Spanners

Social and Technological Networks

Rik Sarkar

University of Edinburgh, 2018.

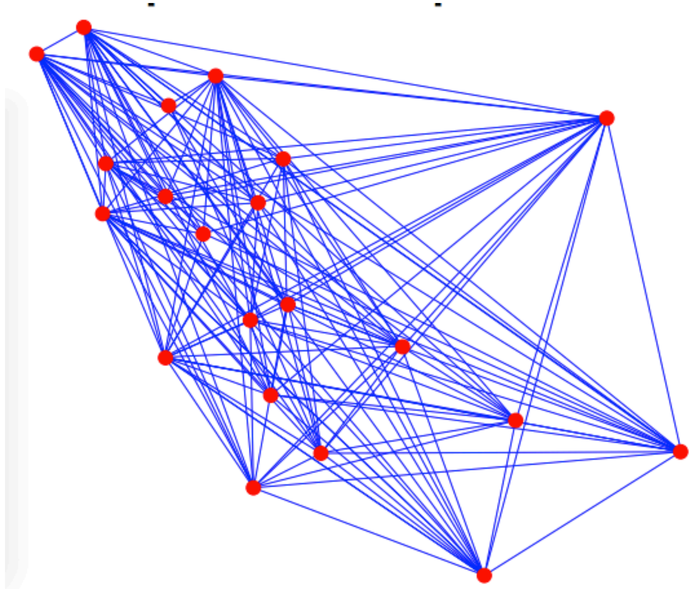
Distances in graphs

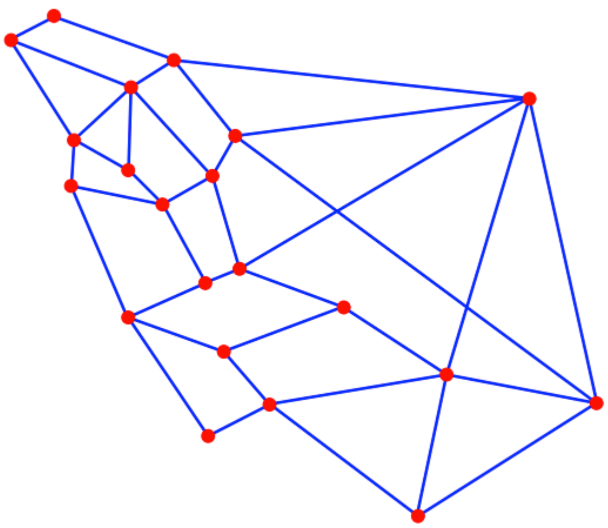
- Suppose we are interested in finding distances, shortest paths etc in a weighted graph G

- The problem: A graph can have n^2 edges.
- Any computation is expensive
- Storage is expensive

Idea: use a “similar” graph with fewer edges

- A spanning graph H of a connected graph G:
  - H is connected and has the same set of vertices
- Construct an H with fewer edges





Stretch

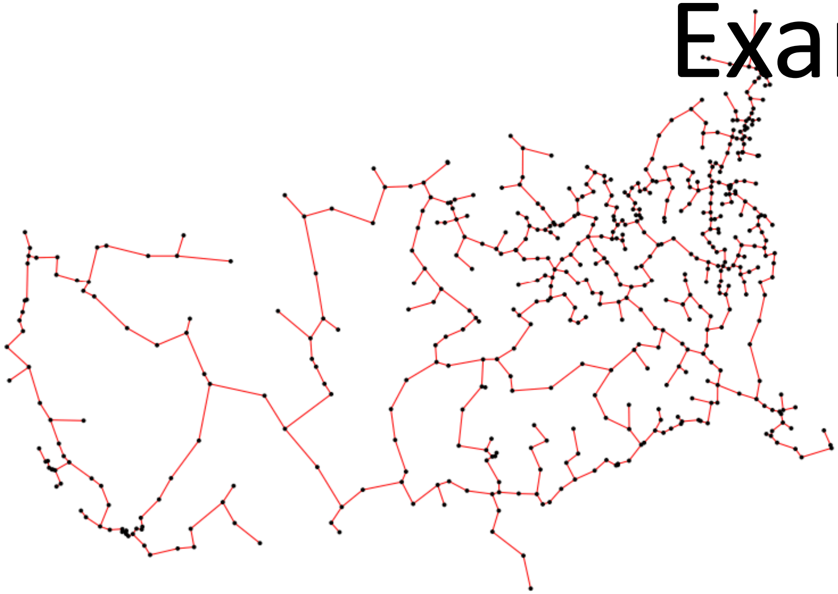
- Suppose d_G is the shortest path distance in G
- Suppose $d_H(u, v) = s \cdot d_G(u, v)$
- S is called the stretch of distance between u,v

