

Exercise 7.5

(i) When branching we find a node with $\text{degree} > 2$. If no such node exists, the solution is feasible and there is no reason for branching. A branch is then created for each edge e incident to the node. In the branch for edge e , it is required, that e is *not* in the solution.

The branch-and-bound tree is shown in figure 1.

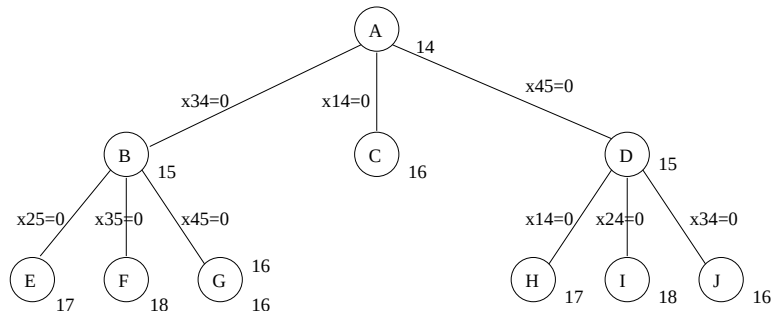


Figure 1: Branch-and-bound tree for exercise 7.5.

The 1-tree lower bound is calculated by first finding a minimum spanning tree in the network except that node 1 is not included. When this is done, node 1 is added by adding it using the 2 shortest edges incident to node 1.

For node A in the branch tree, this gives the 1-tree lower bound shown in figure 2. Node 4 is the only node with $\text{degree} > 2$, hence this node is selected for branching. This gives rise to 3 branches, B, C and D, where edge 3 – 4, 1 – 4 and 4 – 5, respectively are excluded from the solution. The calculated lower bounds for the three branches are shown in figure 2. Values are 15, 16 and 15 respectively.

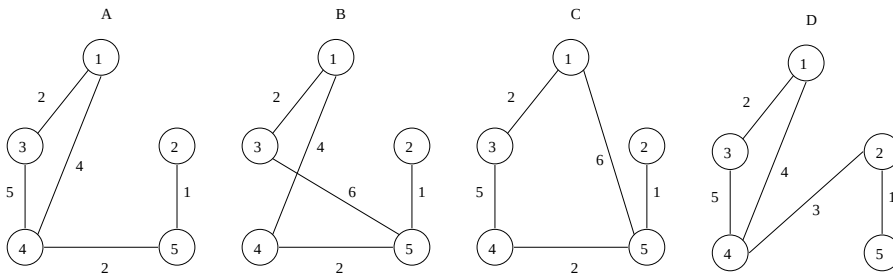


Figure 2: 1-tree lower bound for branches A, B, C and D.

Since both B and D have lower bound = 15, we arbitrarily pick B. In this case

node 5 is the only node with degree >2 , hence this is the node we branch on. This gives 3 new branches E, F and G where edge $2 - 5$, $3 - 5$ and $4 - 5$ are excluded from the solution. The calculated lower bound for the three branches are shown in figure 3. Values are 17, 18 and 16 respectively.

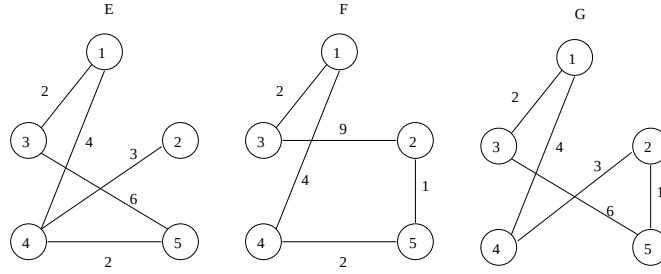


Figure 3: 1-tree lower bound for branches E, F and G.

Since G is feasible, it is pruned and it is a global upper bound. Thus C, E and F can be pruned, since $UB \leq LB$, leaving only branch D to be examined.

The only node with degree >2 in the network corresponding to branch D is node 4. So three branches are created, H, I and J, where edge $1 - 4$, $2 - 4$ and $3 - 4$, respectively are excluded from the solution. The calculated lower bounds for the three branches are shown in figure 4. Values are 17, 18 and 16 respectively which can all be pruned since $UB \leq LB$.

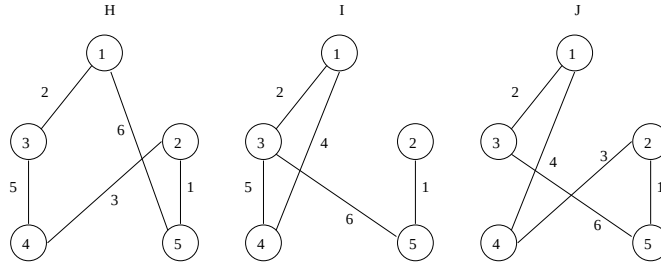


Figure 4: 1-tree lower bound for branches H, I and J.

(ii) Throwing away the subtour elimination constraints gives the assignment relaxation (see page 7-8) used as lower bound. A branching rule would be to identify a subtour (if no subtour exists, the found solution is optimal) and create a branch for each edge in the subtour. In the branch for edge e it is required, that e is *not* in the solution. This can be achieved by inserting a sufficiently large number in place of the original number in the assignment problem.

Solve the assignment problem for the given instance by subtracting the lowest number in each row and the least number in each column:

$$\begin{pmatrix} - & 7 & 6 & 3 \\ 3 & - & 6 & 9 \\ 2 & 3 & - & 1 \\ 7 & 9 & 4 & - \end{pmatrix} \rightarrow \begin{pmatrix} - & 4 & 3 & 0 \\ 0 & - & 3 & 6 \\ 1 & 2 & - & 0 \\ 3 & 5 & 0 & - \end{pmatrix} \rightarrow \begin{pmatrix} - & 2 & 3 & 0 \\ 0 & - & 3 & 6 \\ 1 & 0 & - & 0 \\ 3 & 3 & 0 & - \end{pmatrix}$$

A zero assignment can then be identified as $x_{21} = 1$, $x_{32} = 1$, $x_{43} = 1$ and $x_{14} = 1$. The cost of this solution is (using the original matrix) $3+3+4+3=13$. The solution represents a feasible (and thus optimal) solution since there are no subtours. (Thus no need to branch.)