Randomized Algorithms 2017A – Lecture 6
Distance Oracles and Distance Labeling via Embedding*

Robert Krauthgamer

1 Distance Oracles

Goal: Preprocess a graph $G = (V, E)$ with edge lengths $l : E \to \mathbb{R}_+$ into a (small) data structure that can answer in time $O(1)$ queries about the distance $d = d_G$ (between any two vertices $u, v \in V$).

We denote $n = |V|$ and $m = |E|$.

Naive solution: Store all $\binom{n}{2}$ distances in a matrix/array, with direct access in time $O(1)$.

Can one "compress" the information, perhaps at the expense of accuracy, i.e., the distances are only approximated?

Theorem 1 [Thorup-Zwick, 2001]: There is an algorithm that preprocesses an integer $k \geq 2$ and a graph G in expected time $O(kmn^{1/k})$ and produces a data structure of expected size $O(kn^{1+1/k})$ words that can be used to answer distance queries in time $O(k)$ and approximation factor $2k - 1$.

Remark: We will ignore the preprocessing time, and focus on storage (space). In particular, we assume the shortest path between every two vertices is computed, and essentially use only the fact that distances satisfy the triangle inequality (i.e., it holds for every n-point metric space).

Algorithm Prep(G,k):

1. $A_0 = V$; $A_k = \emptyset$.
2. for $i = 1, \ldots, k-1$
3. construct A_i by including each $u \in A_{i-1}$ independently with probability $1/n^{1/k}$.
4. for every $v \in V$
5. for $i = 0, \ldots, k-1$
6. store $d(v, A_i) = \min\{d(v, w) : w \in A_i\}$ and the minimizer w as $p_i(v)$
7. set $d(v, A_k) = \infty$.

*These notes summarize the material covered in class, usually skipping proofs, details, examples and so forth, and possibly adding some remarks, or pointers. The exercises are for self-practice and need not be handed in. In the interest of brevity, most references and credits were omitted.

8. store $B(v) = \cup_{i=0}^{k-1}\{w \in A_i \setminus A_{i+1} : \ d(v,w) < d(v, A_{i+1})\}$ in a hash table that answers whether $w \stackrel{?}{\in} B(v)$ and if so, what is its distance to v, in $O(1)$ worst-case time.

Intuition of preprocessing:

The sets A_i are subsamples of V at different "levels", and provide "landmarks".

Each "pivot" $p_i(v)$ is just the level i landmark closest to v.

What is the "bunch" set $B(v)$? sort V by distance from v, and partition it into k levels (rings) determined by positions roughly $n^{1/k}, n^{2/k}, \ldots$; store $n^{1/k}$ random vertices from each ring (but coordinated for different "centers" v).

Analysis of preprocessing storage:

The main concern is $\sum_v |B(v)|$, as the pivots can be stored in $O(kn)$ space. The total space bound $O(kn^{1+1/k})$ follows from the next lemma.

Lemma 2: For every $v \in V$, we have $\mathbb{E}[|B(v)|] \leq O(kn^{1/k})$.

The proof was seen in class.

Exer: Show how to implement the preprocessing algorithm in the claimed runtime.

Hint: To compute the pivots of level i, run Dijkstra (single-source shortest paths) from "all" of A_i (e.g. connect all of A_i to a new source vertex s with length 0 edges). To compute the bunches, check for each w to which bunches $B(v)$ it belongs by running Dijkstra from w but truncating it, and bound the expected number of times each edge is traversed.

Algorithm Query(u,v):

1. $i = 0$; $w = u$ // throughout $w = p_i(u)$
2. while $w \notin B(v)$
3. $\quad i = i + 1$
4. $\quad (u, v) = (v, u)$ // swap ["ignore" at first]
5. $\quad w = p_i(u)$
6. return $d(u, w) + d(w, v)$

The runtime is obviously $O(k)$.

Analysis of query algorithm: The entire $A_{k-1} \subseteq B(v)$, hence some answer is always returned, and the number of $u - v$ swaps is at most $k - 1$.

Lemma 3: At each iteration (including swap of u, v), the distance $d(w, u)$ increases by at most $\Delta = d(u, v)$.

The proof was seen in class.

The lemma implies the approximation factor (strech bound), since we start with $d(w_0, u_0) = 0$, and at the final i we have $d(w_i, u_i) \leq i \cdot \Delta \leq (k-1)\Delta$, and thus also $d(w_i, v_i) \leq d(w_i, u_i) + d(u_i, v_i) \leq k\Delta$.

2 Distance labeling via embedding into ℓ_∞

Goal: Preprocess a graph $G = (V, E)$ with edge lengths $l : E \to \mathbb{R}_+$ to create a (small) label for each vertex, so that the distance $d = d_G$ (between any two vertices $u, v \in V$) can be computed from their labels.

Remark: We actually require that the evaluation algorithm does not depend on G, i.e., a single evaluation algorithm for the entire family of graphs of size n.

Embedding and distortion: An *embedding* of a metric space (V, d) into $(\mathbb{R}^s, \ell_\infty)$ is a map $f : V \to R^s$. Its (bi-Lipschitz) *distortion* is the least $D \geq 1$ such that

$$\forall x, y \in V, \qquad d(x, y) \leq \|f(x) - f(y)\|_\infty \leq D \cdot d(x, y).$$

By scaling f, we can instead "move" D to the LHS. The definition extends to embedding into any metric space, such as ℓ_1 or ℓ_2.

