ISSN 1342-2812

Research Reports on Mathematical and Computing Sciences

Robustness of
Greedy Type Minimum Evolution Algorithms

Takeya Shigezumi

Dec 2005, C–218

Department of
Mathematical and
Computing Sciences
Tokyo Institute of Technology

SERIES C: Computer Science

Abstract

For a phylogeny reconstruction problem, Desper and Gascuel [2] proposed Greedy Minimum Evolution algorithm (in short, GME) and Balanced Minimum Evolution algorithm (in short, BME). Both of them are faster than the current major algorithm, Neighbor Joining (in short, NJ); however, less accurate when an input distance matrix has errors. In this paper, we prove that BME has the same optimal robustness to such errors as NJ but GME does not. Precisely, we prove that if the maximum distance error is less than a half of the minimum edge length of the target tree, then BME reconstruct it correctly. On the other hand, there is some distance matrix such that maximum distance error is less than $\frac{2}{\sqrt{n}}$ of the minimum edge length of the target tree, for which GME fails to reconstruct the target tree.

1 Introduction

A phylogeny reconstruction problem is to determine an evolutionary tree representing relationships between a set of species from a distance matrix expressing the "closeness" of each pair of species. Recently, the phylogeny reconstruction is not only used for biology, but also for clustering various kinds of data, such as languages[10], music, etc.[11] To state the phylogeny construction problem formally, we need some notations. For any n, we use $L = \{1, \ldots, n\}$ to denote a set of n species. A distance matrix $D = (d_{ij})$ over L , for each pair i and j in L, a distance $d_{ij} \geq 0$. Note that $d_{ij} = d_{ji}$ and that $d_{ii} = 0$. An evolutionary tree over L is a tree such that its set of leaves is L and all internal vertices are of degree three. For any tree T and for any i and j in L, let d^T_{ij} denote the length of the unique path between i and j, i.e., the sum of the length of edges on the path. Now the phylogeny construction problem is stated as follows.

Instance: A distance matrix $D = (d_{ij})$ over $L = \{1, \ldots, n\}$.

Question: Find a tree T^* minimizing the following square error:

$$\sum_{i<j} \left| d_{ij} - d^{T^*}_{ij} \right|^2$$

Day [8] showed that this problem is NP-hard in general. It is, however, possible to "reconstruct" a tree in polynomial time if a distance matrix is sufficiently close enough to the one for a given tree. Note that for any tree T, this T itself is the unique solution for the phylogeny construction problem on the distance matrix $D^T = (d^T_{ij})$. Several polynomial time algorithms have been proposed for reconstructing the original tree T from D^T. Among them, Neighbor-Joining[4], its variants BIONJ [5] and WEIGHBOR[6], and Fitch-Margoliash as implemented by Felsenstein [9] are well known. NJ and its variants run in $O(n^3)$ for reconstructing a tree of n leaves. Though polynomial, this running time is not sufficient enough for constructing a large evolutionary trees, and several attempts have been made to improve the efficiency .

Recently, Desper and Gascuel [2] introduced Greedy Minimum Evolution (GME) algorithm based on the ordinary least squares (OLS) branch length estimation and Balanced Minimum Evolution (BME) algorithm based on a branch length estimation scheme of Pauplin [3]. BME runs in $O(n^2\mathrm{diam}(T))$ where diam(T) is a diameter of the target tree, and GME runs in $O(n^2)$, which is optimal considering that the size of the input distance matrix is $O(n^2)$. Moreover, BME has an important property that the output tree has no edge with negative length [1]. Since edges with negative length are biologically meaningless, this property is also an advantage of BME.

In practice, a distance matrix has errors, that is, some distance δ_{ij} differs from the original distance d_{ij}^T. For measuring such errors, Atteson [7] introduced the notation of "l_∞ radius." We say that a distance matrix $\Delta = (\delta_{ij})$ has l_∞ *radius* α to an original distance $D^T = (d_{ij}^T)$ if

$$\|D^T - \Delta\|_\infty = \max_{i<j} |d_{ij}^T - \delta_{ij}| < \alpha \min_{e \in E(T)} \{l(e)\},$$

where $E(T)$ is the set of edges of T and $l(e)$ denotes the length of an edge e. An algorithm has l_∞ *radius* α if it yields the original T for any distance matrix of l_∞ radius α to D^T. Note that no algorithm has l_∞ radius larger than $\frac{1}{2}$. Because if we allow errors larger than $\frac{1}{2}l(e)$, there exists one distance matrix made from two different trees with such errors. Atteson showed that NJ has l_∞ radius $\frac{1}{2}$.

Desper and Gascuel showed the relationship between BME length estimation and weighted least squares branch length estimation [1] as a theoretical foundation of BME. However, there is still no theoretical analysis about l_∞ radius for GME and BME. We analyze BME and GME, and obtain the following two theorems, which claim that BME has l_∞ radius $\frac{1}{2}$ but GME does not.

Theorem 1. *BME has l_∞ radius $\frac{1}{2}$, i.e., if the input distance matrix Δ satisfies*

$$\left\| D^{T^*} - \Delta \right\|_\infty < \frac{1}{2} \min_{e \in E(T^*)} \{l(e)\}$$

BME outputs T^* .

Theorem 2. *Let α_G be an l_∞ radius of GME. Then,*

$$\alpha_G < \frac{2}{\sqrt{n}}.$$

According to these theorems, BME has the same robustness as NJ, and it is faster than NJ. On the other hand, GME is faster than BME, it is less accurate in some cases. Our proof is obtained by a precise analysis of the proof of Desper and Gascuel [1] for the correctness of BME on error free distance matrix. Desper and Gascuel also proposed Balanced Nearest Neighborhood Interchange (BNNI) algorithm to improve a phylogeny tree into minimal one. However, according to our theorem, we do not need to use BNNI when BME is used for a distance matrix Δ with sufficiently small $\|D^{T^*} - \Delta\|_\infty$ error.

