Optimal Phylogenetic Reconstruction

Constantinos Daskalakis
EECS Department
UC Berkeley
Berkeley, CA 94720
costis@eecs.berkeley.edu

Elchanan Mossel
Department of Statistics
UC Berkeley
Berkeley, CA 94720
mossel@stat.berkeley.edu

Sébastien Roch
Department of Statistics
UC Berkeley
Berkeley, CA 94720
sroch@stat.berkeley.edu

ABSTRACT

One of the major tasks of evolutionary biology is the reconstruction of phylogenetic trees from molecular data. The evolutionary model is given by a Markov chain on the true evolutionary tree. Given samples from this Markov chain at the leaves of the tree, the goal is to reconstruct the evolutionary tree.

It is well known that in order to reconstruct a tree on n leaves, sequences of length $\Omega(\log n)$ are needed. It was conjectured by M. Steel that for the CFN evolutionary model, if the mutation probability on all edges of the tree is less than $p^* = (\sqrt{2}-1)/2^{3/2}$, then the tree can be recovered from sequences of length $O(\log n)$. This was proven by the second author in the special case where the tree is "balanced". The second author also proved that if all edges have mutation probability larger than p^* then the length needed is $n^{\Omega(1)}$. This "phase-transition" in the number of samples needed is closely related to the phase transition for the reconstruction problem (or extremality of free measure) studied extensively in statistical physics, probability and computer science.

Here we complete the proof of Steel's conjecture and give a reconstruction algorithm using optimal (up to a multiplicative constant) sequence length. Our results further extend to obtain an optimal reconstruction algorithm for the Jukes-Cantor model with short edges. All reconstruction algorithms run in polynomial time.

Categories and Subject Descriptors: F.2 [Theory of Computation]: Analysis of Algorithms and Problem Complexity; J.3 [Computer Applications]: Life and Medical Sciences—*Biology and genetics*.

General Terms: Algorithms, Theory.

Keywords: Phylogenetics, CFN model, Ising model, phase transitions, reconstruction problem, Jukes Cantor.

1. INTRODUCTION

In this paper we establish a central conjecture in algorithmic Phylogeny [26]: we show that every phylogenetic tree with short edges on n leaves can be reconstructed from sequences of length $O(\log n)$. This result is optimal up to a multiplicative constant.

Phylogeny Background. Phylogenies are used in evolutionary biology to model the stochastic evolution of genetic data on the ancestral tree relating a group of species. The leaves of the tree correspond to (known) extant species. Internal nodes represent extinct species while the root of the tree represents the most recent ancestor to all species in the tree. Following paths from the root to the leaves, each bifurcation indicates a speciation event whereby two new species are created from a parent. We refer the reader to [9, 24] for excellent introductions to Phylogeny.

The underlying assumption is that genetic information evolves from the root to the leaves according to a Markov model on the tree. This genetic information may consist of DNA sequences, proteins, etc. Suppose for example that the genetic data consists of (aligned) DNA sequences and let us follow the evolution of the first letter in all sequences. This collection, named the first *character*, evolves according to Markov transition matrices on the edges. The root is assigned one of the four letters A, C, G and T. Then this letter evolves from parents to descendants according to the Markov matrices on the edges connecting them.

The vector of the i'th letter of all sequences is called the i'th *character*. It is further assumed that the characters are i.i.d. random variables. In other words, each site in a DNA sequence is assumed to mutate independently from its neighbors according to the same mutation mechanism. Naturally, this is an over-simplification of the underlying biology. Nonetheless, the model above may be a good model for the evolution of some DNA subsequences and is the most popular evolution model in molecular biology, see e.g. [9, 24]. One of the major tasks in molecular biology, the *reconstruction of phylogenetic trees*, is to infer the topology of the (unknown) tree from the characters (sequences) at the leaves (extant species).

In this paper we will be mostly interested in two mutation models, the Cavender-Farris-Neyman (CFN) model [3, 8, 23], and the Jukes-Cantor (JC) model [14].

In the CFN model the character states are 0 and 1 and their a priori probability at the root is $1/2$ each (the 0 and 1 originally corresponded to the Purine and Pyrimidine groups). To each edge e corresponds a mutation parameter $p(e)$ which is the probability that the character mutates along the edge e. In the JC model the character states are A, C, G and T with a priori probability $1/4$ each. To each edge e corresponds a mutation parameter $p(e)$ and it is assumed that every state mutates with probability $p(e)$ to each of the other states.

The reconstruction problem. A problem that is closely related to the *phylogenetic problem* is that of inferring the *ancestral state*, i.e. the character state at the root of the tree, given the character states at the leaves. This problem was studied earlier in statistical physics, probability and computer science under the title of the *reconstruction problem*, or the *extremality of the free Gibbs measure*.

STOC'06, May21–23, 2006, Seattle, Washington, USA.

See [25, 11, 10]. The reconstruction problem for the CFN model was analyzed in [2, 6, 12, 1, 15]. In particular, the role of the reconstruction problem in the analysis of the mixing time of Glauber dynamics on trees was established in [1, 15].

Roughly speaking, the reconstruction problem is *solvable* when the correlation between the root and the leaves persists no matter how large the tree is. When it is unsolvable, the correlation decays to 0 for large trees. The results of [2, 6, 12, 1, 15] show that for the CFN model, if for all e, it holds that $p(e) \leq p_{\max} < p^*$ then the reconstruction problem is solvable, where

$$p^* = \frac{\sqrt{2}-1}{\sqrt{8}}.$$

If, on the other hand, for all e it holds that $p(e) \geq p_{\min} > p^*$ and the tree is balanced in the sense that all leaves are at the same distance from the root, then the reconstruction problem is unsolvable. Moreover in this case, the correlation between the root state and any function of the character states at the leaves decays as $n^{-\Omega(1)}$.

Our results. M. Steel [26] conjectured that when $0 < p_{\min} \leq p(e) \leq p_{\max} < p^*$ for all edges e, one can reconstruct with high probability the phylogenetic tree from $O(\log n)$ characters. Steel's insightful conjecture suggests that there are deep connections between the reconstruction problem and phylogenetic reconstruction.

This conjecture has been proven to hold for trees where all the leaves are at the same distance from the root in [20]. It is also shown there that the number of characters needed when $p(e) \geq p_{\min} > p^*$ for all e is $n^{\Omega(1)}$. The second result intuitively follows from the fact that the topology of the part of the tree that is close to the root is essentially independent of the characters at the leaves if the number of characters is not at least $n^{\Omega(1)}$.

The basic intuition behind Steel's conjecture is that, since, in the regime where $p(e) \leq p_{\max} < p^*$, there is no decay of the quality of reconstructed sequences, it should be as easy to reconstruct deep trees as it is to reconstruct shallow trees. In [5] (see also [19, 7]) it is shown that "shallow" trees can be reconstructed from $O(\log n)$ characters if all mutation probabilities are bounded away from 0 and $1/2$ (the results of [5] also show that in this regime, any tree can be recovered from sequences of polynomial length). The same high-level reasoning has also yielded a complete proof that $O(\log n)$ characters suffice for a "homoplasy-free" mutation model when all edges are short [22].

Here we give a complete proof of Steel's conjecture. We show that, if $0 < p_{\min} \leq p(e) \leq p_{\max} < p^*$ for all edges e of the tree, then the tree can be reconstructed from $c(p_{\min}, p_{\max})(\log n + \log 1/\delta)$ characters with error probability at most δ. This result implies that sequences of logarithmic length suffice to reconstruct phylogenetic trees in the Jukes-Cantor model, when all the edges are sufficiently short.

1.1 Definitions and results

Let T be a tree. Write $\mathcal{V}(T)$ for the nodes of T, $\mathcal{E}(T)$ for the edges of T and $\mathcal{L}(T)$ for the leaves of T. If the tree is rooted, then we denote by $\rho(T)$ the root of T. Unless stated otherwise, all trees are assumed to be *binary* (all internal degrees are 3) and it is further assumed that $\mathcal{L}(T)$ is labeled.

Let T be a tree equipped with a length function on its edges, $d : \mathcal{E}(T) \to \mathcal{R}_+$. d will also denote the induced path metric on $\mathcal{V}(T)$: $d(v,w) = \sum\{d(e) : e \in \mathtt{path}_T(v,w)\}$, for all $v, w \in \mathcal{V}(T)$, where $\mathtt{path}_T(x,y)$ is the path (sequence of edges) connecting x to y in T.

We will further assume below that the length of all edges is bounded between f and g for all $e \in E$. In other words, for all $e \in \mathcal{E}(T)$, $f \leq d(e) \leq g$.

