

# Efficient Algorithms for Inverting Evolution

Martin Farach\*

Sampath Kannan†

January 11, 1999

## Abstract

Evolution can be mathematically modelled by a stochastic process that operates on the DNA of species. Such models are based on the established theory that the DNA sequences, or genomes, of all extant species have been derived from the genome of the common ancestor of all species by a process of random mutation and natural selection.

A stochastic model of evolution can be used to construct phylogenies, or evolutionary trees, for a set of species. Maximum Likelihood Estimations (MLE) methods seek the evolutionary tree which is most likely to have produced the DNA under consideration. While these methods are intellectually satisfying, they have not been widely accepted because of their computational intractability.

In this paper, we address the intractability of MLE methods as follows. We introduce a metric on stochastic process models of evolution. We show that this metric is meaningful by proving that in order for any algorithm to distinguish between two stochastic models that are close according to this metric, it needs to be given many observations. We complement this result with a simple and efficient algorithm for inverting the stochastic process of evolution, that is, for building a tree from observations on two-state characters. (We have used the same techniques in a subsequent paper to solve the problem for multistate characters, and hence for building a tree from DNA sequence data.) The tree we build is provably close, in our metric, to the tree generating the data and gets closer as more observations become available.

Though there have been many heuristics suggested for the problem of finding good approximations to the most likely tree, our algorithm is the first one with a guaranteed convergence rate, and further, this rate is within a polynomial of the lower-bound rate we establish. Ours is also the first polynomial-time algorithm which is *proven* to converge at all to the correct tree.

---

\*Rutgers University; [farach@cs.rutgers.edu](mailto:farach@cs.rutgers.edu); <http://www.cs.rutgers.edu/~farach>; Supported by an NSF Career Advancement Award and an Alfred P. Sloan Research Fellowship.

†University of Pennsylvania; [kannan@central.cis.upenn.edu](mailto:kannan@central.cis.upenn.edu); <http://www.cis.upenn.edu/~kannan/home.html>; Supported by NSF CCR 96-19910 and NSF SGER 9612829

# 1 Introduction

The evolutionary history of a set of species is modeled as a tree, called a *phylogeny*, whose leaves are bijectively labeled by the species. Reconstructing the phylogeny for a set of species is one of the fundamental problems of computational biology, for which a large set of methods exists (see [8, 9, 16] for surveys). Since the early 80’s –with the advent of rapid sequencing technology– the data for phylogeny construction methods has primarily been biomolecular sequences such as DNA.

The first step in processing sequence data for a set of species is to identify positions in each of the sequences that are believed to have a common origin. This is done by an algorithm for multiple sequence alignment. After the alignment step the sequence data can be thought of as a matrix with rows indexed by the species and columns indexed by aligned positions in the sequence.

The phylogeny construction problem can then be stated, somewhat imprecisely, as follows:

**Input:** Matrix  $M[i, j]$  where row  $i$  corresponds to the sequence of species  $i$ ’s sequence on a gene.

**Output:** A phylogeny which best *fits*  $M$ .

The above problem can be viewed as the problem of inverting a time-dependent process of evolution. Thus in order to precisely describe the degree of fit between a phylogeny and the given data we need a model for the forward direction, i.e., for evolution.

In the next subsection we describe a statistical model for evolution. The goodness of fit of a tree to the given data will be defined in terms of the model.

## 1.1 A Stochastic Model of Evolution

In this paper, we consider the most widely studied class of stochastic models for this problem. Each model in this class can be represented by a weighted tree. We will call a tree representing a model a *Cavender-Farris Tree*, named after the biologists who first proposed this class of models [3, 7]. We assume that we have two-state characters with states 0 and 1.

A Cavender-Farris tree (CFT) is a rooted tree where the root has a probability  $P_r \in [0, 1]$  and each edge  $e$  has some probability  $P_e \in (0, 1/2)$ . We interpret an  $n$  leaf CFT as a random source of vectors from  $\{0, 1\}^n$  (referred to in the literature as *patterns*) as follows. The root gets labeled ‘1’ with probability  $P_r$ . When a node gets labeled with a bit, it broadcasts that bit to all of its children. If a bit  $b$  is being broadcast down edge  $e = (u, v)$ , then  $v$  gets labeled  $\bar{b}$  with probability  $P_e$ , and  $b$  with probability  $1 - P_e$ . Thus each edge has some probability of producing an observable mutation. The vector of labels at the leaves is taken to be the output of a CFT.

Now, any  $\{0, 1\}^n$  vector has some positive probability of being produced as the output of any CFT. Let  $S$  be a CFT. Then we define the *output distribution of  $S$* , denoted  $\mathcal{P}_S$ , as the probability distribution on  $\{0, 1\}^n$  such that for any  $\vec{x} \in \{0, 1\}^n$ ,  $\mathcal{P}_S(\vec{x})$  is the probability of seeing  $\vec{x}$  as the output of  $S$ . Consider Figure 1. Fixing  $P_u = 1$ , the probability of the vector 010 at the leaves is the sum of two terms for the two choices for the state of  $v$ . This probability is equal to  $(1 - p)(1 - q)r(1 - s) + pq(1 - r)(1 - s)$ .