- The idea is to have compressed network H with small stretch and few edges

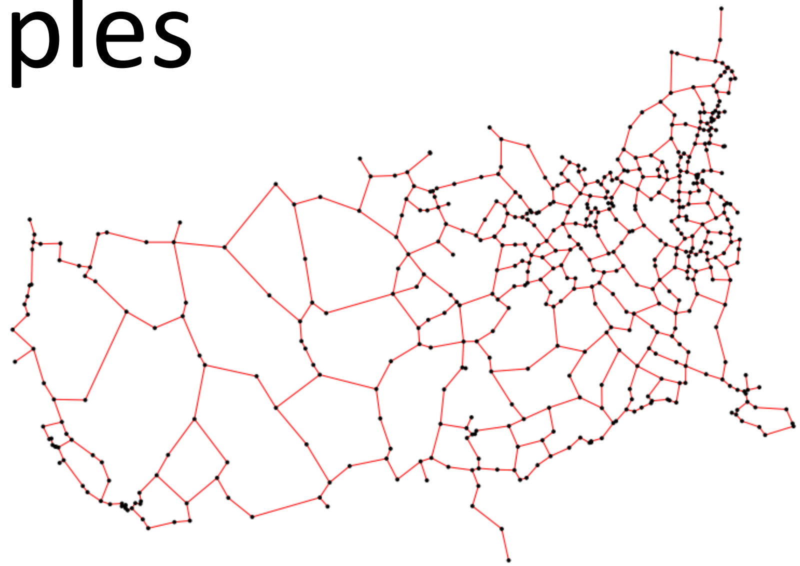
Spanners

- Suppose d_G is the shortest path distance in G
- H is a t-spanner of G if:
- $d_H(u, v) \leq t \cdot d_G(u, v)$
  - A multiplicative spanner
  - The stretch of the spanner is t
- More generally, H is a (α, β)-spanner of G if:
- $d_H(u, v) \leq \alpha \cdot d_G(u, v) + \beta$

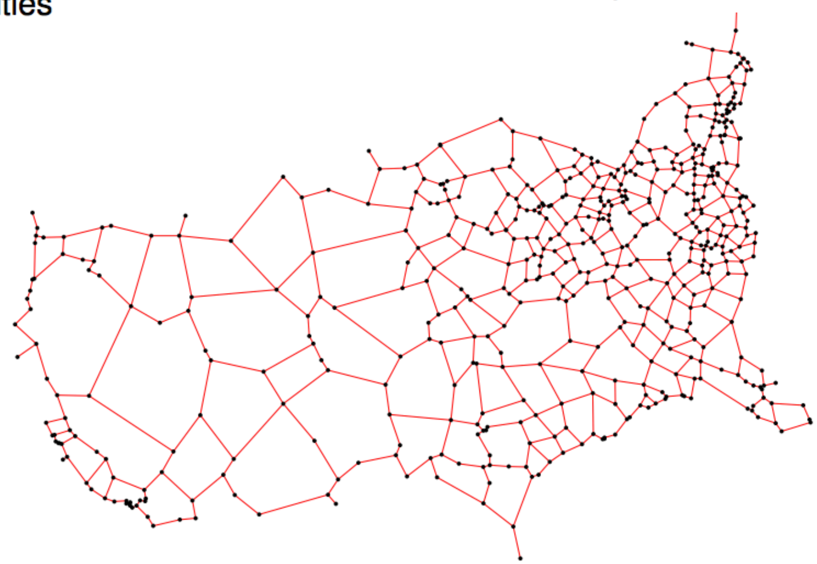
Examples



10-spanner for 532 US-cities



3-spanner for 532 US-cities



2-spanner for 532 US-cities

Images from:
http://cs.yazd.ac.ir/farshi/Teaching/Spanner-3932/Slides/GSN-Course.pdf

Examples

- Compress road maps and still find good paths
- Compress computer/communication networks and get smaller routing tables
- “Bridges” are part of spanner

- Small set of distances among moving objects.
  - To detect possible collisions
  - A “short edge” must always be in the spanner
  - Thus, we need to only check edges in the spanner

Simple greedy algorithm

- Given graph G=(V, E) and stretch t
- We want to construct H=(V, E')
- Sort all edges in E by length
- Proceed from shortest to longest edge
  - Take edge e=(u,v)
  - If $d_H(u, v) > t \cdot d_G(u, v)$, add e to E'
- Output H

Geometric spanner

- Suppose we have only a set of points in the plane, and no graph (e.g. position of robots)

- Then the same algorithm applies
  - With G as the complete graph with, planar distance as the edge length

H is a spanner

- Claim: $d_H(u, v) \leq t \cdot d_G(u, v)$

- If (u,v) is an edge in G, then this holds by construction.
- If not, suppose P is the path between them of length $d_G(u, v)$
- For each edge $(a, b) \in P$, the claim holds.
  - Therefore. It holds for the sum of their lengths.