We now define the evolution process on a rooted tree equipped with a path metric d. The process is determined by a rooted tree $T = (V,E)$ equipped with a path metric d and a *mutation rate matrix* Q. We will be mostly interested in the case where $Q = \left(\begin{smallmatrix} -1 & 1 \\ 1 & -1 \end{smallmatrix}\right)$ corresponding to the CFN model and in the case where Q is a 4×4 matrix given by $Q_{i,j} = 1 - 4\delta(i = j)$ corresponding to the Jukes-Cantor model. To edge e of length $d(e)$ we associate the mutation matrix $M^e = \exp(d(e)Q)$.

In the mutation model on the tree T rooted at ρ each vertex iteratively chooses its state from the state at its parent by an application of the Markov transition rule M^e, where e is the edge connecting it to its parent. We assume that all edges in E are directed away from the root. Thus the probability distribution on the tree is the probability distribution on $\{0,1\}^V$ ($\{A,C,G,T\}^V$) given by $\overline{\mu}[\sigma] = \pi(\sigma(\rho)) \prod_{(x \to y) \in E} M^{(x \to y)}_{\sigma(x),\sigma(y)}$, where π is given by the uniform distribution at the root, so that $\pi(0) = \pi(1) = 1/2$ for the CFN model and $\pi(A) = \pi(C) = \pi(G) = \pi(T) = 1/4$ for the JC model. We let the measure μ denote the marginal of $\overline{\mu}$ on the set of leaves which we identify with $[n]$. Thus $\mu(\sigma) = \sum\{\overline{\mu}(\tau) : \forall i \in [n], \tau(i) = \sigma(i)\}$. The measure μ defines the probability distribution at the leaves of the tree.

We note that both for the CFN model and for the JC model, the mutation matrices M^e are in fact very simple. For the CFN model, with probability $p(e) = (1 - \exp(-2d(e)))/2$, there is a mutation and, otherwise, there is no mutation. Similarly for the JC model with probability $p(e) = (1 - \exp(-4d(e)))/4$ each of the three possible mutations occur. In particular, defining

$$g^* = \frac{\log 2}{4}, \tag{1}$$

we may formulate the result on the reconstruction problem for the phase transition of the CFN model as follows: "If $d(e) \leq g < g^*$ for all e then the reconstruction problem is solvable."

We will be interested in reconstructing phylogenies in this regime. The objective is to reconstruct the underlying tree T whose internal nodes are unknown from the collection of sequences at the leaves. Since for both the CFN model and the JC model, the distribution $\overline{\mu}[\sigma]$, described above, is independent of the location of the root we can only aim to reconstruct the underlying un-rooted topology. Let $\mathcal{T}$ represent the set of all *binary topologies* (i.e. unrooted undirected binary trees) and $\mathcal{M}^{CFN}_{f,g}$ the family of CFN mutation matrices, as described above, which correspond to distances d satisfying:

$$0 < f \leq d \leq g < g^*,$$

where g^* is given by (1) and f is an arbitrary positive constant. Let $\mathcal{T} \otimes \mathcal{M}^{CFN}_{f,g}$ denote the set of all unrooted phylogenies, where the underlying topology is in $\mathcal{T}$ and all mutation matrices on the edges are in $\mathcal{M}^{CFN}_{f,g}$. Rooting $T \in \mathcal{T} \otimes \mathcal{M}^{CFN}_{f,g}$ at an arbitrary node, let μ_T be the measure at the leaves of T as described above. It is well known, e.g. [5, 4] that different elements in $\mathcal{T} \otimes \mathcal{M}^{CFN}_{f,g}$ correspond to different measures; therefore we will identify measures with their corresponding elements of $\mathcal{T} \otimes \mathcal{M}^{CFN}_{f,g}$. We are interested in finding an efficiently computable map Ψ such that $\Psi(\sigma^1_\partial, \ldots, \sigma^k_\partial) \in \mathcal{T}$, where $\left\{\sigma^i_\partial\right\}_{i=1}^k$ are k characters at the leaves of the tree. Moreover, we require that for every distribution $\mu_T \in \mathcal{T} \otimes \mathcal{M}^{CFN}_{f,g}$, if $\sigma^1_\partial, \ldots, \sigma^k_\partial$ are generated independently from μ_T, then with high probability $\Psi(\sigma^1_\partial, \ldots, \sigma^k_\partial) = T$. The problem of finding an efficiently computable map Ψ (with small value of k) is called the *phylogenetic reconstruction problem* for the CFN model. The phylogenetic reconstruction problem for the JC model is defined similarly. In [5], it is shown that there exists a polynomial time algorithm that reconstructs the topology from

$k = \mathrm{poly}(n, 1/\delta)$ characters, with probability of error δ. Our results are the following.

THEOREM 1. *Consider the CFN model on binary trees where all edges satisfy: $0 < f \le d(e) \le g < g^*$. Then there exists a polynomial time algorithm that reconstructs the topology of the tree from $k = c(f,g)(\log n + \log 1/\delta)$ characters with error probability at most δ; in particular, $c(f,g) = \frac{c(g)}{f^2}$. Moreover, the value g^* given by (1) is tight.*

Our algorithm can be used to reconstruct phylogenies from DNA sequences if we replace all occurrences of characters 'A' and 'G' by '0' and all occurrences of 'C' and 'T' by '1'. Note that this mapping is in accordance with the biological role of purines and pyrimidines if we interpret symbol '0' as purine and symbol '1' as pyrimidine. In fact, we prove the following.

COROLLARY 1. *Consider the JC model on binary trees where all edges satisfy*

$$0 < f \le d(e) \le g < g^*_{\mathrm{JC}}, \text{ where } g^*_{\mathrm{JC}} := g^*/2.$$

Then there exists a polynomial time algorithm that reconstructs the topology of the tree from $c'(f,g)(\log n + \log 1/\delta)$ characters with error probability at most δ; $c'(f,g) = \frac{c'(g)}{f^2}$.

Theorem 1 and Corollary 1 extend also to cases where the data at the leaves is given with an arbitrary level of noise. For this "Robust Phylogenetic Reconstruction Problem" both values g^* and g^*_{JC} are tight.

1.2 Organization of the Paper

The paper is organized as follows. We start with an overview of the algorithm and techniques used in Section 2. In Section 3 we provide a high-level description of our analysis of the reconstruction algorithm. Proofs for the combinatorial part of the argument can be found in sections 4 and 5. Proofs for the probabilistic part of the argument are omitted from the extended abstract. All proofs can be found in the full version of the paper at `http://arxiv.org/abs/math.PR/0509575`.

2. ALGORITHM OVERVIEW

Our reconstruction algorithm has two components. The probabilistic part, which consists in reconstructing estimates of sequences at internal nodes, borrows heavily from the work of [20] where Steel's conjecture is proved for the special case of balanced trees. The main tool there is recursive majority, as detailed in Subsection 2.1. Our main contribution lies in the combinatorial component of the algorithm, which is significantly more involved than in [20]. The combinatorial component is detailed in Subsection 2.2.

2.1 Properties of the majority function

In this subsection we quote some of the results we are using from [20] and explain briefly how they are used in our reconstruction algorithm. The results of [20] are stated assuming that the character values are ± 1 instead of $0/1$. Furthermore, instead of using the mutation probability $0 \le p(e) \le 1/2$, they use $\theta(e) = 1 - 2p(e)$ which satisfies $0 \le \theta(e) \le 1$. Note that in terms of θ we have reconstruction solvability whenever $\theta(e) \ge \theta > \theta_*$ for all e where $2\theta_*^2 = 1$.

For the CFN model both the majority algorithm [11] and recursive majority algorithm [17] are effective in reconstructing the root value. (For other models in general, most simple reconstruction algorithms are not effective all the way to the reconstruction threshold [18, 21, 13].)

The function $\mathrm{Maj} : \{-1,1\}^d \to \{-1,1\}$ is defined by

$$\mathrm{Maj}(x_1,\ldots,x_d) = \mathrm{sign}\left(\sum_{i=1}^{d} x_i + 0.5\omega\right)$$

where ω is ± 1 with prob $1/2$ and is independent of the x_i. In other words, the Maj outputs the majority value unless there is a tie in which case it outputs ± 1 with probability $1/2$ each.