CFTs have the following straightforward reinterpretation as Poisson processes. Imagine a Poisson process with rate  $\lambda$  where the events are the changes of state of a character. This process

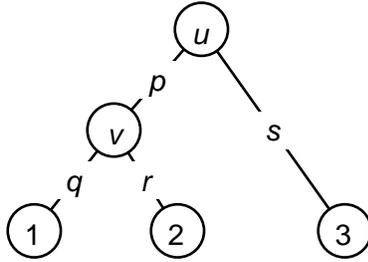


Figure 1: Tree describing a stochastic process. Labels on the edges represent probabilities of change between the endpoints.

proceeds along each edge  $e$  for time  $w(e)$ . The probability of observing a change of state between the end-points of  $e$  is  $P_e$  and is equal to the probability that an odd number of events occur in time  $w(e)$ . Normalizing by choosing  $\lambda = 1/2$ , we find that  $P_e = (1 - e^{-w(e)})/2$  and  $w(e) = -\ln(1 - 2P_e)$ .

Note that we are not claiming that evolution proceeds by a Poisson process with a fixed rate constant. We are merely stating that CFTs can be so interpreted. Evolution may proceed at different actual rates in different parts of the CFT but we simply think of the times on each edge as being scaled to make the rates uniform throughout the tree. However, since our input and output trees are in the probability domain, the scaling is purely conceptual and internal to our algorithm. We do assume that evolution proceeds at the same “rate” for each position of the DNA, that is, each character column is produced by the same CFT. The mapping between the time domain  $w(e)$  above and observable changes in the DNA will be crucial to our algorithm. This mapping is the only place in the algorithm which needs modification to deal with four state data.

The output of a CFT is a vector in  $\{0, 1\}^n$  which can be viewed as a character assigning a 0 or 1 state to each of the leaves. We assume that we observe  $k$  such vectors (which we also call characters or samples or observations) from  $k$  independent runs of the same CFT.

The model we have described is substantially similar to the model discussed by Felsenstein [8] and by Steel, Hendy, and Penny [14], and is the basis for most MLE methods for tree construction. The assumption in this model that the characters are generated by *i.i.d.* processes is questionable from the biological point of view, but is a simplifying assumption that is made in much of the literature on maximum likelihood methods. See [13, Ch 3 and 5] for discussions of biological conditions under which these assumptions might be appropriate. Given this model the computational problem can be described as below.

**Problem:** MLE construction of evolutionary trees

**Input:** Sample  $n$ -vectors (referred to as points)  $M = \langle \vec{c}_1, \dots, \vec{c}_k \rangle$  generated from some unknown  $n$ -leaf CFT  $S$ . This  $M$  is the same as the  $M$  in the phylogeny construction problem described in the introduction with  $n$  being the number of species and  $k$  being the number of (binary) characters.

**Output:** The CFT  $\hat{S}$  which is most likely to have generated the  $k$  sample points.

A comment is in order about the desired output in the problem above. Although the desired maximization criterion is  $\Pr[\hat{S}|M]$  we make the assumption of a uniform distribution on the prior probabilities for the models and seek instead to maximize probability  $\Pr[M|\hat{S}]$ . This is common in many maximum likelihood estimation problems and, in particular, is standard in phylogeny construction.

The state of knowledge with respect to computing the most likely tree is extremely primitive. Kashyap and Subas [10] describe a method for finding the parameters of the most likely tree on 3 species and then describe a heuristic for extending this to more than 3 species. Felsenstein [8] describes methods for finding parameters for a fixed topology as well as a computationally expensive heuristic for finding the most likely tree. Steel, Hendy, and Penny [14] view the problem as an inverse problem and show that an exponential-sized Hadamard matrix can be used to find the least-squares-best-fit tree.

It is unreasonable to expect that any algorithm for maximum likelihood estimation can reconstruct the original CFT  $S$  exactly from a finite amount of data. Thus we would like algorithms that “converge” to  $S$  as more and more samples are observed. To make the notion of convergence precise we need a metric on CFTs, which we introduce later in this paper.

There has been significant effort expended by computational biologist in coming up with heuristics for this problem. In this paper, we present the first known polynomial time algorithm that converges to the true tree at a rate that is at most a polynomial factor slower than the best possible rate that we establish. Since the results in this paper first appeared [6], there has been progress in showing that other methods converge. We discuss these recent results in Section 4

One further point about our algorithm is the following: We assume that the data we are given is produced by a CFT that we call the *true tree*. We will actually prove that the tree produced by our algorithm converges to the true tree as more and more observations are available. It has been shown that maximum likelihood methods are *consistent* and therefore the most likely tree also converges to the true tree<sup>1</sup>. Thus the output of our algorithm also converges to the most likely tree, but the main point is that it converges to the true tree.

Our result can be viewed in the setting of PAC-learning of distributions introduced by [11]. Viewing  $S$  as the target concept, we can show that polynomially many samples are sufficient to produce a hypothesis  $\hat{S}$  from the class of CFTs that is “close” to  $S$  with high probability.

The rest of this paper is organized as follows. In Section 2 we set up the metric on CFTs and show a bound on the maximum possible convergence rate for any algorithm, regardless of its computational complexity. In Section 3 we present the algorithm for maximum likelihood estimation and its analysis. Finally, in Section 4 we discuss implications of this work, recent results and directions for future work.

## 2 A metric on Cavender-Farris Trees

We are interested in converging on the CFT  $S$  via some algorithm on  $k$  sample points  $\vec{c}_1, \dots, \vec{c}_k$  drawn from the distribution  $\mathcal{P}_S$ . This suggests that we should measure the distance between two CFTs by the difference in their output distributions. While this makes common sense, it is a radical departure within the area of phylogeny construction, where the *topology* is the *sine qua non* for

---

<sup>1</sup>However, as stated in the abstract no previous algorithm has been analyzed to converge to the most likely tree.

biologists. Biologists ultimately want to know the actual relationships amongst the input species. Nonetheless, we will justify this approach by showing a lower bound on the learnability of CFTs in terms of their distance under the following function.