2 Preliminaries

We call T^* the target tree being studied with an edge set $E(T^*)$ and edge weight function $l : E(T^*) \to \mathbb{R}$. Let $l_{\min}^{T^*} = \min_{e \in E(T^*)}\{l(e)\}$ and $\Delta = (\delta_{ij})$ be an estimate of the evolutionary distance between leaves i and j. Let $E = (\varepsilon_{ij})$ be an error matrix given by following equation:

$$\varepsilon_{ij} = \delta_{ij} - d_{ij}^{T^*},$$

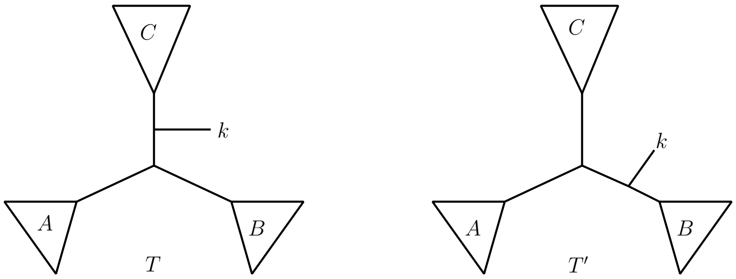

Figure 1: Positions the leaf k is inserted

and ε_0 be the maximum error such that $\varepsilon_0 = \max_{i,j} |\varepsilon_{ij}|$. For a given tree T, we say P_{ij}^T be a number of edges on the path connecting i and j on the tree T, and say $r(A)$ for any subtrees A of T be a root vertex of the subtree. Let P_{AB}^T be a number of edges from $r(A)$ to $r(B)$ for any disjoint subtrees A and B . We will treat a subtree A as a set of leaves for a simpler notation.

Pauplin [3] showed BME principle to estimate the edge length by modifying the OLS analytical formulae. Under this principle , the tree length is estimated as $\hat{l}(T, \Delta) = \sum_{i,j} 2^{1-P_{ij}^T} \delta_{ij}$. Consider the tree T in Figure 1, where k is inserted between subtrees C and $A \cup B$, and assume that we have the size $\hat{l}(T, \Delta)$ of this new tree. If we know the size of T, we can quickly calculate the size of T' using the following equation:

$$\hat{l}(T', \Delta) = \hat{l}(T, \Delta) + \frac{1}{4} \left\{ \left(\delta_{AC}^T + \delta_{kB}^T \right) - \left(\delta_{AB}^T + \delta_{kC}^T \right) \right\}. \tag{1}$$

In this formula, δ_{AB}^T is defined as follows. If A and B each contains only one leaf, denoted as a and b, respectively, then $\delta_{AB}^T = \delta_{ab}$, while if one of the two, say B, is made of two subtrees denoted as B_1 and B_2, then $\delta_{AB}^T = \frac{1}{2} \left(\delta_{AB_1}^T + \delta_{AB_2}^T \right)$.

Under the OLS length estimation principle, (1) is represented as follows:

$$\begin{aligned} \hat{l}(T', \Delta) - \hat{l}(T, \Delta) = \frac{1}{2} \big[&(\lambda - \lambda') \left(\Delta_{k|A} + \Delta_{B|C} \right) \\ &+ (\lambda' - 1) \left(\Delta_{A|B} + \Delta_{k|C} \right) + (1 - \lambda) \left(\Delta_{A|C} + \Delta_{k|B} \right) \big] \end{aligned} \tag{2}$$

where $\lambda = \frac{(|A|+|B||C|)}{(|A|+|B|)(|C|+1)}$, $\lambda' = \frac{(|A|+|B||C|)}{(|A|+|C|)(|B|+1)}$ and $\Delta_{A|B} = \frac{1}{|A||B|} \sum_{i \in A, j \in B} \delta_{ij}$.

We here introduce BME algorithm by Desper and Gascuel [2]. This algorithm starts with the tree of three leaves and adds the leaf to minimize the estimated tree length.

BME algorithm is as follows:

- For $k = 3$, initialize the matrix of balanced average distances between distant-2 subtrees and the array counting the number of leaves par subtree. Form T_3 with leaf set $\{1, 2, 3\}$.

- For $k = 4$ to n,

  1. Compute all $\delta_{kS}^{T_{k-1}}$ balanced average distances;

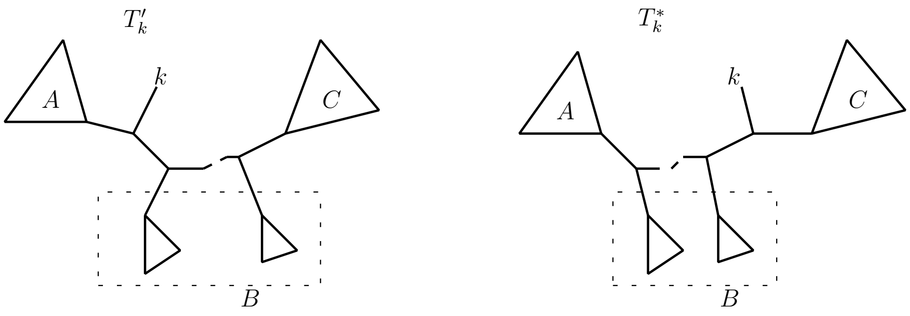

Figure 2: T_k^* and T_k'

2. Starting from an initial edge e_0 of T_{k-1}, set $f(e_0) = 0$, and recursively search each edge e to obtain $f(e)$ from (1).
3. Select the best edge by minimizing f, insert k on the edge to form T_k, and update the balanced average distance between every pair of distant-2 subtrees as well as the number of leaves per subtree.

- Return T_n.

The time complexity of BME is $O(n^2 \mathrm{diam}(T^*))$. GME is almost same but using (2) and average distance $\Delta_{A|B}$ instead of the equation 1 and δ^T_{AB}. The time complexity of GME is $O(n^2)$.

3 Robustness of BME

BME algorithm chooses the position of the leaf k step by step. Let T_k^* be a tree over $\{1, \ldots, k\}$ induced from the target tree T^*. The proof of Theorem 1 is by the induction of k. In $k = 3$, there is only one topology T_3^*. Let us consider the k-th round. By the induction hypothesis, BME reconstructs the target tree topology T_{k-1}^*. BME chooses T_k minimizing $\hat{l}(T_k, \Delta_k)$, where Δ_k is a distance matrix on $\{1, 2, \ldots, k\}$ induced from Δ.

