CS466/666, Fall 2011: Assignment 4

Out: November 2, Due: November 16, 5pm

1. **Minimum enclosing ball:** (10 points) The *Minimum enclosing ball* problem consists of finding, given a set of n points in d dimensions, the smallest ball B^* that contains them all. B^* is usually described by giving its centerpoint $c(B^*)$ and its radius $r(B^*)$.

   Give a deterministic 2-approximation algorithm (with respect to the radius) for finding the minimum enclosing ball that has run-time $O(nd)$.

   Remarks: (a) Computing the distance between any two points takes $\Theta(d)$ time. So the run-time of your algorithm is really "$O(n)$ distance computations". (b) There exists a polynomial-time algorithm to find the minimum enclosing ball, but it is complicated and slow for larger d. Knowing this algorithm will not help (and quite possibly hurt) in answering the question.

2. **Euclidean Travelling Salesman:** (6 points) Suppose the vertices of a TSP instance are distinct points in the plane, and the weight of an edge is the Euclidean distance between its endpoints. Show that in any optimal tour, no two edges in the tour cross.

3. **Random algorithm for Independent Set:** (12 points)

   (a) Consider the following algorithm: Randomly permuate the vertices in V. Then for each vertex $v \in V$ (in this random order), add v to I if none of the neighbours of v is in I already.

   Show that the expected size of the resulting independent set is at least $\sum_{v \in V} \frac{1}{\deg(v)+1}$.

   (b) Give a deterministic poly-time algorithm that finds an independent set of size at least $\sum_{v \in V} \frac{1}{\deg(v)+1}$. In other words, de-randomise the algorithm from part (a).

   (c) (Bonus) Which algorithm performs better: the algorithm $\mathcal{A}_1$ from part (a), or the algorithm $\mathcal{A}_2$ from Assignment 2 that finds an independent set of expected size at least $n^2/4m$?

   In other words, either prove $n^2/4m \geq \sum_{v \in V} \frac{1}{\deg(v)+1}$ for all graphs, or prove $n^2/4m \leq \sum_{v \in V} \frac{1}{\deg(v)+1}$ for all graphs, or give examples of two connected graphs G_1, G_2 (with a large number of vertices) such that $n^2/4m < \sum_{v \in V} \frac{1}{\deg(v)+1}$ for G_1 and $n^2/4m > \sum_{v \in V} \frac{1}{\deg(v)+1}$ for G_2.

4. k**-Vertex-Cover:** (12 points) A different way to phrase the Vertex Cover problem is to ask for the maximum number of edges that can be covered with a set of k vertices. Thus, given a graph and k, we want to find a set $C \subset V$ of k vertices such that as many edges as possible have an endpoint in C.

   Give a randomized algorithm that finds a set $C \subset V$ such that the expected size of C is at most k and the expected number of covered edges is at least $\frac{3}{4}OPT$, where OPT is the maximum number of edges that can be covered with k vertices. (Hint: Randomized rounding. You will need to formulate a suitable IP first.)