Optimal Sequence Length Requirements for Phylogenetic Tree Reconstruction with Indels

Arun Ganesh
arunganesh@berkeley.edu
University of California, Berkeley
Berkeley, California, USA

Qiuyi (Richard) Zhang
10zhangqiuyi@berkeley.edu
University of California, Berkeley
Berkeley, California, USA

ABSTRACT

We consider the phylogenetic tree reconstruction problem with insertions and deletions (indels). Phylogenetic algorithms proceed under a model where sequences evolve down the model tree, and given sequences at the leaves, the problem is to reconstruct the model tree with high probability. Traditionally, sequences mutate by substitution-only processes, although some recent work considers evolutionary processes with insertions and deletions. In this paper, we improve on previous work by giving a reconstruction algorithm that simultaneously has $O(\text{poly}\log n)$ sequence length and tolerates constant indel probabilities on each edge. Our recursively-reconstructed distance-based technique provably outputs the model tree when the model tree has $O(\text{poly}\log n)$ diameter and discretized branch lengths, allowing for the probability of insertion and deletion to be non-uniform and asymmetric on each edge. Our polylogarithmic sequence length bounds improve significantly over previous polynomial sequence length bounds and match sequence length bounds in the substitution-only models of phylogenetic evolution, thereby challenging the idea that many global misalignments caused by insertions and deletions when p_{indel} is large are a fundamental obstruction to reconstruction with short sequences.

We build upon a signature scheme for sequences, introduced by Daskalakis and Roch, that is robust to insertions and deletions. Our main contribution is to show that an averaging procedure gives an accurate reconstruction of signatures for ancestors, even while the explicit ancestral sequences cannot be reconstructed due to misalignments. Because these signatures are not as sensitive to indels, we can bound the noise that arise from indel-induced shifts and provide a novel analysis that provably reconstructs the model tree with $O(\text{poly}\log n)$ sequence length as long as the rate of mutation is less than the well known Kesten-Stigum threshold. The upper bound on the rate of mutation is optimal as beyond this threshold, an information-theoretic lower bound of $\Omega(\text{poly}(n))$ sequence length requirement exists.

CCS CONCEPTS

• **Mathematics of computing → Probabilistic inference problems**; • **Theory of computation → Design and analysis of algorithms**.

KEYWORDS

Phylogenetic reconstruction, distance methods, sequence length requirements

ACM Reference Format:
Arun Ganesh and Qiuyi (Richard) Zhang. 2019. Optimal Sequence Length Requirements for Phylogenetic Tree Reconstruction with Indels. In *Proceedings of the 51st Annual ACM SIGACT Symposium on the Theory of Computing (STOC '19), June 23–26, 2019, Phoenix, AZ, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3313276.3316345

1 INTRODUCTION

The phylogenetic tree reconstruction problem is a fundamental problem in the intersection of biology and computer science. Given a sample of DNA sequence data, we attempt to infer the phylogenetic tree that produced such samples, thereby learning the structure of the hidden evolutionary process that underlies DNA mutation. The inference of phylogenies from molecular sequence data is generally approached as a statistical estimation problem, in which a model tree, equipped with a model of sequence evolution, is assumed to have generated the observed data, and the properties of the statistical model are then used to infer the tree. Various approaches can be applied for this estimation, including maximum likelihood, Bayesian techniques, and distance-based methods [25].

Many stochastic evolution models start with a random sequence at the root of the tree and each child inherits a mutated version of the parent sequence, where the mutations occur i.i.d. in each site of sequence. The most basic model is the Cavender-Farris-Neyman (CFN) [6, 20] symmetric two-state model, where each sequence is a bitstring of 0/1 and mutations are random i.i.d. substitutions with probability $p_{sub}(e)$ for each edge e. More complicated molecular sequence evolution models (with four states for DNA, 20 states for amino acids, and 64 states for codon sequences) exist but typically, the theory that can be established under the CFN model can also be established for the more complex molecular sequence evolution models used in phylogeny estimation [12]. For simplicity, we will work with sequences that are bitstrings of 0/1 only.

In addition to computational efficiency, a reconstruction algorithm should also have a small *sequence length requirement*, the minimum sequence length required to provably reconstruct the model tree with high probability. Many methods are known to be statistically *consistent*, meaning that they will provably converge

STOC '19, June 23–26, 2019, Phoenix, AZ, USA

ACM ISBN 978-1-4503-6705-9/19/06…$15.00
https://doi.org/10.1145/3313276.3316345

to the true tree as the sequence lengths increase to infinity, including maximum likelihood [24] and many distance-based methods [3, 25]. Methods that require only $O(\text{poly}(n))$ sequence length are known as *fast converging* and recently, most methods have been shown to be fast converging under the CFN model, including maximum likelihood [24] (if solved exactly) and various distance-based methods [4, 11, 12, 17, 19, 23, 26]. More impressively, some algorithms achieve $O(\text{poly}\log(n))$ sequence length requirement but require more assumptions on the model tree and they always need a tighter upper bound on g, the maximum edge length (in the CFN model, the length of an edge is defined as $\lambda(e) = -\ln(1 - 2p_{sub}(e))$) [4, 7, 17, 18, 22, 23]. Specifically, these methods are based on reconstruction of ancestral sequences and can provably reconstruct when g is smaller than what is known as the Kesten-Stigum threshold, which is $\ln(\sqrt{2})$ [18] for the CFN model. Intuitively, when g is small, the edges have a small enough rate of mutation that allows for a concentration effect on estimators of ancestral sequences; however, when g is past a certain threshold, a phase transition occurs and reconstruction becomes significantly harder. Essentially matching information-theoretic lower bounds show that $\Omega(\log(n))$ sequence lengths are needed to reconstruct when g is smaller than the Kesten-Stigum threshold and $\Omega(\text{poly}(n))$ lengths are needed beyond this threshold [24].

One of the biggest drawbacks of the CFN model is that it assumes that mutations only occur as a substitution and that bitstrings remain aligned throughout the evolutionary process down the tree. This allows for an relatively easy statistical estimation problem since each site can be treated as an i.i.d. evolution of a bit down the model tree. In practice, global misalignments from insertions and deletions (indels) breaks this site-independent assumption. Therefore, most phylogenetic tree reconstruction algorithms must first apply a multiple sequence alignment algorithm before attempting to reconstruct the sequence under a purely substitution-induced CFN model, with examples being CLUSTAL[13], MAFFT[15], and MUSCLE[9]. However, the alignment process is based on heuristics and lacks a provable guarantee. Even with a well-constructed pairwise similarity function, the alignment problem is known to be NP-hard [10]. Furthermore, it has been argued that such procedures create systematic biases [16, 27].

Incorporating indels directly into the evolutionary model and reconstruction algorithm is the natural next step but the lack of site-wise independence presents a major difficulty. However, in a breakthrough result by [8], the authors show that indels can be handled with $O(\text{poly}(n))$-length sequences, using an alignment-free distance-based method. Also, in [1], the authors provide an $O(\text{poly}\log(n))$-length sequence length requirement for tree reconstruction but can only handle indel probabilities of $p_{indel} = O(1/\log^2(n))$, which is quite small since a string of length $O(\log(n))$ will only experience $O(1)$ indels as it moves $O(\log(n))$ levels down the tree. Similarly, [2] provides a method for reconstruction given k-length sequences as long as $p_{indel} = O(k^{-2/3}(\log n)^{-1})$ for k sufficiently large.

In this paper, we almost close the gap by showing that provable reconstruction can be done in polynomial time with $O(\text{poly}\log(n))$ sequence length rather than $O(\text{poly}(n))$ sequence length, even when the probability of insertion and deletion are non-uniform, asymmetric, and p_{ins}, p_{del} are bounded by a constant. We do this by first constructing signature estimators that exhibit 1) robustness to indel-induced noise in expectation and 2) low variance when our mutation rate is below the Kesten-Stigum threshold. Then, these reconstructed signatures are the key components of a distance estimator, inspired by estimators introduced in [22], that uses $O(\log(n))$ conditionally independent reconstructed signatures to derive a concentration result for accurately estimating the distance between any nodes a, b. Our bounds on the rates of substitution, insertion and deletion are essentially optimal since we show that as long as our overall mutation rate is less than the Kesten-Stigum threshold, we can reconstruct with a sequence length requirement of $O(\text{poly}\log(n))$, matching lower bounds up to a constant in the exponent. Our result implies that the noise introduced by insertions and deletions can be controlled in a similar fashion as the noise introduced by substitutions, breaking the standard intuition that misalignments caused by indels would inherently lead to a significantly higher sequence length requirement.

In Section 2, we provide a general overview of our model and methods. In Section 3, we demonstrate that tree reconstruction with poly-logarithmic sequence length requirement for the symmetric case when the insertion and deletion probabilities are the same. In Section 4, we extend our results to the asymmetric case and then we conclude with future directions.

2 PRELIMINARIES

2.1 Model and Methods

For simplicity, we'll consider the following model of phylogenetic evolution, which we call **CFN-Indel**, as seen in [1, 8]. We note that our analysis can be extended to the more general sequence models, such as GTR, with standard techniques. We start with a tree T with n leaves, also known as the model tree. There is a length k bitstring at the root chosen uniformly at random from all length k bitstrings. Each other node in T inherits its parent's bitstring, except the following perturbations are made simultaneously and independently for each edge e in the tree,

- Each bit is flipped with probability $p_{sub}(e)$.
- Each bit is deleted with probability $p_{del}(e)$.
- Each bit inserts a random bit to its right with probability $p_{ins}(e)$.

Definition 1. *For an edge e, we denote that* **length** *or* **rate of mutation** *of that edge as*

$$\lambda(e) = -[\ln(1-2p_{sub}(e))+\ln(1-p_{del}(e))-\frac{1}{2}\ln(1+p_{ins}(e)-p_{del}(e))]$$

Note that higher $p_{ins}(e), p_{del}(e), p_{sub}(e)$ leads to a higher edge length and that this $\lambda(e)$ is nonnegative: the only possibly negative term is the term $\frac{1}{2}\ln(1 + p_{ins}(e) - p_{del}(e))$, and if this term is negative, its absolute value is less than the term $-\ln(1 - p_{del}(e))$. We provide some intuition for this definition: to estimate distances between nodes, our algorithm will look at the correlation between their bitstrings. We can show the correlation between two bitstrings decays by roughly $(1 - 2p_{sub}(e))(1 - p_{del}(e))$ for each edge e on the path between the corresponding nodes, justifying the first two terms in the above definition. The term $\frac{1}{2}\ln(1+p_{ins}(e)-p_{del}(e))$ is included to match a normalization term in our definition of correlation, which is needed to account for differing bitstring lengths throughout the tree. Thus, $\lambda(e)$ represents the rate of change that an edge produces in the sequence evolution process and therefore captures a notion of the length of an edge. For two nodes, a, b, let $P_{a,b}$ denote the unique path between them and let $d(a,b)=\sum_{e\in P_{a,b}}\lambda(e)$ be our true distance measure between a, b. Note that d is a tree metric or an **additive** distance matrix.

