6.838: Geometric Computing

Spring 2005
Problem Set 3
Due: Thursday, April 21

Mandatory Part (note: this problem set does not have the optional parts)

Hint: The solution to the first two problems involves very similar techniques. You might want to think about them in parallel.

Problem 1. Fast Approximate Near Neighbor in l_1

Construct a new data structure for the Approximate Near Neighbor problem in $\Re^d$ under l_1 norm. Your data structure should have the following parameters:

- Approximation factor: $O(d)$
- Space: $O(dn)$
- Query time: $O(d)$

Your data structure can be randomized.

Problem 2. LSH in $\Re^d$ under l_1

In the class, we have seen how to embed $\{0 \dots M\}^d$ equipped with l_1 norm, into the Hamming space $\{0,1\}^{Md}$. This automatically yields a randomized data structure solving a c-approximate Near Neighbor with query time $O(dMn^{1/c})$, for $c = 1 + \epsilon > 1$.

Show how to extend the latter data structure so that it works for points in $\Re^d$ (again, the distance is defined by the l_1 norm). Your data structure should support queries in time $O((d \log n/\epsilon)^{O(1)} n^{1/c})$.

Problem 3. $(1,2) - B$ metrics

In the class, we have seen how to construct an exact embedding of a given metric $M = (X, D)$, $|X| = n$, into l_∞^n. In this problem we consider embeddings of a special subclass of metrics called $(1,2) - B$ *metrics*. A metric is a $(1,2) - B$ metric if it satisfies the following two very particular conditions:

1. All non-zero distances are either 1 or 2
2. For any point $p \in X$, the number of points $q \in X$ such that $D(p,q) = 1$ is at most B.

Show that there is a constant C such that any metric M satisfying the above conditions can be embedded exactly into l_∞^d where $d = CB \log n$.

Hint: Use probabilistic method, similar to the proof of Matousek's theorem.

Note: You might wonder: why anyone would be interested in $(1, 2) - B$ metrics ? It turns out that it is possible to show that, for a certain constant $A > 1$, it is NP-hard to find an A-approximate solution the Traveling Salesman Problem for such metrics (this is a much stronger fact than the NP-hardness of the *exact* TSP showed in the Intro to Algorithms class). This remains true even if B is constant.

The embedding implies that the problem is equally hard even if the metric is induced by n points living in l_∞ with dimension $d = O(\log n)$. So, any A-approximation algorithm for this problem is unlikely to run in time $2^{2^{o(d)}}$. Otherwise, we would have an algorithm solving an NP-hard problem in time $2^{2^{o(d)}} = 2^{2^{o(\log n)}} = 2^{n^{o(1)}}$, i.e., in sub-exponential time, which is conjectured to be impossible.

So, the problem of approximately solving TSP in d-dimensional l_∞ norm suffers from doubly exponential dependence on d. This is a "super-curse of dimensionality" !