For any $T_k' \neq T_k^*$, we prove the following inequality holds if $\varepsilon_0 < \frac{1}{2} l_{\min}^{T^*}$.

$$\hat{l}(T_k', \Delta_k) - \hat{l}(T_k^*, \Delta_k) = \sum_{ij} 2^{1-P_{ij}^{T_k'}} \varepsilon_{ij} - \sum_{ij} 2^{1-P_{ij}^{T_k^*}} \varepsilon_{ij} + \hat{l}(T_k', D_k^{T^*}) - \hat{l}(T_k^*, D_k^{T^*}) > 0.$$

We call the value of the first two terms of the last equation the error difference and the value of the last two terms the tree difference.

We here provide two lemmas about the error difference and tree differnece. Let T_k' and T_k^* , and name subtrees A, B and C of T_{k-1} as shown in Figure 2. Let $p = P_{AC}^{T_{k-1}}$, i.e., number of edges on the path of T_{k-1} between the root of A and the root of B.

Error difference

First, we provide a lemma for the error difference.

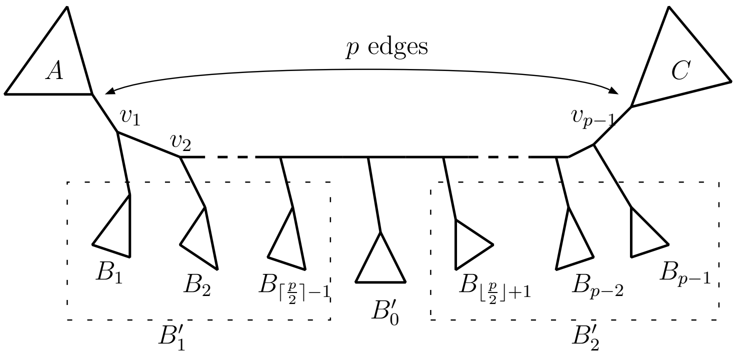

Figure 3: T_{k-1}

Lemma 1.

$$\sum_{ij} 2^{1-P_{ij}^{T'_k}} \varepsilon_{ij} - \sum_{ij} 2^{1-P_{ij}^{T^*_k}} \varepsilon_{ij}$$
$$\geq -3\varepsilon_0 + 2 \cdot 2^{-p}\varepsilon_0 + \begin{cases} 3 \cdot 2^{-\frac{p}{2}}\varepsilon_0 & (p : \mathit{even}) \\ 2 \cdot 2^{-\lfloor \frac{p}{2} \rfloor}\varepsilon_0 & (p : \mathit{odd}) \end{cases}$$

Proof. We consider the tree T_{k-1} as shown in Figure 3. Name the vertices on the path from $r(A)$ to $r(C)$ $v_1, v_2, \ldots, v_{p-1}$. Let B_i the set of leaves which connects to the path from $r(A)$ to $r(C)$ at the vertex v_i for $i \in \{1, 2, \ldots, p-1\}$, and $B = \bigcup_{i=1}^{p-1} B_i$. Let $B'_1 = \bigcup_{i=1}^{\lceil \frac{p}{2} \rceil -1} B_i$, $B'_2 = \bigcup_{i=\lfloor \frac{p}{2} \rfloor +1}^{p-1} B_i$ and $B'_0 = B_{\frac{p}{2}}$ when p is even. If p is odd, $B'_0 = \phi$.

Now we analyze the error difference in every case that each of the leaf i and j belongs to A, B, C or $\{k\}$. Let $E_{XY} = \sum_{i \in X, j \in Y} 2^{1-P_{ij}^{T'_k}} \varepsilon_{ij} - \sum_{i \in X, j \in Y} 2^{1-P_{ij}^{T^*_k}} \varepsilon_{ij}$.

- $i, j \in \{1, \ldots, k-1\}$.
  - $i \in A, j \in C$: By $P_{ij}^{T'_k} = P_{ij}^{T^*_k} = P_{ij}^{T_{k-1}} + 1$, the error difference is 0.
  - $i \in A, j \in B$: The number of edges between i and j satisfies $P_{ij}^{T'_k} = P_{ij}^{T_{k-1}} + 1$ and $P_{ij}^{T^*_k} = P_{ij}^{T_{k-1}}$ in T'_k and T^*_k respectively. In the worst case $\varepsilon_{ij} = \varepsilon_0$. The error difference is
  $$E_{AB} = -\frac{1}{2} \sum_{i \in A, j \in B} 2^{1-P_{ij}^{T_{k-1}}} \varepsilon_{ij} \geq -\frac{1}{2} \sum_{i \in A, j \in B} 2^{1-P_{ij}^{T_{k-1}}} \varepsilon_0.$$
  - $i \in B, j \in C$: By the same analysis of E_{AB} with the the worst case $\varepsilon_{ij} = -\varepsilon_0$,
  $$E_{BC} \geq -\frac{1}{2} \sum_{i \in B, j \in C} 2^{1-P_{ij}^{T_{k-1}}} \varepsilon_0.$$
  - $i, j \in X$, $(X = A, B$ or $C)$: $E_{XX} = 0$ by $P_{ij}^{T'_k} = P_{ij}^{T^*_k} = P_{ij}^{T_{k-1}}$.

	worst case error difference (lower bound)	the worst case input ε_{ij}
$i \in A, j \in C$	0	
$i \in A, j \in B$	$-\frac{1}{2} \sum_{i \in A, j \in B} 2^{1-P_{ij}^{T_{k-1}}} \varepsilon_0$	$\varepsilon_{ij} = \varepsilon_0$
$i \in B, j \in C$	$-\frac{1}{2} \sum_{i \in B, j \in C} 2^{1-P_{ij}^{T_{k-1}}} \varepsilon_0$	$\varepsilon_{ij} = -\varepsilon_0$
$i, j \in X (X = A, B, C)$	0	
$i \in A, j = k$	$-(2^{p-1} - 1) \sum_{i \in A} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0$	$\varepsilon_{ij} = -\varepsilon_0$
$i \in B'_0, j = k$	0	
$i \in B'_1, j = k$	$\sum_{1 \le i < \frac{p}{2}} \left[-(2^{p-2i} - 1) \sum_{i \in B_i} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0 \right]$	$\varepsilon_{ij} = -\varepsilon_0$
$i \in B'_2, j = k$	$\sum_{\frac{p}{2} < i \le p-1} \left[(2^{p-2i} - 1) \sum_{i \in B_i} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0 \right]$	$\varepsilon_{ij} = \varepsilon_0$
$i \in C, j = k$	$(2^{-(p-1)} - 1) \sum_{i \in C} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0$	$\varepsilon_{ij} = \varepsilon_0$

Table 1: A lower bound of error difference in each cases.

