Network Optimization (Fall 2008) – Brief Solutions to Homework 7

1. There is a small typo in the Bellman equations. It should read

$$d(j) = \min\{\, d(i) + c_{ij} \,:\, (i,j) \in AI(j)\} \text{ for all } j \in N.$$

   (a) The shortest path distances $d(\cdot)$ satisfy $d(j) \leq d(i) + c_{ij}$ for every arc $(i,j) \in A$. For the particular arc (i,j) where $pred(j) = i$, we have $d(j) = d(i) + c_{ij}$.

   (b) If $d(\cdot)$ satisfy the Bellman equations, they clearly satisfy the shortest path optimality conditions $d(j) \leq d(i) + c_{ij}, \forall\ (i,j) \in A$. We still need to show that the distance labels represent actual shortest path distances. Consider the *acyclic* graph defined by the arcs (i,j) for which $d(j) = d(i) + c_{ij}$ (tight arcs). Except the source node s, every node has an indegree of at least 1 (if not, that node is unreachable from any s, and will have an infinite distance label – we can leave out this trivial case here). Hence, there will be a directed path from node s to every other node i, and hence the distance labels $d(\cdot)$ represent actual shortest path distances.

2. In the bipartite graph, the longest path (in terms of number of arcs) will have at most $2n_1$ arcs (starting from a node in N_2, the path visits each node in N_1 and finishes in a different node in N_2, thus contributing *two* arcs per node in N_1). Hence, the main loop in the FIFO label correcting algorithm needs to be run only $2n_1$ times, and the algorithm runs in $O(n_1 m)$ time. Of course, the difference in running time will be significant when n_1 is much smaller than n_2, and $s \in N_2$.

   More generally, if the maximum number of arcs in any path in the network is K, then the FIFO label correcting algorithm runs in $O(mK)$ time.

3. See course web page for MATLAB code `FIFO_label_correcting.m`.