**Definition:** Let  $S_0$  and  $S_1$  be two CFTs on the same  $n$  leaves. Then we define the *var-distance*,  $V(S_0, S_1)$ , as the variational distance  $Var(\mathcal{P}_{S_0}, \mathcal{P}_{S_1}) = \sum_{\vec{x} \in \{0,1\}^n} |\mathcal{P}_{S_0}(\vec{x}) - \mathcal{P}_{S_1}(\vec{x})|$ .

As noted above, we must justify the use of this measure of distance between trees. Further, we must show that this function is well behaved. For example, while  $V$  is symmetric and satisfies triangle inequality, it is not obvious that it is a metric, since there may be  $S_1 \neq S_2$  such that  $V(S_1, S_2) = 0$ . In particular, if we allowed edge probabilities to range in  $(0, 1/2]$  instead of  $(0, 1/2)$ , the variational distance of CFTs would not be a metric. Note that an edge probability of  $1/2$  on an edge  $(u, v)$  corresponds to a situation where the states of  $v$  are completely uncorrelated with the states of  $u$ , a biologically unrealistic situation.

In the next two subsections we prove the following.

1. That there is a lower bound on the learnability of CFTs in terms of their  $V$  distance.
2. That  $V$  is a metric on CFTs.

## 2.1 An Information-Theoretic Lower Bound

In this section we will show that CFTs that are close in var-distance cannot be distinguished based on a small number of observations by any method, no matter how computationally expensive.

Suppose  $S_0$  and  $S_1$  are two CFTs such that  $V(S_0, S_1) \leq \epsilon$ . Let  $A$  be any decision procedure that is given  $k$  samples from one of these two CFTs and decides whether the samples were drawn from  $S_0$  or  $S_1$ . Without loss of generality, suppose that  $A$  outputs 0 if it decides that the samples were drawn from  $S_0$  and 1 if it guesses that the samples were drawn from  $S_1$ .

There are two kinds of errors that are made by  $A$ . Let  $e_0(A)$  be the probability that  $A$  outputs 1 when the samples are drawn from  $S_0$  and  $e_1(A)$  be the probability that  $A$  outputs 0 when the samples are drawn from  $S_1$ . Let  $e(A) = \max(e_0(A), e_1(A))$ .

**Lemma 1** For  $S_0, S_1$  and  $A$  as above,  $e(A) \geq \frac{1-k\epsilon}{2}$ .

**Proof:** We first prove a general statement. Let  $D_0$  and  $D_1$  be any two distributions such that  $Var(D_0, D_1) \leq \delta$ . Suppose a decision procedure  $A$  is given one sample from either  $D_0$  or  $D_1$  and is asked to decide on which distribution the sample came from. Let  $e_0(A), e_1(A), e(A)$  be defined as above.

We claim that  $e(A) \geq (1 - \delta)/2$ . To see this, suppose  $P$  is the set of points on which  $A$  outputs 1. Then  $e_0(A) + e_1(A) = \Pr[P|D_0] + 1 - \Pr[P|D_1] \geq 1 - \delta$ . Since  $e(A) \geq (e_0(A) + e_1(A))/2$  the claim follows.

By our assumptions  $Var(\mathcal{P}_{S_0}, \mathcal{P}_{S_1}) \leq \epsilon$ . We observe  $k$  samples from one of  $S_0$  or  $S_1$ . We will view these  $k$  samples as one sample from an appropriate Cartesian product of distributions. To this end, let  $D^{\otimes k}$  denote the  $k$ -fold cross-product of the distribution  $D$ . Thus  $D^{\otimes k}(\vec{x}_1, \dots, \vec{x}_k) = \prod_{i=1}^k D(\vec{x}_i)$ .

We can show that if  $\text{Var}(\mathcal{P}_{S_0}, \mathcal{P}_{S_1}) \leq \epsilon$ , then  $\text{Var}(\mathcal{P}_{S_0}^{\otimes k}, \mathcal{P}_{S_1}^{\otimes k}) \leq k\epsilon$ , as follows

$$\begin{aligned}
\text{Var}(\mathcal{P}_{S_0}^{\otimes k}, \mathcal{P}_{S_1}^{\otimes k}) &= \sum_{\vec{x}_1, \dots, \vec{x}_k \in (\{0,1\}^n)^k} \left| \prod_{i=1}^k \mathcal{P}_{S_0}(\vec{x}_i) - \prod_{i=1}^k \mathcal{P}_{S_1}(\vec{x}_i) \right| \\
&\leq \sum_{\vec{x}_1 \in \{0,1\}^n} |\mathcal{P}_{S_0}(\vec{x}_1) - \mathcal{P}_{S_1}(\vec{x}_1)| \left( \sum_{\vec{x}_2, \dots, \vec{x}_k} \prod_{i=2}^k \mathcal{P}_{S_0}(\vec{x}_i) \right) \\
&\quad + \sum_{\vec{x}_2, \dots, \vec{x}_k \in (\{0,1\}^n)^{k-1}} \left| \prod_{i=2}^k \mathcal{P}_{S_0}(\vec{x}_i) - \prod_{i=2}^k \mathcal{P}_{S_1}(\vec{x}_i) \right| \left( \sum_{\vec{x}_1 \in \{0,1\}^n} \mathcal{P}_{S_1}(\vec{x}_1) \right) \\
&= \sum_{\vec{x}_1 \in \{0,1\}^n} |\mathcal{P}_{S_0}(\vec{x}_1) - \mathcal{P}_{S_1}(\vec{x}_1)| + \sum_{\vec{x}_2, \dots, \vec{x}_k \in (\{0,1\}^n)^{k-1}} \left| \prod_{i=2}^k \mathcal{P}_{S_0}(\vec{x}_i) - \prod_{i=2}^k \mathcal{P}_{S_1}(\vec{x}_i) \right| \\
&= \text{Var}(\mathcal{P}_{S_0}, \mathcal{P}_{S_1}) + \text{Var}(\mathcal{P}_{S_0}^{\otimes k-1}, \mathcal{P}_{S_1}^{\otimes k-1})
\end{aligned} \tag{1}$$

By induction the last expression in Eq. 3 can be seen to be no greater than  $k\epsilon$ . The lemma follows from this fact and the claim proved above.  $\blacksquare$

## 2.2 Proof of Metricity

We prove that  $V$  is well behaved by showing that  $\mathcal{P}_S$  determines  $S$ , as follows.

