion (known).

Outline

1. Low-distortion low-dimensional embeddings

2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Outline

1. Low-distortion low-dimensional embeddings

2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Multicommodity flow problem

Multicommodity flow

- Flow network $G(V, E)$, edge $e \in E$ has capacity $c_e \geq 0$.
- k commodities $K_1, K_2, \ldots, K_k$
- $K_i = (s_i, t_i, d_i)$: s_i and t_i is the source and sink of commodity i, $d_i \geq 0$ is the demand.
- The flow of commodity i along edge (u, v) is $f_i(u, v)$.
- Find *maxflow* – the largest λ for which it is possible to simultaneously flow λd_i between s_i and $t_i \forall i$, satisfying:
  - Capacity constraints: $\sum_{i=1}^{k} f_i(u, v) \leq c(u, v)$
  - Flow conservation: $\sum_{w \in V} f_i(u, w) = 0 \quad \text{when} \quad u \neq s_i, t_i$
  - Demand satisfaction: $\sum_{w \in V} f_i(s_i, w) = \sum_{w \in V} f_i(w, t_i) = d_i$.

The problem is NP-complete for integer flows, even for only two commodities.

Applications to multicommodity flow - Results

- The gap between the max-flow and the min-cut in a multicommodity flow problem is upper bounded by the *least distortion* with which a particular metric (associated with the network) can be embedded in l_1.
- This metric is defined via the Linear Programming dual of a program for the max-flow.
- This is the basis for a unified and simple proof to a number of old and new results on multicommodity flows.

Outline

1. Low-distortion low-dimensional embeddings

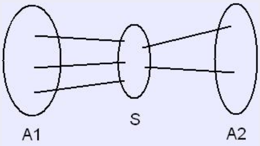
2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Separator problem

Separator

- Undirected graph $G = (V, E)$.
- A *separator* $S \subseteq V$ of G partitions V into two parts $A_1 \subseteq V$ and $A_2 \subseteq V$
  such that $A_1 + S + A_2 = V$, and no edge joins vertices in A_1 and A_2
- (A_1, S, A_2) is called a *separation* of G.
- Goal: minimize $|S|$, maintaining an appropriate balance between A_1 and A_2.



Applications to separators - Results

Low-dimensional graphs have small separators:

A d-dimensional graph G has a set S of $O(dn^{1-\frac{1}{d}})$ vertices which separates the graph,
so that no component of $G \backslash S$ has more than $(1 - \frac{1}{d+1} + o(1))n$ vertices.

Outline

1. Low-distortion low-dimensional embeddings
2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions
3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Graph decomposition problem

Graph decomposition

- A decomposition of a graph $G = (V, E)$ is a partition of its vertex set into subsets (blocks).
- The *diameter* of the decomposition is the least δ: any two vertices belonging to the same connected component of a block are at distance $\leq \delta$ in the graph.
- Usually: decompose a weighted graph into a specified number of subgraphs such that these subgraphs have balanced sums of vertex weights and minimal sums of edge weights.

Applications to graph decompositions - Results

- The vertices of any d-dimensional graph can be $(d+1)$-colored so that each monochromatic connected component has diameter $\leq 2d^2$.
- They can also be covered by 'patches' so that each r-sphere (r – any positive integer) in the graph is contained in at least one patch, while no vertex is covered more than $d+l$ times. The diameter of each such patch is $\leq (6d+2)dr$.
- Moreover, the patches may be $(d+1)$-colored so that equally colored patches are at distance ≥ 2.
- That is, there exist low-diameter decompositions with parameters depending on the dimension alone.

Outline

1. Low-distortion low-dimensional embeddings

2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Outline

1. Low-distortion low-dimensional embeddings

2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions

3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Applications to multicommodity flow - Results

Near-tight cuts for multicommodity flow problems can be found in deterministic polynomial time.

Outline

1. Low-distortion low-dimensional embeddings
2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions
3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Applications to separators - Results

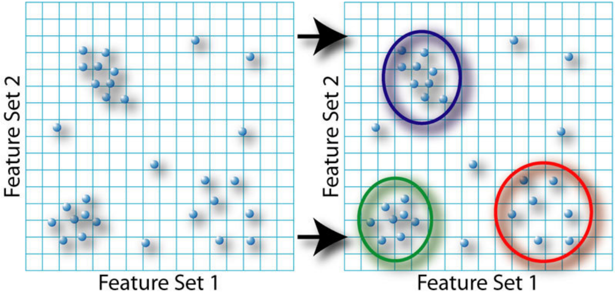
Given an isometric ($c = 1$) embedding of G in d dimensions, a balanced separator of size $O(dn^{1-\frac{1}{d}})$ can be found in random polynomial time.

Outline

1. Low-distortion low-dimensional embeddings
2. Structural Consequences
   - Structural consequences to multicommodity flow problems
   - Structural consequences to separators
   - Structural consequences to graph decompositions
3. Algorithmic Consequences
   - Algorithmic consequences to multicommodity flow problems
   - Algorithmic consequences to separators
   - Algorithmic consequences to clustering

Clustering problem

Clustering: the partitioning of a set of points into subsets (*clusters*), so that the points in the same cluster tend to be much closer than points in distinct clusters.



- Key problem in pattern-recognition.
- *Easy* in low-dimensional Euclidean space.
- *Very difficult* in high-dimensional/non- Euclidean spaces.

Applications to clustering - Results

- Low-dimensional, small-distortion representation of statistical data offers a new approach to clustering.
- Tested in search for patterns among protein sequences.
- Metric space: all known proteins.
- Thm 2 $\rightsquigarrow$ the distance between any 2 points in the space can be evaluated in a single time unit!

Open Problems

- Many. . .
- Quoting J. Matoušek: *Amazing* progress in the area during the last 5 years.

Conjecture (stated by A. Gupta et al. in FOCS '99)

Excluded-minor graph families can be embedded into l_1 with distortion dependent only upon the set of excluded minors.

References

N. Linial, E. London, and Y. Rabinovich.
The geometry of graphs and some of its algorithmic applications.
FOCS 1994, Combinatorica 1995.

P. Indyk.
Algorithmic applications of low-distortion geometric embeddings.
FOCS 2001.

Elias Koutsoupias.
Personal Communication.

A. Gupta, I. Newman, Y. Rabinovich, A. Sinclair.
Cuts, trees and l_1 embeddings.
FOCS 1999.

Thank you!