As with all phylogenetic reconstruction guarantees, we assume upper and lower bounds $0<\lambda_{min}\leq\lambda(e)<\lambda_{max}$ (in literature, λ_{min} is often denoted with f and λ_{max} with g). Similar to [22], we work in the Δ-branch model, where for all edges e, we have $\lambda(e)=\tau_e\lambda_{min}$ for some positive integer τ_e. The *phylogenetic reconstruction problem* in the CFN-Indel model is to reconstruct the underlying model tree T with high probability given the bitstrings at the leaves.

In this paper, we establish the claim that reconstruction can be done with $k=O(\log^{\kappa} n)$ bits given that λ_{max} is the well known Kesten-Stigum threshold and $\lambda_{min}=\Omega(1/\text{poly}\log(n))$. Note that this allows $p_{sub}, p_{del}, p_{ins}$ to be constant and that the upper bound λ_{max} is information-theoretically optimal. We will first state the main theorem when $p_{ins}(e)=p_{del}(e)$ for all edges e and our model tree T is balanced, which we call the *symmetric* case.

Theorem 2.1. *Assume that $\lambda_{max}\leq\ln(\sqrt{2})$ is less than the Kesten-Stigum threshold and $\lambda_{min}=\Omega(1/poly\log(n))$. Furthermore, assume that we are in the symmetric case where $p_{ins}(e)=p_{del}(e)$ for all edges e. Then for a sufficiently large κ, with $O(\log^{\kappa} n)$ sequence length, there is an algorithm TreeReconstruct that can reconstruct the phylogenetic tree with high probability under the CFN-Indel model.*

Our high level idea is to estimate the additive distance matrix $d(a,b)=\sum_{e\in P_{a,b}}\lambda(e)$ and use standard distance-based methods in phylogeny to reconstruct the tree. As in [8], our estimator of the distance relies on correlation calculations using blocks of consecutive sequence sites of length $l=\lfloor k^{1/2+\zeta}\rfloor$ for some small constant $0<\zeta<1/2$[1]. Therefore, we have approximately $L=\lfloor\frac{k}{l}\rfloor$ total disjoint blocks. For the bitstring at node a and $1\leq i\leq L$ let $\Delta_{a,i}$ be the signed difference between the number of zeroes in bits $(i-1)l+1$ to il of the bitstring at node a (for convenience, we call this block i of node a) and the expected number of zeroes, $\frac{l}{2}$.

Definition 2. *For a node a, we define the* **signature** *of the corresponding sequence, s_a, as a vector with the i-th coordinate as*

$$s_{a,i}=\Delta_{a,i}/\sqrt{l}$$

We will use the signature of a sequence as the only information used in distance computations. Specifically, we note that signatures of two nodes far apart in the tree should have a low correlation, whereas the signatures of two nodes close together should have a high correlation. To realize this intuition, we prove concentration of signature correlations even under indel-induced noise and show that this concentration property can be applied recursively up the tree for signature reconstruction.

Note that signatures are robust to indels as a single indel can only slightly change a signature vector, although it can have a global effect and change many signature coordinates at once. This leads to the somewhat accurate intuition that indels can introduce more noise than substitutions, as they can only produce local changes. However, because each coordinate of a signature, $s_{a,i}$, is an average over blocks of large size, we can control the indel-induced noise by 1) showing the signature is almost independent coordinate-wise and 2) applying concentration to produce an accurate distance estimator. Next, we introduce a novel analysis of indel-induced noise in signature reconstruction that allows for recursion up the tree. Specifically, we show that the indel-induced noise decays at about the same rate as the signal of the correlations between nodes, as we move down the tree.

Putting it together, we are able to recursively reconstruct the signatures for all nodes using simply leaf signatures, showing that these estimators have low variance of $O(\log^2 n)$ as long as the edge length is less than the Kesten-Stigum threshold. Intuitively, this phenomenon occurs because the number of samples increases at a faster rate than the decay in correlation. By averaging over signatures in a sequence, this reduces the noise to $O(1/\text{poly}\log(n))$. Finally, we show that this is sufficient for a highly accurate distance estimator via a Chernoff-type bound with $O(\log n)$ conditionally independent estimators to achieve tight concentration, leading to a recursive reconstruction algorithm via a simple distance-based reconstruction algorithm.

In the asymmetric case, let $\mathcal{D}_{max}$ be the maximum depth of our model tree. When $p_{ins}(e)\neq p_{del}(e)$, we see that if $p_{del}(e)>p_{ins}(e)+\kappa\frac{\log\log n}{\mathcal{D}_{max}}$, our model will generate a nearly zero-length bitstring with high probability, as noticed in [1]. Therefore, reconstruction is impossible. Furthermore, note that if $\mathcal{D}_{max}>k^2$, where k is the sequence length at the root, then the standard deviation in leaf sequence lengths due to insertion and deletion is on the order of $\Theta(\sqrt{\mathcal{D}_{max}})=\Omega(k)$ even if $p_{del}(e)=p_{ins}(e)$. Again, we encounter nearly zero-length bitstrings with decent probability.

Otherwise, for any constants $\alpha,\beta>0$ if we have $\mathcal{D}_{max}\leq\log^{\alpha} n$ and $|p_{del}(e)-p_{ins}(e)|\leq\frac{\beta\log\log n}{\mathcal{D}_{max}}$, we show that if κ is large enough[2] our algorithm can still reconstruct the underlying model tree, albeit through a more complicated analysis.

Theorem 2.2. *Assume that $\lambda_{max}\leq\ln(\sqrt{2})$ and $\lambda_{min}=\Omega(1/poly\log(n))$. Also assume for some constants α,β that $|p_{ins}(e)-p_{del}(e)|\leq\frac{\beta\log\log n}{\mathcal{D}_{max}}$ for all edges e, where $\mathcal{D}_{max}\leq\log^{\alpha} n$ is the maximum depth of the model tree. Then for a sufficiently large constant κ, when the root has $O(\log^{\kappa} n)$ sequence length, there is an algorithm TreeReconstruct that can reconstruct the phylogenetic tree with high probability under the CFN-Indel model.*

Throughout the paper, we will use the following observation:

Observation 1. *Let X be any random variable, and $\mathcal{E}$ an event which occurs with probability $1-n^{-\Omega(\log n)}/B$, where B is any upper bound on $|X|$. Then $\mathbb{E}[X|\mathcal{E}]$ and $\mathbb{E}[X]$ differ by at most $n^{-\Omega(\log n)}$.*

In this paper, all random variables we use can be upper bounded in magnitude by $O(\text{poly}(n))$. Thus for events $\mathcal{E}$ which occur with probability $1-n^{-\Omega(\log n)}$, we may use $\mathbb{E}[X|\mathcal{E}]$ and $\mathbb{E}[X]$ interchangeably as they only differ by at most $n^{-\Omega(\log n)}$, which will not affect

[1] The best setting of ζ will depend on other parameters introduced in the paper. For simplicity, the reader may wish to think of ζ as 1/4.

[2] We note that the dependence of our κ value on α, β does not match the previously mentioned lower bounds.

any of our calculations. However, interchanges will often still be justified in proofs.

3 RECONSTRUCTION WITH BALANCED TREES AND SYMMETRIC PROBABILITIES

In this section, we are in the symmetric case and assume $p_{del}(e) = p_{ins}(e)$ for every edge. We also assume that the model tree is perfectly balanced. Both these assumptions will be relaxed later and our results are extended to the asymmetric case in the next section.

We first demonstrate that some regularity conditions on the underlying bitstrings in the model tree hold with high probability. Given these regularity conditions, we show that the concentration for an recursive signature estimator provides a good distance estimator between any two nodes in the tree. Finally, we present our final distance-based reconstruction algorithm. Unless otherwise specified, proofs assume κ is a sufficiently large constant depending only on ζ, λ_{min} and the parameters ϵ, δ in the lemma/theorem statements. Our algorithmic construction will fix values of ϵ, δ, and thus works for some sufficiently large κ.

3.1 High Probability Tree Properties

Before we begin to describe our method for reconstructing the tree, we observe a few regularity properties about the bitstrings in the tree and prove that these properties hold with high probability.

Definition 3. *For an edge (a, b) from parent node a to child node b, we say that the j_bth bit of the bitstring at b is **inherited** from the j_ath bit of the bitstring at a if:*

- *The j_ath bit of the bitstring at a does not participate in a deletion on the edge (a, b).*
- *The number of insertions minus the number of deletions on the edge (a, b) in bits 1 to $j_a - 1$ of the bitstring at a is $j_b - j_a$.*

For a that is an ancestor of b, we extend this definition by saying that the j_bth bit of b is inherited from the j_ath bit of the bitstring at a if for the unique a-b path $x_0 = a, x_1, \ldots x_k = b$, there are $j_0 = j_a, j_1 \ldots j_k = j_b$ such that for any i, the j_ith bit of the bitstring at x_i is inherited from the j_{i-1}th bit of the bitstring at x_{i-1}.

To account for the case where $a = b$, we say that for any node a, the jth bit of a's bitstring is inherited from the jth bit of a's bitstring.

Definition 4. *For any two nodes a, b, the j_ath bit of the bitstring at a and the j_bth bit of the bitstring at b are **shared** if both are inherited from the jth bit of the bitstring at the least common ancestor of a and b for some j.*

Definition 5. *For any two nodes a and b, we say that the jth bit of the bitstring at a **shifts** by m bits on the path from a to b if there is j' such that the $|j' - j| = m$ and the jth bit of the bitstring at a and the j'th bit of the bitstring at b are shared.*

It will simplify our analysis to assume all bitstrings are length at least k, which might not happen if the length of the root bitstring is k. Instead, we will let the length of the root bitstring be $2k$, and then by the following lemma all of the leaf bitstrings have k bits to look at.

Lemma 1. *If the bitstring at the root has length $2k$, then with probability $1 - n^{-\Omega(\log n)}$, the bitstring at all nodes have length at least k and at most $4k$.*

The next two regularity properties show that the bit shifts are in fact also small and that the number of excess zeros on a consecutive sequence of bits is small. They both follow from independence and concentration of binomials.

Lemma 2. *With probability $1 - n^{-\Omega(\log n)}$, no bit shifts by more than $4 \log^2 n\sqrt{k}$ bits on any path in the tree.*

Lemma 3. *With probability $1 - n^{-\Omega(\log n)}$, for all nodes a, the number of zeroes in any consecutive sequence of length m sequence in a's bitstring differs from $m/2$ by at most $\sqrt{m} \log n$. Consequently, $|s_{a,i}| \le O(\log n)$.*

We defer the proofs of Lemmas 1, 2, and 3 to Section A.