**Lemma 2** *If  $S_1$  and  $S_2$  are distinct CFTs, then  $\mathcal{P}_{S_1} \neq \mathcal{P}_{S_2}$ . Furthermore, we can reconstruct  $S_1$  from  $\mathcal{P}_{S_1}$ .*

**Proof:** We define  $\mathcal{D}_S$  to be the *difference metric* of  $S$ , that is, we take  $\mathcal{D}_S(i, j)$  to be the probability that leaves  $i$  and  $j$  have different bits. Given  $\mathcal{P}_S$ , we can compute  $\mathcal{D}_S$  as

$$\mathcal{D}_S(i, j) = \sum_{\vec{x} \in \{0,1\}^n, \vec{x}_i \neq \vec{x}_j} \mathcal{P}_S(\vec{x}).$$

Now, we define  $\mathcal{T}_S$ , the *time metric* of  $S$ , as

$$\mathcal{T}_S(i, j) = -\ln(1 - 2\mathcal{D}_S(i, j)).$$

**Claim 2.1**  *$\mathcal{T}_S$  is an additive metric, that is, it is a path metric on a tree.*

**Proof:** Let  $W$  be the weighted tree isomorphic to  $S$  such that if an edge  $e$  has change probability  $P_e$  in  $S$ , then its image has weight  $w_e = -\ln(1 - 2P_e)$  in  $W$ . Then consider any two leaves  $i$  and  $j$  which are connected by a two-edge path, and let  $v$  be the node between them. Now  $\mathcal{D}_S(i, j) = P_{(i,v)} + P_{(v,j)} - 2P_{(i,v)}P_{(v,j)}$  and so  $\mathcal{T}_S(i, j) = -\ln(1 - 2(P_{(i,v)} + P_{(v,j)} - 2P_{(i,v)}P_{(v,j)})) = -\ln(1 - 2P_{(i,v)}) - \ln(1 - 2P_{(v,j)}) = w_{(i,v)} + w_{(v,j)}$ . By induction, for any pair of leaves  $i, j$ ,  $\mathcal{T}_S(i, j)$  is the sum of the weights of the edges on the path between them in  $W$ , thus proving the claim.  $\blacksquare$

We can conclude from the claim that  $W$  is unique and efficiently constructible from  $\mathcal{T}_S$ . Furthermore, we can reconstruct  $S$  from  $W$  by noting that  $P_e = (1 - e^{-w_e})/2$ . This establishes the lemma.  $\blacksquare$

A similar lemma is proved in [14].

### 3 A Simple Distance-Based Method and Its Analysis

Lemma 2 suggests the following algorithm. Let  $H(\vec{x}_1, \vec{x}_2)$  be the Hamming distance between vectors  $\vec{x}_1$  and  $\vec{x}_2$ .

**Algorithm Outline:**

1. Input  $M = \begin{pmatrix} \vec{s}_1 \\ \vdots \\ \vec{s}_n \end{pmatrix} = (\vec{c}_1, \dots, \vec{c}_k)$ .
2. For all  $i, j$ , set  $\mathcal{D}^*(i, j) = H(s_i, s_j)/k$ , that is,  $\mathcal{D}^*$  is our estimate of  $\mathcal{D}_S$ . In the analysis below, we will estimate  $\mathcal{E}(i, j)$  which represents the tolerance in the value  $\mathcal{D}^*(i, j)$ . In other words,  $\mathcal{D}_S(i, j)$  will be in  $\mathcal{D}^*(i, j) \pm \mathcal{E}(i, j)$  with high probability.
3. For all  $i, j$ , set  $\mathcal{T}^*(i, j) = -\ln(1 - 2\mathcal{D}^*(i, j))$ . Then  $\mathcal{T}^*$  is our estimate of  $\mathcal{T}_S$ . We will make  $\mathcal{E}(i, j)$  small enough that with high probability  $\mathcal{T}_S(i, j)$  lies in  $\mathcal{T}^*(i, j) \pm \delta$  for all  $i, j$ , for some constant  $\delta > 0$ .
4. While we know that  $\mathcal{T}^*$  is close to being additive – we will show that as  $k$  increases  $\mathcal{T}^*$  approaches  $\mathcal{T}_S$  – it need not be exactly additive. Use an additive fitting algorithm [1] which produces a tree close to  $\mathcal{T}^*$ . Call this tree  $\hat{\mathcal{T}}$ . Invert the edge weights back into probabilities to produce  $\hat{\mathcal{S}}$ . Output  $\hat{\mathcal{S}}$ .

