

HOMEWORK 2
SA405, FALL 2018
INSTRUCTOR: FORAKER

1. 2.40 (this is a harder problem than the others):

The network one can use is to have nodes for each beam length plus a dummy sink node which we denote by 0, i.e.,

$$V = \{0, 4, 7, 9, 10, 14, 18, 23, 30\}.$$

We interpret the 0 node as beams of length “zero”. There will be arcs from every length node to every other shorter length node, i.e.,

$$A = \{(\ell, k) | \ell \in V, \ell > k\}.$$

Thus, the arc $(7, 4) \in A$ but $(4, 7) \notin A$. The trick to understanding the model is to understand what each arc represents. In particular, each arc, (ℓ, k) for $\ell, k \in \{4, 7, 9, 10, 14, 18, 23, 30\}$ represents the decision **to store enough beams of length ℓ to satisfy demand for beams with length less than or equal to ℓ but STRICTLY greater than k** . Define d_ℓ and p_ℓ as the demand and price for beams of length ℓ . Further, let $d_0 = p_0 = 0$ (these aren’t really needed – see below). Then, the arc costs are the setup cost plus the cost of storing the larger beam lengths to satisfy demand for smaller beams. Note that we assume that scrap is discarded (so a 30 ft beam can only make one 4 ft beam, not 7). More precisely,

$$c_{\ell k} = 100 + p_\ell \sum_{\{j: \ell < j < k\}} d_j.$$

For example,

$$c_{30,14} = 100 + 38 \cdot (125 + 90 + 75) = 11120, c_{18,7} = 100 + 25 \cdot (75 + 100 + 110 + 60) = 7285.$$

Solving a shortest path problem on this network with these costs from the node 30 to the node 0 will find the minimum cost storage plan.

2. 2.41: The linear program uses the arc flow variable x_{ij} . Denote the node for Hillbilly, MO by s and the node for Beverly Hills, CA by t . To make the problem a circulation, we add the dummy arc, (t, s) to the network. Note that the capacity on the dummy arc, (t, s) is unbounded. In the LP formulation below we are effectively maximizing the flow on the dummy return arc (t, s) :

$$\begin{array}{ll} \min & -x_{ts} \\ \text{s.t.} & x_{s1} + x_{s3} - x_{ts} = 0 \\ & x_{12} + x_{14} - x_{s1} = 0 \\ & x_{2t} - x_{12} - x_{32} = 0 \\ & x_{32} + x_{34} - x_{s3} = 0 \\ & x_{4t} - x_{14} - x_{34} = 0 \\ & x_{ts} - x_{2t} - x_{4t} = 0 \\ & 0 \leq x_{s1} \leq 9 \\ & 0 \leq x_{s3} \leq 8 \\ & 0 \leq x_{12} \leq 5 \\ & 0 \leq x_{14} \leq 7 \\ & 0 \leq x_{32} \leq 10 \\ & 0 \leq x_{34} \leq 7 \\ & 0 \leq x_{2t} \leq 10 \\ & 0 \leq x_{4t} \leq 12 \\ & x_{ts} \geq 0 \end{array}$$

3. 2.42 (Note that this is slightly different than the shortest path presented in class). The graph is as given with costs $c_{ij} = d_{ij}$. Let n denote the number of nodes. For supply demands, set

$$b_i = \begin{cases} n-1 & i = 1 \\ -1 & \text{otherwise.} \end{cases}$$

There are no arc capacities.

4. 2.44: For the case when $\sum_i s_i > \sum_j d_j$, add a dummy node, t , plus arcs from each supply node s_i to the dummy node t . The arcs have no capacity and no cost. The demand at t is $b_t = -(\sum_i s_i - \sum_j d_j)$. For the case when $\sum_i s_i < \sum_j d_j$, add a dummy node, t , plus arcs from each demand node d_j to the dummy node t . The arcs have no capacity and no cost. The supply at t is $b_t = (\sum_j d_j - \sum_i s_i)$.
5. Consider a minimum cost flow problem on a network, $G = (V, A)$ with arc costs $\{c_{ij} : (i, j) \in A\}$, arc capacities $\{u_{ij} : (i, j) \in A\}$, and node supply/demands, $\{b_i : i \in V\}$. As discussed in class, the mathematical program associated with this can be formulated using decision variables x_{ij} for each $(i, j) \in A$ as:

$$\begin{aligned} \min \quad & \sum_{(i,j) \in A} c_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{\{j:(i,j) \in A\}} x_{ij} - \sum_{\{j:(j,i) \in A\}} x_{ji} = b_i \quad \forall i \in V \\ & 0 \leq x_{ij} \leq u_{ij} \quad \forall (i, j) \in A. \end{aligned}$$

Using simplex would entail finding a feasible solution to the above problem. Formulate a maximum flow problem that, when solved, can be used to (correctly) generate a feasible flow or indicate that the min. cost flow problem is not feasible.

Solution: Add a dummy source node s and sink node t to the graph. Add arcs from s to every supply node (i.e., every node i with $b_i > 0$) and from every demand node (i.e., every node with $b_i < 0$) to t . More precisely, the arc set is now

$$A \cup \{(s, i) | b_i > 0\} \cup \{(i, t) | b_i < 0\}.$$

The arc capacities are as given for arcs in A . For the new arcs the arc capacity is the supply for (s, i) arcs and the negative demand for (i, t) arcs. More precisely, the capacities for the new arcs are:

$$u_{ij} = \begin{cases} b_j & | i = s, \\ -b_i & | j = t. \end{cases}$$

Now consider a feasible flow that has all the new arcs at maximum capacity. Such a flow satisfies all supply constraints as the flow out of supply nodes will equal the flow in (which is constrained to be the supply). The demand constraints are satisfied as the flow into demand nodes will be constrained to be equal to the flow to t (which is constrained to be the demand). All other constraints (capacity and node balance) are also satisfied. Thus, **if the maximum flow returns a flow where all new arcs are at maximum capacity, the original problem is feasible and we have a feasible flow.** However, if the maximum flow returns a flow where a new arc is NOT at maximum capacity, then the original problem was not feasible. For a contradiction, suppose the original problem were feasible and consider the node with a new arc not at maximum capacity. The node must be a supply or a demand. Suppose the node were a demand. Then the feasible flow for the original problem could be used at this node with the flow out of the node being used for the new dummy arc to t . The flow out will be at the maximum capacity and the returned maximum flow is not optimal, a contradiction.