3.2 Distance Estimator

We define $\mathcal{E}_{reg}$ to be event that the high-probability regularity assumptions that are proven in Lemma 1, 2, and 3 all hold. Using correlations of signatures, we can define the distance estimator of two leafs a, b, $\widetilde{C}(a, b)$ analogously to [8].

$$\widetilde{C}(a, b) = \frac{2}{L} \sum_{i=1}^{L/2} s_{a,2i+1} s_{b,2i+1}$$

In the case when indels do not occur, standard techniques can be used to show that for any leafs a, b, $\widetilde{C}(a, b)$ has an expectation that exponentially decays with respect to $d(a, b)$. The exponential decay comes from the observation that since the mutations can be viewed as a Markov transition from one state to the other, the correlations between states exponentially decays. Therefore, a back-of-the-envelope calculation gives $\mathbb{E}[\widetilde{C}(a, b)] \approx \exp(-\sum_{e \in P(a,b)} \lambda(e))\mathbb{E}[\widetilde{C}(a, a)] \approx \exp(-d(a, b))$.

We show that in the presence of indels, such an expectation still holds with $O(1/\text{poly}\log(n))$ relative error. Key to our surprisingly small relative error, even when the insertion and probability errors are as large as a constant, is the observation that the indel-induced noise also decays exponentially with respect to $d(a, b)$.

Lemma 4. *For any two nodes a, b in the tree, and any i, $\mathbb{E}[s_{a,i} s_{b,i}] = \frac{1}{4}(1 \pm O(\log^{-\kappa\zeta+2} n)) \exp(-d(a, b))$*

We defer the proof to Section A.

Corollary 5. *For any two nodes a, b,*

$$\mathbb{E}[\widetilde{C}(a, b)] = \frac{1}{4}(1 \pm O(\log^{-\kappa\zeta+2} n)) \exp(-d(a, b))$$

Despite the indels, we can show that the odd-index blocks are almost independent conditioned on $\mathcal{E}_{reg}$, i.e. in our analysis, we introduce a shift-invariant blockwise-independent signature scheme that is provably similar to our actual signature scheme. Thus, with high probability, we can also derive a tight concentration of our distance estimator that only uses signature correlations. We defer the proof and details to Section A.

Lemma 6. *Let $\delta > 0$ be any constant and $\epsilon = \Omega(\log^{\max\{-\kappa(1/2-\zeta)+2\delta+6, -\kappa\zeta/2+\delta+3\}}(n))$. Then for nodes a, b such that $d(a, b) < \delta \log\log(n)$, then $|-\ln(4\widetilde{C}(a, b)) - d(a, b)| < \epsilon$ with probability at least $1 - n^{-\Omega(\log n)}$.*

3.3 Signature Reconstruction

The lemmas proven so far show that the estimator $-\ln(4\widetilde{C}(a,b))$ suffices to reconstruct the tree up to height $\delta \log\log n$. To reconstruct past this height, we will come up with a recursive estimator of the signatures of internal nodes, and then using this estimator build a more robust distance estimator for nodes at height above $\delta \log\log n$.

Let L_h be the nodes at height $h = 0, ..., \log n$, where L_0 contains all the leaves. For a node $a \in L_h$, let A be the set of leaves that are descendants of a and we expect $|A| = 2^h$. Define the following estimator of $s_{a,i}$

$$\hat{s}_{a,i} = \frac{1}{|A|} \sum_{x \in A} e^{d(x,a)} s_{x,i}$$

In the next few lemmas, we demonstrate that this signature estimator exhibits 1) robustness to indel-induced noise in expectation and 2) low variance when λ_{max} is below the Kesten-Stigum threshold. The ultimate purpose of signature reconstruction is to introduce a distance estimator that uses $O(\log(n))$ conditionally independent reconstructed signatures to derive a concentration result for the distance between any nodes a, b. The definition of $\hat{s}_{a,i}$ comes from similar intuition to that of Lemma 4: we expect that each edge on the path from a to some descendant x adds multiplicative decay to the correlation between a's and x's bitstrings, so as the next lemma formalizes, it should be that $\mathbb{E}[s_{x,i}] \approx s_{a,i} e^{-d(x,a)}$.

Lemma 7. *Let $\mathcal{E}$ denote the bitstring at the node a and* $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. *Then, for a leaf x that is a descendant of a,* $\mathbb{E}[s_{x,i}|\mathcal{E}] = e^{-d(x,a)}(s_{a,i} + \nu_{a,i})$, *where* $|\nu_{a,i}| \leq 8k^{1/4} \log^2 n/\sqrt{l} = O(\log^{-\kappa\zeta/2+2} n)$.

We defer the proof to the full paper.

Corollary 8. *Let $\mathcal{E}$ denote the bitstring at the node a and* $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. *Then* $\mathbb{E}[\hat{s}_{a,i}|\mathcal{E}] = s_{a,i} + \nu_{a,i}$, *where* $|\nu_{a,i}| \leq O(\log^{-\kappa\zeta/2+2} n)$.

Next, we show that the signature estimator $\hat{s}_{a,i}$ has $O(\log^2 n)$ variance, which relies on the fact that the variance reduction due to averaging is greater than the variance increase due to mutation when the mutation rate is less than the Kesten-Stigum threshold. It might be surprising that as we move up the tree, the variance of the estimator stays unchanged. However, since the correlations between two nodes exponentially decays in the distance, each term in the signature estimator becomes more "independent", allowing for a tight variance bound.

Lemma 9. *Let $\mathcal{E}$ denote the bitstring at the node a and* $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. *Then,* $\mathbb{E}[\hat{s}^2_{a,i}|\mathcal{E}] = O(\log^2 n)$ *as long as* $\lambda_{max} < \ln\sqrt{2}$.

Proof.

$$\mathbb{E}[\hat{s}^2_{a,i}|\mathcal{E}] = \frac{1}{|A|^2} \sum_{x,y \in A} e^{d(x,a)+d(y,a)} \mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}]$$

To analyze $\mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}]$, let $x \wedge y$ be the least common ancestor of x, y and let $\mathcal{E}'$ denote the bitstring of $x \wedge y$. Note that conditioned on $\mathcal{E}'$, $s_{x,i}, s_{y,i}$ are independent and by Lemma 7:

$$\begin{aligned}
\mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}] &= \mathbb{E}[\mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}, \mathcal{E}']] \\
&= \mathbb{E}[\mathbb{E}[s_{x,i}|\mathcal{E}']\mathbb{E}[s_{y,i}|\mathcal{E}']] \\
&= e^{-d(x,y)} \mathbb{E}[(s_{x\wedge y,i} + \delta_{x\wedge y,i})^2|\mathcal{E}]
\end{aligned}$$

Then, since $\mathcal{E}_{reg}|\mathcal{E}$ is a high probability event and noting that the quantity $(s_{x\wedge y,i} + \delta_{x\wedge y,i})^2$ is at most $\log^2 n$ and thus conditioning on an event that happens with probability $1 - n^{-\Omega(\log n)}$ does not change its expectation by more than $n^{-\Omega(\log n)}$, we get $\mathbb{E}[(s_{x\wedge y,i} + \delta_{x\wedge y,i})^2] \leq O(\log^2 n)$ and thus $\mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}] \leq e^{-d(x,y)} \cdot O(\log^2 n)$.

This gives:

$$\begin{aligned}
\mathbb{E}[\hat{s}^2_{a,i}|\mathcal{E}] &\leq \frac{1}{|A|^2} \sum_{x,y \in A} e^{d(x,a)+d(y,a)} \mathbb{E}[s_{x,i} s_{y,i}|\mathcal{E}] \\
&\leq O(\log^2 n) \frac{1}{|A|^2} \sum_{x,y \in A} e^{2d(a, x\wedge y)}
\end{aligned} \tag{1}$$

Now, for a fixed x, note that 1/2 of $y \in A$ satisfies $d(a, x \wedge y) \leq \lambda_{max}$ and and 1/4 of them satisfies $d(a, x\wedge y) \leq 2\lambda_{max}$ and so on. Since $e^{2\lambda_{max}} < 2$,

$$\frac{1}{|A|} \sum_{y \in A} e^{2d(a, x\wedge y)} \leq \left[(1/2)e^{2\lambda_{max}} + (1/4)e^{4\lambda_{max}} + ...\right] = O(1)$$

Finally, by symmetry,

$$\mathbb{E}[\hat{s}^2_{a,i}|\mathcal{E}] \leq O(\log^2 n) \frac{1}{|A|} \sum_{x \in A} \frac{1}{|A|} \sum_{y \in A} e^{2d(a, x\wedge y)} = O(\log^2 n)$$

□

3.4 Distance Estimators

Distance computations can be done with reconstructed signatures by analogously defining

$$\hat{C}(a,b) = \frac{2}{L} \sum_{i=1}^{L/2} \hat{s}_{a,2i+1} \hat{s}_{b,2i+1}$$

Although we may use $\hat{C}(a,b)$ directly as an estimator for the distance between a, b, the variance in the reconstructed signature is still too high for the necessary concentration. To provide the concentration, we use many conditionally independent estimators.

For two nodes a, b, consider creating the distance estimator $\hat{d}(a,b)$ as follows. For some height $\Delta h = \delta \log\log n$, consider the nodes that are descendants of a, b exactly Δh below a, b respectively; order them arbitrarily as $a_1, ..., a_{2^{\Delta h}}$ and $b_1, ..., b_{2^{\Delta h}}$, as in Figure 1. Next, compute $\widetilde{d}(a_j, b_j) = -\ln(e^{d(a_j,a)+d(b_j,b)} 4\hat{C}(a_j, b_j))$, which is an estimator for $d(a,b)$. Note that we have $2^{\Delta h} = \Omega(\log n)$ of these estimators. We will aggregate these estimators in order to derive high probability concentration of the aggregate around $-\ln(4\widetilde{C}(a,b))$, allowing us to use Lemma 6 to show concentration of the aggregate around the true distance.

