ECS 122A Homework

Hardy Jones

999397426

Professor Gysel

Fall 2014

1. We can use the Floyd-Warshall algorithm to do the brunt of the work here. We just need to modify some parts of it, and perform some post computation.

   Given a graph G where
   $$G.V = \{C_1, C_2, \ldots, C_n\}$$
   $$G.E = \{H_{i,j} \mid C_i \text{ is connected to } C_j \text{ by a highway}\}$$

   We can construct most of the algorithm the same, and just change the final case in the recursive formula.

   $$D^{(k)}[i,j] = \begin{cases} 0 & \text{if } k = 0 \text{ and } i = j \\ t(H_{i,j}) & \text{if } k = 0 \text{ and } H_{i,j} \in G.E \\ \infty & \text{if } k = 0 \text{ and } i \neq j \text{ and } H_{i,j} \notin G.E \\ min(D^{(k-1)}[i,j], D^{(k-1)}[i,k] + t(C_k) + D^{(k-1)}[k,j]) & \text{if } k \geq 1 \end{cases}$$

   Using this formulation, we construct FLOYD-WARSHALL-MODIFIED similar to the method for constructing FLOYD-WARSHALL in class – taking care to use the modified cases where appropriate.

   Since this gives us back an $n \times n$ array with just the times from one city to another (including travel times between cities), we still need to add on the time necessary to travel from each depot to the highway.

   For each i, j entry in the array, where $i \neq j$, we need the time from the depot to the highway in C_i and the time from the highway to the depot in C_j. Then we can just return the array.

   Our completed algorithm is:

function MOVE-FREIGHT(G, t, d)
    $n \leftarrow |G.E|$
    $D \leftarrow$ FLOYD-WARSHALL-MODIFIED(G, t)
    for $i \leftarrow 1$ to n **do**
        for $j \leftarrow 1$ to n **do**
            if $i \neq j$ **then**
                $D[i, j] = D[i, j] + d(C_i) + d(C_j)$
            end if
        end for
    end for
    return D
end function