Overview of Lecture from 4/6/2006.

We began by observing that the algorithm we saw last class for the k-center problem did not depend at all on the points being in the plane. In fact, the algorithm works for any arbitrary distance function.

What do I mean by a distance function? Well, a distance function is defined on a set of points, which we will call X, and should satisfy

1. \( d(x,x) = 0 \), for all x.
2. \( d(x,y) \geq 0 \), for all y.
3. \( d(x,y) = d(y,x) \).
4. \( d(x,z) \leq d(x,y) + d(y,z) \), for all x, y, and z.
   (this is the triangle inequality)

(the fancy word for X and such a distance function is a "metric space")

In the example we did, X was the plane. But, X could be anything, such as a set of documents, web pages, computers, or something more abstract.

In this case, we would allow both the sites and the centers to be chosen from X.

I then showed that it is NP-hard to get any approximation factor better than 2 for the k-centers problem. There were two main ideas in the proof:

1. We can reduce the vertex-cover problem to the k-center problem, and
2. We can set up this reduction so that all distances in
the k-center problem are either 1 or 2. So, the radius achieved by a set of centers will either be 1 or 2. In such a problem, any algorithm that approximates the solution by a factor better than 2 allows one to distinguish between these two cases.

We will arrange the reduction so that there are k centers with covering radius at most 1 if and only if the graph has a vertex cover of size k.

Here's the reduction.

Let $G = (V, E)$ be a graph.

We will let $X = V \cup E$, so the points in our abstract space correspond to both vertices and edges. We will let $E$ be the set of sites, and set the distance function $d$ so that

1. for vertices $u$ and $v$, $d(u,v) = 1$, unless $u = v$, in which case $d(u,v) = 0$.

2. for edges $e$ and $f$, $d(e,f) = 2$, unless $e = f$, in which case $d(e,f) = 0$.

3. for an edge $e$ and vertex $v$, $d(e,v) = 1$ if $v$ is an endpoint of $e$, and $d(e,v) = 2$ otherwise.

The only condition on distance functions that we really need to check is that these distances satisfy the triangle inequality. But, this is trivial because for distinct $x$, $y$ and $z$, all distances between $x$, $y$ and $z$ will be 1 or 2, and these distances will always satisfy the triangle inequality. (this is why 2 is so important, if we had made the distances 1 and 2.01, this would not be true)

Now, you can observe that if $G$ has a vertex-cover $S$, a subset of $V$, such that $|S| = k$, then $S$ gives $k$ centers
in X such that every site in E is within distance 1 of these centers. Similarly, if such centers exist, then they contain a vertex cover of size at most k.

The next topic in lecture was a 2-approximation algorithm for vertex cover. Here's how it went.

We began by defining a set of edges F subset of E to be independent if all the edges in F have distinct endpoints. We then defined a maximal independent set of edges F to be an independent set such that for all edges e in E-F, F union {e} is not independent.

We observed that it is easy to find a maximal independent set of edges--just go through all the edges, greedily adding them if you can.

We then showed:

1. For every vertex-cover S, and every independent set of edges F, |F| <= |S|.

2. If F is a maximal independent set of edges, then the endpoints of S form a vertex cover.

So, our 2-approximation algorithm was to greedily choose a maximal independent set of edges F, and set S to be the set of all endpoints of edges in F. This gives |S| = 2|F|.

On the other hand, if S* is a vertex cover of minimum size, then we know |F| <= |S*|, which implies

|S| = 2|F| <= 2|S*|.

We then looked at how the algorithm performed on the following graph. Note that this graph has 2k+1 vertices,
and it has a vertex-cover of size $k+1$. On the other hand, if our algorithm makes a poor choice of edges, then it finds a vertex cover of size $2k$, which is off by a factor approaching 2.

Rather than asking if the algorithm could be improved, I suggested that it is unlikely. In particular,
1. A result of Dinur and Safra implies that it is NP hard to approximate vertex-cover by a factor better than 1.36, and
2. Any approximation by a constant factor less than 2, would violate something called the Unique Games Conjecture. It might be false, but it is definitely difficult to break.

We finished class by observing that even though maximum independent set is reducible to minimum vertex cover, our approximation algorithm for vertex cover does not provide an approximation of independent set. (recall that $S$ is a vertex cover iff $V-S$ is an independent set). In the example above, the graph has an independent set of size $k$, but the complement of the 2k vertices our algorithm finds has size just 1.
This is reasonable. A theorem of Hastad implies that it is hard to approximate maximum independent set to any factor $n^{(1-\epsilon)}$, for any $\epsilon$. For example, it is hard to distinguish a graph in which the largest independent set has $n^{(1/100)}$ vertices from one that has an independent set of $n^{(99/100)}$ vertices.