DEFINITION 1. *Let $T = (V,E)$ be a tree rooted at ρ with leaf set ∂T. For functions $\theta' : E \to [0,1]$ and $\eta' : \partial T \to [0,1]$, let $CFN(\theta',\eta')$ be the CFN model on T where $\theta(e) = \theta'(e)$ for all e which is not adjacent to ∂T, and $\theta(e) = \theta'(e)\eta'(v)$ for all $e = (u,v)$, with $v \in \partial T$. Let*

$$\widehat{Maj}(\theta',\eta') = \mathbf{E}[+Maj(\sigma_{\partial T})|\sigma_\rho = +1] = \mathbf{E}[-Maj(\sigma_{\partial T})|\sigma_\rho = -1],$$

where σ is drawn according to $CFN(\theta',\eta')$.

For functions θ and η as above, we abbreviate by writing $\min\theta$ for $\min_{e\in E}\theta(e)$, $\max\eta$ for $\max_{v\in\partial T}\eta(v)$, etc. The function $\widehat{\mathrm{Maj}}$ measures how well majority calculates the character value at the root of the tree.

THEOREM 2. *[20] Let b and $\theta_{\min}$ be such that $b\theta_{\min}^2 > h^2 > 1$. Then there exist $\ell(b,\theta_{\min})$, $\alpha(b,\theta_{\min}) > h^\ell$ and $\beta(b,\theta_{\min}) > 0$, such that any $CFN(\theta,\eta)$ model on the ℓ-level b-ary tree satisfying $\min\theta \ge \theta_{\min}$ and $\min\eta \ge \eta_{min}$ must also satisfy:*

$$\widehat{Maj}(\theta,\eta) \ge \min\{\alpha\eta_{\min}, \beta\}. \tag{2}$$

The previous theorem allows to reconstruct the root state of an l-level balanced binary tree given values at the leaves. The estimate is guaranteed to have a positive correlation with the true value. For a balanced tree that contains more than l levels, the theorem can be applied recursively and, this way, one can estimate internal states deep inside the tree with correlation at least β.

Our main use of reconstructed sequences is in estimating distances between internal nodes of the tree. Note that, if nodes u and v obtain sequences $\sigma_u, \sigma_v \in \{\pm 1\}^k$ by the CFN model, then the following quantity

$$\widehat{\mathrm{Dist}}\,(\sigma_u,\sigma_v) = -\frac{1}{2}\log\left[\left(\frac{1}{k}\sum_{t=1}^{k}\sigma_u^t\sigma_v^t\right)_+\right], \tag{3}$$

measures the correlation between sequences σ_u and σ_v and serves as an estimate of how far nodes u and v are in the tree, i.e. it is an estimate of $d(u,v)$. However, we do not know the true sequences at the internal nodes of the tree, so we will apply (3) to reconstructed sequences $\hat{\sigma}_u, \hat{\sigma}_v$. The effect of this is discussed in the following comments regarding our application of Theorem 2:

1. Below, we consider general trees. In particular, when estimating the sequence at an internal node u, we apply Theorem 2 to a subtree "below" u, but this subtree is not balanced. This can be tackled by "completing" the subtree into a balanced tree and assuming that all added edges have length 0.

2. We are not given access to the true sequences at the internal nodes, but only estimated sequences. By Theorem 2, the reconstruction procedure introduces a "bias" in the sequence. One may think of this bias as an extra edge in the Markov model. Therefore, our estimate of the distance is itself biased (upwards). However, as shown in Figure 1, a correct estimate of the length of the internal edge of a *quartet* (i.e. a tree on four leaves) can still be obtained because the estimation of the length of the internal edge is unaffected by the biases at the leaves of the quartet.

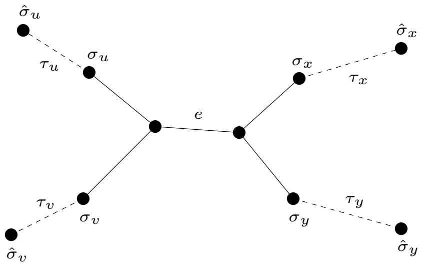


Figure 1: The estimation of $d(e)$ is not affected by the bias at the leaves, represented as dashed edges.

3. To apply Theorem 2, all edges in the subtree below the reconstructed node need to be "small enough". In our algorithm, some of the edges in these subtrees are actually paths in the true tree so we need to verify that these paths are sufficiently short.
4. For $\widehat{\mathrm{Dist}}\,(\hat{\sigma}_u, \hat{\sigma}_v)$ to be a "good" estimate of $d(\hat{u}, \hat{v})$ (where by $\hat{u}$ and $\hat{v}$ we denote the "biased" images of nodes u and v; see Figure 1), we need the biases at u and v to be independent. For this to hold, the subtrees we use to reconstruct sequences at u and v must be "disjoint", i.e. the path between them must go "above" u and v. This task is nontrivial when only partial information is known about the topology.

To use Theorem 2, it is important to make sure that properties 3 and 4 above hold whenever we perform a sequence reconstruction. One last point to note is that, given that we use $O(\log n)$ length sequences, only small (in fact $O(1)$) distances can be estimated with accuracy. This follows from standard concentration inequalities. Therefore, given a reconstructed subforest of the true tree and estimated sequences at its internal nodes, we only get local metric information about the "rest" of the tree.

2.2 The Reconstruction Algorithm

Recall that, in a binary tree, a *cherry* is a pair of leaves at graph distance 2. At a high level, our reconstruction algorithm proceeds from a simple idea: it builds the tree one level of cherries at a time. To see how this works, imagine that we had access to a "cherry oracle", i.e. a function $C(u, v, T)$ that returns the parent of the pair of leaves $\{u, v\}$ if the latter forms a cherry in the tree T (and say 0 otherwise). Then, we could perform the following "cherry picking" algorithm:

```
• Currently undiscovered tree: $T' := T$;
• Repeat until $T'$ is empty,
    – For all $(u, v) \in \mathcal{L}(T') \times \mathcal{L}(T')$, if
      $w := C(u, v, T') \neq 0$, set $\mathrm{Parent}(u) := \mathrm{Parent}(v) :=$
      $w$;
    – Remove from $T'$ all cherries discovered at
      this step;
```

Unfortunately, the cherry oracle cannot be simulated from short sequences at the leaves. Indeed, short sequences provide only local metric information on the structure of the tree. Nevertheless, the above scheme can be roughly followed by making a number of modifications, which we now describe briefly. The description of the algorithm uses the following notation and conventions:

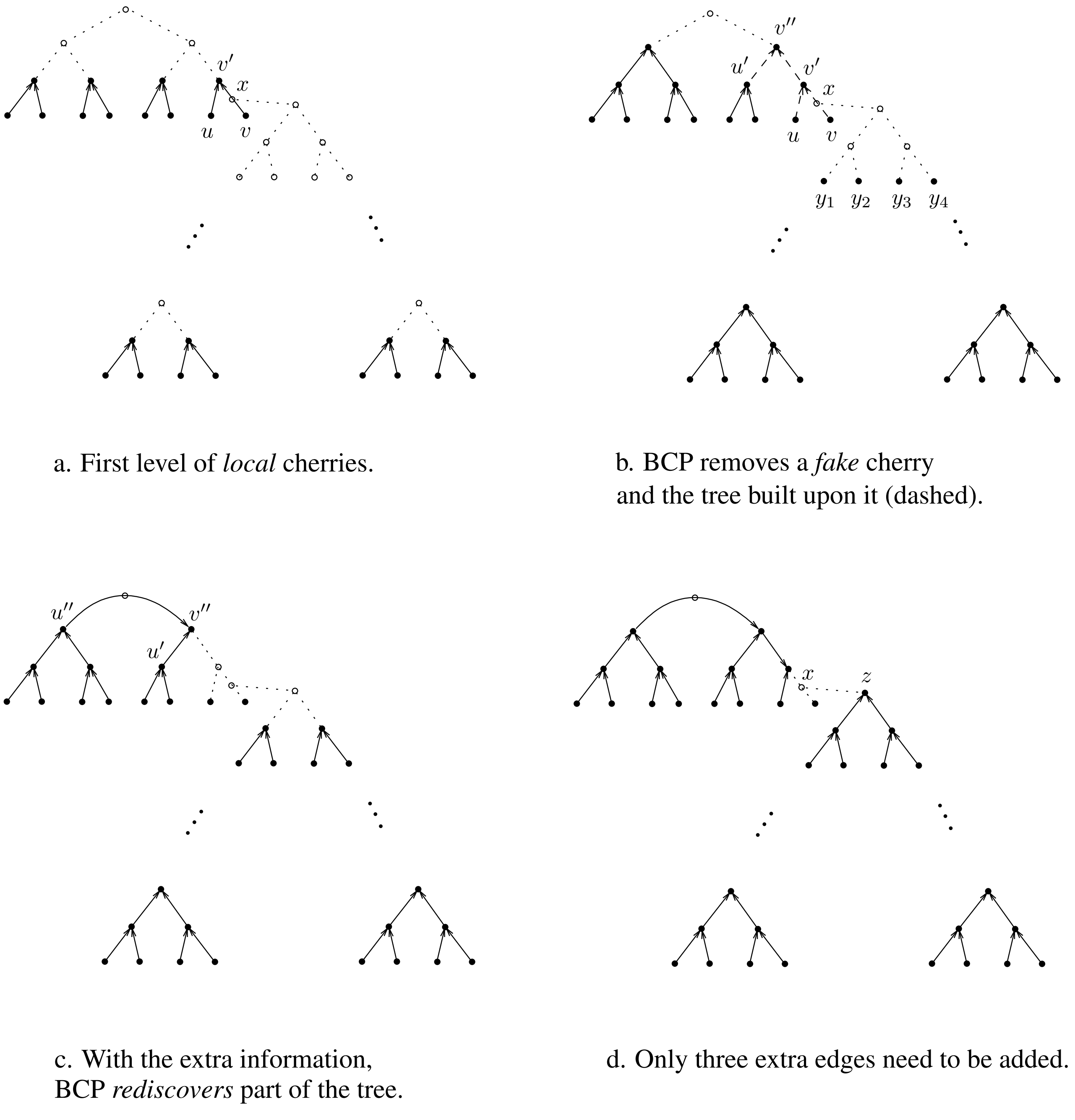
- $T^{\mathrm{Child}}_{\leq w}$ is the tree made of the descendants of w as defined by the descendance function Child.
- A *g-cherry* is a cherry where both edges have length less or equal to g.
- Let $M > 0$. Let T be a tree and F be the subforest of T where we keep all the leaves and only those nodes with the following property: they are on a path of length at most M between two leaves of T. We say that a pair of leaves $\{u, v\}$ is an *M-local g-cherry* in T if $\{u, v\}$ is a g-cherry in F and there are at least two other leaves u', v' s.t.
$$\max\{d(u, u'), d(u, v'), d(v, u'), d(v, v')\} \leq M$$
(the leaves u', v' will act as "witnesses" of the cherry $\{u, v\}$).
- A *pseudoleaf* is a current active node. The set of active nodes at iteration i of the algorithm will be denoted by $\widehat{L}_i$ and will correspond to the leaves of the currently undiscovered part of the tree.

