Efficient Algorithms for Inverting Evolution

Martin Farach* Sampath Kannan†

March 26, 1996

Abstract

Evolution is a stochastic process which operates on the DNA of species. The evolutionary process leaves tell-tale signs in the DNA which 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 widely accepted and intellectually satisfying, they have been computationally intractable.

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 stochatic models that are close according to this metric, it needs to be given a lot of observations. We complement this result with a simple and efficient algorithm for inverting the stochastic process of evolution, that is, for building the tree from observations on the DNA of the species. Our result can be viewed as a result on the PAC-learnability of the class of distributions produced by tree-like processes.

Though there have been many heuristics suggested for this problem, 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 the first polynomial-time algorithm which is guaranteed to converge at all to the correct tree.

*Rutgers University; 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; http://www.cis.upenn.edu/~kannan/home.html

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. Broadly, these methods fall into three categories — distance-based methods, character-based methods, and likelihood methods. Excellent surveys of the known methods can be found in [7, 8, 14]. 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.

Distance methods start by mapping the input DNA into a table of pairwise distances for the species. The problem is to then construct a tree which fits the distances as well as possible. Many heuristics are known for this problem [12] and for some optimization criteria, algorithms exist with provably good performance [1, 4]. For a detailed survey of the methods in this class, see [6, 14]. *Prima facie* it appears that distance-based methods lose information in converting the original sequence data into a matrix of distances. In this paper we refute this belief by showing that a simple *distance-based* algorithm lies at the core of a procedure for finding the tree that is most likely to have given rise to the input *sequence* data.

A *character* is an equivalence relation on the set S of species, partitioning the species set into distinct equivalence classes which are called *states*. Most character data is drawn from aligned biomolecular sequences. For such data each aligned position represents a character. The alignment of sequences drawn from a set of species is a difficult task since it is not clear how an objective function should be chosen and since for most reasonable choices of this function the optimization problem is NP-hard. However all character-based and likelihood methods assume that aligned data is available and we will make the same assumption.

Given a set of species S of size n whose DNA has been mapped into a set of characters C of size k, the information about the state of each character on each species can be described by an $n \times k$ matrix M where M_{ij} represents the state of the i^{th} species on the j^{th} character. The most popular character-based criterion, *parsimony*, seeks to find a phylogentic tree for the species which requires the minimum number of mutations to produce the data set M. A survey of parsimony and its variants can be found in [6]. In mathematical terms, we get the Steiner Tree Problem under the Hamming metric. Parsimony, as well as all its standard variants, are NP-hard [3] but heuristics for parsimony are some of the most widely used tree construction methods.

Despite its popularity, parsimony suffers from the fact that it is not a *consistent* method [2, 5]. Informally, a method is consistent if the tree it constructs converges to the true tree as more and more data becomes available, which in the case of phylogeny means as k goes to infinity. There is no reason to believe that the tree which gives the minimum number of mutations would be the true tree. This would only be the case if mutations are so expensive that they are quite rare. However, mutations are quite common.

1.1 A Stochastic Model of Evolution

It has therefore been recognized that a deeper model of evolution needs to be incorporated into phylogeny construction methods. By far the most interesting methods produced to date are the so called *Maximum Likelihood Estimation* methods (MLE). In MLE methods, the input matrix is

understood to be the outcome of a random process, that random process being evolution. Any evolutionary random process induces a distribution over the space of data, and so the question becomes that of finding, from amongst all evolutionary random processes, the one which is most likely, given the data.

We must therefore address two question. First, what types of random processes are "evolutionary," and second, how do we find the best such process for the data. 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 *Cavendar-Farris Tree*, named after the biologists who first proposed this class of models.. For ease of presentation, we assume that we have two-state characters with states 0 and 1. We will discuss generalizations of this class in the conclusions.

A Cavendar-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 $P_e \in (0,.5)$. 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 it gets labeled b with probability $1 - P_e$. Thus each edge has some probability of producing a mutation. The leaf vector is taken to be the output of an CFT.