So, we proceed with the analogous construction that Roch presents for accuracy amplification for the substitution-only model

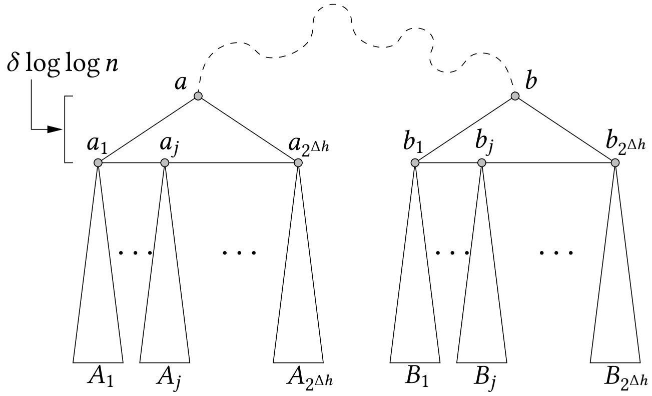

Figure 1: Concentration via conditionally independent estimators. Slightly modified from a figure in [22]

[22]. Consider the set of estimates $S_h(a,b) = \left\{\widetilde{d}(a_j, b_j)\right\}_{j=1}^{2^h}$. We will use a median-like measure to aggregate these estimates. For each j, we let r_j be the minimum radius of an interval centered on $\widetilde{d}(a_j, b_j)$ that captures at least 2/3 of the other points in $S_h(a,b)$.

$$r_j = \inf\left\{ r > 0 : \left| \{ j' \neq j : |\widetilde{d}(a_j, b_j) - \widetilde{d}(a_{j'}, b_{j'})| \leq r\} \right| \geq \frac{2}{3} 2^h \right\}$$

Then, if $j^* = \arg\min_j r_j$, then our distance estimator is $\hat{d}(a,b) = \widetilde{d}(a_{j^*}, b_{j^*})$.

Lemma 10 (Deep Distance Computation: Small Diameter). *For any constant $\delta > 0$, let a, b be nodes at height at least $\delta \log\log n$ such that $d(a,b) \leq \delta \log\log n$. If $\lambda_{max} < \ln(\sqrt{2})$, $|\hat{d}(a,b) - d(a,b)| < \epsilon$ with high probability.*

Proof. Let ξ denote the set of variables that are the underlying true bitstrings at nodes in $\{a_j\}_{j=1}^{2^{\Delta h}}$ and $\{b_j\}_{j=1}^{2^{\Delta h}}$. Throughout this proof, we will condition ξ unless otherwise stated. Notice that we can translate all high probability results even upon conditioning. Note if the unconditioned probability $\Pr(\mathcal{E}_{reg}) > 1 - n^{-\Omega(\log n)}$, the law of total expectation and a simple Markov bound shows us that $\Pr(\Pr(\mathcal{E}_{reg}|\xi) > 1 - n^{-\Omega(\log n)}) > 1 - n^{-\Omega(\log n)}$, where the outer probability is taken over instantiation of ξ. This allows us to condition and establish independence of $\hat{s}_{a_j,i}$ and $\hat{s}_{b_j,i}$, while preserving high probability results.

By Lemma 7, with high probability, $\mathbb{E}[\hat{s}_{a_j,i}] = s_{a_j,i} + \delta_{a_j,i}$ where $|\delta_{a_j,i}| \leq O(\log^{-\kappa\zeta/2+2} n)$. Furthermore, by Lemma 3, $|s_{a_j,i}| \leq O(\log n)$ with high probability. Symmetrically, these bounds hold for b_j. Therefore, we see that

$$\begin{aligned}
\mathbb{E}[\hat{C}(a_j, b_j)] &= \frac{2}{L}\sum_{i=1}^{L/2} \mathbb{E}[\hat{s}_{a_j,2i+1}\hat{s}_{b_j,2i+1}] \\
&= \frac{2}{L}\sum_i s_{a_j,2i+1}s_{b_j,2i+1} + O(\log^{-\kappa\zeta/2+3} n) \qquad (2)\\
&= \widetilde{C}(a_j, b_j) + O(\log^{-\kappa\zeta/2+3} n)
\end{aligned}$$

Furthermore, we can bound the variance by using Lemma 9. We first bound the covariance of terms in $\hat{C}(a_j, b_j)$:

Lemma 11. *Conditioned on ξ, for $i \neq i'$,*

$$\mathrm{Cov}(\hat{s}_{a_j,2i+1}\hat{s}_{b_j,2i+1}, \hat{s}_{a_j,2i'+1}\hat{s}_{b_j,2i'+1}) = O(\log^{-\kappa\zeta/2+5} n)$$

The proof is deferred to the full paper. Then the variance is bounded as follows:

$$\begin{aligned}
\mathrm{Var}(\hat{C}(a_j, b_j)) &= O(1/L^2)\left[\sum_i \mathrm{Var}(\hat{s}_{a_j,2i+1}\hat{s}_{b_j,2i+1})\right. \\
&\quad \left. + \sum_{i \neq i'} \mathrm{Cov}(\hat{s}_{a_j,2i+1}\hat{s}_{b_j,2i+1}, \hat{s}_{a_j,2i'+1}\hat{s}_{b_j,2i'+1})\right] \\
&\leq O(1/L^2)\sum_i \mathbb{E}[\hat{s}^2_{a_j,2i+1}]\mathbb{E}[\hat{s}^2_{b_j,2i+1}] \qquad (3)\\
&\quad + O(\log^{-\kappa\zeta/2+5} n) \\
&= O(\log^2 n/L) + O(\log^{-\kappa\zeta/2+5} n) \\
&= O(\log^{-\kappa\zeta/2+5} n)
\end{aligned}$$

Therefore, we can make the estimator variance $1/\mathrm{poly}\log(n)$. Since $d(a_j, b_j) \leq 2\delta\log\log n + d(a,b) \leq 3\delta\log\log n$, by Lemma 6, we can guarantee w.h.p. that $(1+\epsilon)e^{-d(a_j,b_j)} \geq 4\widetilde{C}(a_j, b_j) \geq (1-\epsilon)e^{-d(a_j,b_j)}$ when k is chosen with a large enough κ. Since ϵ is $\Omega(1/\mathrm{poly}\log(n))$, we see that $\mathbb{E}[4\hat{C}(a_j, b_j)] \in (1-2\epsilon, 1+2\epsilon)e^{-d(a_j,b_j)}$ with constant probability by a Chebyshev bound for a fixed j. Therefore, we conclude that $|-\ln(4\hat{C}(a_j, b_j)) - d(a_j, b_j)| < 2\epsilon$ with probability at least 5/6. Since $d(a_j, b_j) = d(a,b) + d(a, a_j) + d(b, b_j)$, we have $|\widetilde{d}(a_j, b_j) - d(a,b)| < 2\epsilon$ with probability at least 5/6.

Finally, since a_j, b_j provide $\Omega(\log n)$ independent estimators of $d(a,b)$, by Azuma's inequality, we can show that at least 2/3 of all a_j, b_j satisfies $|\widetilde{d}(a_j, b_j) - d(a,b)| < 2\epsilon$ with probability at least $1 - 2^{-\Omega(\log n)} = 1 - n^{-\Omega(1)}$. In particular, this means that there exists j such that $r_j < 4\epsilon$ and for all j such that $|\widetilde{d}(a_j, b_j) - d(a,b)| > 6\epsilon$, we must have $r_j \geq 4\epsilon$. Therefore, we conclude that $|\widetilde{d}(a_{j^*}, b_{j^*}) - d(a,b)| = |\hat{d}(a,b) - d(a,b)| < 6\epsilon$. □

3.5 Reconstruction Algorithm

The algorithm for reconstruction is ultimately based from our ability to apply signature reconstruction and derive well-concentrated distance estimators in an inductive process. The base case would be to simply use the sequences at the leaves and the basic distance function to reconstruct the tree up to $O(\log\log n)$ height, after which we use our signature reconstruction algorithm to produce a reconstructed distance function that provides high accuracy throughout the entire process.

In the previous section, we showed that if two nodes are $O(\log\log n)$ distance apart, then distance estimators will concentrate to the mean with *poly* $\log n$ sequence length. The recursive argument depends crucially that we can detect closeby nodes so that we only use statistically accurate distance estimators. Fortunately, testing for the size of the diameter of two nodes, on whether it is larger than or less than $O(\log\log n)$, is viable by the same concentration properties of our various distance estimators.

Lemma 12. *Let $\delta > 0$ be any constant and $\epsilon = \Omega(1/poly\log(n))$. Then, for nodes a, b, if $d(a,b) > r + \epsilon$ with $r = \delta \log\log n$ and $k > \log^{\kappa}(n)$, then $-\ln(4\widetilde{C}(a,b)) > r$ with probability at least $1 - n^{-\Omega(\log n)}$.*

Proof. Follows analogously to Lemma 6. □

Definition 6. *For two nodes a, b that are more than $h = \delta \log\log n$ up the tree, we define $T(S_h(a,b), r) = 1$ if at least half of $S_h(a,b)$ is bounded by r and 0 otherwise.*

$$T(S_h(a,b), r) = \mathbb{1}\left\{|[-r, r] \cap S_h(a,b)| \geq \frac{1}{2}2^h\right\}$$

Lemma 13. *[Deep Computation: Diameter Test] Let a, b be nodes and we choose $r = O(\log\log n)$. If $k > poly\log(n)$, then with high probability, $T(S_h(a,b), r) = 1$ when $d(a,b) < r - \epsilon$ and $T(S_h(a,b), r) = 0$ when $d(a,b) > r + \epsilon$.*

Proof. The case when $d(a,b) < r - \epsilon$ follows directly from the proof of Lemma 10. When $d(a,b) > r + \epsilon$, note that the only change to the proof of Lemma 10 is that instead of calling Lemma 6 to upper and lower bound $\widetilde{C}(a_j, b_j)$, we use Lemma 12 to deduce that $\widetilde{d}(a_j, b_j) \geq r + \epsilon/2$ still holds with constant probability by Chebyshev and $T(S_h(a,b), r) = 0$ occurs with high probability using Azuma's inequality bound over all pairs (a_j, b_j). □

With the diameter test, we can ensure that all distance computations are accurate with an additive error of ϵ by using a diameter condition and also guarantee that all close enough nodes have distances computed. Therefore, we can compute an accurate localized distance matrix. With that, we use the traditional Four Point Method to determine quartets. The standard technique, called the Four Point Method, to compute quartet trees (i.e., unrooted binary trees on four leaves) is based on the Four Point Condition [5]. The underlying combinatorial algorithm we use here is essentially identical to the one used by Roch in [22].

Definition 7. *(From [11]) Given a four-taxon set $\{a, b, c, d\}$ and a dissimilarity matrix D, the Four Point Method (FPM) infers tree $ab|cd$ (meaning the quartet tree with an edge separating a, b from c, d) if $D(a,b) + D(c,d) \leq \min\{D(a,c) + D(b,d), D(a,d) + D(b,c)\}$. If equality holds, then the FPM infers an arbitrary topology.*

If $D(a,b)$ is a dissimilarity matrix that has maximum deviation from $d(a,b)$ by an additive error of $\epsilon < f/2$, where f is the minimum non-zero entry in d, then FPM will always infer the true quartet, as in Figure 2. In this case, setting $\epsilon < \lambda_{min}/2$ will allow for correct short quartet inference and therefore this implies that $\lambda_{min} = \Omega(1/\text{poly}\log(n))$ in order for the sequence length requirement to still be polylogarithmic (in general it will depend inverse polynomially on ϵ).