The high-level idea of the algorithm, which we call BLINDFOLDED CHERRY PICKING (BCP), is to apply the cherry picking scheme above using the local metric information obtained from short sequences at the leaves and reconstructed sequences at internal nodes as outlined in the previous section. However, because of the local nature of our information, some of the cherries we pick will turn out *not* to be cherries. This only becomes apparent once a larger fraction of the tree is reconstructed, at which point a subroutine identifies the "fake" cherries and removes them.

Consider for instance the tree depicted in Figure 2a. It is made of a large complete binary tree (on the right) with a small 3-level complete binary tree attached to its root (on the left). All edges have length g, except (v, x), (x, v') and the two edges attached to the root which have length $g/2$. In the figure, the subtree currently discovered by BCP is made of solid arrows and full circles. The remaining (undiscovered) tree is in dotted lines and empty circles. Assume that the length of the sequences at the leaves allows us to estimate accurately distances up to $5g$ (the actual constants used by the algorithm can be found later).

a. Suppose we join into cherries all pairs of leaves that "look" like g-cherries in this local metric, i.e. they form a g-cherry in all quartets of leaves. In particular, these selected pairs of leaves form $5g$-local g-cherries. We are guaranteed to find all true g-cherries (actually we keep only those for which we can locate the top of the cherry). However, consider pairs of leaves such as u, v for which there is no local evidence that it does not form a cherry. Even though u, v is not a cherry, it is joined into a cherry by BCP. Figure 2a depicts the current forest after the first iteration of BCP. The top of the cherries are called pseudoleaves. Before proceeding further, we apply the majority function of the previous section to obtain reconstructed sequences at all pseudoleaves and recompute the local metric.

b. We subsequently proceed to join local g-cherries one level at a time, reconstructing internal sequences as we do so. After many iterations, we find ourselves in the situation of Figure 2b where most of the large complete tree has been reconstructed (assume for now that edges (u', v''), (v', v''), (u, v'), (v, v') represented in dashed lines are present). Now, the new information coming from sequences at $y_1, \ldots, y_4$ provides evidence that (u, v', v) is not a cherry and that there is in fact a node x on edge (v, v'). For example, the quartet $\{y_1, y_2, u, v\}$ suggests that u, v forms a cherry with a $3g/2$-edge, which cannot hold in a g-cherry. At this point, we remove the "fake" cherry (u, v', v) as well as all cherries built upon it, here only (u', v'', v'). Note that we have removed parts of the tree that were in fact reconstructed correctly (e.g., the path between u and u') in order to maintain properties 3 and 4 which are essential for the sequence re-



a. First level of *local* cherries.

b. BCP removes a *fake* cherry and the tree built upon it (dashed).

c. With the extra information, BCP *rediscovers* part of the tree.

d. Only three extra edges need to be added.

Figure 2: Illustration of BCP's unraveling.

construction component to work properly. This aspect of the algorithm is explained in more detail in Section 3.

c. Subsequently, BCP continues to join local cherries and "rediscovers" the parts of the tree that we removed. For instance, in Figure 2c, the edge (u', v'') is reconstructed again but this time it forms a cherry with (u'', v'') rather than (v', v'').

d. Eventually, the full tree is correctly reconstructed except maybe for a few (at most 3) remaining edges. Those are easy to add separately. For example in Figure 2d only the three edges around x remain to be uncovered. Note that the reconstructed tree has a root which is different from that of the original tree (i.e. the two trees differ as directed trees).

We now summarize the various routines needed for the reconstruction. Details are provided in figures 3, 4, 5, and 6.

1. DISTEST. This routine relies on the reconstructed sequences (computed in the routine SEQREC) and computes distances between pseudoleaves as indicated in Section 2.1 (see Figure 1). Details are omitted from this extended abstract.

2. CHERRYID. This routine identifies local cherries – these are pairs of pseudoleaves that appear as cherries under the local metric in all quartets.

3. SEQREC. This routine reconstructs sequences as indicated in Section 2.1. Details are omitted from this extended abstract.

4. FAKECHERRY. This routine identifies "fake" cherries. The routine is rather involved – see Figure 5 for the algorithm and Section 5 for proof of its correctness.

5. BUBBLE. This routine cleans up "fake" cherries and removes all cherries built upon "fake" cherries.

6. FOURPOINT. This routine uses the four-point method to find the right quartet split of a given quadruple of pseudoleaves and also computes estimates of the lengths of the edges using (5 times) a scheme similar to DISTEST. Details are omitted from this extended abstract.

3. ANALYSIS

In this section, we establish that BCP reconstructs the phylogeny correctly. There are two main technical aspects to the proof. The probabilistic part follows [20]. We focus rather on the combinatorial part where the novelty and complexity of BCP lies. There, we first establish a number of combinatorial properties of the current forest $\widehat{\mathcal{F}}_i$ grown by BCP. We then prove that the "correctly reconstructed subforest" of $\widehat{\mathcal{F}}_i$ increases in size at every iteration.

3.1 Preliminaries

The following notation will be used in the proofs: T is the phylogenetic tree that produced the data; $0 < f < g < +\infty$ are lower and upper bounds on the length of every edge in T, where $g < g^*$; $k = c \log n$ is the number of samples available at the leaves, where $c = c(f, g, \delta)$ is determined by the proof, δ being the probability of error. The proof yields a bound of $\exp(-c''(f, g)k) \leq n^{-\gamma}$ on the probability of failure for each distance estimation. We will have at most n^{10} distance estimations and will therefore require that $n^{10-\gamma} \leq \delta$.

In the following discussion, a *subtree* refers to a subgraph of a tree induced by a subset of the nodes. (We sometimes apply this definition to a directed tree, in which case we actually refer to the undirected version of the tree.)

