# Topics in Algorithms 2007

Ramesh Hariharan

# Tree Embeddings

# Solving Graph Metric Problems

- How do we make the problem easier?
- Convert into a tree!!

# Projections

- Can vertices in a given weighted graph G be mapped to vertices in any edge-weighted tree H so that all distances (i.e., shortest paths) only increase, but not by too much?
- Distortion: max stretch over all edges in G

# H vs G

- Two cases
  - $|G|=|H|$
  - $|G|<|H|$
- Two vertices in G cannot map to the same vertex in H

# A Simple Case

- G=unweighted cycle of length n and |H|=|G|
- How much is the distortion?
- >=n-1?

# Proof of the Simple Case

- Embedding of G in H must cancel out (why??)
- Take each e in G, map to H and then map back to G: two cases
  - Result is e itself ➔ G maps to itself ➔ contradiction (why??)
  - Result is some other path in G between the endpoints of e ➔ distortion for e is n-1
    (why??)

# Generalization

- Embedding of unweighted G into H has distortion at least g-1 if X(H)<X(G) and |H|=|G| (g is girth of G, X() is E-V+1)

# Proof

- Take each e in G, map to H and then map back to G: two cases
  - Result is e itself ➔ each cycle in G maps to itself
  - Result is some other path in G between the endpoints of e ➔ distortion for e is g-1
    (why??)

# Proof

- If each cycle in G maps to itself in G->H ->G
  - X(G)  independent cycles in G map to at most X(H) independent cycles in H
  - X(H) independent cycles in H cannot map to more than X(H) fundamental cycles in G
  - Contradiction

# Further Generalization

- Embedding of unweighted G into H has distortion at least $g_k$-1 if X(H)<=X(G)-k and |H|=|G| ($g_k$ is the length of the kth smallest cycle in G, X() is E-V+1)

# Proof

- Take each e in G, map to H and then map back to G: two cases
  - Result is e itself ➔ each cycle in G maps to itself
  - Result is some other path in G between the endpoints of e
    - ➔ the cycle formed by the path + e is NOT amongst the k-1 shortest cycles ➔ distortion for e is at least $g_k$ -1 (why??)
    - ➔ the cycle formed by the path + e IS amongst the k-1 shortest cycles ➔ each cycle in G maps to itself plus a combination of the k-1 shortest cycles

# Proof

- If each cycle in G maps (in G->H ->G) either to itself or to itself plus a combination of the smallest k-1 cycles of G
  - After G->H ->G there are at least X(G)-k+1 independent cycles in G (Why?)
  - G->H has only X(H)= X(G)-k basis cycles
  - Contradiction!!

# Steiner Points

- Will $|H|>|G|$ help?

# Further Generalization

- Embedding of unweighted *G* into H has distortion at least $g_k/3-4/3$ if X(H)=X(G)-k and |H|>=|*G*| ($g_k$ is the length of the kth smallest cycle in *G*, X() is E-V+1)

# Problem

- Take each e in G, map to H and then map back to G: map back is not defined for vertices in H-G
  - Add extra degree 2 vertices to G (many choices, pick any one), i.e., artificially define map-back for H-G
  - Add degree two vertices to H so each edge in H has length at most 1 (technical)
  - E-V+1 is in G or in H as a result of the above unchanged

# Problem

- Distances between new vertices in G may not expand in H
  - Translate to distances in terms of original vertices in G

# Problem

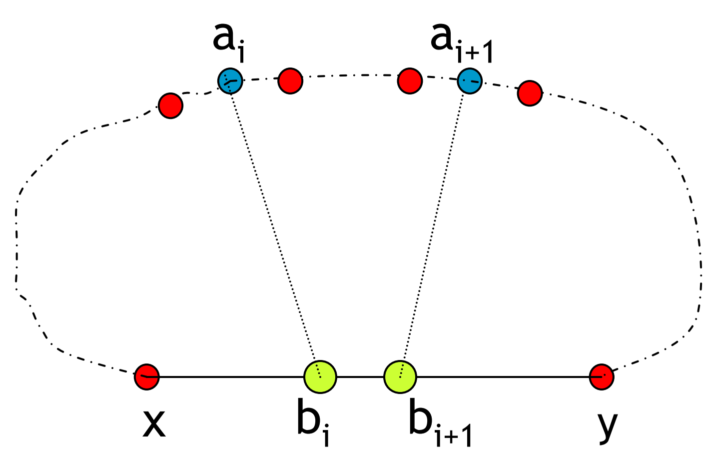
- Take each e=(x,y) in G, map to H and then map back to G:
  - We get a sequence on vertices x a1 a2 a3….y;  two cases
    - This cancels to just e itself ➔ each cycle in G maps to itself ➔ contradiction as before
    - This gives some other path in G between the endpoints of e
      - ➔ the cycle formed by the path + e IS amongst the k-1 shortest cycles ➔ each cycle in G maps to itself plus  a combination of the k-1 shortest cycles ➔ contradiction as before
      - ➔ the cycle formed by the path + e is NOT amongst the k-1 shortest cycles (so length >= $g_k$) ➔ cannot claim distortion at least $g_k$ -1 as before because the distance between ai and ai+1 in G need not be less than that in H (unless both vertices were in G originally)

