Notes On Improved Bandwidth Approximation for Trees

July 9, 2007

These notes outline the basic definitions and theorems we looked at in class when studying [1]. An intutive sketch of the proof of the main theorem is given.

1 Definitions

The following where the definitions of the basic elements of the work in this paper.

Definition 1.1. *The Bandwidth Minimization problem is the following: given an undirected graph $G = (V, E)$, find a one-one mapping of vertices $f : V \to [n]$ whose bandwidth, which is defined to be:*

$$max_{(i,j)\in E}|f(i) - f(j)|, \tag{1}$$

is minimum.

The Bandwidth Minimization problem is hard to even approximate on trees. In fact, it is NP-hard to approximate it to within any constant even when the input graph is a caterpillar of maximum degree 3.

We consider a tree $T = (V, E)$ with n vertices and l leaves, which we root at an arbitrary vertex r. This imposes an ancestor-descedent relationship on the vertex set of T. Let $d(u, v)$ be the number of edges in the unique path between u and v in T.

Definition 1.2. *The capterpillar dimension of a rooted tree T denoted by $k(T)$ is: for a tree with a single vertex, $k(T) \leq k + 1$ if there exists paths $P_1, P_2, ..., P_k$ beginning at the root and pairwise edge-disjoint such that each component T_j of $T - F(P_1) - E(P_2) - ... - E(P_k)$ has $k(T_j) \leq k$, where $T - F(P_1) - E(P_2) - ... - E(P_k)$ denotes the tree T with the edges of the P_i's removed, and the components T_j are rooted at the unique vertex lying on some P_i.*

The collection of edge-disjoint path in the above recursive definition form a partition of E, and are called caterpillar decomposition of T.

Definition 1.3. *A tree volume $Tvol(S)$ of a k-point metric S is a product of the lengths of the edges of the minimum spanning tree of S.*

Definition 1.4. *The local density D of a graph G is defined to be $max_{u\in V}|N(v, d)|/2d$, where $N(v, d)$ is the set of vertices in G at distance at most d from the vertex v.*

2 Theorems

To devolop some intution consider a simple algorithm for producing a linear arrangement of a rooted tree: let $\phi(r) = 0$ and $\phi(v) = d(r, v)$. This is not a one-to-one map, but we can make it so by arranging the set of vertices falling on a particular position in some arbitrary or random fashion.

This algorithm turns out to be poor since there are examples with bandwidth about $sqrt(n)$ and the algorithm gives $O(n)$.

Algorithm Random Lengths: For each path P_i in the caterpillar decomposition, choose a rate R_i independently and uniformly. For each edge in P_i, let its length be R_i. Let the dstance function using these edge length be denoted by d', and let $\phi(v) = d'(r, v)$

The map $\phi : V \rightarrow R$ induces a linear order on the vertices in V. The map f is the natural conversion of ϕ into a map from V to $[n]$ thus: $f(i) = j$ if $|\{v \in V | \phi(v) < \phi(i)\}| = j$.

Theorem 2.1. *The bandwidth of the output produced by Random Lengths on catilpillars is a logn approximation to the optimal bandwidth.*

Theorem 2.2. *Random Lengths is a $O(log^2 n\sqrt{k(T)})$-approximation algorithm for the bandwidth tree problem.*

The times where the algorithm does bad is when there is a tie in the ordering. The author bounds the probability for this to happen and correspondingly the effect:

Lemma 2.3. *For any set S of k points, the probability that all the points of S fall in some integer interval is bounded above by $O(\sqrt{k(T)})^{k-1}/Tvol(S)$.*

The proof of the lemma uses a lemma obtained by raveling a theorem of Leader and Radcliffe:

Lemma 2.4. *Let X_i be independent random variables, where X_i takes a value uniformly from the set $[d_i, 2d_i]$, where $d_i \in Z$. Then there exists a constant $c \geq 0$ (independent of the d_i's) such that for any unit open iinterval $I \subset R$,*

$Pr(\sum X_i \in I) \leq \frac{c}{(\sum d_i^2)^{1/2}}$

At first, we observe that any two vertices $u, v \in S$ with least common ancestor r', then $d(v, r')/d(u, r')$ lies between 1/2 and 2. This follows simply from the fact that edges are being stretched by at most a factor of 2.

The authors then give an ordering on the vertices of S by traversal of tree T. Let $S = v_1, ..., v_k$, where v_i is visited int this traversal before v_{i+1}. The rate for a path is chosen when a vertex belonging to it is visited for the first time, randomly. Finally, we fix $I \in (a, a+1)$ into which v_1 falls,where $a \in Z$. The probability that v_{i+1} also falls into I is $3c\sqrt{k(T)}/d(v_i, v_{i+1})$. This is shown by consideringthe time at which v_i is visited. We let x be the least common ancestor and v_i and v_{i+1}. Clearly, x is distinct from both these vertices, since they both lie in S, and thus are unrelated. Hence it has children y and z which are ancestors of v_i and v_{i+1}, respectively, and y lies to the left of z. This implies that x and v_{i+1} must have still not been seen. Further, the total contribution of the paths is $d(x, v_{i+1}) \geq d(x, v_i)/2 \geq d(v_i, v_{i+1})/3$. Now the position of v_{i+1}, conditional on the events up to when v_i is visited, is the sum of at most $k(T)$ independent random variables X_j, being the contribution of the path between x and v_{i+1}, and thus lying in some range $[d_j, 2d_j]$, where $\sum d_j = d(x, v_{i+1}$. Thus the chance that v_{i+1} lies in I iss at most $3c\sqrt{k(T)}/d(v_i, v_{i+1})$.

References

[1] Anupam Gupta *Improved Bandwidth Approximation for Trees*