Once quartet splits are determined accurately, any quartet-based tree-building algorithm can be used. For simplicity, we will use a cherry picking algorithm that simply identifies a, b as a cherry if they are always on the same side of all quartet splits. Then, we can reconstruct the ancestors of these cherries and simply recurse. This is the high-level summary of our reconstruction algorithm 1.

Definition 8. *A quartet $Q = \{a, b, c, d\}$ of L_h is* **r-short** *if*

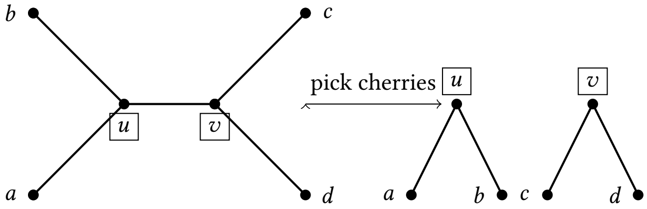

Figure 2: The Four Point Method on $\{a, b, c, d\}$ with distance matrix D will infer the correct quartet as long as $D(a,b)$ do not differ from the underlying additive distance $d(a,b)$ by more than $\lambda_{min}/2$. We then use quartet splits to pick cherries and recurse on ancestors (i.e. a, b)

- *When $h = 0$, we use $D(x,y) = -\ln(4\widetilde{C}(x,y))$ and*

$$\max_{x,y \in Q} D(x,y) \leq r$$

- *When $h > \delta \log\log n$, we use $D(x,y) = \hat{d}(x,y)$ and*

$$\min_{x,y \in Q} T(S_h(x,y), r) = 1$$

Lemma 14. *If a quartet Q is $O(\log\log n)$-short, $\lambda_{min} = \Omega(1/poly\log(n))$, and $k > poly\log(n)$, then with high probability FPM with the corresponding distance matrix D will return the true quartet tree.*

Proof. Since FPM will return the true quartet tree when the distance matrix is $\epsilon < \lambda_{min}/2$ away from the true distance matrix, this follows directly from combining Lemma 12, 6 for the case when $h = 0$. We use Lemma 13, 10 for the case when $h > \delta \log\log n$. □

When we recursively build this tree up, it is crucial that we can calculate distances between nodes in each sub-tree that is reconstructed so far. This is because our distance estimators $\hat{d}(a,b)$ requires knowledge or a good estimate of all distances in the subtree under a and b in order to calculate the $O(\log n)$ conditionally independent low-variance distance estimates. Fortunately, it suffices to have a good enough estimate of these distances when distances can only take only discrete integer multiples of λ_{min}, as assumed in our Δ-branch model. Under this assumption, we can ascertain the distances by simply rounding to the nearest integer multiple of λ_{min} as long as $\epsilon < \lambda_{min}/2$. The estimation of all relevant distances is based of a very simple three-point rule.

Definition 9 (Three-Point Rule). *For a triplet of nodes a, b, c that meet at x and a dissimilarity matrix D, we define $\hat{D}(a,x)$ to be the estimator*

$$\hat{D}(a,x) = \frac{1}{2}[D(a,b) + D(a,c) - D(b,c)]$$

We are now ready to present the final algorithm, TreeReconstruct. We assume that the input to the algorithm are just the leaf signatures $s = \{s_x\}$ and λ_{min}.

Theorem 2.1. *Assume that $\lambda_{max} \leq \ln(\sqrt{2})$ is less than the Kesten-Stigum threshold and $\lambda_{min} = \Omega(1/poly\log(n))$. Furthermore, assume that we are in the symmetric case where $p_{ins}(e) = p_{del}(e)$ for all*

Algorithm 1 Tree Reconstruction With Signatures

```
Require: Leaf signatures $\{s_x\}$ for all $x \in L_0$, $\lambda_{min}$

function TreeReconstruct($\{s_x\}, \lambda_{min}$)
    Apply FPM to $2\delta \log\log n$-short quartets of $L_0$ with $D(a,b) = -\ln(4\widetilde{C}(a,b))$ to infer splits
    Use quartet splits and the three-point rule to build multiple subtrees up to $\delta \log\log n$ height by iteratively identifying cherries.
    Estimate and fill in distances within each subtree using the three-point rule, rounded to the nearest $\lambda_{min}$ multiple.
    for $h \leftarrow \delta \log\log n$ to $\log n$ do
        Apply FPM to $2\delta \log\log n$-short quartets on $L_h$ with $D(a,b) = \hat{d}(a,b)$ to infer splits.
        Identify cherries as pairs of vertices that only appear on the same side of quartet splits.
        Estimate and fill in the edge lengths on cherries using the three-point rule on a short quartet, rounded to nearest $\lambda_{min}$ multiple.
        Add cherries, with edges containing estimated decay rates, to the tree
    end for
    return the resulting tree $t$
end function
```

edges e. Then for a sufficiently large κ, with $O(\log^\kappa n)$ sequence length, there is an algorithm TreeReconstruct *that can reconstruct the phylogenetic tree with high probability under the CFN-Indel model.*

Proof. Set $\epsilon = \lambda_{min}/3 = \Omega(1/\text{poly}\log(n))$. Then, by Lemma 14, all quartets queries made by TreeReconstruct are correct. Note that a, b that are neighbors (i.e. for which a, b is a cherry) appear on the same side of all short quartets. Furthermore, if $a, b \in L_h$ are not cherries and $d(a,b) < O(\log\log n)$, then there must exists x, y, such that $\{a, x, b, y\}$ is $O(\log\log n)$-short and a, b are not on the same side of the split. Otherwise, if $d(a,b) = \Omega(\log\log n)$ is large, then by Lemma 12 and Lemma 13, none of the quartets involving a, b will be considered $O(\log\log n)$-short. Finally, since all short quartets are considered, we conclude that all cherries picked are correct.

Lastly, we note that we can estimate distances up to error $\epsilon < \lambda_{min}/2$, which allows us to round to the nearest multiple of λ_{min} to get noiseless distance reconstruction with high probability. Therefore, all estimated decay rates or lengths are correct, as we reconstruct up the tree, by using Lemma 10, and noting that if all distances in D are accurate up to error ϵ, then the Three-Point Rule is accurate up to error at most $3\epsilon/2 < \lambda_{min}/2$. □

4 UNBALANCED TREES AND ASYMMETRIC PROBABILITIES

In this section, we show how to relax the assumption that the tree is completely balanced. Instead, we consider trees which are approximately balanced. In particular, let $\mathcal{D}(a)$ denote the number of edges on the path from a to the root. Then for $\mathcal{D}_{max}$ such that $\mathcal{D}_{max} \le \log^\alpha n$ for a constant α, we assume all nodes satisfy $\mathcal{D}(a) \le \mathcal{D}_{max}$.

We also relax the assumption that $p_{del}(e) = p_{ins}(e)$ for every edge, instead assuming that $|p_{ins}(e) - p_{del}(e)|$ is bounded by $\beta \frac{\log\log n}{\mathcal{D}_{max}}$ for a constant β. As mentioned in Section 2, up to the constants α, β these assumptions are optimal, i.e. for a fixed κ and sufficiently large α or β, reconstruction with high probability is not possible due to significant loss of bitstring length down the tree. We will assume for simplicity of presentation the length of the root bitstring is $\Theta(\log^\kappa n)$, but the analysis easily generalizes to the case where the root has a larger bitstring.

Again, in all proofs we assume κ is a sufficiently large constant depending only on the fixed values $\alpha, \beta, \zeta, \lambda_{min}$ and parameters δ, ϵ in the lemma/theorem statements (which will be fixed by our algorithmic construction).

4.1 Tree Properties

Let k_a be the number of bits in the bitstring at vertex a, and $k_r = \Theta(\log^\kappa n)$ specifically be the number of bits in the root bitstring. Let $L = \lfloor k_r^{1/2-\zeta} \rfloor$ for some small constant $\zeta > 0$. The length of a block at the root l_r will be $\lfloor k_r/L \rfloor$ as before.

Then, define $\eta(a) = \prod_{e \in P_{r,a}} (1 + p_{ins}(e) - p_{del}(e))$. Note that the expected position of bit j of r in a conditioned in the bit not being deleted is $j\eta(a)$. Thus, we will define the length of a block in bitstring a to be $l_a = \lfloor l_r \eta(a) \rfloor$, and the ith block of the bitstring at node a to be bits $(i-1)l_a + 1$ to il_a of the bitstring. Note that by the assumption that $|p_{ins}(e) - p_{del}(e)| \le \beta \frac{\log\log n}{\mathcal{D}_{max}}$, $\log^{-\beta}(n) \le \eta(a) \le \log^\beta(n)$ for all a.

Key to our algorithmic construction is the fact that not only do we expect $k_a = k_r \eta(a)$, but that this concentrates very tightly.

Lemma 15. *With probability $1 - n^{-\Omega(\log n)}$ for all vertices a, $k_r(1 - \frac{\mathcal{D}(a) \cdot \log^{\beta/2+1}(n)}{\log^{\kappa/2-2} n}) \le k_a/\eta(a) \le k_r(1 + \frac{\mathcal{D}(a) \cdot \log^{\beta/2+1}(n)}{\log^{\kappa/2-2} n})$*

The proof is deferred to the full paper. For the purposes of analysis, it will be convenient to define the *normalized shift* of a bit.

Definition 10. *For any two nodes a and b, we say that the jth bit of the bitstring at a has a* ***normalized shift*** *of m bits on the path from a to b if there is some j' such that $|j'/\eta(b) - j/\eta(a)| = m$ and the jth bit of the bitstring at a and the j'th bit of the bitstring at b are shared.*

Lemma 16. *With probability $1 - n^{-\Omega(\log n)}$, no bit has a normalized shift of more than $4\log^{\alpha+1} n\sqrt{k_r}$ bits on any path in the tree.*

The proof is deferred to the full paper. Analogously to before, we will define $\mathcal{E}_{reg}$ to be the intersection of the high probability events described in Lemmas 15, 16, and 3.

We define $s_{a,i}$ analogously to before, letting it be $1/\sqrt{l_a}$ times the signed difference between the number of zeroes in the ith block of the bitstring of the bitstring at node a and half the length of ith block, and note that Lemma 3 still applies. However, we do not know the true block lengths, so even for the leaves we cannot exactly compute $s_{a,i}$.