# Proof

Consider G

Take images of $a_i$ and $a_{i+1}$ on e=xy (distances between images are proportional to those in H)

- Distance $b_i a_i$ is <= $g_k/3-4/3$ (Why?)
- Distance $b_i a_i$ is <= $g_k/3-4/3$ (Why?)
- Distance $a_i a_{i+1}$ is <= $g_k/3-1/3+2$, likewise for $b_i b_{i+1}$ (Why?)
- Total is <= $g_k-1$, i.e., the cycle $b_i a_i a_{i+1} b_{i+1} b_i$ IS amongst the smallest k-1 cycles
- Each cycle maps to itself plus a linear combination of small cycles ➔ contradiction as before



# Exercise

- Complete the proof

- Is it tight for a cycle G? Is there a graph H with |H|>|G| so that the embedding has distortion as low as |G|/3?

# Way Forward

- How about embedding not on to a single less complex graph but to a probability distribution of less complex graphs
- What is the maximum expected stretch?

# Cycle to Paths

- Take all n paths, each with prob 1/n
- How large is the expected stretch for any edge? <=2!!

# Application

- K-medians
- Find k centers is a graph so sum of all vertices of distance to nearest center is minimized
- Suppose you have an A approximate algorithm for trees
- And also an embedding on to a prob dist of trees so that the maximum expected stretch is B
- Then we can claim an expected approx factor AB on graphs (Why?)
- How do we convert expectation to a high probability bound?
- How about running time?

# Probabilistic Lower Bounds

- Why do the previous arguments fail for probability distributions over graphs in H?
- In each graph in H, some edge has a large distortion, but the prob weight on this graph is low.

# Two Player Games

- Fix G and some algorithm to embed G into a tree
- For all prob dist over trees. there exists an edge for which expected distortion >= c
- 2 player game
  - You choose prob dist P over trees
  - I choose an edge e
  - Value of this game is the expected distortion for e wrt P
  - I win if the value >=c otherwise you win

# Two Player Games

- Flip the game
- There exists an edge e, such that for all prob distributions over trees, the expected distortion for e >= c
- 2 player game
  - I choose an edge e
  - You choose prob dist P over trees
  - Value of this game is the expected distortion for e wrt P
  - I win if the value >=c otherwise you win

# Two Games

- Game A
  - You choose prob dist P over trees
  - I choose an edge e
  - Value of this game is the expected distortion for e wrt P
  - I win if the value >=c otherwise you win
- Game B
  - I choose an edge e
  - You choose prob dist P over trees
  - Value of this game is the expected distortion for e wrt P
  - I win if the value >=c otherwise you win

# Which game has a higher value?

- Game A >= Game B

# Two Games

- Game A
  - You choose prob dist P over trees
  - I choose an edge e
  - Value of this game is the expected distortion for e wrt P
  - I win if the value >=c otherwise you win
- Game B
  - I choose a probability distribution Q over edges
  - You choose a tree
  - Value of this game is the expected distortion for e wrt Q
  - I win if the value >=c otherwise you win

# Which game has a higher value?

- Game A >= Game B !!
- Von Neuman's Principle, Yao's Lemma

# Probabilistic Lower Bounds

- For all prob dist P over trees, there exists edge e with large expected distortion
- There exists prob dist Q over edges, such that for all trees, the expected distortion is large
- Pick Q, show that for all trees, the expected distortion is large
- Eg, Q is uniform, simply show that the average distortion when embedding into any tree is large

# Another Example: Randomized→Average Case

- Show that no randomized algorithm can have good performance for all inputs
- Show that for any deterministic algorithm, the average performance over all inputs is not good

# Probabilistic Lower Bounds

- Find a graph G such that the average distortion when embedding into any tree is large, say log n/3
- Show this for any graph which has
  - >=2n edges
  - Smallest cycle > K
  - At least half the edges must have distortion > K/3
  - Average distortion must be greater than K/3