- $i \in \{1, \ldots, k-1\}, j = k$
  - $i \in A$: The number of edges between i and j is $P_{ij}^{T'_k} = P_{ij}^{T_k^*} - p + 1$. With the worst case $\varepsilon_{ij} = -\varepsilon_0$, the error difference is

$$E_{A\{k\}} \ge -(2^{p-1} - 1) \sum_{i \in A} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0.$$

  - $i \in B'_0$: By $P_{ij}^{T'_k} = P_{ij}^{T_k^*}$, $E_{B'_0\{k\}} = 0$.
  - $i \in B'_1$: We can obtain the error difference with the worst case $\varepsilon_{ij} = -\varepsilon_0$,

$$E_{B'_1\{k\}} \ge \sum_{1 \le i < \frac{p}{2}} \left[-(2^{p-2i} - 1) \sum_{i \in B_i} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0 \right].$$

  - $i \in B'_2$: Same as $i \in B'_1$, with the worst case $\varepsilon_{ij} = \varepsilon_0$,

$$E_{B'_2\{k\}} \ge \sum_{\frac{p}{2} < i \le p-1} \left[(2^{p-2i} - 1) \sum_{i \in B_i} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0 \right].$$

  - $i \in C$: Same as $i \in A$, with the worst case $\varepsilon_{ij} = \varepsilon_0$,

$$E_{C\{k\}} \ge (2^{-(p-1)} - 1) \sum_{i \in C} 2^{1-P_{ik}^{T_k^*}} \varepsilon_0.$$

Summary of above worst cases is shown in Table 1. We here introduce the following lemma to treat summations such as: $\sum_{i \in X, j \in Y} 2^{1-P_{ij}^{T}} \varepsilon_0$.

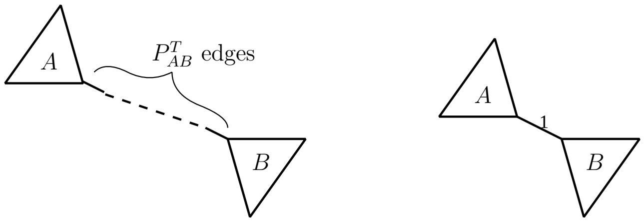

Figure 4: Left:Subtrees A and B, Right:T_{AB}

Lemma 2. *For any disjoint subtrees A and B of T,*

$$\sum_{i \in A, j \in B} 2^{1-P_{ij}^T} = 2^{1-P_{AB}^T}.$$

Proof.
Let T_{AB} as shown in the right hand side of Figure 4, there is an edge connecting the root of subtrees A and B. It is easy to see that $P_{ij}^T = P_{ij}^{T_{AB}} - 1 + P_{AB}^T$. So we obtained

$$\sum_{i \in A, j \in B} 2^{1-P_{ij}^T} = 2^{1-P_{AB}^T} \sum_{i \in A, j \in B} 2^{1-P_{ij}^{T_{AB}}}.$$

For T_{AB}, we can consider the distance $(d_{ij}) = \begin{cases} 1 & i \in A \text{and } j \in B \\ 0 & otherwise \end{cases}$, and

$$\sum_{i \in A, j \in B} 2^{1-P_{ij}^{T_{AB}}} = \sum_{i,j \in A \cup B, i<j} 2^{1-P_{ij}^{T_{AB}}} d_{ij}$$

This is the length of T_{AB}, 1. There is an edge connecting subtrees A and B has weight 1, and all other edges have weight 0. ■

Corollary 1. *For a subtree A, C and a set of leaves B according to a family of subtrees as shown in Figure 5,*

$$\sum_{i \in A, j \in B} 2^{1-P_{ij}^T} = 2^{1-P_{A(B \cup C)}^T} - 2^{1-P_{AC}^T}.$$

Proof. As shown in Lemma 2,

$$\sum_{i \in A, j \in B} 2^{1-P_{ij}^T} = \sum_{i \in A, j \in B \cup C} 2^{1-P_{ij}^T} - \sum_{i \in A, j \in C} 2^{1-P_{ij}^T} = 2^{1-P_{A(B \cup C)}^T} - 2^{1-P_{AC}^T}.$$

■

By summing up error difference in Table 1 and applying Lemma 2 and

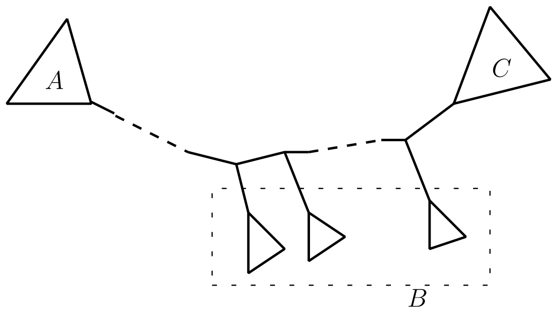

Figure 5: Subtrees A,B and C

Corollary 1,

$$
\begin{aligned}
&\sum_{i<j} 2^{1-P_{ij}^{T_k'}}\varepsilon_{ij} - \sum_{i<j} 2^{1-P_{ij}^{T_k^*}}\varepsilon_{ij} \\
&\geq -\frac{1}{2}\left(2^0 - 2^{1-p}\right)\varepsilon_0 - \frac{1}{2}\left(2^0 - 2^{1-p}\right)\varepsilon_0 \\
&\quad - \left(2^{p-1} - 1\right)2^{1-(p+1)}\varepsilon_0 + \left(2^{-p+1} - 1\right)2^{1-2)}\varepsilon_0 \\
&\quad - \sum_{1 \leq i < \frac{p}{2}} \left[(2^{p-2i} - 1)\left\{2^{1-(p-i+1)} - 2^{1-(p-i+2)}\right\}\varepsilon_0\right] \\
&\quad + \sum_{\frac{p}{2} < i \leq p-1} \left[(2^{p-2i} - 1)\left\{2^{1-(p-i+1)} - 2^{1-(p-i+2)}\right\}\varepsilon_0\right] \\
&= -2\varepsilon_0 + 4 \cdot 2^{-p}\varepsilon_0 \\
&\quad - \left[\sum_{1 \leq i < \frac{p}{2}} 2^{-i} - \sum_{\frac{p}{2} < i \leq p-1} 2^{-i}\right]\varepsilon_0 \\
&= -2\varepsilon_0 + 4 \cdot 2^{-p}\varepsilon_0 - \left[1 - 2^{1-\lceil \frac{p}{2} \rceil} - \left\{2^{-\lfloor \frac{p}{2} \rfloor} - 2^{1-p}\right\}\right]\varepsilon_0 \\
&= -3\varepsilon_0 + 2 \cdot 2^{-p}\varepsilon_0 + \begin{cases} 3 \cdot 2^{-\frac{p}{2}}\varepsilon_0 & (p : \text{even}) \\ 2 \cdot 2^{-\lfloor \frac{p}{2} \rfloor}\varepsilon_0 & (p : \text{odd}) \end{cases}.
\end{aligned}
$$

■

Tree difference

Next, we provide a lemma for the tree difference.

