Homework 7

Sahil Palnitkar
CSCE 411

35.2-3)
Let the optimal tour at the step I as $H_i^*$ and the tour made by the heuristic as $H_i$. Let the vertex $u_i$ be added such that it is nearest to $v_i$. The cost function satisfies the triangle inequality, giving us: $c(H_i) \leq c(H_{i-1} + 2c)$. So, $c(H_i) \leq 2 \sum_I c(u_i, v_i)$
The nodes and edges are added in closest point heuristic the same way as Prim's algorithm. Therefore, the cost of the MST produced by Prims algorithm is equal to $\sum_I c(u_i, v_i)$.
From the textbook, we see that $c(MST) \leq c(H^*)$. From that we get.
$c(H) \leq 2c(MST) \leq 2c(H^*)$.
So it is proved that a heuristic returns a tour whose total cost is not more than twice the cost of an optimal tour.

35.4-3)
For each vertex v, we will randomly and independently place v in S with a probability ½ and in V – S with probability ½. For an edge $e_i$, we define the indicator random variable $Y_i = I\{e_i$ crossing a cut. For an edge $e_i$ to cross a cut, its vertices u,v have to be in S and V – S respectively. The probability of such an event is Pr{$e_i$ crossing a cut} = Pr{u in S and v in V-S} OR {u in V-S and v in S} = ½ x ½ + ½ x ½ = ½. According to lemma 5.1 in the textbook $E[Y_i] = ½$. Let Y be the number of edges crossing a cut, so that Y = Y1 + Y2 + … + Yn, n = |E|.

With this we get
$E[Y] = E[\sum^n_{i=1} Y_i]$
$= \sum^n_{i=1} E[Y_i]$ (because of expectation linearity)
$= \sum^n_{i=1} ½$
$= ½ n$

Let c* be the weight of the max-cut. The upper bound of c* is the total number of edges, which is $c^* \leq n$.
We get $c^* \leq n$
$= 2 \times ½ n$
$= 2E[Y]$

Therefore we get $c^*/E[Y] \leq 2$ (as this is a maximization problem)
Therefore this is a random 2-approximation problem.

35.4-4)

Let $x : V \rightarrow R_{\geq 0}$ be an optimal solution for the linear programming relaxation with condition 35.19 removed. Suppose there exists v from V such that $x(v) > 1$. Let x' be such that $x'(u) = x(u)$ for $u \neq v$ and $x'(v) = 1$.
The conditions 35.18 and 35.20 are still satisfied for x'.
But $\sum_{(u \text{ in } V)} w(u)x(u) = \sum_{(u \text{ in } V - v)} w(u)x'(u) + w(v)x(v)$
$> \sum_{(u \text{ in } V - v)} w(u)x'(u) + w(v)x'(v)$
$= \sum_{(u \text{ in } V)} w(u)x'(u)$

which contradicts the assumption that x minimized the objective function. So it must be true that $x(v) \leq 1$ for all v in V. So the condition 35.19 is redundant, proved.

26.2-6)
We can modify this similarly to as it is done in section 26.1. We can create an extra vertex $s^{hat}_i$ for each i and place it between s and $s_i$. Remove, the edge from $(s, s_i)$ and add edges $(s, s^{hat}_i)$ and $(s^{hat}_i, s_i)$. In the same way, we create an extra vertex $t^{hat}_i$ and place it between t and $t_i$. Remove the edges $(t_i, t)$ and add edges $(t_i, t^{hat}_i)$ and $(t^{hat}_i, t)$. Assign $c(s^{hat}_i, s_i) = p_i$ and $c(t^{hat}_i, t) = q_i$. If a flow that satisfies the constraints exists, it will assign $f(s^{hat}_i, s_i) = p_i$. By flow conservation, this implies that $\sum_{(v \text{ in } V)} f(s_i, v) = p_i$. Similarly, we must have $f(t_i, t^{hat}_i) = q_i$ so by flow conservation we get $\sum_{(v \text{ in } V)} f(v, t_i) = q_i$