Now, any $\{0,1\}^n$ vector has some positive probability of being produced as the output of any CFT. Let S be an 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

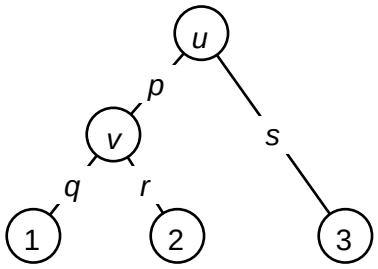

Figure 1: Tree describing a stochastic process. Labels on the edges represent probabilities of change between the endpoints.

leaves is the sum of two terms for the two choices for the state of v. This probability is equal to $(1-p)(1-q)r(1-s) + pq(1-r)s$.

CFTs have the following straightforward reïnterpretation as a Poisson processes. Imagine a Poisson process with rate λ where the events are the changes of state of a character. This process 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, though 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 an 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 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 [7] and by Steel, Hendy, and Penny [13], and is the basis for all MLE methods. Given this model the computational problem can be described as below.

Problem: MLE construction of evolutionary trees

Input: Sample points $M = \langle c_1, \ldots, c_k \rangle$ generated from some unknown n-leaf CFT S.

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 [9] 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 [7] 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 [13] view the problem as an inverse problem and show that an exponential-sized Hadamard matrix can be used to find exact values of the parameters.

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. However, even asymptotic convergence (to the true tree) has not been formally proven for any of these heuristic methods and nothing has been shown about convergence rates or computational complexity. We present a 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.

Our result can be viewed in the setting of PAC-learning of distributions introduced by [10]. 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 and directions for future work.

2 A metric on Cavendar-Farris Trees

We are interested in converging on the CFT S via some algorithm on k sample points $c_1, \ldots, c_k$ 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 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 $(0, .5]$ instead of $(0, .5)$, the variational distance of CFTs would not be a metric.

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 $D^{\otimes k}(x_1, \ldots, x_k) = \prod_{i=1}^{k} D(x_i)$. We can show that if $Var(\mathcal{P}_{S_0}, \mathcal{P}_{S_1}) \leq \epsilon$, then $Var(\mathcal{P}_{S_0}^{\otimes k}, \mathcal{P}_{S_1}^{\otimes k}) \leq k\epsilon$. This derivation is shown in the appendix where by induction the last expression in Eq. 1 can be seen to be no greater than $k\epsilon$. The lemma follows from this fact and the claim proved above. ■

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 *$\mathcal{T}_S$ in 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. ■

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. ■

A similar lemma is proved in [13].

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. 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.

2. 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 $\delta > 0$.

3. 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 $\mathcal{D}^*(i,j)$ is bounded away from $1/2$ for all pairs of species i, j. This is because as $\mathcal{D}^*(i,j)$ approaches $1/2$, $\mathcal{T}^*(i,j)$ approaches ∞ 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. Formally, we assume that there is a constant $\alpha > 0$ such that $\mathcal{D}^*(i,j) \leq 1/2 - \alpha$ for all i, j.

We would like to estimate $\mathcal{D}^*(i,j)$ 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 δ 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 α and δ. If we ensure that $\mathcal{E}^*(i,j) \leq t \ \forall i, j$ then we get the required uniform tolerances of δ 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 i, j, $\Pr[\mathcal{E}^*(i,j) > t] \leq n^{-4}$.*

Proof: The proof follows from a direct application of the Chernoff bound inequalities [11]. As a corollary the probability that $\mathcal{E}^*(i,j) > t$ for *any* i, j, 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_∞-best-fit additive tree.

Input: An $n \times n$ matrix M representing pairwise distance estimates among n points.