Lemma 3.

$$\hat{l}(T_k', D_k^{T^*}) - \hat{l}(T_k^*, D_k^{T^*}) \geq l_{\min}^{T_k^*}\left(p - 2 + 2 \cdot 2^{-p}\right).$$

Proof. Desper and Gascuel [1] defined the split metric $D^{X|Y} = (d_{ij}^{X|Y})$ by $d_{ij}^{X|Y} = 1$ when $i \neq j$ and $|\{i,j\} \cap X| = 1$, and $d_{ij}^{X|Y} = 0$ otherwise. Let $S(T)$

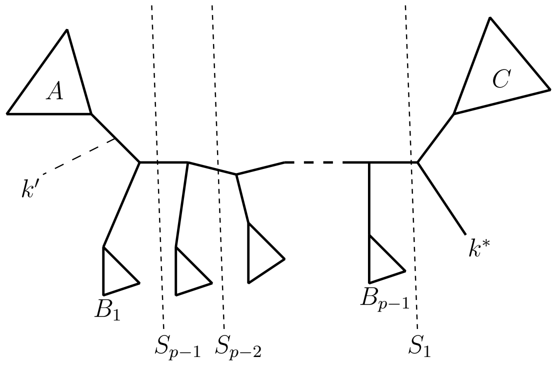

Figure 6: Splits in $S(T_k^*)\backslash S(T_k')$. k^* and k' represents the position of the leaf k in T_k^* and T_k' respectively.

be a set of all splits of T. Using this split metric, the tree difference $\hat{L}(T_k', T_k^*)$ is represented in:

$$\hat{l}(T_k', D_k^{T^*}) - \hat{l}(T_k^*, D_k^{T^*}) = \sum_{X|Y \in S(T_k^*)} l(X|Y) \left\{ \hat{l}(T_k', D_k^{X|Y}) - \hat{l}(T_k^*, D_k^{X|Y}) \right\}$$

where $l(X|Y)$ is a length of an edge induced by the split $X|Y$. For a split $X|Y$ which is in both $S(T_k')$ and $S(T_k^*)$, the difference $\hat{l}(T_k', D_k^{X|Y}) - \hat{l}(T_k^*, D_k^{X|Y}) = 0$. Let us suppose the split $X|Y$ is in $S(T_k^*)$ but not in $S(T_k')$. First, we will consider how many splits are in $S(T_k^*)\backslash S(T_k')$, and later, we will consider how much difference for one split belonging to $S(T_k^*)\backslash S(T_k')$.

Consider T_k' and T_k^* as shown in Figure 2. Name the subtrees as shown in Figure 3. It is easy to see that the splits $A \cup B_1 \cup B_2 \cup \ldots \cup B_{p-1} | \{k\} \cup C$, $A \cup B_1 \cup B_2 \cup \ldots \cup B_{p-2} | B_{k-1} \cup \{k\} \cup C$, $\ldots$, $A \cup B_1 | B_2 \cup \ldots \cup B_{p-1} \cup \{k\} \cup C$ are all in $S(T_k^*)\backslash S(T_k')$. We call these splits S_1, S_2, $\ldots$, S_{p-1}, respectively. (Figure 6)

Now, fix a split $S_t = X|Y$, and let us consider the split difference $\hat{l}(T_k', D_k^{S_t}) - \hat{l}(T_k^*, D_k^{S_t})$. Following the proof of Desper and Gascuel, we shall color the leaves in black and white according to S_t. The leaves in $X = A \cup B_1 \cup, \ldots, \cup B_{p-t}$ are colored in black and the others ($Y = B_{p-t+1}, \cup, \ldots, B_{p-1} \cup \{k\} \cup C$) are in white. The split difference between T_k' and T_k^* is represented as a sum of differences of the series of topological transformations $T_k' = T_k^1 \to T_k^2 \to \cdots T_k^p = T_k^*$,

$$\hat{l}(T_k', D_k^{S_t}) - \hat{l}(T_k^*, D_k^{S_t}) = \sum_{i=1}^{p-1} \left\{ \hat{l}(T_k^i, D_k^{S_t}) - \hat{l}(T_k^{i+1}, D_k^{S_t}) \right\}.$$

The trees of the above transformation are shown in Figure 7, k^i represents the position of the leaf k in T_k^i.

We here present a lemma about the difference of one transformation under a split distance.

Lemma 4. *For $1 \leq i \leq p-t$, the following equation holds.*

$$\hat{l}(T_k^i, D_k^{S_t}) - \hat{l}(T_k^{i+1}, D_k^{S_t}) = 2^{-p+i+t-1}. \tag{3}$$

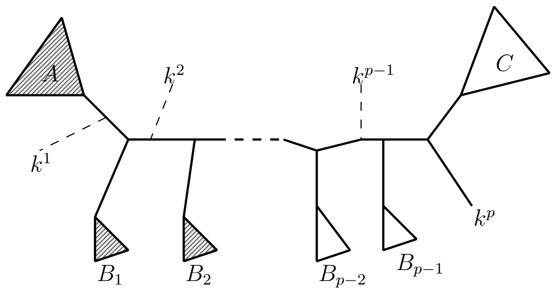

Figure 7: k^i represents the position of the leaf k in T_k^i.