DEFINITION 2 (EDGE DISJOINTNESS [19]). *Denote by* $\mathtt{path}_T(x, y)$ *the path (sequence of edges) connecting* x *to* y *in* T. *We say that two subtrees* T_1, T_2 *of* T *are* edge disjoint *if*

$$\mathtt{path}_T(u_1, v_1) \cap \mathtt{path}_T(u_2, v_2) = \emptyset,$$

for all $u_1, v_1 \in \mathcal{L}(T_1)$ *and* $u_2, v_2 \in \mathcal{L}(T_2)$. *We say that* T_1, T_2 *are* edge sharing *if they are not edge disjoint. (If* T_1 *and* T_2 *are directed, we take this definition to refer to their underlying undirected version.)*

DEFINITION 3 (COLLISIONS). *Suppose that* T_1 *and* T_2 *are edge disjoint rooted subtrees of* T. *We say that* T_1 and T_2 collide at distance d, *if the path* $\mathtt{path}_T(\rho(T_1), \rho(T_2))$ *has non-empty intersection with* $\mathcal{E}(T_1) \cup \mathcal{E}(T_2)$ *and the length of the shortest path between* T_1 *and* T_2 *is at most* d. *In other words,* T_1 *and* T_2 *collide at distance* d, *if the shortest path between* T_1 *and* T_2 *is of length at most* d *and this path does not contain either* $\rho(T_1)$ *or* $\rho(T_2)$.

DEFINITION 4 (FIXED SUBFOREST). *Let* $\mathcal{F}$ *be a rooted directed edge disjoint subforest of* T *with implicit descendance relationship* Child. *Let* $u \in \mathcal{V}(\mathcal{F})$. *We say that* u *is* fixed *if* $T^{\mathrm{Child}}_{\leq u}$ *is fully reconstructed (or in other words,* $T^{\mathrm{Child}}_{\leq u}$ *can be obtained from* T *by removing (at most) one edge adjacent to* u*). Note that descendants of a fixed node are fixed themselves. We denote by* $\mathcal{F}^*$ *the (directed) subforest of* $\mathcal{F}$ *made of all fixed nodes of* $\mathcal{F}$. *We say that* $\mathcal{F}^*$ *is the* maximal fixed subforest *of* $\mathcal{F}$.

3.2 Probabilistic Analysis

Assume that g satisfies the inequality $2e^{-2g} > 1$, which defines the space of values of g for which full reconstruction with $O(\log n)$ samples at the leaves is not forbidden by [20]. Also, fix the constant $\varepsilon < f/2$ such that if $g' = g + \varepsilon$ then g' satisfies $2e^{-2g'} > 1$. Fix $\varepsilon_2 < \varepsilon/8$. In both subroutines DISTEST and FOURPOINT, we take a number of samples large enough so that all distances smaller or equal to $25g$ are computed within ε_2 with high probability when the biases at endpoints are independent. The next lemma bounds the error on estimated distances between pseudoleaves in the presence of collisions. The proof is omitted from this extended abstract.

PROPOSITION 1 (LOCAL METRIC). *Suppose that* $\mathcal{F} = \{T_1, T_2, \ldots, T_\alpha\}$ *is a forest of rooted full binary trees with the following properties:*

1. [Edge Disjointness] $\{T_1, T_2, \ldots, T_\alpha\}$ *is an edge disjoint subforest of* T.
2. [Edge Lengths] *All edges in* $\mathcal{F}$ *have length at most* g'.
3. [Collisions] *There is no collision at distance* $20g$ *in* $\mathcal{F}$.

Then, using routine DISTEST *to estimate the distances between every pair of roots of trees in* $\mathcal{F}$, *the following property is satisfied by the estimated distance* $\hat{d}$ *with probability at least* $1 - n^{-\gamma+2}$, *where* $\gamma = \gamma(f, g, \epsilon_2)$:

$$\hat{d}(u, v) \leq 12g \vee d(u, v) \leq 12g \Rightarrow |d(u, v) - \hat{d}(u, v)| < \varepsilon_2.$$

3.3 Combinatorial Analysis

The following proposition establishes a number of properties of the forest grown by BCP. The proof can be found in Section 5. Recall that $\widehat{L}_i$ is the set of pseudoleaves (i.e. active nodes) at iteration i (see the algorithm in Figure 3 for a precise definition).

PROPOSITION 2 (PROPERTIES OF $\widehat{\mathcal{F}}_i$). *The following properties hold at the beginning of* BCP*'s i-th iteration,* $\forall i \geq 1$:

1. [Edge Disjointness] $\widehat{\mathcal{F}}_i = \left\{ T^{\mathrm{Child}}_{\leq u} \,:\, u \in \widehat{L}_i \right\}$ *is an edge disjoint subforest of* T.

```
Algorithm Blindfolded Cherry Picking (Input: samples at the leaves; Output: estimated topology)
  • 0) Initialization: $i := 0$; $j := n$; $\widehat{L}_0 := [n]$; $\forall \alpha \in [n]$, $\hat{\sigma}_\alpha := \sigma_\alpha$;
  • 1) Distance Estimation: For all $(u, v) \in \widehat{L}_i \times \widehat{L}_i$, set $\hat{d}_i(u, v) := \text{DistEst}(u, v)$ [see text];
  • 2) Cherry Identification: $\widehat{L}_{i+1} := \widehat{L}_i$; $\widehat{C}_i := \emptyset$; For all $(u_0, v_0) \in \widehat{L}_i \times \widehat{L}_i$ such that $u_0 < v_0$, apply CherryID $(u_0, v_0)$;
  • 3) Sequence Reconstruction: For all $(u, w, v) \in \widehat{C}_i$, set $\hat{\sigma}_w := \text{SeqRec}(u, w, v)$ [see text];
  • 4) Fake Cherry Detection: For all $(u_0, u_1) \in \widehat{L}_{i+1} \times \widehat{L}_{i+1}$ with $u_0 < u_1$, perform FakeCherry$(u_0, u_1)$;
  • 5) Termination: If $|\widehat{L}_{i+1}| \leq 3$, join pseudoleaves in $\widehat{L}_{i+1}$ (star if 3, single edge if 2) and compute the length of the missing edges; Output the reconstructed tree; Else, set $i := i + 1$, and go to Step 1.
```

Figure 3: Algorithm Blindfolded Cherry Picking.

```
Algorithm CherryID (Input: pair of pseudoleaves $(u_0, v_0)$);
  • IsCherry := TRUE;
  • Test 1 [Distance less than $2g + \varepsilon_2$]: If $\hat{d}_i(u_0, v_0) > 2g + \varepsilon_2$, then IsCherry := FALSE;
  • Test 2 [Local cherry]: Let $R_{5g}$ be the set of all $(u_1, v_1) \in \widehat{L}_i \times \widehat{L}_i$ such that $u_1 < v_1$, $\{u_0, v_0\} \cap \{u_1, v_1\} = \emptyset$, and $\max\left\{\hat{d}_i(x_0, x_1) \,:\, x_\iota \in \{u_\iota, v_\iota\}\right\} \leq 5g + \varepsilon_2$. Then:
      – If $R_{5g}$ is empty, then IsCherry := FALSE; Otherwise, perform FourPoint$(u_0, v_0, u_1, v_1)$; If $(u_0, v_0)$ is not a $(g+\varepsilon_2)$-cherry in $\{u_0, v_0, u_1, v_1\}$, then set IsCherry := FALSE;
  • If IsCherry = TRUE,
      – Set $j := j+1$ and $w := j$; Add $w$ to $\widehat{L}_{i+1}$, add $(u_0, w, v_0)$ to $\widehat{C}_i$, and remove $u_0, v_0$ from $\widehat{L}_{i+1}$; Update parenting relationships; Let $\hat{\gamma}(u_0, w)$ and $\hat{\gamma}(v_0, w)$ be the estimated lengths of edges $(u_0, w)$ and $(v_0, w)$.
```

Figure 4: Subroutine CherryID.

```
Algorithm FakeCherry (Input: pseudoleaves $u_0, u_1$);
  • For $\iota = 0, 1$, set $T_\iota := T^{\text{Child}}_{\leq u_\iota}$ and denote $C_\iota$ the set of cherries in $T_\iota$;
  • Compute all pairwise distances $\hat{d}$ between $T_0$ and $T_1$ using DistEst (some of these distances are actually wrong);
  • $\forall (\kappa_0, \kappa_1) \in C_0 \times C_1$ with $\kappa_\iota = (x_\iota, z_\iota, y_\iota)$, set $\hat{d}_M(\kappa_0, \kappa_1) = \max\{\hat{d}(v_0, v_1) \,:\, v_\iota \in \{x_\iota, y_\iota\}\}$;
  • For $\iota = 0, 1$, if $u_{1-\iota}$ is not a leaf, do
      – Set $\text{Stop}_\iota$ := FALSE;
      – Inside Loop: For all $\kappa_\iota = (x_\iota, z_\iota, y_\iota) \in C_\iota$,
          ∗ Set $C' := \{\kappa \in C_{1-\iota} \,:\, \hat{d}_M(\kappa_\iota, \kappa) \leq 25g\}$; Break from Inside Loop if empty;
          ∗ While $C' \neq \emptyset$ and $\text{Stop}_\iota$ = FALSE,
              · Let $\kappa = (x, z, y)$ be the lowest cherry in $C'$;
              · [Collision Test 1] Let $w$ be the node at the intersection of the triplet $\{x_\iota, x, y\}$ (note that it may be that $w \neq z$); use the four point method on $\{x_\iota, x, y\}$ to compute the distance between $x$ and $w$, say $h$ (using a scheme similar to that in routine DistEst); if $|h - \hat{\gamma}(x, z)| > 2\varepsilon_2$ then set $\text{Test}_1$ := TRUE;
              · [Collision Test 2] Perform the previous step again with $y_\iota$ rather than $x_\iota$ and $\text{Test}_2$ rather than $\text{Test}_1$;
              · If in both $\text{Test}_1$ = TRUE and $\text{Test}_2$ = TRUE, then set $\text{Stop}_\iota$ := TRUE and set $w_{1-\iota} := z$; otherwise remove $\kappa$ from $C'$.
  • For $\iota = 0, 1$,
      – If $\text{Stop}_\iota$ = TRUE, perform Bubble$(w_{1-\iota}, u_{1-\iota})$.
```

