Phylogenetic Placement of Aligned Genomes and Metagenomes with Non-tree-like Evolutionary Histories

Md Alamin
Michigan State University
Computer Science and Engineering
East Lansing, Michigan, USA

Kevin J. Liu
kjl@msu.edu
Computer Science and Engineering
Ecology, Evolution, and Behavior Program
Genetics and Genome Sciences
Michigan State University
East Lansing, Michigan, USA

ABSTRACT

Phylogenetic placement is the computational task that places a query taxon into a reference phylogeny using computational analysis of biomolecular sequence data or other evolutionary characters. A chief advantage of phylogenetic placement over one-shot phylogenetic reconstruction is greatly reduced computational requirements, and the former has been applied in many different topics in phylogenetics. One of the more recent applications has been enabled by rapid advances in biomolecular sequencing technology: classification of genomes, metagenomes, and metagenome-assembled genomes (MAGs) in large-scale datasets produced by next-generation sequencing. A number of methods have been developed for this purpose, and all share the common simplifying assumption that a phylogenetic tree suffices for modeling the evolutionary history of all genomes and/or metagenomes under study. Another parallel development in today's post-genomic era is a greater understanding of the prevalence and importance of non-tree-like evolution in the Tree of Life – the evolutionary history of all life on Earth – which in fact may not be a tree at all. More general graph representations such as phylogenetic networks have therefore been proposed, and a new generation of phylogenetic network reconstruction methods are under active development. But the simplifying assumption made by phylogenetic tree placement methods is fundamentally at odds with the non-tree-like evolutionary histories of many microbes and other organisms. The consequences of this conflict are poorly understood.

In this study, we conduct a comprehensive performance study to directly assess the impact of non-tree-like evolution on phylogenetic tree placement of genomes and metagenomes. Our study includes *in silico* simulation experiments as well as empirical data analyses. We find that the topological accuracy of phylogenetic tree placement degrades quickly as genomic sequence evolution becomes increasingly non-tree-like. We then introduce a new statistical method for phylogenetic network placement of genomes and metagenomes, which we refer to as NetPlacer version 0. Initial experiments with NetPlacer provide a proof-of-concept, but also point to the need for greater computational scalability. We conclude with thoughts on algorithmic techniques to enable fast and accurate phylogenetic network placement.

BCB '23, September 3–6, 2023, Houston, TX, USA

ACM ISBN 979-8-4007-0126-9/23/09…$15.00
https://doi.org/10.1145/3584371.3612981

CCS CONCEPTS

• **Applied computing** → *Computational genomics*; *Computational biology*; *Molecular sequence analysis*; *Molecular evolution*; *Computational genomics*; *Bioinformatics*; *Population genetics.*

KEYWORDS

phylogenetic placement, phylogenetic network, horizontal gene transfer, reticulate evolution, simulation study, Neisseria, Helicobacter, metagenome, metagenomics, metagenome assembled genome

ACM Reference Format:
Md Alamin and Kevin J. Liu. 2023. Phylogenetic Placement of Aligned Genomes and Metagenomes with Non-tree-like Evolutionary Histories. In *14th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '23), September 3–6, 2023, Houston, TX, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3584371.3612981

1 INTRODUCTION

Phylogenetic placement is the problem which seeks to place a new taxon into an existing or reference phylogeny, typically via computational analysis of biomolecular sequence data. This problem has been traditionally studied in the context of phylogenetics and systematics, including large-scale phylogenetic reconstruction [42], dynamically updated phylogenies [17, 35], and biodiversity research [6]. Thanks to rapid advances in next generation sequencing technology, computational phylogenetics has seen many major advances, and new applications of phylogenetic placement have emerged. In particular, phylogenetic placement methods are increasingly used in genomic and metagenomic studies.

One particularly important task in genomics and metagenomics is to classify organisms that are present in a sequenced sample. Classical approaches like BLAST-based sequence analysis [1, 43] are widely used for taxonomic classification of next-generation sequencing (NGS) read data and assembled biomolecular sequences and related computational tasks [41].

