CMPSCI 711: More Advanced Algorithms

Section 3-2: Grid Based Algorithms

Andrew McGregor

Last Compiled: April 29, 2012

Geometric Streams

- Consider a stream of points:

$$P = \langle p_1, \ldots, p_n \rangle$$

  Now suppose each $p_i \in \{1, \ldots, \Delta\}^d$. For this lecture assume $d = 2$.
- What properties of P can we compute in sub-linear space? Want to support both insertions and deletions.

Outline

Warm-Up: Diameter

Probabilistic Embedding

Diameter

- *Goal:* Given a sequence of points P from $\{1, \ldots, \Delta\}^2$, estimate

$$D = \max_{p,q \in P} \|p - q\|_1 \ .$$

- *Idea:*
  - Impose square grids $G_0, \ldots, G_k$ of side-lengths

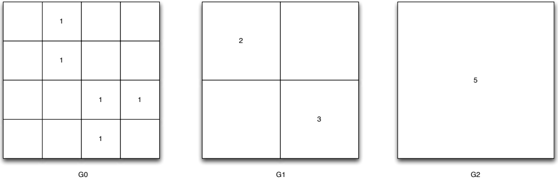
$$1, (1+\epsilon), (1+\epsilon)^2, \ldots, (1+\epsilon)^k = \Delta \ .$$

  - Consider the points at resolution of each grid.
  - Can use G_i to approximate distances up to $\pm 4(1+\epsilon)^i$ so knowing points in G_i gives a $(1+\epsilon)$ approx if $i \leq \log_{1+\epsilon}(\epsilon D/4)$.
  - For $i = \log_{1+\epsilon}(\epsilon D/4)$, there $\leq (D/(\epsilon D/4))^2 = 16/\epsilon^2$ non-empty cells
- *Algorithm:*
  - Use F_0 estimator to approx. $\#$ non-empty cells and let j be minimum value such that G_j contains $\leq k = 16/\epsilon^2$ non-empty cells.
  - Use k sparse-recovery algorithm to find all the non-empty cells in G_j.

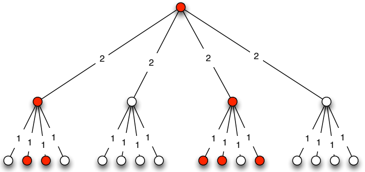
Outline

Tree Embedding

- Let $G_0, \ldots, G_k$ of side-lengths $1, 2, 2^2, \ldots, 2^k = \Delta$



- Consider a tree T where the leaves correspond to cells in G_0 and internal nodes correspond to cells in G_i for $i > 0$. There is an edge of weight 2^i between a cell c in G_i and a cell c' in G_{i+1} if $c \subset c'$.



- By construction $d_T(p, q) = \sum_{i \geq 0} 2^{i+1} I[p, q \text{ in different cells of } G_i]$

Tree Distance versus Original Distance

- Easy to see $d_T(p, q) \geq \|p - q\|_1$.
- Can we say $d_T(p, q)$ isn't much bigger than $\|p - q\|_1$? No, but perhaps randomization would help...

Random Offsets

- Let $G_0, \ldots, G_k$ of side-lengths

$$1, 2, 2^2, \ldots, 2^k = \Delta$$

  but use a random offset $r = (r_1, r_2) \in \{1, 2, \ldots \Delta\}^2$. E.g., rather than cell $[5, 8] \times [17, 21]$ we have $[5 + r_1, 8 + r_1] \times [17 + r_2, 21 + r_2]$.
- Expected distance is:

$$\begin{aligned} \mathbb{E}\left[d_T(p,q)\right] &= \sum_{i=0}^{\log \Delta - 1} 2^{i+1} \cdot \mathbb{P}\left[p, q \text{ in different cells of } G_i\right] \\ &\leq \sum_{i=0}^{\log \Delta - 1} 2^{i+1} \cdot \frac{\|p-q\|_1}{2^i} \\ &= O(\log \Delta) \cdot \|p-q\|_1 \end{aligned}$$

- Hence, we have

$$\|p-q\|_1 \leq d_T(p,q) \leq \mathbb{E}\left[d_T(p,q)\right] \leq O(\log \Delta)\|p-q\|_1$$

Application: Minimum Spanning Trees

- *Goal:* Estimate weight $w(T)$ of minimum spanning tree T of P.
- Construct probabilistic embedding of P and let T' be the minimum Steiner tree for P in embedding.
- Weight of T' in the embedding is:

$$w(T') = \sum_{i=0}^{L-1} 2^i \cdot (\# \text{ non-zero cells in } G_i)$$

where L is the lowest level where there is only 1 non-zero cell.
- Then,

$$w(T) \leq 2w(T') \quad \text{and} \quad \mathbb{E}\left[w(T')\right] \leq O(\log \Delta) w(T)$$

- Hence $w(T')$ is a $O(\log \Delta)$ approximation to weight of MST.
- Can $(1+\epsilon)$ approximate $w(T')$ with an F_0 estimator.