Figure 5: Subroutine FakeCherry.

```
Algorithm Bubble (Input: node $w$, pseudoleaf $u$);
  • Add the children of $w$ to $\widehat{L}_{i+1}$;
  • Set $z := w$;
  • While $z \neq u$,
      – Add Sister$(z)$ to $\widehat{L}_{i+1}$;
      – Set $z$ := Parent$(z)$.
  • Remove $u$ from $\widehat{L}_{i+1}$;
```

Figure 6: Subroutine Bubble.

2. [Edge Lengths] $\forall u \in \widehat{L}_i$, *$T^{\text{Child}}_{\leq u}$ is a rooted full binary tree with edge lengths at most g'.*
3. [Weight Estimation] *The estimated lengths of the edges in $\widehat{\mathcal{F}}_i$ are within ε_2 from their right values.*
4. [Collisions] *There is no collision at distance $20g$.*

The next proposition establishes that, in a precise sense, the algorithm makes progress at every iteration. The proof can be found in Section 5.

PROPOSITION 3 (PROGRESS). *Let*

$$\widehat{\mathcal{F}}_i = \left\{ T^{\text{Child}}_{\leq u} \,:\, u \in \widehat{L}_i \right\}$$

(where $\widehat{L}_i$ is taken at the beginning of iteration i) for all $i \geq 0$ with corresponding maximal fixed subforest $\widehat{\mathcal{F}}^*_i$. Then for all $i \geq 0$ (before the termination step), $\widehat{\mathcal{F}}^*_i \subseteq \widehat{\mathcal{F}}^*_{i+1}$ and $|\mathcal{V}(\widehat{\mathcal{F}}^*_{i+1})| > |\mathcal{V}(\widehat{\mathcal{F}}^*_i)|$.

4. PROOF OF THE MAIN THEOREM

Proof of Theorem 1: By Proposition 2, the current forest is correctly reconstructed. By Proposition 3, after $O(n)$ iterations, there remain at most three nodes in $\widehat{L}_i$ and at that point, from Proposition 2, $\widehat{\mathcal{F}}_i \equiv \widehat{\mathcal{F}}^*_i$. Therefore the remaining task is to join the remaining pseudoleaves and there is only one possible topology. So when the BCP algorithm terminates, it outputs the tree T (as an undirected tree) with high probability and all estimated edges are within ε_2 of their correct value. This concludes the proof. The tightness of the value $g^* = \frac{\log 2}{4}$ is justified by the polynomial lower bound [20] on the number of required characters if the mutation probability p on all edges of the tree satisfies $2(1-2p)^2 < 1$. ■

5. COMBINATORIAL ANALYSIS: PROOFS

Proof of Proposition 2:

$i = 0$: The set $\widehat{L}_0$ consists of the leaves of T. The claims are therefore trivially true.

$i > 1$: Assume the claims are true at the beginning of the i-th iteration. By doing a step-by-step analysis of the i-th iteration, we show that the claims are still true at the beginning of the $(i+1)$-st iteration. The following lemma follows from Proposition 1.

LEMMA 1 (CORRECTNESS OF DISTEST). *After the completion of step 1, for all $u, v \in \widehat{L}_i$:*

$$\hat{d}_i(u,v) \leq 12g \vee d(u,v) \leq 12g \Rightarrow |d(u,v) - \hat{d}_i(u,v)| < \varepsilon_2.$$

Proof: From the induction hypothesis (Claim 4), it follows that in the beginning of the i-th iteration there is no collision at distance $20g$. So the claim follows from Proposition 1. (A small detail to note is that the sequences at the nodes of the forest were reconstructed in different steps of the algorithm. However, the subtrees that were used for the reconstruction of each node are exactly those in the statement of Proposition 1.) ■

Next, we analyze the routine CHERRYID.

LEMMA 2 (CORRECTNESS OF CHERRYID). *Let u, v be the input to* CHERRYID. *Let $T' = T - \widehat{\mathcal{F}}_i$ (keeping the nodes in $\widehat{L}_i$) at the beginning of the i-th iteration. Then we have the following.*

- *If $\{u, v\}$ is a $5g$-local g-cherry in T', then it passes all screening tests in* CHERRYID.

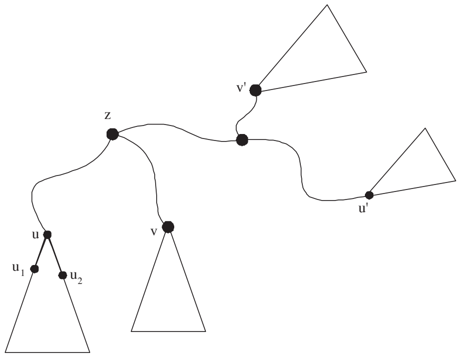


Figure 7: Estimating distance $d(u,z)$.

- *If $\{u, v\}$ is not a $(5g + 2\varepsilon_2)$-local $(g + 2\varepsilon_2)$-cherry in T', then it is rejected by at least one of the tests in* CHERRYID.

Proof: This result is implied by the following claim. Every time FOURPOINT is called by CHERRYID, on 4 nodes u, v, u', v' where $\{u, v\}$ is the candidate cherry and $\{u', v'\}$ is the witness, then

- the trees rooted at u, v, u', v' do not collide,
- the split returned by FOURPOINT is the correct split,
- all edge lengths of the quartet joining u, v, u', v' are estimated within ε_2 of their correct value.

We now prove this claim. The subroutine FOURPOINT is called by CHERRYID when the following assumptions are satisfied.

- $\hat{d}_i(u,v) \leq 2g + \varepsilon_2$,
- $\max\left\{\hat{d}_i(u,u'), \hat{d}_i(u,v'), \hat{d}_i(v,u'), \hat{d}_i(v,v')\right\} \leq 5g + \varepsilon_2$.

From Lemma 1, it follows that the above estimated distances are within ε_2 of their correct values. An application of the triangle inequality gives $d(u',v') < 11g$ so that $|\hat{d}_i(u',v') - d(u',v')| < \varepsilon_2$ as well. In fact, all pairwise distances of nodes in the set $\{u, v, u', v'\}$ are smaller than $11g$. Hence, by the induction hypothesis (Claim 4), the four trees rooted at u, v, u', v' do not collide. Therefore, from Proposition 1 and the fact that the quartet joining u, v, u', v' has width at most $11g$, the split of nodes u, v, u', v' is found correctly by the four point method and the length of the internal edge of the quartet is estimated within ε_2 of its correct value.

It remains to show that all other edges of the quartet are estimated within ε_2 of their correct value. Above, we have established that the quartet split computed for the nodes u, v, u', v' is correct. Also, by the induction hypothesis (Claim 1) the trees rooted at u, v, u', v' are edge disjoint subtrees of T. Suppose the quartet joining u, v, u', v' is as depicted in Figure 7 and we are estimating $d(u,z)$. Without loss of generality, assume the algorithm applies the four point method to the set of nodes $\{u_1, u_2, v, v'\}$. It is easy to see that every pair of nodes in the set $\{u_1, u_2, v, v'\}$ has distance $< 7g$ and so the width of the quartet is $< 7g$. Thus, Proposition 1 can be applied and the internal edge of the quartet, i.e. (u,z), is estimated within ε_2 of its correct value.■

We are now in a position to prove claims 1, 2, and 3.