Matsen et al. [26] were early proponents of phylogeny-aware alternatives. As they noted, phylogenetic analyses of metagenomic data offer several key advantages that can complement taxonomic classification. First, phylogenetic placement explicitly accounts for phylogenetic relatedness, which can be a confounding factor in

downstream analyses if not properly accounted for [9, 11]. Furthermore, fine grained evolutionary relationships can add substantial insight into originating processes that underlie present day snapshots of microbial genetics. One of the first methods in this class was pplacer [26]. Other methods have been since developed to address the phylogenetic placement problem, such as EPA-ng [5], SEPP [28], TIPP [31], APPLES [4], and APPLES-2 [3]. All of these methods focus purely on phylogenetic tree placement. This requires a critical assumption: that phylogenetic relationships in a sample or study are purely tree-like.

But it is well understood that horizontal gene transfer (HGT) has played an important role in prokaryotic genome evolution throughout the Tree of Life [33]. Furthermore, the importance of reticulate evolution in other microbes has gained greater appreciation in recent years [25]. The prevalence of non-tree-like evolution in metagenomic samples is fundamentally at odds with the simplifying assumption inherent to phylogenetic tree placement. The consequences are not well understood, and solutions are not at hand. These gaps are partly due to the lack of quantitative experiments to assess the impacts of model violation on state-of-the-art phylogenetic placement algorithms, as well as the lack of alternative methods that relax the simplifying assumption of tree-like evolution of genomes and metagenomes.

In this study, we directly address both gaps. (1) We conduct a comprehensive performance study to quantify the impact of non-tree-like evolution on phylogenetic tree placement of genomes and metagenomes. (2) We introduce NetPlacer version 0, a new statistical method for phylogenetic network placement of genomes and metagenomes. To avoid ambiguity, we refer to traditional phylogenetic placement – where the reference phylogeny is restricted to be a tree – as "tree placement", and more general phylogenetic placement where the reference phylogeny is a more general phylogenetic network as "network placement". Our study focuses on phylogenetic placement using aligned biomolecular sequences, and sets the stage for generalization to other applications of phylogenetic placement.

2 METHODS

2.1 Preliminaries

We begin with relevant background and definitions. A phylogenetic tree $T = (V, E)$ is a connected acyclic graph where, for every pair of vertices $v, w \in V$, there is a unique path between v and w; furthermore, leaf nodes (or leaves) in the tree are uniquely labeled by a set of taxa Ξ, as described below. Phylogenetic trees can be of two types: rooted and unrooted. In a rooted tree, there is a unique root node $r \in V$ indicating the most recent common ancestor of all taxa in the tree, and the edge set E consists of directed edges. The root r has in-degree 0 and out-degree 2 or greater, internal nodes have in-degree 1 and out-degree 2 or greater, and leaf nodes (or leaves) have in-degree 1 and out-degree 0; each leaf node is uniquely labeled by a taxon in the set of taxa Ξ. A rooted tree is binary if the root and all internal nodes have out-degree exactly 2. In an unrooted tree, the edge set E consists of undirected edges and every node is either a leaf node if it has degree 1 or an internal node if it has degree 3 or greater; an unrooted tree is binary if all internal nodes have degree exactly 3.

Phylogenetic placement is the computational problem that places a query taxon into a backbone phylogeny using computational analysis of biomolecular sequence data and other character data. In the context of phylogenetic tree placement, the problem is defined as follows. The problem input consists of a backbone tree T on a set of reference taxa S where the number of reference taxa is $n = |S|$, a query taxon q, and a multiple sequence alignment for $S \cup \{q\}$. The problem output is a placement tree P_q that is obtained by attaching a leaf edge representing q to an existing edge in T such that a phylogenetic criterion is optimized.