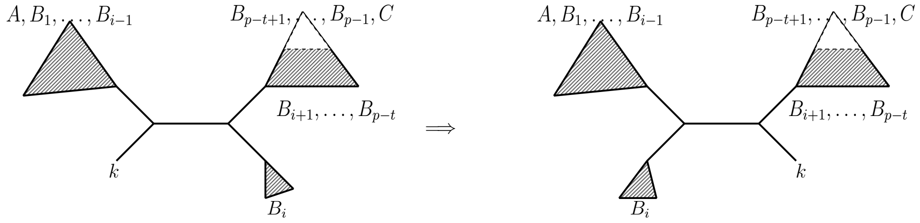

Figure 8: Transition from T_k^i to T_k^{i+1} under the split metric $D_k^{S_t}$.

The transform $T_k^i \to T_k^{i+1}$ is shown in Figure 8.

Proof. When we consider about δ^T function on $\Delta^{X|Y}$ distances, we can treat the monotonically colored subtree as a leaf in that color. Choose the two adjacent leaves x_1, x_2 in a monotonically black subtree X. Consider a tree T' obtained from T by deleting a subtree x_1, x_2 and inserting a new black leaf x_1' instead of x_1, x_2. Let $X' = (X \backslash \{x_1, x_2\}) \cup \{x_1'\}$ For any subtree Y being disjoint to X (Note that Y may be monotonically or not), we can obtain the following equations:

$$
\begin{aligned}
&\delta_{XY}^T = \sum_{i \in X, j \in Y} 2^{1-P_{ij}^{T_{XY}}} \Delta_{ij} \\
&= \sum_{\substack{i \in X \backslash \{x_1, x_2\}, \\ j \in Y}} 2^{1-P_{ij}^{T_{XY}}} \Delta_{ij} + \sum_{j \in Y} \left\{ 2^{1-P_{x_1 j}^{T_{XY}}} \Delta_{x_1 j} + 2^{1-P_{x_2 j}^{T_{XY}}} \Delta_{x_2 j} \right\} \\
&= \sum_{\substack{i \in X' \backslash \{x_1'\}, \\ j \in Y}} 2^{1-P_{ij}^{T'_{X'Y}}} \Delta_{ij} + \sum_{j \in Y} \left\{ 2^{1-(P_{x_1' j}^{T'_{X'Y}}+1)} \Delta_{x_1 j} + 2^{1-(P_{x_1' j}^{T'_{X'Y}}+1)} \Delta_{x_2 j} \right\} \\
&= \sum_{\substack{i \in X' \backslash \{x_1'\}, \\ j \in Y}} 2^{1-P_{ij}^{T'_{X'Y}}} \Delta_{ij} + \sum_{j \in Y} \left\{ 2^{1-P_{x_1' j}^{T'_{X'Y}}} \Delta_{x_1' j} \right\} \\
&= \sum_{i \in X', j \in Y} 2^{1-P_{ij}^{T'_{X'Y}}} \Delta_{ij} = \delta_{X'Y}^{T'}.
\end{aligned}
$$

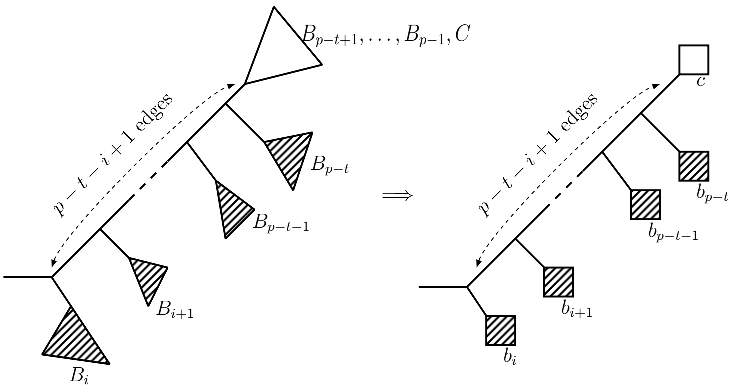

Figure 9: $B_i, B_{i+1}, \ldots, B_{p-1}, C$ in T_i

Since both x_1 and x_2 are colored in black, for any j in Y, $\Delta_{x_1 j} = \Delta_{x_2 j}$. Continue replacing new adjacent leaves in X' to a leaf until a set of leaves becomes a singleton set.

Now we consider (3):

$$\frac{1}{4}\left\{\delta^{T^i}_{(A,B_1,\ldots,B_{i-1})\{k^i\}} + \delta^{T^i}_{B_i(B_{i+1},\ldots,B_{p-1},C)} - \delta^{T^i}_{(A,B_1,\ldots,B_{i-1})B_i} - \delta^{T^i}_{\{k^i\}(B_{i+1},\ldots,B_{p-1},C)}\right\} = \frac{1}{4}\left\{1 + 2^{-p+i+t} - 1 + 2^{-p+i+t}\right\}.$$

Under the split distance S_t, the leaves in $A, B_1, \ldots, B_{p-t}$ is colored in black, and the leaves in $B_{p-t+1}, \ldots, B_{p-1}, C$ and the leaf k is colored in white. When disjoint subtrees X and Y are monotonically colored in black and white respectively, $\delta^T_{XY} = 1$. Both X and Y are monotonically colored in the same color, $\delta^T_{XY} = 0$. Note that $\delta^{T^i}_{(A,B_1,\ldots,B_{i-1})\{k^i\}}$ is the first case and $\delta^{T^i}_{(A,B_1,\ldots,B_{i-1})B_i}$ is the second one.

Let us consider $\delta^{T_i}_{B_i(B_{i+1},\ldots,B_{p-1},C)}$ and $\delta^{T^i}_{\{k^i\}(B_{i+1},\ldots,B_{p-1},C)}$ under the split distance S_t. Since the each subtree of $B_i, B_{i+1}, \ldots, B_{p-1}, C$ and $\{k\}$ is monotonically colored, we can treat them as leaves as shown in Figure 9. Say the white leaf made from $B_{p-t+1}, \ldots, B_{p-1}$ and C c, the black leaf made from B_j b_j for $j \in \{i, i+1, \ldots, p-t\}$. Now, $\delta^{T_i}_{B_i(B_{i+1},\ldots,B_{p-1},C)}$ is:

$$\delta^{T_i}_{B_i(B_{i+1},\ldots,B_{p-1},C)} = \delta^{T_i}_{\{b_i\}\{b_{i+1},\ldots,b_{p-t},c\}} = 2^{-(p-t-i)}.$$