#### 3.1 Goodness of Estimates

In the following we will assume that  $\mathcal{D}_S(i, j)$  is bounded away from  $1/2$  for all pairs of species  $i, j$ . This is because as  $\mathcal{D}_S(i, j)$  approaches  $1/2$ ,  $\mathcal{T}_S(i, j)$  approaches  $\infty$  and small tolerances in the probability estimates blow-up into large tolerances in the distance estimates. For most biological data sets this is a very reasonable assumption. The situation where  $\mathcal{D}_S(i, j) \geq 1/2$  corresponds to a situation where there is no correlation or even negative correlation between species  $i$  and  $j$  which is a biological absurdity. Formally, we assume that there is a constant  $\alpha > 0$  such that  $\mathcal{D}_S(i, j) \leq 1/2 - \alpha$  for all  $i, j$ .

We would like to estimate  $\mathcal{D}_S(i, j)$  well enough to ensure that when these estimates are translated into the time domain we can assert that  $\mathcal{T}^*(i, j)$  is at most  $\delta$  away from  $\mathcal{T}_S(i, j)$  for all  $i, j$  with high probability. Given the nature of the function that is used to map from the probability domain to the time domain, the tightest constraints on probability estimation arise for  $\mathcal{D}^*(i, j) = 1/2 - \alpha$ .

Let  $p = 1/2 - \alpha$  and let  $t = t(\alpha, \delta)$  be the allowable tolerance in the value of  $p$ . We can solve for  $t$  from the equation

$$\delta = -\ln(1 - 2(p + t)) + \ln(1 - 2p)$$

Thus  $t$  is a constant that is dependent only on  $\alpha$  and  $\delta$ . (The exact choice for the value of  $t$  is made later in the proof of Theorem 1.) If we ensure that  $\mathcal{E}(i, j) \leq t \forall i, j$  then we get the required uniform tolerances of  $\delta$  around each time estimate.

A simple appeal to the Chernoff bound tells us the number of samples required to make  $\mathcal{E}(i, j)$  small enough.

**Lemma 3** If  $k = 6 \ln n / t^2$  samples are observed then for any given  $i, j$ ,  $\Pr[\mathcal{E}(i, j) > t] \leq n^{-4}$ .

**Proof:** The proof follows from a direct application of the Chernoff bound inequalities [12]. As a corollary the probability that there exist  $i$  and  $j$  such that  $\mathcal{E}(i, j) > t$ , is no more than  $n^{-2}$ . ■

Because there is a deterministic function mapping probabilities to times the following lemma is immediate.

**Lemma 4** With probability at least  $1 - n^{-2}$ ,  $|\mathcal{T}_S(i, j) - \mathcal{T}^*(i, j)| \leq \delta$  for all  $i, j$ .

The next stage involves application of an additive-fitting method taken from [1]. We now describe the problem considered by [1] and the result they obtain.

**Problem:**  $L_\infty$ -best-fit additive tree.

**Input:** An  $n \times n$  matrix  $\mathcal{T}^*$  representing pairwise distance estimates among  $n$  points.

**Output:** An edge-weighted tree  $T$  such that  $\max_{i,j} |T[i, j] - \mathcal{T}^*[i, j]|$  is minimized, where  $T[i, j]$  represents the distance in the tree between leaves representing species  $i$  and  $j$ .

Let  $\epsilon = \min_T \max_{i,j} |T[i, j] - \mathcal{T}^*[i, j]|$ . In [1] it is shown that finding a tree  $T$  whose  $L_\infty$  distance from the given matrix is  $\epsilon$  is NP-hard. Fortunately, it is also shown that there is an efficient algorithm that finds a tree  $T'$  such that  $\max_{i,j} |T'[i, j] - \mathcal{T}^*[i, j]| \leq 3\epsilon$ .

In our case, we know that with probability greater than  $1 - n^{-2}$  there is a tree within  $\delta$  of  $\mathcal{T}^*$ . Thus the algorithm of [1] will find a tree within  $3\delta$  of the given estimates. We apply the algorithm to compute the tree  $\hat{T}$ . Finally we convert  $\hat{T}$  into  $\hat{S}$ .

### 3.2 Analysis of Algorithm

Our goal is to show that the variational distance between  $\hat{S}$  and  $S$  is small. (We will sometimes prove statements about the equivalent time domain trees –  $\hat{T}$  and  $\mathcal{T}_S$  – but these statements translate directly into the probability domain.) We need some terminology. For any edge  $e$  in a rooted tree the set of leaves in the subtree that lies below  $e$  will be denoted  $B(e)$ .

**Definition:** Given two rooted (edge-weighted) trees  $T_1$  and  $T_2$ , the *homeomorphic collapse*,  $hc(T_1 < T_2)$  is obtained from  $T_1$  by collapsing every edge  $e \in T_1$  such that  $B(e) \neq B(e')$  for any edge  $e'$  in  $T_2$ .

Note that  $hc(T_1 < T_2)$  and  $hc(T_2 < T_1)$  are isomorphic in their topologies but may differ in edge weights. We first make the following simple observation.

**Observation 1 :**

$$V(T_1, T_2) \leq V(T_1, hc(T_1 < T_2)) + V(hc(T_1 < T_2), hc(T_2 < T_1)) + V(hc(T_2 < T_1), T_2).$$

The observation follows simply from the fact that the variational distance defines a metric. However, it serves an important purpose. We can view  $hc(T_1 < T_2)$  and  $hc(T_2 < T_1)$  as being simultaneously isomorphic in topology to both  $T_1$  and  $T_2$ . Thus all variational distances on the right hand side of the above inequality can be thought of as being computed between isomorphic trees. This fact will be used in the argument that follows.