LEMMA 3 (CLAIMS 1, 2, AND 3). *At the beginning of the $(i+1)$-st iteration, claims 1, 2, and 3 of the induction hypothesis hold.*

Proof: Since FAKECHERRY only removes edges from the current forest, it is enough to prove that after the completion of Step 3 the resulting forest satisfies claims 1, 2, and 3.

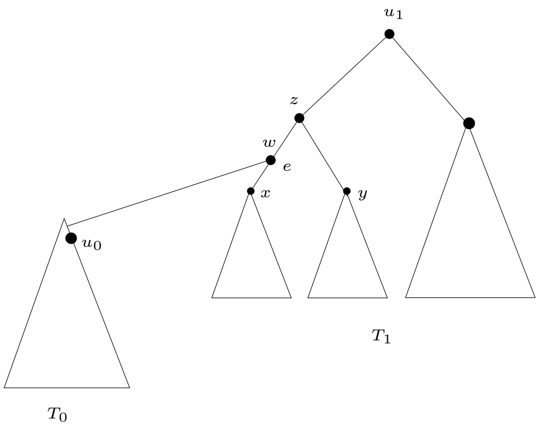


Figure 8: Illustration of routine FAKECHERRY.

Claim 1. Suppose the resulting forest is not edge disjoint. Also, suppose that, along the execution of Step 2, the forest stopped being edge disjoint when cherry (x, z, y) was added to $\widehat{C}_i$. Then one of the following must be true:

1. There is a pseudoleaf $z' \in \widehat{L}_i \cap \widehat{L}_{i+1}$ such that $\mathtt{path}_T(x, y)$ is edge sharing with $T^{\mathrm{Child}}_{\leq z'}$. Then it is not hard to see that there is a collision in $\left\{T^{\mathrm{Child}}_{\leq u} \,:\, u \in \widehat{L}_i\right\}$ at distance $3g$ which contradicts the induction hypothesis (Claim 4).

2. There is a pseudoleaf $z' \in \widehat{L}_{i+1} \backslash \widehat{L}_i$ such that $\mathtt{path}_T(x, y)$ is edge sharing with $T^{\mathrm{Child}}_{\leq z'}$. We can distinguish the following subcases.

   - $(x', z', y') \in \widehat{C}_i$ and $\mathtt{path}_T(x, y)$ is edge sharing with $\mathtt{path}_T(x', y')$: in this case $xy|x'y'$ is not the correct split and, by Lemma 2, it is not hard to see that CHERRYID rejects $\{x, y\}$ when performing Test 2.
   - Otherwise, it is not hard to see that there is a collision at distance $3g$ in $\left\{T^{\mathrm{Child}}_{\leq u} \,:\, u \in \widehat{L}_i\right\}$, which contradicts the induction hypothesis (Claim 4).

Claim 2. Follows directly from the description of the algorithm: a cherry (u, x, v) is added to $\widehat{C}_i$ only if $d(u, x)$ and $d(v, x)$ are estimated to be at most $g + \varepsilon_2$, so that the true edge lengths are less than g' by Lemma 1 and the choice of ε_2.

Claim 3. This follows from proof of Lemma 2. ■

It remains to prove Claim 4. This follows immediately from the following analysis of FAKECHERRY.

LEMMA 4 (COLLISION REMOVAL). *Let $u_0, u_1 \in \widehat{L}_{i+1}$ after step 3 of the i-th iteration. Suppose $T_0 \equiv T^{\mathrm{Child}}_{\leq u_0}$ and $T_1 \equiv T^{\mathrm{Child}}_{\leq u_1}$ collide at distance $20g$. Then after an application of* FAKECHERRY *the remaining subtrees of T_0 and T_1 do not collide at distance $20g$.*

Proof: Suppose by contradiction that, after applying FAKECHERRY, there is a collision between two remaining subtrees T_0' and T_1' of T_0 and T_1 respectively. Then, without loss of generality, there exists a path of length at most $20g$ between an internal node u_0 of T_0 and the inside of an edge e of T_1 such that the subpath lying on T_0 is above u_0. See Figure 8. Let x_0, y_0 be the children of u_0 (the case where u_0 is a leaf is similar). Consider the set

$$A_{0\to 1} = \{v \in \mathcal{V}(T_1) \,:\, e \text{ is not in the subtree of } T_1 \text{ rooted at } v\}\,.$$

It is not hard to see that for all $v \in A_{0\to1}$ the reconstructed sequence at node v is positively correlated with the true sequence and the bias is independent of the biases of the reconstructed sequences at x_0 and y_0. Thus, from Section 3.2, it follows that $\forall v \in A_{0\to1} : d(x_0, v) \leq 25g \Rightarrow |\hat{d}_i(x_0, v) - d(x_0, v)| < \varepsilon_2$ and similarly for y_0. Let $A'_{0\to1} \subseteq A_{0\to1}$ be the set that contains the nodes $v \in A_{0\to1}$ such that $\hat{d}_i(x_0, v) \leq 25g$ and $\hat{d}_i(y_0, v) \leq 25g$. Since the collision is at distance $20g$ it follows that $A'_{0\to1}$ is nonempty and in fact contains at least the lower endpoint of edge e and its sibling in T_1. The routine FAKECHERRY scans the cherries of T_1 starting from the lowest cherry and going up and, in fact, only considers cherries formed by pairs of nodes in $A'_{0\to1}$. Therefore, by the proof of Lemma 2 (correctness of weight estimations), it stops when it reaches the cherry formed by the lower endpoint of e and its sibling. It then calls BUBBLE which in turn removes e. Note that since T is a tree, there is only one path between T_0 and T_1 and, therefore, at most one fake cherry can be found by FAKECHERRY. Also, from the proof of Proposition 3 below, it follows that FAKECHERRY does not stop before reaching this fake cherry. This leads to a contradiction. ■

This concludes the proof of Proposition 2. ■

Proof of Proposition 3: We first argue that $\widehat{\mathcal{F}}_i^* \subseteq \widehat{\mathcal{F}}_{i+1}^*$. Note that the only routine that removes edges is BUBBLE when called by FAKECHERRY. Because $\widehat{\mathcal{F}}_i^*$ is fully reconstructed, it suffices to show that collisions identified by FAKECHERRY are actual collisions or lie "above" an actual collision—i.e. are on a cherry located on the path between the actual collision and the root. Indeed, since BUBBLE removes only edges "above" presumed collisions, this would then imply that no edge in $\widehat{\mathcal{F}}_i^*$ can be removed. We now prove the claim by analyzing the behavior of FAKECHERRY. We use the notation defined in the routine. Consider the collision tests in FAKECHERRY. The key point is to observe the following:

- if cherry $\kappa = (x, z, y)$ is in $\widehat{\mathcal{F}}_i^* \cap T_{1-\iota}$ and $\kappa_\iota = (x_\iota, z_\iota, y_\iota)$, then at least one of x_ι or y_ι has a reconstruction bias that is independent from the bias at both x and y; therefore this "correct" witness will not observe a collision (using Proposition 1);
- if cherry κ is in $(T - \widehat{\mathcal{F}}_i^*) \cap T_{1-\iota}$, then all the cherries above κ (on the path to $u_{1-\iota}$) cannot be in $\widehat{\mathcal{F}}_i^*$ and therefore applying BUBBLE to κ does not modify $\widehat{\mathcal{F}}_i^*$.

This proves that $\widehat{\mathcal{F}}_i^* \subseteq \widehat{\mathcal{F}}_{i+1}^*$. To prove that $|\mathcal{V}(\widehat{\mathcal{F}}_{i+1}^*)| > |\mathcal{V}(\widehat{\mathcal{F}}_i^*)|$, assume $\widehat{\mathcal{F}}_i = \{T_1, \ldots, T_\alpha\}$ and $\mathcal{F}' \equiv T - \widehat{\mathcal{F}}_i = \{T_1', \ldots, T_\beta'\}$. $\mathcal{F}'$ is the forest obtained from T by removing all the edges in the union of the trees $T_1, \ldots, T_\alpha$. The nodes of $\mathcal{F}'$ are all the endpoints of the remaining edges. Since the trees $T_1, \ldots, T_\alpha$ are edge disjoint, the set $\mathcal{F}'$ is in fact a subforest of T. Each leaf v in $\mathcal{F}'$ satisfies exactly one of the following:

- **Collision Node:** v a leaf of $\mathcal{F}'$ that belongs to a path connecting two vertices in $T_a \in \widehat{\mathcal{F}}_i$ but is not the root of T_a (it lies in the "middle" of an edge of T_a).
- **Fixed Pseudoleaf:** v is a root of a fully reconstructed tree $T_a \in \widehat{\mathcal{F}}_i$ (i.e. T_a is also in $\widehat{\mathcal{F}}_i^*$);
- **Colliding Pseudoleaf:** v is a root of a tree $T_a \in \widehat{\mathcal{F}}_i$ that is not in $\widehat{\mathcal{F}}_i^*$ (the tree T_a contains a collision).

