Assignment 3

Aryan Choudhary - 170152
Abhinav Sharma - 180017

A hierarchical metric

Algorithm

Our algorithm is very similar to Huffman Code algorithm shown in lecture and analysis using generic technique. By optimal solution in our answer we mean a solution which satisfies the two required conditions in the question and all the nodes present in tree are either ancestor of at least one pair of nodes or leaf.

Condition 1. τ is consistent with d.

Condition 2. If $\tau 1$ is any other hierarchical metric consistent with d, then $\tau 1(i,j) \leq \tau(i,j)\ \forall\ i,j \in S$.

All definitions are mentioned in the question.

Let S be the set of points for which we want to find an optimal solution. We will first reduce it to a problem of $|S|-1$ points then use the optimal solution of those $|S|-1$ points to construct optimal solution for S.

Lets pick two points $u, v \in S$ such that $d(u,v) \leq d(i,j)\ \forall\ i,j \in S$. We will prove that in all optimal solutions $\tau(u,v) = d(u,v)$ (lemma 1) and $\tau(i,u) = \tau(i,v) \leq min(d(u,i), d(v,i))\ \forall\ i \in S - \{u,v\}$ (lemma 3).

Once we have proved above lemmas the solution is easy. We will remove u, v from S and add new node (say z) such that $d(i,z) = min(d(i,u), d(i,v))\ \forall\ i \in S - \{u,v\}$. This z will mimic all conditions imposed due to u, v on other nodes. This step is similar to removing two least frequency letters in Huffman code and replacing it with one letter of frequency of their sum.

To sum up Formally,
Let T be a new set such that $|T| = |S|-1$, $T = (S - \{u,v\}) \cup \{z\}$.
Lets denote distance functions of T, S as d_T, d_S respectively.
We define distance function d_T in terms of d_S as
$d_T(i,j) = d_S(i,j)\ \forall\ i,j \in T - \{z\}$
$d_T(i,z) = d_T(z,i) = min(d_S(i,u), d_S(i,v))\ \forall\ i \in T$
$d_T(z,z) = 0$

Note - $min(A,B)$ denotes minimum value among A, B.

Since $|T| = |S|-1$. We can always repeat these steps recursively to reduce this problem into smaller problems. Solution for just one point (or two points) is very trivial. For one point we just assign the single node as the only leaf. For two nodes we have 2 leafs and their parent at height equal to distance between them. Anything better for these 2 nodes will just violate the distance constraint between them.

Now lets use optimal solution of T consistent with d_T to create one optimal solution of S consistent with d_S. Let E_T, V_T, H_T be the set of edges, vertices, and height function in tree for T. We will remove z from V_T and add 3 new nodes namely u, v, l where l will be the LCA of u, v in E_S.

Formally, for vertex set
$V_S = (V_T - \{z\}) \cup \{u, v, l\}$

For edges set -
$(i,j) \in E_S \iff (i,j) \in E_T\ \forall\ i,j \in V_T - \{z\}$
$(i,l) \in E_S \iff (i,z) \in E_T\ \forall\ i \in V_T - \{z\}$
We also add (u,l) and (v,l) in E_S. Such that l is parent of both u, v and parent of z is the parent of l.

For height function -
$H_S(i) = H_T(i)\ \forall\ i \in V_T - \{z\}$
$H_S(u) = H_S(v) = 0$
$H_S(l) = d_S(u, v)$, Height of all non leaf node (and hence all ancestors of l) $\geq d_S(u, v)$. (Lemma 2). So we can safely make this assignment.

If height of l is same as height of parent of z we wont add this new node. Instead just label parent of z as l. We will just attach u, v to parent of z.

Summing up in simple words what we have done is, we took optimal solution of T. Removed z from T and joined l with parent of z. Made l at height $d_S(u, v)$ and added leaf nodes u, v as children of l to create one optimal solution for S.

Lemmas

1. In the optimal tree, $\tau(u, v) = d(u, v)$ for all pairs (u, v) such that $d(u, v) \leq d(i, j)\ \forall\ i, j \in\ S$

We will prove this using contradiction.
Lets assume that there exist an optimal tree (Γ) in which $\tau_\Gamma(u, v) < d(u, v)$. Let ℓ be the lca of u and v in this tree. Therefore ,

$$H(\ell) = \tau_\Gamma(u, v) < d(u, v)$$

Now consider another hierarchy tree ($\Gamma 1$) which is almost similar to Γ, with the only difference that $H(\ell) = d(u, v)$. This change might induce some inconsistency in the value of height for the parent of ℓ. In order to fix this, we change the height of parent of ℓ to $d(u, v)$ if its height is less than $d(u, v)$. We continue this process until either we reach a node whose height is greater than or equal to $d(u, v)$ or if we reach the root node. We observe that during this modification, if the height of a node changes, it does not increases beyond $d(u, v)$. So, for each internal node of this new tree $\Gamma 1$, its height is either same as that in Γ or increase upto $d(u, v)$.