A range of methods have been developed to address the phylogenetic placement problem. One class of phylogenetic placement methods utilizes maximum likelihood estimation (MLE). Prominent examples include pplacer [26] and EPA-ng [5]. These methods place a query taxon's leaf edge into the backbone tree such that model likelihood is maximized, where common models include finite-sites substitution models such as the General Time Reversible (GTR) model [34] and nested models. Another class of phylogenetic placement methods are distance based. APPLES [4] is a representative method in this class. APPLES chooses a placement for a query taxon based on computational analysis of a pairwise distance matrix computed on biomolecular sequence data for the reference taxa and query taxon. The distance calculations used for computing the pairwise distance matrix can either be estimated from a multiple sequence alignment or using an alignment-free method. As mentioned above, a simplifying assumption common to existing phylogenetic placement methods is that evolutionary history is strictly tree-like.

In the presence of reticulate evolutionary processes such as horizontal gene transfer (HGT), hybridization and introgression, and genetic recombination, the evolutionary relationships among a set of taxa requires a more complex phylogeny such as a graph-based representation known as a phylogenetic network.

A phylogenetic network χ is defined as a 3-tuple (ψ, λ, γ) which consists of a rooted directed acyclic graph $\psi = (V, E)$, edge lengths λ, and inheritance probabilities γ. The vertices V consist of the the following four classes of vertices. The root r has $\text{indeg}(r) = 0$. Leaf nodes (or leaves) are V_L where $\forall v \in V_L$ $\text{indeg}(v) = 1$ and $\text{outdeg}(v) = 0$. The tree nodes are V_T where $\forall v \in V_T$ $\text{indeg}(v) = 1$ and $\text{outdeg}(v) \geq 2$. The reticulate nodes are V_N where $\forall v \in V_N$ $\text{indeg}(v) = 2$ and $\text{outdeg}(v) = 1$. A phylogenetic network can be called a phylogenetic tree if $V_N = \{\}$.

2.2 Simulation experiments

Genomic dataset simulations. Random model networks with n taxa were sampled using the procedure described in [15], which we briefly recap here. First, a random tree was sampled under a random birth-death process using r8s [36] version 1.81. Branch lengths of the tree were then re-scaled to obtain height $h = 5.0$. To obtain a model network, ϕ reticulation(s) were added to the model phylogeny using the following procedure: for each reticulation, a time t_M was randomly selected such that $0.01 \leq t_M \leq \frac{h}{4}$. Two populations were then selected randomly at time t_M, and a reticulation edge with random orientation between the two populations was added to connect the corresponding pair of tree edges. An outgroup

was added to the resulting network at time 15.0. Our simulation conditions included datasets with $n \in \{50, 100\}$ and $\phi \in \{0, 5, 10\}$.

For each model network, ms [19] was used to conduct simulations under the multi-species coalescent and isolation-with migration (MSC+IM) model. A reticulation at time t_M was modeled using a unidirectional migration event from time $t_M - 0.01$ to $t_M + 0.01$ with migration rate 5.0. A total of 100 local coalescent histories and associated coalescent trees were sampled from each MSC+IM simulation.

Coalescent trees with branch lengths in coalescent units were converted into gene trees with branch lengths in expected numbers of substitutions using equation 3.1 in [14] and scaled mutation rate $\theta \in \{0.02, 0.06, 0.2\}$. Gene tree branch lengths were then deviated away from ultrametricity using the approach of Nakhleh et al. [30] with deviation factor $c = 2.0$.

DNA sequence evolution on each gene tree was simulated under a finite-sites model of substitutions, insertions, and deletions using INDELible version 1.03 [12]. Substitutions were simulated under the General Time-Reversible (GTR) model [34]. GTR model parameter values were based on the study of [24], where base frequency parameters $(\pi_T, \pi_C, \pi_A, \pi_G)$ were set to $(0.3115, 0.1913, 0.3004, 0.1967)$, respectively, and substitution rate parameters $(r_{TC}, r_{TA}, r_{TG}, r_{CA}, r_{CG}, r_{AG})$ were set to $(1.2620, 0.1401, 0.2878, 0.3577, 0.3083, 1.0)$, respectively. Insertions and deletions were simulated according to a power law distribution with insertion/deletion rate 0.004, distribution parameter setting $a = 1.2$, and maximum insertion/deletion length of 50 bp. Ancestral sequence length at the root of each gene tree was set to 300 bp.

