Algorithmic Bioinformatics DD2450, spring 2010, Lecture 11

Lecturer Jens Lagergren
Several current and previous students
will be acknowledged in a separate document.

May 22, 2010

1 Four Point Condition

Consider four points A, B, C and D in an additive metric. One of the following three inequalities must hold, where $D(i,j) = d_T(i,j)$ (see figures 1 - 2):

1. $D(A,B) + D(C,D) \leq D(A,C) + D(B,D) = D(A,D) + D(B,C)$
2. $D(A,C) + D(B,D) \leq D(A,B) + D(C,D) = D(A,D) + D(B,C)$
3. $D(A,D) + D(B,C) \leq D(A,C) + D(B,D) = D(A,B) + D(C,D)$

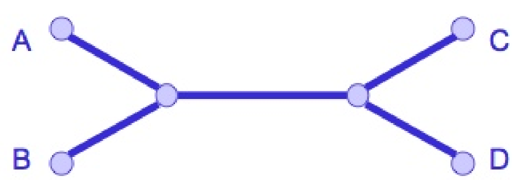

Figure 1: Four point condition - condition 1

Moreover, by observing a quartet it is also possible to derive the following inequality (see figure 3):

$$\max(D(A,B) + D(C,D), D(A,C) + D(B,D), D(A,D) + D(B,C)) -$$
$$- \min(D(A,B) + D(C,D), D(A,C) + D(B,D), D(A,D) + D(B,C))$$
$$\geq 2 \times \text{minimum edge length in } T(D)$$

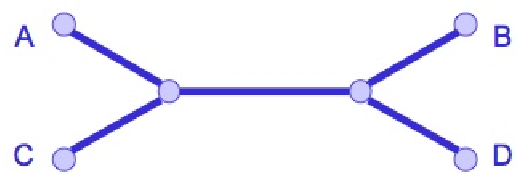

Figure 2: Four point condition - condition 2

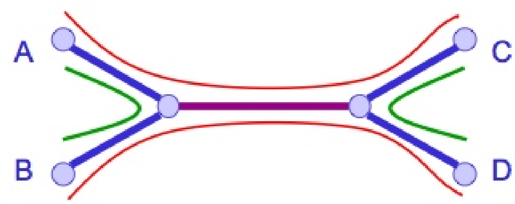

Figure 3: Four point condition - paths in red show $\max(D(A,B) + D(C,D), D(A,C) + D(B,D), D(A,D) + D(B,C))$, paths in green show $\min(D(A,B) + D(C,D), D(A,C) + D(B,D), D(A,D) + D(B,C))$ and path in violet shows $2\times$ minimum edge length in $T(D)$

2 Cherry Identification

Given an additive $n \times n$ distance matrix D let $T = T(D)$.

Idea: Identify a cherry i, j in T and reduce it (i.e. i, j s is obtained by removing i and j from T, alter D s.t. E is obtained and $s = T(E)$). Recursively apply the step and afterwards add i and j to s.

2.1 Version 1

Let

$$w_{ij} = |\{u, v \in \{1, \ldots, n\} \backslash \{i, j\} (D(i,u) + D(j,v)) - (D(i,j) + D(u,v)) > 0\}|$$

Claim

$$w_{ij} = \left(\begin{array}{c} n-2 \\ 2 \end{array} \right) \Leftrightarrow i, j \text{ is a cherry in } T$$

Proof Assume that i, j is a cherry in T and $(u, v) \in \{1, \ldots, n\} \backslash \{i, j\}$. Then i, j, u, v gives a quartet where:

$$(D(i,u) + D(j,v)) - (D(i,j) + D(u,v)) > 0$$

Hence

$$i, j \text{ is a cherry in } T \Rightarrow w_{ij} = \binom{n-2}{2}$$

Now assume that i, j is not a cherry. Then there exists a pair $(u, v) \in \{1, \ldots, n\} \backslash \{i, j\}$ that gives a configuration for which

$$w_{ij} < \binom{n-2}{2}$$

So the equivalency claim holds.

Time complexity The identification takes time $\Omega(n^4)$, which is only reasonable for small instances.

2.2 Version 2

A more efficient algorithm for cherry identification is desirable. One might consider using the following idea:

$$argmin_{i,j} D(i, j)$$

However this method only works for ultra-metric trees e.g. when time is used as edge lengths. It is incorrect in the general case since for certain instances the distance between leaves can be misguiding.