### 3.3 $L_\infty$ Closeness Implies $V$ Closeness

Suppose that  $S$  and  $S'$  are CFTs such that  $L_\infty(\mathcal{D}_S, \mathcal{D}_{S'})$  is “small.” In this section, we will prove that  $V(S, S')$  is correspondingly small. We do so in three steps. In the first two lemmas, we prove the bound for the case when  $S$  and  $S'$  have isomorphic topologies. Finally, we show how to apply these lemmas to the non-isomorphic case by showing that  $S$  and  $S'$  are *almost isomorphic*, in some appropriate sense.

**Lemma 5** *Suppose  $S$  and  $S'$  are two isomorphic CFTs on the leaf set  $[1, \dots, n]$ . Suppose that every pair of corresponding edges  $e \in S$  and  $e' \in S'$  is such that  $|P_e - P_{e'}| \leq \epsilon$  for some  $\epsilon > 0$ . Then  $V(S, S') \leq 2n\epsilon$ .*

**Proof:** The proof is recursive and is based on a “coupling” argument where the process represented by  $S$  and by  $S'$  are coupled as much as their parameters allow. Coupling is a technique where two stochastic processes are made to be highly correlated while still preserving the property that each process satisfies the parameters of its own definition.

We first introduce some notation. Assume that  $S$  and  $S'$  are binary trees. (The proof can easily be extended to the non-binary case.) Renumber the leaves of  $S$  and  $S'$  so that the left-to-right ordering of the leaves is the same as the increasing order of numbers. Let  $r$  be the root of  $S$  and  $x$  and  $y$  be its children. Let  $r', x'$ , and  $y'$  represent the corresponding nodes in  $S'$ . For any node  $v$  let  $S_v$  represent the subtree rooted at  $v$ . Let the number of leaves in  $S_x$  (which is equal to the number of leaves in  $S_{x'}$ ) be  $k$ . Let  $u$  be a string in  $\{0, 1\}^k$  and  $v$  be a string in  $\{0, 1\}^{n-k}$ .

The use of coupling here is as follows. Let  $e$  and  $e'$  be corresponding edges in  $S$  and  $S'$  with  $P_{e'} \geq P_e$  without loss of generality. Setting  $0 \leq \rho := P_{e'} - P_e \leq \epsilon$ , we roll a 3-sided die with probabilities  $P_e$ ,  $\rho$ , and  $1 - P_e - \rho$  for the 3 sides. If the first side shows up, then changes are made on both  $e$  and  $e'$ ; if the second side shows up then a change is made only along  $e'$  and if the third side shows up no change is made along either edge.

By the notation  $\Pr[uv|r_i]$  we will mean the probability that the tree  $S$  (the tree is identified by the node  $r$ ) generates the string  $uv$  at its leaves given that  $r$  is in state  $i$ .

Assume that  $p$  is the probability that the state at  $r$  (and at  $r'$ ) is 1. By considering the coupled processes we assume that both root states are always identical. The variational distance between two trees is unaffected if both root states are fixed at 0 or if both root states are fixed at 1. Thus  $V(S, S') \leq \sum_{uv \in \{0,1\}^n} |\Pr[uv|r_0] - \Pr[uv|r'_0]|$ . By using the independence of the CFT process along two branches once the root value has been fixed, we derive the sequence of relations as follows

$$\begin{aligned}
 V(S, S') &\leq \sum_{uv \in \{0,1\}^n} |\Pr[uv|r_0] - \Pr[uv|r'_0]| \\
 &= \sum_{uv} |(\Pr[u|x_0] \Pr[x_0|r_0] + \Pr[u|x_1] \Pr[x_1|r_0])(\Pr[v|y_0] \Pr[y_0|r_0] + \Pr[v|y_1] \Pr[y_1|r_0]) \\
 &\quad - (\Pr[u|x'_0] \Pr[x'_0|r'_0] + \Pr[u|x'_1] \Pr[x'_1|r'_0])(\Pr[v|y'_0] \Pr[y'_0|r'_0] + \Pr[v|y'_1] \Pr[y'_1|r'_0])| \\
 &\leq 4\epsilon + \sum_{uv} |(\Pr[u|x_0] \Pr[v|y_0]) - (\Pr[u|x'_0] \Pr[v|y'_0])|
 \end{aligned}$$

This last inequality is obtained by observing that there is a probability of at most  $2\epsilon$  that either  $x$  and  $x'$  or  $y$  and  $y'$  have different states in the coupled processes in the two trees. Even when there

is such a difference of state the conditional variational distance can be less than the maximum value of 2, but we conservatively assume that it is in fact 2. In the case where the coupled processes produce the same states at  $x$  and  $x'$  and at  $y$  and  $y'$ , we can assume without loss of generality that this state is 0 for the same reasons that we were able to assume that the roots had state 0.

Continuing the above derivation we have

$$\begin{aligned}
V(S, S') &\leq 4\epsilon + \sum_{uv} |\Pr[u|x_0] \Pr[v|y_0] - \Pr[u|x'_0] \Pr[v|y'_0]| \\
&= 4\epsilon + \sum_{uv} |\Pr[u|x_0] \Pr[v|y_0] - \Pr[u|x'_0] \Pr[v|y_0] + \Pr[u|x'_0] \Pr[v|y_0] - \Pr[u|x'_0] \Pr[v|y'_0]| \\
&\leq 4\epsilon + \sum_{uv} \Pr[u|x_0] (|\Pr[v|y_0] - \Pr[v|y'_0]|) + \sum_{uv} \Pr[v|y'_0] (|\Pr[u|x_0] - \Pr[u|x'_0]|) \\
&\leq 4\epsilon + \sum_v |\Pr[v|y_0] - \Pr[v|y'_0]| + \sum_u |\Pr[u|x_0] - \Pr[u|x'_0]| \\
&\leq 4\epsilon + V(S_x, S_{x'}) + V(S_y, S_{y'})
\end{aligned}$$

and, by using inductive assumptions about the subtrees rooted at the children of the roots, we get the required result.  $\blacksquare$

The next two lemmas are proved in the time domain but apply by the standard transformation ( $p_{ij} = (1 - e^{-t_{ij}})/2$ ) in the probability domain.