Frechet embedding: This is an embedding (map) $f : V \to R^s$ where each coordinate f_i is defined as $f_i : x \to d(x, A_i)$ for some subset $A_i \subseteq V$, where by definition $d(x, A) = \min\{d(x, a) : \ a \in A\}$.

Fact 4: Each coordinate f_i is 1-Lipschitz (nonexpansive), i.e.,

$$|f_i(x) - f_i(y)| \leq d(x, y) \qquad \forall x, y \in V.$$

Proposition 5: Every n-point metric space embeds isometrically (i.e., with distortion 1) into ℓ_∞^n, and thus G admits an exact distance labeling with label-size $O(n)$ words.

Proof: Consider a Frechet embedding with n singleton sets $A_x = \{x\}$. By the above fact, $\|f(x) - f(y)\|_\infty \leq d(x, y)$. For the opposite direction, for every pair $x, y \in X$, we can look at coordinate f_x and get $\|f(x) - f(y)\|_\infty \geq |f_x(x) - f_x(y)| = d(x, y)$.

Question: Can we reduce the dimension? If we allow distortion?

Theorem 6 [Matousek 1996, based on Bourgain 1985]: For every integer $k \geq 2$, every n-point metric space (X, d) embeds with distortion $2k - 1$ into ℓ_∞^s where $s = O(kn^{1/k} \log n)$. This implies a distance labeling with approximation $2k - 1$ and label size $s = O(kn^{1/k} \log n)$.

Proof of Theorem 6: We employ a Frechet embedding whose sets are constructed at random, as follows. Let $q = 1/n^{1/k}$. For each $i = 1, \ldots, k$, construct at random a "group" of $m = \frac{24}{q} \ln n$ sets $A_{i,1}, \ldots, A_{i,m}$ that include every point in V independently with probability $q_i = \min\{1/2, q^i\} = \min\{1/2, 1/n^{i/k}\}$.

Lemma 7: For every $x, y \in V$ there exists i, such that with probability $\geq q/12$,

$$|d(x, A_{i,1}) - d(y, A_{i,1})| \geq \Delta := \tfrac{1}{2k-1} d(x, y).$$

Let's use the lemma to finish the proof of the theorem. For every $x, y \in X$, the probability that all the m random sets (in group i suggested by the lemma) fail this event is at most

$$(1 - q/12)^m < e^{-(q/12)\cdot(24/q)\ln n} = 1/n^2.$$

Finally, apply union bound over the $\binom{n}{2}$ pairs of points.

Proof of Lemma 7 (sketch): Define the following sequence of balls: Let $B_0 = \{x\}$, let B_1 be a (closed) Δ-ball around y, let B_2 be a 2Δ-ball around x and continue this way (with alternating centers) until B_k. Observe that the last two radii add up to $(k-1)\Delta + k\Delta = d(x,y)$.

We claim it is possible to find indices $i \geq 1$ and t such that both

$$|B_t| \geq n^{(i-1)/k} \quad \text{and} \quad |B_{t+1}| \leq n^{i/k}.$$

Intuitively, it just says there is t such that $|B_{t+1}|/|B_t| \leq n^{1/k}$, hence at sampling rate $q^i = 1/n^{i/k}$, there is reasonable probability to "hit" one set and "miss" the other one.

Assume for now the claim is true. Let E_1 be the event that $A_{i,1}$ contains a point from B_t, and E_2 the event that $A_{i,1}$ contains NO point from B_{t+1}. Clearly,

$$\Pr[|d(x, A_{i,1}) - d(y, A_{i,1})| \geq \Delta] \geq \Pr[E_1 \cap E_2] = \Pr[E_1] \cdot \Pr[E_2],$$

because the two events are independent (that's why we used open balls). It is not difficult to verify, using the above claim, that $\Pr[E_2] \geq 1/4$ and $\Pr[E_1] \geq q/3$.

To prove the claim, partition the interval $[1, n]$ to k intervals $I_1, \ldots, I_k$ where $I_i = [n^{(i-1)/k}, n^{i/k}]$. If the sequence $|B_1|, \ldots, |B_k|$ is monotonically increasing, then we can use the pigeon-hole principle ($k+1$ balls and only k intervals) to conclude that some interval has two balls, and in particular two successive balls $|B_t|, |B_{t+1}|$. Otherwise, there is t such that $|B_t| \geq |B_{t+1}|$ then this t and the interval I_i containing $|B_t|$ satisfy the claim.

QED

Exer: Show that every n-point metric space (V, d) embeds into ℓ_1 with distortion $O(\log^2 n)$.

Hint: Use the proof of Theorem 6 with $k = \log n$ to obtain an embedding f into $(\mathbb{R}^s, \ell_1)$ for $s = O(\log^2 n)$.

Exer: Extend this analysis to an embedding into ℓ_2 (instead of ℓ_1) with an even better distortion.

Remark: This is result is stronger not only because the distortion is smaller, but also because of a well-known fact that for finite metrics $\ell_2 \subset \ell_1$ (we may see its proof later).

Theorem 9 [Bourgain 1985]: Every n-point metric space (X, d) embeds into ℓ_2 with distortion $O(\log n)$.

We did not see the proof in class. The basic approach is similar but a more involved analysis is needed to "collect" contributions from all coordinates.