Number of edges

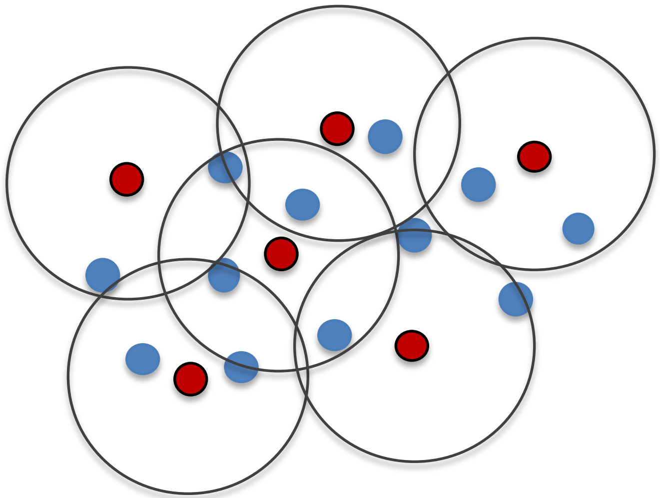
- Theorem:
- The greedily constructed t-spanner has
- $n^{1+\frac{2}{t+1}}$ edges

- Proof: Ommitted

Deformable spanners

- Suppose we have n points in $\mathbb{R}^d$
- We want to compute a good spanner
  - With stretch $(1 + \epsilon)$
  - Number of edges n/ϵ^d

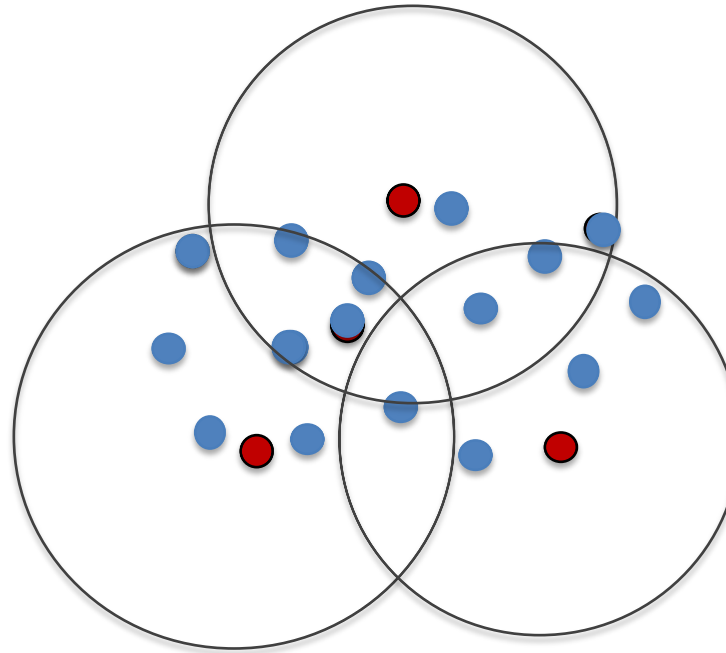
Discrete centers

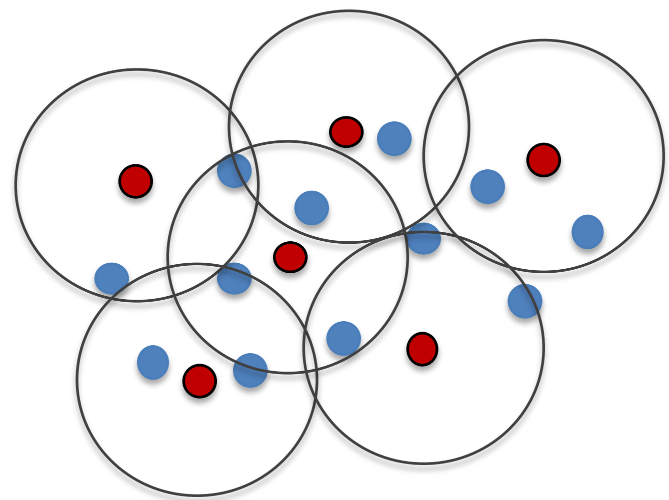


- Give a radius r
- A set S of discrete centers is a subset of V
- Such that:
  - Any point of V is within distance r of some $s \in S$.
  - Any two points $s_1, s_2 \in S$ are at least r apart
- That is, a set of balls with far apart centers, that covers all points