**Lemma 6** *If  $T_1$  and  $T_2$  are isomorphic in topology, each internal node in each tree has degree at least 3, and they have the same leaf set and if for all pairs of leaves  $x, y$ ,  $|T_1[x, y] - T_2[x, y]| < \epsilon$ , then if  $e_1$  and  $e_2$  are corresponding edges in  $T_1$  and  $T_2$ ,  $|w(e_1) - w(e_2)| \leq 2\epsilon$ .*

**Proof:** Suppose for contradiction that  $e_1 \in T_1$  and  $e_2 \in T_2$  are corresponding edges and  $w(e_2) - w(e_1) > 2\epsilon$ . There are two cases.

Case 1:  $e_2 = (u, v)$  is an internal edge with at least two species on each side. Let  $e_1 = (u', v')$ . If there is a leaf  $x$  on the  $u$  side of  $e_2$  and a leaf  $y$  on the  $v$  side of  $e_2$  such that the  $T_2[x, u] - T_1[x, u'] \geq -\epsilon/2$  and  $T_2[y, v] - T_1[y, v'] \geq -\epsilon/2$ , then  $T_2[x, y] - T_1[x, y] > \epsilon$  contradicting the assumption.

Hence, on either the  $u$  side or the  $v$  side (say  $u$ ), for each leaf  $x$ ,  $T_2[x, u] < T_1[x, u'] - \epsilon/2$ . Now for any two such leaves  $x$  and  $y$ ,  $T_2[x, y] < T_1[x, y] - \epsilon$  again contradicting our assumption.

Case 2:  $e_2 = (u, v)$  is an external edge with  $u$  a leaf. Let  $e_1 = (u', v')$  be the corresponding edge with  $u'$  being a leaf in  $T_1$ . For any other species  $x$ ,  $T_2(x, v) < T_1(x, v) - \epsilon$ . Take two such species  $x$  and  $y$  such that the paths in  $T_2$  between  $x$  and  $v$  and between  $y$  and  $v$  are disjoint.  $T_2[x, y] < T_1[x, y] - 2\epsilon$  and we again have a contradiction.  $\blacksquare$

**Lemma 7** *Let  $T_1$  and  $T_2$  be edge-weighted rooted trees on the same leaf set which are not necessarily isomorphic even in topology. Suppose for every pair of leaves  $x, y$   $|T_1[x, y] - T_2[x, y]| \leq \epsilon$ . Then for every pair of corresponding edges  $e$  and  $e'$  in  $T_1$  and  $hc(T_1 < T_2)$  respectively,  $w(e) - w(e') \leq 2\epsilon$ .*

**Proof:** Suppose  $e = (u, v)$  is an edge in  $T_1$  such that its corresponding edge  $e'$  in  $hc(T_1 < T_2)$  has zero weight. Then there exist 4 species  $x, y, p, q$  in  $T_1$  such that  $x, y \in B(e)$  and  $p, q \notin B(e)$ , the

least common ancestor (lca) of  $x$  and  $y$  is  $v$  and if  $T_1$  were rooted at  $v$  then the lca of  $p$  and  $q$  would be at  $u$ . Furthermore  $T_2$  does not contain any edge  $e'$  such that  $x, y \in B(e')$  and  $p, q \notin B(e')$ .

From  $T_1$  we have  $T_1(x, y) + T_1(p, q) + 2w(e) = T_1(p, x) + T_1(q, y)$ . From  $T_2$  we have  $T_2(x, y) + T_2(p, q) \geq T_2(p, x) + T_2(q, y)$ . By using the relationship between interleaf distances in  $T_1$  and  $T_2$  we have  $T_1(x, y) + T_1(p, q) + 4\epsilon \geq T_1(p, x) + T_1(q, y)$ . Thus  $w(e) \leq 2\epsilon$  and the result follows. ■

The proof that the tree produced by our algorithm is close to the true tree with high probability takes the following form.

With probability at least  $1 - n^{-2}$  our estimate  $\mathcal{T}^*$  is within  $\delta$  of the true tree  $\mathcal{T}_S$ . Since we find a tree  $\hat{\mathcal{T}}$  which is within  $3\delta$  of  $\mathcal{T}^*$  in this case, the  $L_\infty$  distance between  $\hat{\mathcal{T}}$  and  $\mathcal{T}_S$  is bounded by  $4\delta$ . By the previous two lemmas in each pair of isomorphic trees we consider, the corresponding edges differ in weight by at most  $8\delta$ .

When each such pair of isomorphic trees is considered in the probability domain the worst discrepancies on the probabilities arise when the two corresponding edges  $e$  and  $e'$  have weights 0 and  $8\delta$  respectively in the time domain. In this case the corresponding probabilities are 0 and  $(1 - e^{-8\delta})/2$ . Using the inequality that  $1 - e^{-x} \leq x$  for  $x \geq 0$ , we get that the latter probability is no more than  $4\delta$ . Thus using Lemma 5 we find that each pair of corresponding trees have variational distance bounded by  $4n\delta$ . Putting all this together we get the following theorem.

