Network Optimization (Fall 2008) – Brief Solutions to Homework 6

1. (a) **False**. The graph in Figure 1 can have two distinct shortest path trees.

   (b) **False**. If we make the arcs in the left-most graph in Figure 2 undirected, the shortest path to node 3 changes.

   (c) **False**. See the second graph in Figure 2.

   (d) **False**. See the third graph in Figure 2.

   (e) **False**. For the last graph in Figure 2, Dijkstra's algorithm will find the shortest path with 3 arcs to node 5, whereas there is another shortest path with 2 arcs.

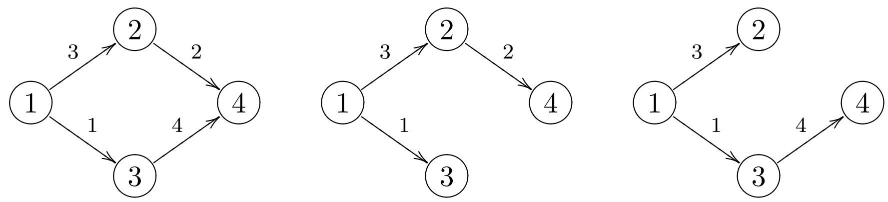

Figure 1: Two distinct shortest paths – 4.21 (a). Network also used for 4.26 (b).

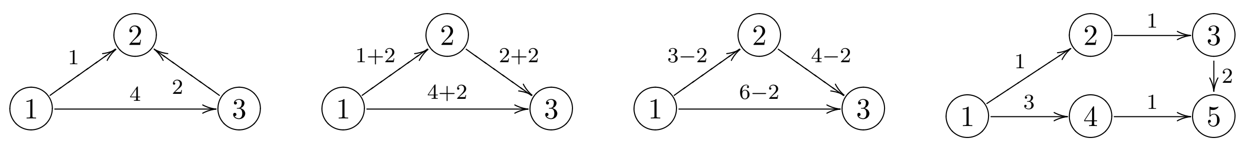

Figure 2: Networks for 4.21 (b),(c),(d), and (e). First network also used for 4.26 (a).

2. (a) **False**. The shortest path length will not be affected if you remove an arc which is not on the shortest path. See first network in Figure 2 – removing $(1,3)$ will not change the shortest path distance from node 1 to node 2.

   (b) **False**. There might be an alternate shortest path in which arc costs are smaller than the one in question. Hence, the shortest path length could remain unaffected. See network in Figure 1 – removing arc $(3,4)$ leaves the shortest path distance to node 4 from node 1 unchanged.

   (c) **True**. Removing such an arc will not affect the shortest path distance from s to t, and hence it cannot even be a vital arc.

   (d) **True**. If all the shortest paths from s to t have two (or more) common (bottleneck) arcs, each of which has the same cost, then each one of them will be a most vital arc.

3. We first show that the shortest path tree remains the same if we add a constant α to the cost of every arc emanating from the source node s (AMO 4.30). Let P be the path in the shortest path tree from s to a node j. After adding α to the costs of all out-arcs of s, the cost of this path will be $c_\alpha(P) = c(P) + \alpha$ (here, $c(P) = \sum_{(i,j)\in P} c_{ij}$). Assume that now there is another path P' from s to j such that $c_\alpha(P') < c_\alpha(P)$. Then it must be true that $c(P') = c_\alpha(P') - \alpha < c(P)$, contradicting the optimality of P in the original network. Note that the new optimal distance labels are given as $d_\alpha(i) = d(i) + \alpha$, where $d(i)$'s are the original optimal distance labels.

   For the case of a network with negative cost only on the out-arcs of s, consider running Dijkstra's algorithm on a modified network obtained by adding $\alpha = -\{\min_j c_{sj} \,|\, c_{sj} < 0\}$ to all c_{sj} values. By the previous result, the shortest path tree thus obtained will be the same as that for the original network. Equivalently, the algorithm will select (and make permanent) the nodes from $\bar{S}$ in the same order.