Instead, for a leaf bitstring let $l'_a = \lfloor k_a/L \rfloor$. Note that this quantity is computable given only the leaf bitstrings as well as the minimum sequence length requirement k_r. Our algorithm will split each leaf bitstring into "pseudo-blocks" of length l'_a, i.e. the ith pseudo-block of leaf a consists of bits $(i-1)l'_a + 1$ to il'_a of the bitstring. Our estimate $\widetilde{s}_{a,i}$ of $s_{a,i}$ is then $1/\sqrt{l'_a}$ times the signed difference between the number of zeroes in the ith pseudo-block of the bitstring at node a. $\widetilde{s}_{a,i}$ is computable for all leaf nodes a since we can compute l'_a easily, so giving a reconstruction algorithm based on the signatures $\widetilde{s}_{a,i}$ gives a constructive result.

Lemma 17. *Conditioned on $\mathcal{E}_{reg}$,*

$$\widetilde{s}_{a,i} = s_{a,i} \pm O(\log^{\alpha/2+\beta/4+5/2-\kappa\zeta/2} n)$$

We defer the proof to the full paper.

4.2 Distance Estimator

We would like to compute the following estimator as before:

$$\widetilde{C}(a,b) = \frac{2}{L}\sum_{i=1}^{L/2} s_{a,2i+1}s_{b,2i+1}$$

But we cannot directly compute $s_{a,i}$, so we use the following estimator instead:

$$\widetilde{C}'(a,b) = \frac{2}{L}\sum_{i=1}^{L/2} \widetilde{s}_{a,2i+1}\widetilde{s}_{b,2i+1}$$

Note that by Lemma 17 and Lemma 3 we immediately get the following Corollary:

Corollary 18. *Conditioned on $\mathcal{E}_{reg}$,*

$$|\widetilde{C}'(a,b) - \widetilde{C}(a,b)| = O(\log^{\alpha/2+\beta/4+7/2-\kappa\zeta/2} n)$$

Lemma 19. *For any two nodes a, b in the tree, and any i,*

$$\mathbb{E}[s_{a,i}s_{b,i}] = \frac{1}{4}(1 \pm O(\log^{-\kappa\zeta+\alpha+1} n))\exp(-d(a,b))$$

The proof follows similarly to the proof of Lemma 4 and is deferred to the full paper.

Corollary 20. *For any two nodes a, b,*

$$\mathbb{E}[\widetilde{C}(a,b)] = \frac{1}{4}(1 \pm O(\log^{-\kappa\zeta+\alpha+1} n))\exp(-d(a,b))$$

Lemma 21. *Let $\delta > 0$ be any constant and $\epsilon = \Omega(\log^{max\{-\kappa(1/2-\zeta)+2\delta+6,-\kappa\zeta/2+\alpha/2+\delta+5/2\}} n)$. Then for nodes a, b such that $d(a,b) < \delta\log\log(n)$, then $|-\ln(4\widetilde{C}(a,b))-d(a,b)| < \epsilon$ with probability at least $1 - n^{-\Omega(\log n)}$.*

Proof. The proof follows exactly as did the proof of Lemma 6. □

Corollary 22. *Let $\delta > 0$ be any constant and $\epsilon = \Omega(\log^{\max\{-\kappa(1/2-\zeta)+2\delta+6,-\kappa\zeta/2+\alpha/2+\beta/4+\delta+7/2\}} n)$. Then for nodes a, b such that if $d(a,b) < \delta\log\log(n)$, then $|-\ln(4\widetilde{C}'(a,b)) - d(a,b)| < \epsilon$ with probability at least $1 - n^{-\Omega(\log n)}$.*

Proof. The proof follows from Lemma 21 and Corollary 18. □

4.3 Signature Reconstruction

For a node a in the tree, let A be the set of leaves that are descendants of a. Let $h(a)$ be the maximum number of edges on the path between a and any of its leaves, i.e. $h(a) = \max_{x' \in A}\mathcal{D}(x') - \mathcal{D}(a)$. For for a leaf $x \in A$ let $h(x) = \max_{x' \in A}\mathcal{D}(x') - \mathcal{D}(x)$, i.e. $h(x)$ is the difference between x's depth and the maximum depth of any leaf. If a is far away from all of its leaf descendants, we will use the following estimator of its signature:

$$\hat{s}_{a,i} = \frac{1}{2^{h(a)}}\sum_{x \in A} e^{d(x,a)}2^{h(x)}\widetilde{s}_{x,i}$$

Note that $\sum_{x \in A} 2^{h(x)} = 2^{h(a)}$. In addition, if the tree below a is balanced then $h(x) = 0$ for all $x \in A$ so the estimator is defined analogously to before.

Lemma 23. *Let $\mathcal{E}$ denote the bitstring at the node a and $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. Then, for a leaf x that is a descendant of a, $\mathbb{E}[\widetilde{s}_{x,i}|\mathcal{E}] = e^{-d(x,a)}(s_{a,i} + \nu_{a,i})$, where $|\nu_{a,i}| = O(\log^{\alpha/2+\beta/4+5/2-\kappa\zeta/2}(n))$.*

The proof follows similarly to the proof of Lemma 7 and is deferred to the full paper.

Corollary 24. *Let $\mathcal{E}$ denote the bitstring at the node a and $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. Then $\mathbb{E}[\hat{s}_{a,i}|\mathcal{E}] = s_{a,i} + \nu_{a,i}$, where $|\nu_{a,i}| \leq O(\log^{\alpha/2+\beta/4+5/2-\kappa\zeta/2}(n))$.*

Lemma 25. *Let $\mathcal{E}$ denote the bitstring at the node a and $\Pr(\mathcal{E}_{reg}|\mathcal{E}) > 1 - n^{-\Omega(\log n)}$. Then, $\mathbb{E}[\hat{s}^2_{a,i}|\mathcal{E}] = O(\log^2 n)$ as long as $e^{2\lambda_{max}} < 2$.*

Proof. Note that the sum over all x, y pairs in A such that $x \wedge y$ is m edges away from a of $2^{h(x)+h(y)}$ is $2^{h(a)-m}$. Then, the proof follows analogously to the proof of Lemma 9. □

As before, we define:

$$\hat{C}(a,b) = \frac{2}{L}\sum_{i=1}^{L/2} \hat{s}_{a,2i+1}\hat{s}_{b,2i+1}$$

For some height $h = \delta\log\log n$, we define $\hat{d}(a,b)$ similarly to before, but splitting the exact definition of $\hat{d}(a,b)$ into three cases:

Case 1: *If a, b both have no leaf descendants less than h edges away from them*, consider the nodes A_h and B_h which are the set of descendants of a, b exactly h edges below a, b respectively. Order A_h arbitrarily and let a_j be nodes of A_h with $1 \leq j \leq 2^h = \log^\delta n$ and similarly for B_h. Again let $\widetilde{d}(a_j, b_j) = -\ln(e^{d(a_j,a)+d(b_j,b)}4\hat{C}(a_j, b_j))$. Let $S_h(a,b) = \left\{\widetilde{d}(a_j, b_j)\right\}_{j=1}^{2^h}$, and

$$r_j = \inf\left\{r > 0 : \left|\{j' \neq j : |\widetilde{d}(a_j, b_j) - \widetilde{d}(a_{j'}, b_{j'})| \leq r\}\right| \geq \frac{2}{3}2^h\right\}$$

.

Then, if $j^* = \arg\min_j r_j$, then our distance estimator is $\hat{d}(a,b) = \widetilde{d}(a_{j^*}, b_{j^*})$.

Case 2: *If a has no leaf descendants less than h edges away but b has some leaf descendant b' which is less than h edges from b*, order A_h arbitrarily and let a_j be nodes of A_h with $1 \leq j \leq$

$2^h = \log^\delta n$. Let $\widetilde{d}(a_j, b') = -\ln(e^{d(a_j,a)+d(b',b)} 4\hat{C}(a_j, b'))$, and $S_h(a, b) = \left\{\widetilde{d}(a_j, b')\right\}_{j=1}^{2^h}$, and define r_j and $\hat{d}(a, b)$ analogously to Case 1.

Case 3: *If a and b both have leaf descendants a', b' less than h edges away*, we just define $\hat{d}(a, b) = -\ln(e^{d(a',a)+d(b',b)} 4\widetilde{C}'(a', b'))$.

Lemma 26 (Deep Distance Computation: Small Diameter). *Let a, b be nodes such that $d(a, b) = O(\log\log n)$. If $\lambda_{max} < \ln(\sqrt{2})$, then $|\hat{d}(a, b) - d(a, b)| < \epsilon$ with high probability.*

Proof. In Case 1, the proof follows as did the proof of Lemma 10 (including an analogous proof of Lemma 11). In Case 2, the proof follows similarly to the proof of Lemma 10, except we also condition on the bitstring at b' and note that $d(a, b') = O(\log\log n)$. In Case 3, the proof follows directly from Corollary 22. □

4.4 Reconstruction Algorithm

Since we have proven statements analogous to those needed to prove Theorem 2.1, the proof of Theorem 2.2 follows very similarly to Theorem 2.1, except that a short quartet needs to be slightly redefined for our purposes.

Definition 11. *A quartet $Q = \{a, b, c, d\}$ is* **r-short** *corresponding to a distance matrix D if for every pair $x, y \in Q$,*

- *When x, y both have leaf descendants x', y' less than $\delta \log\log n$ away, we use $D(x, y) = \hat{d}(x, y)$ and $D(x, y) \leq r$.*
- *Otherwise, we use $D(x, y) = \hat{d}(x, y)$ and $T(S_h(x, y), r) = 1$*

By Lemma 26, we see that $O(\log\log n)$-short quartets can be detected and FPM on these quartets always return the true quartet tree, by an analogous argument to the symmetric case. Note that in the asymmetric case, at each step of the tree reconstruction process, not all nodes will be paired as cherries but at least one cherry will be paired (by looking at the cherry with maximum depth) and we can therefore always ensure progress.

There is, however, a slight issue with directly following the the same reconstruction algorithm because we may join subtrees that are no longer both dangling, which intuitively means that the path between the root of both subtrees goes above them in the real model tree. For example, in the case of balanced trees, if S_1, S_2 are subtrees that are to be joined at some iteration of the algorithm, then we reconstruct the shared ancestor of S_1, S_2 and join the roots of S_1, S_2 as children of the reconstructed ancestor. However, in this case, it might be possible that the ancestor of S_1 is a node in S_2 that is not the root node of S_2!

This non-dangling issue is elaborated in the general tree reconstruction algorithm of [22] and is circumvented with standard reductions to the Blindfolded Cherry Picking algorithm of [7], which essentially allows us to reduce all subtree joining processes to the dangling case. The basic idea is that there exists a re-rooting of our subtrees that reduces to the dangling case and finding the correct re-rooting boils down to some $O(1)$ extra distance computations per iteration. Because our algorithm is a close replica of the general tree reconstruction algorithm of Roch, we refer the reader to the appendix of [22] for details.