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

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

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

3.2 Analysis of Algorithm

Our goal is to show that the variational distance between $\hat{\mathcal{S}}$ and S is small. (We will sometimes prove statements about the equivalent time domain trees – $\hat{\mathcal{T}}$ 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_∞ 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 $[1,\ldots,n]$. 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. If e and e' are corresponding edges in S and S' and $P_{e'} = P_e + \epsilon$, we roll a 3-sided die with probabilities P_e, ϵ, and $1 - P_e - \epsilon$ 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. For typographic convenience we will also denote the fact that a node x is in state i simply by x_i.

Assume that the p is the probability that the state at r (and at r') is 1. By coupling we can 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 in the appendix.

Inequality 2 is obtained by observing that there is a probability of at most 2ϵ 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.

We finish the derivation in the appendix and, by using inductive assumptions about the subtrees rooted at the children of the roots, we get the required result (see Eq. 3). ■

The next two lemmas are proved in the time domain but apply by translation 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. ■

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 δ of the true tree $\mathcal{T}_S$. Since we find a tree $\hat{\mathcal{T}}$ which is within 3δ of $\mathcal{T}^*$ in this case, the L_∞ distance between $\hat{\mathcal{T}}$ and $\mathcal{T}_S$ is bounded by 4δ. By the previous two lemmas in each pair of isomorphic trees we consider, the corresponding edges differ in weight by at most 8δ.

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 8δ respectively in the time domain. In this case the corresponding probabilities are 0 and $(1 - e^{-8\delta})/2$. Using the inequality that $e^{-x} \leq (1 - x)$ for $x \geq 0$, we get that the latter probability is no more than 4δ. Thus using our lemma relating probability discrepancies and variational distance 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 *The variational distance between $\hat{\mathcal{S}}$ and S is upper bounded by $12n\delta$. Taking $k \gg \ln n$, we get that $V(S, \hat{\mathcal{S}}) \leq (12n\sqrt{6 \ln n})/(\alpha\sqrt{k})$.*

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 $t = \sqrt{6 \ln n / k}$. A bit of algebra and the assumption that $k \gg \ln n$ gives the rest of the theorem. ■

4 Conclusion

We have shown that in $O(n^2k)$ time we can construct an CFT $\hat{\mathcal{S}}$ that is within $O(n\sqrt{\ln n}/\alpha\sqrt{k})$ from S, and that no algorithm can give an CFT that is $o(1/k)$ from S, w.h.p. There are three factors which give a gap. First, can we remove the dependence on α? 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 ϵ is as small as possible. Clearly finding the best ϵ is again NP-complete, but we would like an approximation algorithm along the lines of [1] for this more general question.

Second, can the dependence on n be reduced? We expect that our algorithm actually has a sublinear dependence on n, and that the analysis needs tightening. Finally, we conjecture that the true lower bound for this problem is $\Omega(1/\sqrt{k})$, which would show that our algorithm is tight in this regard. Such a lower bound requires a deeper understanding of the distribution of CFT induced distributions over $\{0,1\}^n$ within the space of all such distributions.

The other unresolved issue is that of 0/1 data versus four (or more) state data. We used the 0/1 assumption is producing a mapping between $\mathcal{D}_S$ and $\mathcal{T}_S$. For four state characters, we would typically assume that there is some (known) Markov process which is proceeding along the edges of the CFT for unknown times. Our algorithm will work if the times can be derived from observations on the outcomes. Steel, Hendy and Penny [13] have studied which types of processes are so invertible, though they have not considered in their analysis how tolerances in the probability domain translate into tolerances in the time domain. We expect that it should be fairly straightforward to derive these transformations and apply them to our algorithm in these general cases.

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] J. Cavender. Taxonomy with confidence. *Mathematical Biosciences*, 40:271–280, 1978.

[3] W.H.E. Day, D.S. Johnson, and D. Sankoff. The computational complexity of inferring rooted phylogenies by parsimony. *Mathematical Biosciences*, 81:33–42, 1986.

[4] M. Farach, S. Kannan, and T. Warnow. A robust model for finding optimal evolutionary trees. *Algorithmica*, 13:155–179, 1993.