$\delta^{T^i}_{\{k^i\}(B_{i+1},\ldots,B_{p-1},C)}$ is:

$$\delta^{T^i}_{\{k^i\}(B_{i+1},\ldots,B_{p-1},C)} = \delta^{T_i}_{\{k_i\}\{b_{i+1},\ldots,b_{p-t},c\}} = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots + \frac{1}{2^{p-t-i}} = 1 - 2^{-(p-t-i)}.$$

■

For $p-t+1 \leq i \leq p-1$, it is easy to see that:

$$\hat{l}(T_k^i, D_k^{S_t}) - \hat{l}(T_k^{i+1}, D_k^{S_t}) = 0.$$

Now we obtained

$$\hat{l}(T'_k, D_k^{S_t}) - \hat{l}(T^*_k, D_k^{S_t}) = \sum_{i=1}^{p-t} 2^{-p+i+t-1} = 1 - 2^{-(p-t)}.$$

Finally, the tree difference is

$$\begin{aligned}\hat{l}(T'_k, D_k^{T^*}) - \hat{l}(T^*_k, D_k^{T^*}) &= \sum_{t=1}^{p-1} l^{T^*_k}(S_t)\left[\hat{l}(T'_k, D_k^{S_t}) - \hat{l}(T^*_k, D_k^{S_t})\right] \\ &= \sum_{t=1}^{p-1} l^{T^*_k}(S_t)\left[1 - 2^{-(p-t)}\right] \geq l_{\min}^{T^*_k}\left(p - 2 + 2\cdot 2^{-p}\right).\end{aligned}$$

■

Proof of Theorem 1.

We obtained that for any topology $T'_k \neq T^*_k$, the difference of the estimated tree length is

$$\begin{aligned}&\hat{l}(T'_k, \Delta_k) - \hat{l}(T^*_k, \Delta_k) \\ &\quad \geq l_{\min}^{T^*_k}\left(p-2+2\cdot 2^{-p}\right) - 3\varepsilon_0 + 2\cdot 2^{-p}\varepsilon_0 + \begin{cases} 3\cdot 2^{-\frac{p}{2}}\varepsilon_0 & (p : \text{even}) \\ 2\cdot 2^{-\lfloor\frac{p}{2}\rfloor}\varepsilon_0 & (p : \text{odd})\end{cases}\end{aligned}$$

Under the condition $\varepsilon_0 = \left\|D^{T^*} - \Delta\right\|_\infty < \frac{1}{2}l_{\min}^{T^*} \leq \frac{1}{2}l_{\min}^{T^*_k}$, the estimated difference is larger than 0 for any $p \geq 2$. Note that if $p = 1$, then $T^*_k = T'_k$. ■

4 Robustness of GME

We consider a special case of input to prove theorem 2. If ε_0 exceeds α_G, there is an input for which GME outputs a wrong tree. Same as the proof of theorem 1, let T^*_k be a target tree in k-th round of GME, composed of subtrees A, B, C and a leaf k. Let T'_k be a tree obtained from T^*_k by swapping subtree A and leaf k. GME uses the following equation to compare tree lengths:

$$\begin{aligned}\hat{l}(T'_k, \Delta_k) - \hat{l}(T^*_k, \Delta_k) &= \frac{1}{2}\left[(\lambda - \lambda')\left(\Delta_{k|A} + \Delta_{B|C}\right)\right. \\ &\quad \left. + (\lambda' - 1)\left(\Delta_{A|B} + \Delta_{k|C}\right) + (1-\lambda)\left(\Delta_{A|C} + \Delta_{k|B}\right)\right],\end{aligned}$$

where $\lambda = \frac{(|A|+|B||C|)}{(|A|+|B|)(|C|+1)}$ and $\lambda' = \frac{(|A|+|B||C|)}{(|A|+|C|)(|B|+1)}$.

We obtained the following lemma by the same way as theorem 1.

Lemma 5. *Let e be an edge on T_k^* separating A and B from C and k.*

$$\hat{l}(T_k', \Delta_k) - \hat{l}(T_k^*, \Delta_k) = (1-\lambda')l^{T_k^*}(e) - \begin{cases} 2(1-\lambda')\varepsilon_0 & (|B| \geq |C|) \\ 2(1-\lambda)\varepsilon_0 & (|B| < |C|) \end{cases}$$

Proof. Let $E_{AB} = \frac{1}{|A||B|}\sum_{\substack{i\in A\\ j\in B}} \varepsilon_{ij}$. Note that $-\varepsilon_0 \leq E_{AB} \leq \varepsilon_0$. Let T_k^* and T_k' as shown in Figure 1 and let e be an edge providing a split $AB|C\{k\}$ on T_k^*. The tree difference is as follows:

$$\begin{aligned}
&l(T_k', D^{T_k^*}) - l(T_k^*, D^{T_k^*}) \\
&= \frac{1}{2}\left[(\lambda-\lambda')\left(\Delta_{k|A} + \Delta_{B|C}\right)\right. \\
&\qquad \left.+(\lambda'-1)\left(\Delta_{A|B} + \Delta_{k|C}\right) + (1-\lambda)\left(\Delta_{A|C} + \Delta_{k|B}\right)\right] \\
&= \sum_{X|Y\in S(T_k^*)} l^{T_k^*}(X|Y)\frac{1}{2}\left[(\lambda-\lambda')\left(\Delta_{k|A}^{X|Y} + \Delta_{B|C}^{X|Y}\right)\right. \\
&\qquad \left.+(\lambda'-1)\left(\Delta_{A|B}^{X|Y} + \Delta_{k|C}^{X|Y}\right) + (1-\lambda)\left(\Delta_{A|C}^{X|Y} + \Delta_{k|B}^{X|Y}\right)\right] \\
&= l^{T_k^*}(e)\frac{1}{2}\left[(\lambda-\lambda')(1+1)\right. \\
&\qquad \left.+(\lambda'-1)(0+0) + (1-\lambda)(1+1)\right] \\
&= (1-\lambda')l^{T_k^*}(e).
\end{aligned}$$