5 FUTURE DIRECTIONS

In this paper, we give reconstruction guarantees which are optimal (up to the choice of constants α, β, κ) for a popular model of the phylogenetic reconstruction problem. However, we did not attempt to optimize the constant κ in the exponent of our sequence length requirement. In particular, we note that while we have k bits of information for each sequence, we only use $\tilde{O}(\sqrt{k})$ bits of information about each sequence in our algorithm, so there is some reason to believe methods similar to ours cannot achieve the optimal value of κ. It is an interesting problem to design an algorithm which uses $\Omega(k)$ bits of information and matches our asymptotic guarantees with potentially better constants, but doing so seems challenging given the presence of indels. Furthermore, we operated in the Δ-branch model with discretized edge lengths, which avoids errors accumulating over a series of distance estimations. Without discretized edge lengths, a more refined analysis seems necessary in order to avoid this error accumulation.

In addition, there are other models of theoretical or practical interest, but for which the extension from our results is not immediately obvious. One alternative model which has been studied in the trace reconstruction problem (see e.g. [14]) and which could be extended to the phylogenetic reconstruction problem is to view the bitstrings as infinitely long. The new goal is to design an algorithm which only views the first $k(n)$ bits of each bitstring for as small a function $k(n)$ as possible. This model is well-motivated by practical scenarios, where DNA sequences are large but reading the entire sequence is both inefficient and unnecessary for reconstructing the tree. When we assume the bitstrings are infinitely long, then reconstruction may be possible without the assumptions we made to ensure no leaf bitstrings were empty (i.e., it may be possible to reconstruct trees with maximum depth $\Omega(n)$ or with large differences in the insertion and deletion rates). In Section 4 we crucially used sequence lengths to estimate the positions of blocks in the bitstrings, so even with these assumptions our results do not easily extend to this model.

Lastly, our results are optimal up to constants and build on many techniques for phylogenetic reconstruction with independent and random mutations. However, algorithms which perform well on simulated data generated using independent and random mutations are known to perform relatively poorly on real-world data [21]. An interesting problem is to define a theoretical model for phylogenetic reconstruction with dependent mutations or semi-adversarial mutations which better models this real-world data and to design algorithms for this model. In particular, one advantage of our algorithm is that it uses statistics about large-length blocks of bits which are very robust to the errors introduced by indels. One might hope that this robustness extends to models with dependent and/or semi-random mutations.

A DEFERRED PROOFS FROM SECTION 3

Proof of Lemma 1. Let k_a be the length of the bitstring at node a. For any edge $e = (a, b)$ where a is the parent, k_b is equal to k_a plus the difference between two binomial variables with k_a trials and probability of success $p_{indel}(e)$. Applying Azuma's inequality shows that $|k_b - k_a|$ is at most $2 \log n \sqrt{k_a}$, with probability

$1 - n^{-\Omega(\log n)}$. Then fixing any node v and applying this high probability statement to at most $\log n$ edges on the path from the root to any node a gives that $k < k_a < 4k$ with probability $1 - n^{-\Omega(\log n)}$. The lemma then follows from a union bound over all n nodes. □

Proof of Lemma 2. Note that conditioned on the jth bit of a not being deleted on an edge $e = (a, b)$ where a is the parent, the number of bits by which it shifts on the edge (a, b) is the difference between two binomial variables with j trials and probability of success $p_{indel}(e)$, which is at most $2 \log n\sqrt{j}$ with probability $1 - n^{-\Omega(\log n)}$. By Lemma 1, with probability $1 - n^{-\Omega(\log n)}$ we know that $j \le 4k$ so this is at most $4 \log n\sqrt{k}$. Then, fixing any path and applying this observation to the at most $2 \log n$ edges on the path, by union bound we get that the sum of shifts is at most $4 \log^2 n\sqrt{k}$ with probability $1 - n^{-\Omega(\log n)}$. Applying union bound to all $O(n^2)$ paths in the tree gives the lemma. □

Proof of Lemma 3. For a fixed node a and any consecutive sequence of length m, note that each bit in the bitstring is equally likely to be 0 or 1, by symmetry, and furthermore, each bit is independent since they cannot be inherited from the same bit in ancestral bitstrings. The number of zeros in the sequence, S_0, can be expressed as a sum of m i.i.d. Bernoulli variables. Therefore, by Azuma's inequality we have $\Pr(S_0 - m/2 \ge t\sqrt{m}) \le \exp(-\Omega(t^2))$.

There are $2n-1$ nodes, and by Lemma 1, for each node the number of different consecutive subsequences of the node's bitstring is poly $\log(n)$. Therefore, setting $t = \log n$ and applying a union bound over all $O(n \log^{O(1)}(n))$ subsequences gives the lemma. □

Proof of Lemma 4. Fixing any block i of nodes a, b and letting $\sigma_{a,j}$ denote the jth bit of the bitstring at a:

$$\begin{aligned}
\mathbb{E}[s_{a,i}s_{b,i}] &= \frac{1}{l}\mathbb{E}\left[\left(\sum_{j=(i-1)l+1}^{il} \sigma_{a,j} - \frac{l}{2}\right)\left(\sum_{j'=(i-1)l+1}^{il} \sigma_{b,j'} - \frac{l}{2}\right)\right] \\
&= \frac{1}{l}\mathbb{E}\left[\left(\sum_{j=(i-1)l+1}^{il} \left(\sigma_{a,j} - \frac{1}{2}\right)\right)\left(\sum_{j'=(i-1)l+1}^{il} \left(\sigma_{b,j'} - \frac{1}{2}\right)\right)\right] \\
&= \frac{1}{l}\mathbb{E}\left[\sum_{j=(i-1)l+1}^{il} \sum_{j'=(i-1)l+1}^{il} \left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)\right] \\
&= \frac{1}{l}\sum_{j=(i-1)l+1}^{il} \sum_{j'=(i-1)l+1}^{il} \mathbb{E}\left[\left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)\right]
\end{aligned}$$

Note that if bit j of a's bitstring and bit j' of b's bitstring are not shared, then their values are independent and in particular, since $\mathbb{E}[\sigma_{a,j}] = 1/2$ for any a, j:

$$\mathbb{E}\left[\left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)\right] = \mathbb{E}\left[\left(\sigma_{a,j} - \frac{1}{2}\right)\right]\mathbb{E}\left[\left(\sigma_{b,j'} - \frac{1}{2}\right)\right] = 0$$

Let $\mathcal{E}$ be any realization of the locations where insertions and deletions occur throughout the tree. We will look at $\mathbb{E}[s_{a,i}s_{b,i}|\mathcal{E}]$, leaving the root bitstring, the values of bits inserted by insertions, and the locations of substitutions unrealized. Note that $\mathcal{E}$ fully specifies what bits are shared by block i of a, b, giving:

$$\mathbb{E}[s_{a,i}s_{b,i}|\mathcal{E}] = \frac{1}{l}\sum_{\text{shared } j,j'} \mathbb{E}\left[\left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)\right]$$

Now, note that $\left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)$ is 1/4 if $\sigma_{a,j}, \sigma_{b,j'}$ are the same and $-1/4$ otherwise. Since bit j of a and bit j' of b descended from the same bit, it is straightforward to show (see e.g., [25]) that the probability a, b are the same is $\frac{1}{2}(1 + \prod_{e\in P_{a,b}}(1 - 2p_{sub}(e)))$, giving that $\mathbb{E}\left[\left(\sigma_{a,j} - \frac{1}{2}\right)\left(\sigma_{b,j'} - \frac{1}{2}\right)\right] = \frac{1}{4}\prod_{e\in P_{a,b}}(1 - 2p_{sub}(e))$ if j, j' are shared bits of a, b.

Applying the law of total probability to our conditioning on $\mathcal{E}$, we get that $\mathbb{E}[s_{a,i}s_{b,i}]$ is the expected number of shared bits in block i of a and block i of b times $\frac{1}{4l}\prod_{e\in P_{a,b}}(1 - 2p_{sub}(e))$. So all we need to do is compute the expected number of shared bits which are in block i of a and b. Let $a \wedge b$ be the least common ancestor of a, b. The jth bit in $a \wedge b$ will not be deleted on the path from $a \wedge b$ to a or b with probability $\prod_{e\in P_{a,b}}(1 - p_{del}(e))$. Let ρ_j be the probability that the jth bit of $a \wedge b$ appears in the ith block of both a and b conditioned on it not being deleted. Then the expected number of shared bits is $(\sum_j \rho_j) \cdot \prod_{e\in P_{a,b}}(1 - p_{del}(e))$.

For our fixed block i, call the jth bit of $a \wedge b$ a *good* bit if j is between $(i-1)l + 4\log^2 n\sqrt{k}$ and $il - 4\log^2 n\sqrt{k}$ inclusive. Call the jth bit an *okay* bit if j is between $(i-1)l - 4\log^2 n\sqrt{k}$ and $il + 4\log^2 n\sqrt{k}$ inclusive but is not a good bit. If the jth bit is not good or okay, call it a *bad* bit. Note that $4\log^2 n\sqrt{k} \le l \cdot O(\log^{-\kappa\zeta+2} n)$, which is $o(l)$ if κ is sufficiently large and ζ is chosen appropriately. Then, there are $l \cdot (1 - O(\log^{-\kappa\zeta+2} n))$ good bits and $l \cdot O(\log^{-\kappa\zeta+2} n)$ okay bits for block i. Lemma 2 gives that $\rho_j \ge 1 - n^{-\Omega(\log n)}$ for all good bits. Similarly, $\rho_j \le n^{-\Omega(\log n)}$ for all bad bits. For okay bits, we can lazily upper and lower bound ρ_j to be in $[0, 1]$. This gives:

$$\begin{aligned}
\sum_j \rho_j &= \sum_{\text{good } j} \rho_j + \sum_{\text{okay } j} \rho_j + \sum_{\text{bad } j} \rho_j \\
&= l(1 - O(\log^{-\kappa\zeta+2} n)) + l \cdot O(\log^{-\kappa\zeta+2} n) + n^{-\Omega(\log n)} \\
&= l(1 \pm O(\log^{-\kappa\zeta+2} n))
\end{aligned}$$

Combining this with the previous analysis gives that

$$\mathbb{E}[s_{a,i}s_{b,i}] = \frac{1}{4}(1 \pm O(\log^{-\kappa\zeta+2} n)) \prod_{e\in P_{a,b}} (1 - 2p_{sub}(e))(1 - p_{del}(e))$$

Rewriting this in exponential form and using the definition of $\lambda(e)$ and $d(a, b) = \sum_{e\in P_{a,b}} \lambda(e)$ concludes our proof. □

Proof of Lemma 6. We show how to bound the probability of the error in one direction, the other direction follows similarly.