We need the following definition.

DEFINITION 5 (BUNDLE). *A* bundle *is a group of four leaves such that:*

- *Any two leaves are at topological distance at most* 5*;*

- *It includes at least one cherry.*

A fixed bundle *is a bundle in $\mathcal{F}'$ whose leaves are fixed pseudoleaves.*

We now prove that $\mathcal{F}'$ contains at least one fixed bundle. This immediately implies the second claim. Indeed, it is not hard to see that the cherry in the fixed bundle is found by CHERRYID during the $(i+1)$-st iteration.

LEMMA 5 (FIXED BUNDLE). *Assume Proposition 2 holds at the end of the i-th iteration and let $\mathcal{F}'$ as above have at least two internal nodes. Then, $\mathcal{F}'$ contains at least one fixed bundle.*

Proof: We first make a few easy observations:

1. A tree with 4 or more leaves contains at least one bundle. (To see this: merge all cherries into leaves; repeat at most twice.)
2. Because of Claim 4 in Proposition 2, collision nodes are at distance at least $20g$ from any other leaf in $\mathcal{F}'$. Therefore, if a tree in $\mathcal{F}'$ contains a collision node, then it has > 4 nodes and, from the previous observation, it contains at least one bundle. Moreover, this bundle cannot contain a collision node (since in a bundle all leaves are close).
3. From the previous observations, we get the following: if a tree in $\mathcal{F}'$ contains a collision, then either it has a fixed bundle, or it has at least one colliding pseudoleaf.

It is then easy to conclude. Assume there is no collision node in $\mathcal{F}'$. Then, there cannot be any colliding pseudoleaf either and it is easy to see that $\mathcal{F}'$ is actually composed of a single tree all of which leaves are fixed. Then there is a fixed bundle by Observation 1 above.

Assume on the contrary that there is a collision node. Let T'_b be a tree in $\mathcal{F}'$ with such a node. Then by Observation 3, T'_b either has a fixed bundle, in which case we are done, or it has a colliding pseudoleaf, say v. In the latter case, let T_a be the tree in $\widehat{\mathcal{F}}_i$ whose root is v. The tree T_a contains at least one collision node which it shares with a tree in $\mathcal{F}'$, say $T'_{b'}$. Repeat the argument above on $T'_{b'}$, and so on.

Note that in each step we "exit" a tree $T_c \in \widehat{\mathcal{F}}_i$ via a node that is not the root of $T_c \in \widehat{\mathcal{F}}_i$ and enter a new tree $T_d \in \widehat{\mathcal{F}}_i$ at its root. Since T is a tree, this process cannot continue forever, and we eventually find a fixed bundle. ■

6. CONCLUSION

The proof of Steel's Conjecture [26] provides tight results for the phylogenetic reconstruction. However, many theoretical and practical questions remain:

- Can the results be extended to other mutation models? Can the results be extended to deal with "rates across sites" (see e.g. [9])?
- We have found a tight value g^*_{JC} for the "robust phylogenetic reconstruction problem" for the JC model. What is the optimal value for usual phylogenetic reconstruction? Is it the same as the critical $g_{q=4}$ value for the reconstruction problem for the $q = 4$ Potts model on the binary tree? We note that it is a long standing open problem to find $g_{q=4}$. The best bounds known are given in [21, 16].

7. ACKNOWLEDGMENTS

C.D. is supported by CIPRES (NSF ITR grant # NSF EF 03-31494). E.M. is supported by a Miller fellowship in Statistics and Computer Science, by a Sloan fellowship in Mathematics and by NSF grants DMS-0504245 and DMS-0528488. S.R. is supported by CIPRES (NSF ITR grant # NSF EF 03-31494), FQRNT, NSERC and a Loève Fellowship. S.R. thanks Martin Nowak and the Program for Evolutionary Dynamics at Harvard University where part of this work was done. C. D. and E. M. thank Satish Rao for interesting discussions. E.M. thanks M. Steel for his enthusiastic encouragement for studying the connections between the reconstruction problem and phylogeny.

8. REFERENCES

[1] N. Berger, C. Kenyon, E. Mossel, and Y. Peres. Glauber dynamics on trees and hyperbolic graphs. *Probab. Theory Related Fields*, 131(3):311–340, 2005. Extended abstract by Kenyon, Mossel and Peres appeared in proceedings of 42nd IEEE Symposium on Foundations of Computer Science (FOCS) 2001, 568–578.

[2] P. M. Bleher, J. Ruiz, and V. A. Zagrebnov. On the purity of the limiting Gibbs state for the Ising model on the Bethe lattice. *J. Statist. Phys.*, 79(1-2):473–482, 1995.

[3] J. A. Cavender. Taxonomy with confidence. *Math. Biosci.*, 40(3-4), 1978.

[4] J. Chang. Full reconstruction of markov models on evolutionary trees: identifiability and consistency. *Math. Biosci.*, 137(51–73), 1996.

[5] P. L. Erdös, M. A. Steel, L. A. Székely, and T. A. Warnow. A few logs suffice to build (almost) all trees (part 1). *Random Structures Algorithms*, 14(2):153–184, 1999.

[6] W. S. Evans, C. Kenyon, Y. Y. Peres, and L. J. Schulman. Broadcasting on trees and the Ising model. *Ann. Appl. Probab.*, 10(2):410–433, 2000.

[7] M. Farach and S. Kannan. Efficient algorithms for inverting evolution. *J. ACM.*, 46(4):437–449, 1999.

[8] J. S. Farris. A probability model for inferring evolutionary trees. *Syst. Zool.*, 22(4):250–256, 1973.

[9] J. Felsenstein. *Inferring Phylogenies*. Sinauer, New York, New York, 2004.

[10] H. O. Georgii. *Gibbs measures and phase transitions*, volume 9 of *de Gruyter Studies in Mathematics*. Walter de Gruyter & Co., Berlin, 1988.

[11] Y. Higuchi. Remarks on the limiting Gibbs states on a $(d+1)$-tree. *Publ. Res. Inst. Math. Sci.*, 13(2):335–348, 1977.

[12] D. Ioffe. On the extremality of the disordered state for the Ising model on the Bethe lattice. *Lett. Math. Phys.*, 37(2):137–143, 1996.

[13] S. Janson and E. Mossel. Robust reconstruction on trees is determined by the second eigenvalue. *Ann. Probab.*, 32:2630–2649, 2004.

[14] T. H. Jukes and C. Cantor. Mammalian protein metabolism. In H. N. Munro, editor, *Evolution of protein molecules*, pages 21–132. Academic Press, 1969.

[15] F. Martinelli, A. A. Sinclair, and D. Weitz. Glauber dynamics on trees: boundary conditions and mixing time. *Comm. Math. Phys.*, 250(2):301–334, 2004.

[16] F. Martinelli, A. Sinclair, and D. Weitz. Fast mixing for independent sets, colorings, and other models on trees. In *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms*, pages 449–458, 2004.

[17] E. Mossel. Recursive reconstruction on periodic trees. *Random Structures Algorithms*, 13(1):81–97, 1998.

[18] E. Mossel. Reconstruction on trees: beating the second eigenvalue. *Ann. Appl. Probab.*, 11(1):285–300, 2001.

[19] E. Mossel. Distorted metrics on trees and phylogenetic forests. IEEE Comp. Biol. and Bioinformatics, 2004.

[20] E. Mossel. Phase transitions in phylogeny. *Trans. Amer. Math. Soc.*, 356(6):2379–2404, 2004.

[21] E. Mossel and Y. Peres. Information flow on trees. *Ann. Appl. Probab.*, 13(3):817–844, 2003.

[22] E. Mossel and M. Steel. A phase transition for a random cluster model on phylogenetic trees. *Math. Biosci.*, 187(2):189–203, 2004.

[23] J. Neyman. Molecular studies of evolution: a source of novel statistical problems. In S. S. Gupta and J. Yackel, editors, *Statistical desicion theory and related topics*, pages 1–27. 1971.

[24] C. Semple and M. Steel. *Phylogenetics*, volume 22 of *Mathematics and its Applications series*. Oxford University Press, 2003.

[25] F. Spitzer. Markov random fields on an infinite tree. *Ann. Probability*, 3(3):387–398, 1975.

[26] M. Steel. My Favourite Conjecture. http://www.math.canterbury.ac.nz/∼mathmas/conjecture.pdf, 2001.