**Theorem 1** *With high probability, the variational distance between  $\hat{S}$  and  $S$  is upper bounded by  $12n\delta$ . Taking  $k \gg \ln n$ , we get that  $V(S, \hat{S}) \leq (12n\sqrt{6 \ln n})/(\alpha\sqrt{k})$ .*

**Proof:** The first statement follows from the above discussion. So we must show that  $12n\delta = (12n\sqrt{6 \ln n})/(\alpha\sqrt{k})$ . But  $\delta = -\ln(1 - 2(p + t)) + \ln(1 - 2p)$  and  $t = \sqrt{6 \ln n/k}$ . The assumption that  $k \gg \ln n$  gives that  $t \leq \ln(1 + t)$ , and a bit of algebra gives the rest of the theorem. ■

**Remark.** The assumption that  $k \gg \ln n$  is valid on biological data sets where typically  $k \gg n$ .

## 4 Conclusion

We have shown that in  $O(n^2k)$  time we can construct a CFT  $\hat{S}$  that is within  $O(n\sqrt{\ln n}/\alpha\sqrt{k})$  from  $S$ , and that no algorithm can give a CFT that is  $o(1/k)$  from  $S$ , with high probability. There are three factors which give a gap. First, can we remove the dependence on  $\alpha$  without increasing the dependence on  $n$  or  $k$ ? We expect that this may be possible if the tree fitting algorithm of [1] can be generalized to deal with varying tolerance, i.e, we want a tree-fitting algorithm that takes as input a distance matrix  $D$  and a tolerance matrix  $\mathcal{E}$  and finds a tree,  $T$  such that  $|D[i, j] - T[i, j]| \leq \epsilon\mathcal{E}[i, j]$  and  $\epsilon$  is as small as possible. Clearly finding the best  $\epsilon$  is again NP-complete, but we would like an approximation algorithm along the lines of [1] for this more general question. In [4], an algorithm was shown that has a convergence rate which does not depend on  $\alpha$ , and is polynomial in  $n$  and  $k$ . However this convergence rate is not given exactly.

Second, can the bounds be tightened in terms of  $n$  and  $k$ ? In [2], a much stronger lower bound was shown, in particular  $\Omega(\sqrt{n/k})$ . This lower bound was nearly matched for some classes of trees. We refer the reader to that paper for the details.

Another unresolved issue is that of 0/1 data versus four (or more) state data. This issue was also resolved in [2], where it was shown that if a  $k$  state markov process is *invertible*, that the same bounds hold for learning them. See Steel, Hendy and Penny [14] for an analysis of which types of processes are invertible.

Finally, there is the related but somewhat independent problem of determining the number of samples needed to reconstruct the topology of the CFT. This problem was first considered in [15]. It turns out that an algorithm can be good in terms of metric convergence rate but not topological convergence rate, or vice versa. The current best algorithm for the topology reconstruction problem together with its analysis is given in [5].

## References

- [1] R. Agarwala, V. Bafna, M. Farach, B. Narayanan, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy: Fitting distances by tree metrics. *Proc. of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1996.
- [2] Andris Ambainis, Richard Desper, Martin Farach, and Sampath Kannan. Nearly tight bounds on the learnability of evolution. *Proc. of the 38th IEEE Annual Symp. on Foundation of Computer Science*, pages 524–533, 1997.
- [3] J. Cavender. Taxonomy with confidence. *Mathematical Biosciences*, 40:271–280, 1978.
- [4] Mary Crián, Leslie Ann Goldberg, and Paul Goldberg. Evolutionary trees can be learned in polynomial time in the two-state case. *Proc. of the 39th IEEE Annual Symp. on Foundation of Computer Science*, 1998.
- [5] Péter Erdos, Michael Steel, László Székely, and Tandy Warnow. Constructing big trees from short sequences. *Proc. of 24th International Colloquium on Automata Languages and Programming*, 1997.
- [6] M. Farach and S. Kannan. Efficient algorithms for inverting evolution. *Proc. of the 28th Ann. ACM Symp. on Theory of Computing*, 1996.
- [7] J.S. Farris. Estimating phylogenetic trees from distance matrices. *Am. Nat.*, 106:645–668, 1972.
- [8] J. Felsenstein. Statistical inference of phylogenies. *J. R. Statist. Soc. A*, 1983.
- [9] J. Felsenstein. Phylogenies from molecular sequences: inference and reliability. *Annual Review of Genetics*, 22:521–65, 1988.
- [10] R.L. Kashyap and S. Subas. Statistical estimation of parameters in a phylogenetic tree using a dynamic model of the substitutional process. *J. theor. Biol.*, 47:74–101, 1974.
- [11] Michael Kearns, Yishay Mansour, Dana Ron, Ronitt Rubinfeld, Robert E. Schapire, and Linda Sellie. On the learnability of discrete distributions (extended abstract). *Proc. of the 26th Ann. ACM Symp. on Theory of Computing*, pages 273–282, 1994.

- [12] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.
- [13] M. Nei. *Molecular Evolutionary Genetics*. Columbia University Press, New York, 1987.
- [14] M. Steel, M.D. Hendy, and D. Penny. A discrete fourier analysis for evolutionary trees. *Proceedings of the National Academy of Science*, 91:3339–3343, 1994.
- [15] M. Steel, L. Szekely, and P. Erdos. The number of nucleotide sites needed to accurately reconstruct large evolutionary trees. Technical Report 96-19, DIMACS, 1996. <http://dimacs.rutgers.edu/TechnicalReports/1996.html>.
- [16] D. L. Swofford and G. J. Olsen. Phylogeny reconstruction. In D. M. Hillis and C. Moritz, editors, *Molecular Systematics*, pages 411–501. Sinauer Associates Inc., Sunderland, MA., 1990.