We claim that consistency of Γ with function d implies consistency of $\Gamma 1$ with d.
Proof $\rightarrow$ Since Γ is consistent with d, we know that for every pair of points i and j, $\tau_\Gamma(i, j) = H(lca(i, j)) \leq d(i, j)$. In $\Gamma 1$, all the pair of points (i,j) , for which the value of $H(lca(i, j))$ is unchanged, the above condition still remains the same and for the rest of the pairs of points , the value of of $H(lca(i, j)) = d(u, v)$ and therefore $\tau_{\Gamma 1}(i, j)$ for those pairs is equal to $d(u, v)$. Since, $d(u, v)$ is the minimum value of distance function for all pairs of points in S, therefore

$$\tau_{\Gamma 1}(i, j) = min(d(x_1, x_2)\ \forall\ x_1, x_2\ in\ S$$

$$\tau_{\Gamma 1}(i, j) \leq d(x_1, x_2)\ \forall\ x_1, x_2\ in\ S$$

Here, (i, j) are those pairs for which we increased $\tau_{\Gamma 1}(i, j)$.

Hence, consistency of $\Gamma \implies$ consistency of $\Gamma 1$.
Now, we know that $\Gamma 1$ is also a hierarchical tree consistent with given function d. Along with $\tau_{\Gamma 1}(u, v) = d(u, v)$ for $\Gamma 1$. $\tau_\Gamma(u, v) < d(u, v)$ for Γ. Therefore, this contradicts the condition 2 for Γ to be an optimal tree.
Hence, it is proved that in the optimal tree for u and v where $d(u, v) \leq d(i, j)\ \forall\ i, j \in S$, $\tau(u, v) \geq d(u, v)$. Due to the condition 1 for an optimal tree $\tau(u, v) \leq d(u, v)$. Hence, $\tau(u, v) = d(u, v)$

2. In the optimal tree, $\tau(i, j) \geq d(u, v)\ \forall\ i, j\ \in S$, where u, v is a pair of nodes such that $d(u, v) \leq d(i, j)\ \forall\ i, j \in\ S$

In simple words height of all non leaf nodes in all optimal solutions of T are more than or equal to $d_S(u, v)$. Proof is by contradiction similar to one presented in lemma 1 (we proceed by increasing heights of non leaf nodes as long as they are less than $d_S(u, v)$. If we end up increasing height of atleast one non leaf node. We have in turn increased height of LCA of atleast one pair of nodes (since all non leaf nodes are LCA of atleast one pair of nodes).

This would contradict the fact that it was one of the optimal solution of T.

3. In the optimal tree, $\tau(i,u) = \tau(i,v) \leq min(d(u,i), d(v,i)) \ \forall\ i \in S - \{u,v\}$**, where** u,v **is a pair of nodes such that** $d(u,v) \leq d(i,j) \ \forall\ i,j \in\ S$.

Let $B = S - \{u,v\}$
Let i be any node in B. Let L be the $LCA(i,u)$ therefore height of $L = \tau(i,u)$.
Height of L cannot be less than $d(u,v)$. It contradicts the lemma 2.
Now, Height of $L \geq d(u,v)$.
Let K be the $LCA(u,v)$, because of lemma 1 height of K is $d(u,v)$.
Because L, K are both ancestors of u and height of $L \geq$ height of $K \implies L$ is an ancestor of K (or $L = K$).
Because K is an ancestor of v and L is an ancestor of $K \implies L$ is an ancestor of v as well.
Because L is an ancestor of v and i. $\implies \tau(i,v) \leq$ height of L $= \tau(i,u)$.

We can work out similar arguments to show $\tau(i,u) \leq \tau(i,v)$.
Hence, $\tau(i,u) = \tau(i,v)$.

Because $\tau(i,u) \leq d(i,u)$ and $\tau(i,v) \leq d(i,v)$ in optimal tree. This proves our claim that
$\tau(i,u) = \tau(i,v) \leq min(d(u,i), d(v,i)) \ \forall\ i \in S - \{u,v\}$.