Now, the value $\Delta_{A|B}$ calculated with input matrix Δ, i.e. $\hat{\Delta}_{A|B} = \frac{1}{|A||B|}\sum_{i\in A, j\in B}\delta_{ij}$, satisfies:

$$\hat{\Delta}_{A|B} = \frac{1}{|A||B|}\sum_{i\in A, j\in B}\delta_{ij} = \frac{1}{|A||B|}\sum_{i\in A, j\in B}(d_{ij} + \varepsilon_{ij}) = \Delta_{A|B} + E_{A|B}.$$

The total difference is ,

$$\begin{aligned}
&\hat{l}(T_k', \Delta_k) - \hat{l}(T_k^*, \Delta_k) \\
&= \frac{1}{2}\left[(\lambda-\lambda')\left(\Delta_{k|A} + \Delta_{B|C} + E_{kA} + E_{BC}\right)\right. \\
&\qquad +(\lambda'-1)\left(\Delta_{A|B} + \Delta_{k|C} + E_{AB} + E_{kC}\right) \\
&\qquad \left.+(1-\lambda)\left(\Delta_{A|C} + \Delta_{k|B} + E_{AC} + E_{kB}\right)\right] \\
&= l(T_k', D^{T_k^*}) - l(T_k^*, D^{T_k^*}) + \frac{1}{2}\left[(\lambda-\lambda')(E_{kA} + E_{BC})\right. \\
&\qquad \left.+(\lambda'-1)(E_{AB} + E_{kC}) + (1-\lambda)(E_{AC} + E_{kB})\right] \\
&\geq l(T_k', D^{T_k^*}) - l(T_k^*, D^{T_k^*}) - \left[|\lambda-\lambda'| + |\lambda'-1| + |1-\lambda|\right]\varepsilon_0 \\
&= (1-\lambda')l^{T_k^*}(e) - \begin{cases} 2(1-\lambda')\varepsilon_0 & (|B| \geq |C| \iff \lambda \geq \lambda') \\ 2(1-\lambda)\varepsilon_0 & (|B| < |C| \iff \lambda < \lambda') \end{cases}
\end{aligned}$$

■

If $|B| \geq |C$ and $\varepsilon_0 < \frac{1}{2} l^{T_k^*}(e)$, GME chooses T_k^*. However, for A, B, C and ε_0 satisfying $|B| < |C|$ and $\varepsilon_0 > \frac{(1-\lambda')}{2(1-\lambda)} l^{T_k^*}(e)$, GME chooses T_k'. Consider when the leaf k is the last leaf. in this case, FASTNNI after GME cannot correct this fault.

Let $F(a, b, c)$ be:

$$F(a, b, c) = \frac{(1-\lambda')}{2(1-\lambda)} = \frac{(a+b)(c+1)(ab+c)}{(a+c)(b+1)(ac+b)}$$

where a, b and c are the cardinality of A, B and C respectively. By the definition of α_G, if $\varepsilon_0 < \alpha_G \min_{e \in E(T^*)} l^{T^*}(e)$ then GME succeeds to output T^*, we obtained $\alpha_G \leq \min_{b<c, a+b+c+1=n} F(a, b, c)$. We observed that $F(a, b, c)$ is minimized when $a = \sqrt{n}, b = \sqrt{n} - 4$ and $c = n - 2\sqrt{n} + 3$. Now theorem 2 is obtained:

$$\begin{aligned} \alpha_G \leq \min_{\substack{0<a, 0<b<c, \\ a+b+c+1=n}} F(a, b, c) &\leq F(\sqrt{n}, \sqrt{n} - 4, n - 2\sqrt{n} + 3) \\ &= \frac{2}{\sqrt{n}} - \frac{2}{n} - \frac{5}{n\sqrt{n}} + \frac{12}{n^2} + \ldots < \frac{2}{\sqrt{n}}. \end{aligned}$$

5 Concluding Remarks

We proved BME has l_∞ radius $\frac{1}{2}$ and GME has l_∞ radius less than $\frac{2}{\sqrt{n}}$. Desper and Gascuel presented BNNI and FASTNNI to modify an initial tree to the minimal one. However, if $\varepsilon_0 < \frac{1}{2} \min_{e \in E(T^*)} l(e)$, we does not need to use BNNI. If $\varepsilon_0 > \frac{1-\lambda'}{2(1-\lambda)} \min_{e \in E(T^*)} l(e)$, and GME fails to put the last leaf, FASTNNI cannot correct such fault.

References

[1] R. Desper, O. Gascuel. Theoretical Foundation of the Balanced Minimum Evolution Method of Phylogenetic Inference and Its Relationship to Weighted Least-Squares Tree Fitting. *J. of Mol. Biol. Evol.* 2004 21(3): 587–598.

[2] R. Desper, R. Gascuel. Fast and accurate phylogeny reconstruction algorithms based on the minimum evolution principle. *J. Comp. Biol.* 2002 9: 687–705.

[3] Y. Pauplin. Direct Calculation of a Tree Length Using a Distance Matrix. *J Mol Evol.* 2000 51(1): 41–7.

[4] N. Saitou, M. Nei. The Neighbor-Joining Method: a New Method for Reconstructing Phylogenetic Trees. *Mol Biol Evol.* 1987 4(4): 406–25.

[5] O. Gascuel. BIONJ: an Improved Version of the NJ Algorithm Based on a Simple Model of Sequence Data. *Mol Biol Evol.* 1997 14(7):685–95.

[6] Bruno WJ, Socci ND, Halpern AL. Weighted Neighbor Joining: a Likelihood-Based Approach to Distance-Based Phylogeny Reconstruction. *Mol Biol Evol.* 2000 17(1):189–97.

[7] K. Atteson. The Performance of Neighbor-Joining Algorithms of Phylogeny Reconstruction. *Proceedings of the Third Annual International Conference on Computing and Combinatorics, Lecture Notes In Computer Science Vol. 1276* 1997 :101 – 110.

[8] W. Day, D. Sankoff. Computational Complexity of Inferring Phylogenies by Compatibility. *Systematic Zoology*,1986 35(2):224–229.

[9] J. Felsenstein. An Alternating Least Squares Approach to Inferring Phylogenies from Pairwise Distances *Systematic Biology*,1997 46(1):101–111.

[10] D. Benedetto, E. Caglioti and V. Loreto. Language Trees and Zipping. *Physical Review Letters* 2002 88(4):048702-1–04872-4.

[11] R. Cilibrasi, P. Vitányi. Clustering by Compression. *IEEE Transactions on Information Theory*, 2005 51(4):1523–1545.