3 Neighbor Joining (NJ)

- Let $S_D(i, j) = (n-2)D(i, j) - \sum_k (D(i, k) + D(j, k))$
- Identify sibling leafs
  - i.e. take $argmin_{i \cdot j} S_D(i, j)$
- Reduce i, j to a "new leaf" a with distances
  - $D(a, x) = (D(i, x) + D(j, x))/2$
- Call NJ recursively on the new matrix
- Add i and j below a in the tree returned
- See figure 4 - 6

Time complexity $O(n^3)$, for an $n \times n$ distance matrix D.

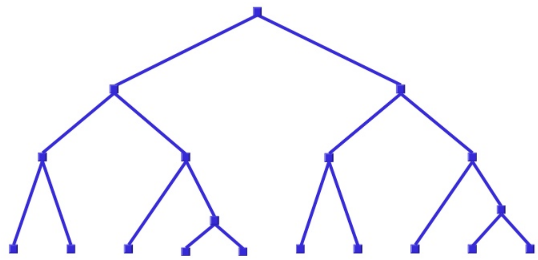

Figure 4: The tree T

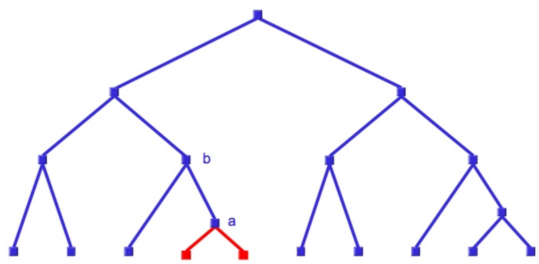

Figure 5: A cherry

3.1 The Proof

- See figures 7 and 8
- Reduce i, j to new taxa a: $E(a,x) \leftarrow (D(i,x) + D(j,x))/2$
- $l_S(a,b) \leftarrow l_T(b,a) + (l_T(a,i) + l_T(a,j))/2$
- $d_S(x,a)$
  $= d_S(x,b) + l_T(b,a) + (l_T(a,i) + l_T(a,j))/2$
  $= d_T(x,b) + l_T(b,a) + (l_T(a,i) + l_T(a,j))/2$
  $= D(x,b) + (l(b,i) + l(b,j))/2$
  $= (D(x,i) + D(x,j))/2$

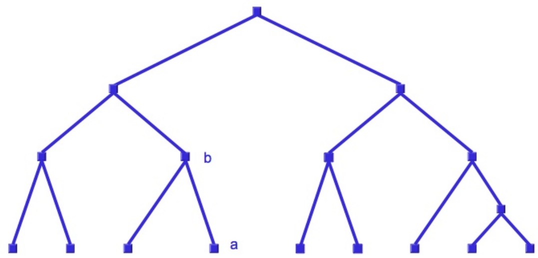

Figure 6: The tree S

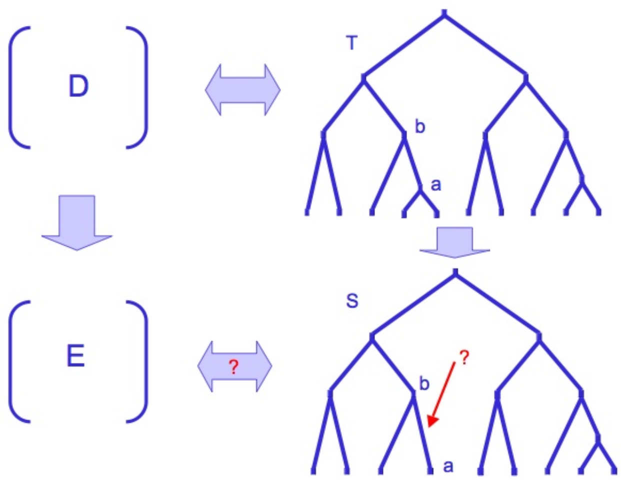

Figure 7: NJ - The proof

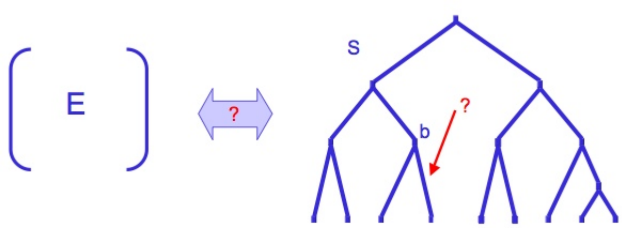

Figure 8: NJ - The proof (continued)