6.891 Approximation Algorithms February 25, 2000

Problem Set 2

Due Date: March 10, 2000

1. Consider the following problem:

   MAX k-CUT

   - **Input:**
     - An undirected graph $G = (V, E)$
     - Weights $w_{ij} \geq 0$ for all $(i, j) \in E$
   - **Goal:** Partition the vertex set V into k parts $V_1, \ldots, V_k$ so as to maximize the weight of all edges whose endpoints are in different parts (i.e. $\max_{(i,j)\in E: i\in V_a, j\in V_b, a\neq b} w_{ij}$).

   Give a randomized $\frac{k-1}{k}$-approximation algorithm for the MAX k-CUT problem.

2. Consider the linear programming relaxation for the MAX SAT problem that we saw in class:

$$\begin{array}{lll} \text{Max} & \displaystyle\sum_{j=1}^{m} w_j z_j & \\ & \displaystyle\sum_{i\in P_j} y_i + \sum_{i\in N_j} (1 - y_i) \geq z_j & \forall C_j = \displaystyle\bigvee_{i\in P_j} x_i \bigvee_{i\in N_j} \bar{x}_i \\ & 0 \leq y_i \leq 1 & 1 \leq i \leq n \\ & 0 \leq z_j \leq 1 & 1 \leq j \leq m. \end{array}$$

   Prove that using randomized rounding with the linear function $f(y_i) = \frac{1}{2} y_i + \frac{1}{4}$ also gives a $\frac{3}{4}$-approximation algorithm for MAX SAT.

3. The *minimum-capacity multicommodity flow problem* is the following:

   - **Input** An undirected graph $G = (V, E)$, and k pairs of terminals s_i, t_i.
   - **Goal** A multicommodity flow in which one unit of flow is shipped from each s_i to t_i so as to minimize the total flow crossing any edge.

   Assume that we can find an optimal vertex solution of the following linear program in polynomial time:

$$
\begin{array}{lll}
\text{Min} & W & \\
\text{subject to:} & \displaystyle\sum_{P \in \mathcal{P}_i} x_P \geq 1 & \forall i \\
& \displaystyle\sum_{P : e \in P} x_P \leq W & \forall e \in E \\
& x_P \geq 0, &
\end{array}
$$

where $\mathcal{P}_i$ is the set of paths P from s_i to t_i.

(a) (Almost trivial) Argue that the linear program above is a relaxation of the minimum-capacity multicommodity flow problem.

(b) Let W^* be the optimal value of the linear program. Show that if $W^* \geq \ln n$, then we can obtain in randomized polynomial time a solution of value no more than $W^* + c\sqrt{W^* \ln n}$ with high probability.

(c) (Bonus problem) Show how to solve the LP above in polynomial time by giving an equivalent, polynomially-sized LP.

4. Using randomized rounding and first fit, give a randomized polynomial-time algorithm for bin packing that uses $\rho \cdot OPT(I) + k$ bins for some $\rho < 2$ and some small constant k.