[5] J. Felsenstein. Cases in which parsimony or compatibility methods will be positively misleading. *Syst. Zool.*, 22:240–249, 1978.

[6] J. Felsenstein. Numerical methods for inferrinng evolutionary trees. *The Quarterly Review of Biology*, 57(4), 1982.

[7] J. Felsenstein. Statistical inference of phylogenies. *J. R. Statist. Soc. A*, 1983.

[8] J. Felsenstein. Phylogenies from molecular sequences: inference and reliability. *Annual Review of Genetics*, 22:521–65, 1988.

[9] 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.

[10] 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.

[11] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[12] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogentic trees. *Mol. Biol. Evol.*, 4:406–424, 1987.

[13] 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.

[14] 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.

Appendix

Derivation for the lower bound:

$$
\begin{aligned}
Var(\mathcal{P}_{S_0}^{\otimes k}, \mathcal{P}_{S_1}^{\otimes k}) &= \sum_{x_1,\ldots,x_k} |\prod \mathcal{P}_{S_0}(x_i) - \prod \mathcal{P}_{S_1}(x_i)| \\
&\leq \sum_{x_1} |\mathcal{P}_{S_0}(x_1) - \mathcal{P}_{S_1}(x_1)|(\sum_{x_2,\ldots,x_k} \prod \mathcal{P}_{S_0}(x_i)) + \sum_{x_2,\ldots,x_k} |\prod \mathcal{P}_{S_0}(x_i) - \prod \mathcal{P}_{S_1}(x_i)|(\sum_{x_1} \mathcal{P}_{S_1}(x_1)) \\
&= \sum_{x_1} |\mathcal{P}_{S_0}(x_1) - \mathcal{P}_{S_1}(x_1)| + \sum_{x_2,\ldots,x_k} |\prod \mathcal{P}_{S_0}(x_i) - \prod \mathcal{P}_{S_1}(x_i)| \qquad (1)
\end{aligned}
$$

Derivation for the upper bound:

$$
\begin{aligned}
V(S, S') &\leq \sum_{uv \in \{0,1\}^n} |\Pr[uv|r_0] - \Pr[uv|r'_0]| \\
&= \sum_{uv} |(\Pr[u|x_0]\Pr[x_0|r_0] + \Pr[u|x_1]\Pr[x_1|r_0])(\Pr[v|y_0]\Pr[y_0|r_0] + \Pr[v|y_1]\Pr[y_1|r_0]) \\
&\quad - (\Pr[u|x'_0]\Pr[x'_0|r'_0] + \Pr[u|x'_1]\Pr[x'_1|r'_0])(\Pr[v|y'_0]\Pr[y'_0|r'_0] + \Pr[v|y'_1]\Pr[y'_1|r'_0])| \\
&\leq 4\epsilon + \sum_{uv} |(\Pr[u|x_0]\Pr[v|y_0]) - (\Pr[u|x'_0]\Pr[v|y'_0])| \qquad (2) \\
&= 4\epsilon + \sum_{uv} |\Pr[u|x_0]\Pr[v|y_0] - \Pr[u|x'_0]\Pr[v|y_0] + \Pr[u|x'_0]\Pr[v|y_0] - \Pr[u|x'_0]\Pr[v|y'_0]| \\
&\leq 4\epsilon + \sum_{uv} \Pr[u|x_0](|\Pr[v|y_0] - \Pr[v|y'_0]|) + \sum_{uv} \Pr[v|y'_0](|\Pr[u|x_0] - \Pr[u|x'_0]) \\
&\leq 4\epsilon + \sum_{v} |\Pr[v|y_0] - \Pr[v|y'_0]| + \sum_{u} |\Pr[u|x_0] - \Pr[u|x'_0]| \\
&\leq 4\epsilon + V(S_x, S_{x'}) + V(S_y, S_{y'}) \qquad (3)
\end{aligned}
$$