Discrete center hierarchy

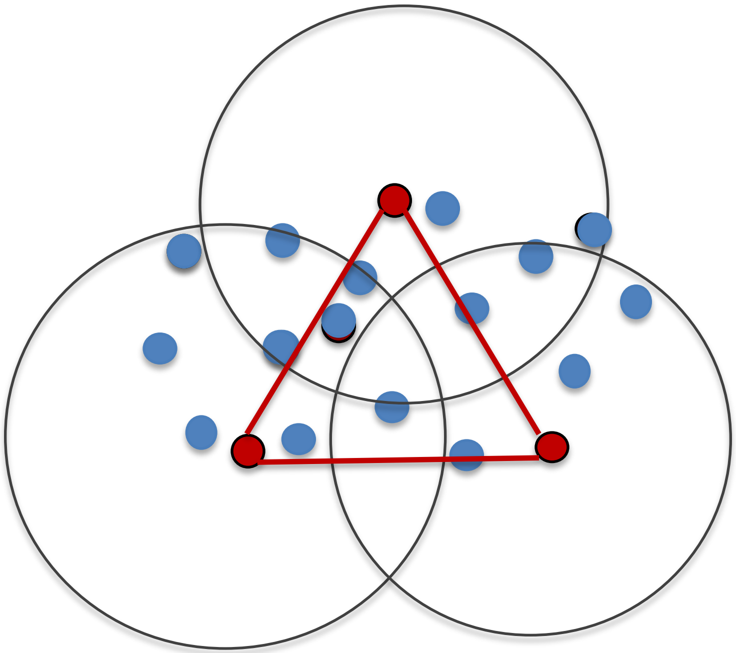
- Compute a set S_i of discrete centers
  - For each $r = 2^i$
  - Such that $S_i \subseteq S_{i-1}$
- Start from smallest distance between a pair of points
  - At this lowest level each node is a center
- Highest level is diameter of the set

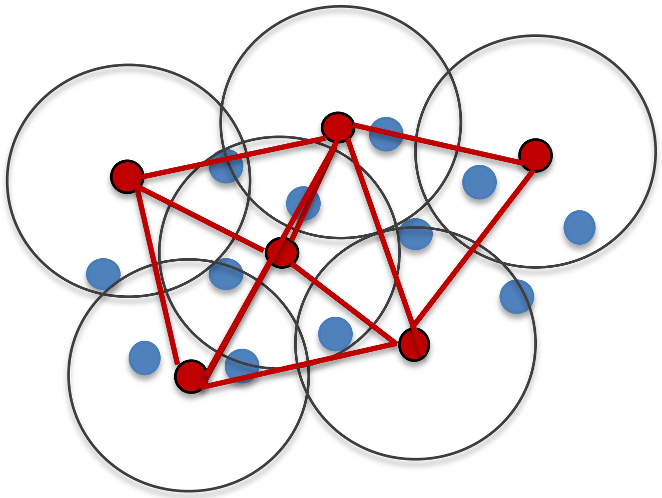




Spanner

- Suppose $s, t \in S_i$ are centers
- Add edge (s, t) if $|st| \leq c \cdot 2^i$
  - For $c = 4 + 16/\epsilon$
- Take the union of edges created at all levels
- To get a graph G





Theorems

- G is a $(1+\epsilon)$ spanner
  - That is, for any two points p and q, there is a path in G of length at most $(1+\epsilon)|pq|$
- G has n/ϵ^d edges
- If the ratio of diameter to smallest distance is α, then each node has $O((\log \alpha)/\epsilon^d)$ edges

Useful properties

- Applies to metrics of bounded doubling dimension
- Relatively small number of edges
- Each node has a small number of edges
  - Efficient in checking for collisions and near neighbors
  - Each robot has to keep small amount of information
- Can be updated easily as nodes move, join, leave
  - Hence the name "deformable"
- Multi-scale simplification of the network
  - Gives a summary of the network at different scales
  - An important topic in current algorithms and ML
  - Computation for large datasets need simplified data

- There are other more complex algorithms
- Areas of research:
  - Specialized graphs
  - Fault–tolerant spanners
  - Dynamic spanners – for changing graphs
  - …

Course

- No class on Friday 23rd Nov
- No office hours Thursday 22nd Nov

- Final class on Tuesday 27th: review
  - We will discuss the course in general
  - What to expect in exam