Let $j_{a,i}$ be the index where the $(i-1)l + 1$th bit of the bitstring of $a \wedge b$, a and b's least common ancestor, ends up in the bitstring at a (or if it is deleted on the path from $a \wedge b$ to a, where it would have ended up if not deleted). i.e., the bits in the ith block of $a \wedge b$ appear in positions $j_{a,i}$ to $j_{a,i+1} - 1$ of the bitstring at a.

Let $s^*_{a,i}$ be defined analogously to $s_{a,i}$, except instead of looking at the bits in the ith block of a, we look at bits $j_{a,i}$ to $j_{a,i+1} - 1$ (we still use the multiplier $\frac{1}{\sqrt{l}}$). Note that conditioned on $\mathcal{E}_{reg}$, $s^*_{a,i}$ and $s_{a,i}$ differ by $O(\frac{k^{\frac{1}{4}}\log^2 n}{\sqrt{l}}) = O(\log^{-\kappa\zeta/2+2} n)$, so $s^*_{a,i}s^*_{b,i}$

and $s_{a,i}s_{b,i}$ differ by $O(\log^{-\kappa\zeta/2+3} n)$. Furthermore, $s^*_{a,i}s^*_{b,i}$ is completely determined by the bits in the ith block of $a \wedge b$ and substitutions, insertions, and deletions on the path from a to b in positions corresponding to the ith block of $a \wedge b$. For $i \neq i'$, these sets of determining random variables are completely independent, so the random variables $\{s^*_{a,i}s^*_{b,i}\}_i$ are independent.

Define $\widetilde{C}^*(a,b)$ analogously to $\widetilde{C}(a,b)$, except using $s^*_{a,i}$ instead of $s_{a,i}$. $\widetilde{C}^*(a,b)$ and $\widetilde{C}(a,b)$ (and their expectations conditioned on $\mathcal{E}_{reg}$) differ by $O(\log^{-\kappa\zeta/2+3} n)$. By rearranging terms and applying Lemma 4 we get:

$$\begin{aligned}
&\Pr[-\ln(4\widetilde{C}(a,b)) > d(a,b) + \epsilon] \\
&= \Pr[4\widetilde{C}(a,b) < e^{-d(a,b)-\epsilon}] \\
&= \Pr[\widetilde{C}(a,b) < \frac{1}{4}e^{-d(a,b)} - \frac{1}{4}(1-e^{-\epsilon})e^{-d(a,b)}] \\
&= \Pr[\widetilde{C}(a,b) < \mathbb{E}[\widetilde{C}(a,b)] - \frac{1}{4}(1-e^{-\epsilon} \pm \\
&\quad O(\log^{-\kappa\zeta+2} n))e^{-d(a,b)}] \\
&= \Pr[\widetilde{C}(a,b) < \mathbb{E}[\widetilde{C}(a,b)|\mathcal{E}_{reg}] - (1-e^{-\epsilon} \pm O(\log^{-\kappa\zeta+2} n)) \\
&\quad (\frac{1}{4}e^{-d(a,b)})] \\
&\leq \Pr[\widetilde{C}(a,b) < \mathbb{E}[\widetilde{C}(a,b)|\mathcal{E}_{reg}] - (1-e^{-\epsilon} \pm O(\log^{-\kappa\zeta+2} n)) \\
&\quad (\frac{1}{4}e^{-d(a,b)})|\mathcal{E}_{reg}] + n^{-\Omega(\log n)} \\
&\leq \Pr[\widetilde{C}^*(a,b) < \mathbb{E}[\widetilde{C}^*(a,b)|\mathcal{E}_{reg}] - (1-e^{-\epsilon} \\
&\quad \pm O(\log^{-\kappa\zeta/2+\delta+3} n)(\frac{1}{4}e^{-d(a,b)})|\mathcal{E}_{reg}] + n^{-\Omega(\log n)}
\end{aligned} \tag{4}$$

Note that conditioned on $\mathcal{E}_{reg}$ no $s^*_{a,i}s^*_{b,i}$ exceeds $O(\log^2 n)$ in absolute value, so the difference in $\mathbb{E}[\widetilde{C}^*(a,b)]$ induced by conditioning on an additional value of $s_{a,2i+1}s_{b,2i+1}$ is $O(\log^2 n/L)$. Azuma's and an appropriate choice of κ then gives:

$$\begin{aligned}
&\Pr[\widetilde{C}^*(a,b) < \mathbb{E}[\widetilde{C}^*(a,b)|\mathcal{E}_{reg}] - (1-e^{-\epsilon} \pm \\
&O(\log^{-\kappa\zeta/2+\delta+3} n))\frac{1}{4}e^{-d(a,b)}|\mathcal{E}_{reg}] \\
&\leq \exp\left(-\frac{((1-e^{-\epsilon} \pm O(\log^{-\kappa\zeta/2+\delta+3} n))\frac{1}{4}e^{-d(a,b)})^2}{(L/2-1)O(\log^2 n/L)^2}\right) \\
&= \exp(-\Omega(L\epsilon e^{-2d(a,b)}/\log^4 n)) \\
&\leq \exp(-\Omega(\epsilon \log^{\kappa(1/2-\zeta)-2\delta-4} n)) \\
&\leq n^{-\Omega(\log n)}
\end{aligned} \tag{5}$$

Combining (4) and (5) gives the desired bound.

□

ACKNOWLEDGEMENTS

We are thankful to Satish Rao for discussions on the generalization to imbalanced trees and other helpful comments, Efe Aras for helpful comments on the presentation, an anonymous reviewer for pointing out an error in the proofs of Lemmas 6 and 10 in a previous version of the paper, and other anonymous reviewers for suggested revisions. Both authors supported by NSF Award CCF-1535989.

REFERENCES

[1] Alexandr Andoni, Mark Braverman, and Avinatan Hassidim. 2010. Phylogenetic reconstruction with insertions and deletions. *Preprint* (2010).

[2] Alexandr Andoni, Constantinos Daskalakis, Avinatan Hassidim, and Sebastien Roch. 2012. Global alignment of molecular sequences via ancestral state reconstruction. *Stochastic Processes and their Applications* 122, 12 (2012), 3852–3874.

[3] Kevin Atteson. 1999. The performance of neighbor-joining methods of phylogenetic reconstruction. *Algorithmica* 25, 2-3 (1999), 251–278.

[4] Daniel G Brown and Jakub Truszkowski. 2012. Fast phylogenetic tree reconstruction using locality-sensitive hashing. In *Workshop on Algorithms for Bioinformatics (WABI)*. Springer, 14–29.

[5] Peter Buneman. 1974. A note on the metric properties of trees. *Journal of Combinatorial Theory (B)* 17 (1974), 48–50.

[6] James A Cavender. 1978. Taxonomy with confidence. *Mathematical biosciences* 40, 3-4 (1978), 271–280.

[7] Constantinos Daskalakis, Elchanan Mossel, and Sébastien Roch. 2011. Evolutionary trees and the Ising model on the Bethe lattice: a proof of Steel's conjecture. *Probability Theory and Related Fields* 149, 1-2 (2011), 149–189.

[8] Constantinos Daskalakis and Sebastien Roch. 2010. Alignment-free phylogenetic reconstruction. In *Annual International Conference on Research in Computational Molecular Biology*. Springer, 123–137.

[9] Robert C Edgar. 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research* 32, 5 (2004), 1792–1797.

[10] Isaac Elias. 2006. Settling the intractability of multiple alignment. *Journal of Computational Biology* 13, 7 (2006), 1323–1339.

[11] Peter L. Erdös, Michael A. Steel, Laszlo Székely, and Tandy Warnow. 1999. A few logs suffice to build (almost) all trees (i). *Random Structures and Algorithms* 14 (1999), 153–184.

[12] Peter L. Erdös, Michael A. Steel, Laszlo Székely, and Tandy Warnow. 1999. A few logs suffice to build (almost) all trees (ii). *Theoretical Computer Science* 221 (1999), 77–118.

[13] Desmond G Higgins and Paul M Sharp. 1988. CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene* 73, 1 (1988), 237–244.

[14] Nina Holden, Robin Pemantle, and Yuval Peres. 2018. Subpolynomial trace reconstruction for random strings
and arbitrary deletion probability. In *Proceedings of the 31st Conference On Learning Theory (Proceedings of Machine Learning Research)*, Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet (Eds.), Vol. 75. PMLR, 1799–1840. http://proceedings.mlr.press/v75/holden18a.html

[15] Kazutaka Katoh, Kazuharu Misawa, Kei-ichi Kuma, and Takashi Miyata. 2002. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic acids research* 30, 14 (2002), 3059–3066.

[16] Ari Löytynoja and Nick Goldman. 2008. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science* 320, 5883 (2008), 1632–1635.

[17] Radu Mihaescu, Cameron Hill, and Satish Rao. 2013. Fast phylogeny reconstruction through learning of ancestral sequences. *Algorithmica* 66, 2 (2013), 419–449.

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

[19] Luay Nakhleh, Usman Roshan, Katherine St. John, Jerry Sun, and Tandy Warnow. 2001. Designing fast converging phylogenetic methods. *Bioinformatics* 17, suppl_1 (2001), S190–S198.

[20] Jerzy Neyman. 1971. Molecular studies of evolution: a source of novel statistical problems. In *Statistical decision theory and related topics*. Academic Press, New York and London, 1–27.

[21] Michael G Nute, Ehsan Saleh, and Tandy Warnow. 2018. Benchmarking Statistical Multiple Sequence Alignment. *bioRxiv* (2018). https://doi.org/10.1101/304659 arXiv:https://www.biorxiv.org/content/early/2018/04/20/304659.full.pdf

[22] Sébastien Roch. 2008. Sequence length requirement of distance-based phylogeny reconstruction: Breaking the polynomial barrier. In *Foundations of Computer Science, 2008. FOCS'08. IEEE 49th Annual IEEE Symposium on*. IEEE, 729–738.

[23] Sébastien Roch. 2010. Towards extracting all phylogenetic information from matrices of evolutionary distances. *Science* 327, 5971 (2010), 1376–1379.

[24] Sébastien Roch and Allan Sly. 2017. Phase transition in the sample complexity of likelihood-based phylogeny inference. *Probability Theory and Related Fields* 169, 1 (01 Oct 2017), 3–62.

[25] Tandy Warnow. 2018. *Computational Phylogenetics: An introduction to designing methods for phylogeny estimation*. Cambridge University Press, Cambridge UK.

[26] Tandy Warnow, Bernard M.E. Moret, and Katherine St. John. 2001. Absolute convergence: true trees from short sequences. In *Proceedings of SODA*. 186–195.

[27] Karen M Wong, Marc A Suchard, and John P Huelsenbeck. 2008. Alignment uncertainty and genomic analysis. *Science* 319, 5862 (2008), 473–476.