The final step of the genomic data simulation procedure was to concatenate sequences across all loci in a simulation, resulting in concatenated unaligned sequence length of around 30 kb for each simulated dataset. True multiple sequence alignments (MSAs) on all loci were similarly concatenated to obtain the concatenated true MSA.

Metagenomic dataset simulations. Metagenomic datasets were simulated by coupling genomic dataset simulations with an additional metagenomic data simulation procedure. The latter used CAMISIM [13] with default settings. The CAMISIM pipeline incorporates the following stage to simulate NGS short read data from simulated multi-locus sequences for query taxa: the ART read simulator version 2.3.6 [18] was used to generate Illumina 2×150 bp paired-end reads from individual genomes with a HiSeq 2500 error profile which has been trained on the MBARC-26 training dataset [37]. The reads were generated with 10X coverage.

Experimental replication and summary statistics. For each model condition, the simulation procedure was repeated to obtain 10 replicate datasets. Results are reported across all replicate datasets in each model condition. Table 1 lists model parameter values and summary statistics for the model conditions in the simulation study. Supplementary Table S1 shows statistics on true gene tree discordance in the simulations.

Phylogenetic tree placement methods. The performance of phylogenetic tree placement was evaluated using a leave-one-out approach. For the genomic datasets, the experimental procedure consisted of the following steps. (1) Unaligned sequences S for the set of taxa Ξ were aligned using MAFFT [21] version 7.305 with default settings, resulting in an estimated MSA A. (2) Using the MSA A as input, RAxML [38] version 8.2.12 was used to perform MLE under the GTR+Γ substitution model and reconstruct the reference tree T_{REF}. The tree T_{REF} was outgroup rooted to facilitate topological comparisons against ground truth (described below); the outgroup was then discarded and otherwise not utilized in our experiments. (3) Each taxon $\xi \in \Xi$ was chosen as the query taxon q in turn. The aligned sequence a_q representing q was removed from A to obtain the reference MSA A_{REF}. The leaf edge for the query taxon q was contracted in the reference tree T_{REF}, and branch lengths of the resulting tree were re-estimated using FastME [23] analysis of the reference MSA A_{REF}; we refer to this tree as the backbone tree T. (4) Using the query sequence a_q, the reference MSA A_{REF}, and backbone tree T as input, APPLES [4] version 2.0.5 with default settings was used to place the query taxon q into the backbone tree T, resulting in the placement tree P_q. (5) Steps 3 through 5 were repeated for all other taxa as query.

The experimental procedure for metagenomic datasets required several changes compared to the genomic experiment procedure. (1-3) The first three steps of the metagenomic experiment procedure were identical to the genomic experiment procedure's first three steps. (4) The query sequence a_q for the query taxon q was used to to simulate the metagenomic NGS data (see "Metagenomic dataset simulations" above). (5) We then used metaSpades [32] version 3.13.0 with default settings to assemble NGS reads into contigs.

Table 1: *Model parameter values and summary statistics for each model condition.* The 50- and 100-taxon model conditions were named 50.A through 50.I and 100.A through 100.I, respectively. Each model condition utilized a fixed setting for the number of taxa ("# of taxa"), the scaled mutation rate θ ("Mutation rate"), and the number of reticulations ("# of retic"); additionally, the simulations utilized migration rate 5.0 and indel rate 0.004. Average sequence length of the true alignment ("True MSA length"), average normalized Hamming distance ("ANHD") across all pairs of aligned sequences in the true MSA and "Gappiness" which is the proportion of the MSA consisting of indels are reported as an average across all experimental replicates in each model condition ($n = 10$).

Model condition	# of taxa	Mutation rate	# of retic	True MSA length	ANHD	Gappiness
50.A	50	0.02	0	31158.6	0.0848	0.0365
50.B	50	0.02	5	31113.7	0.0844	0.0352
50.C	50	0.02	10	31113.6	0.0846	0.0356
50.D	50	0.06	0	33612.8	0.2181	0.1070
50.E	50	0.06	5	33685.6	0.2175	0.1085
50.F	50	0.06	10	33476.8	0.2168	0.1042
50.G	50	0.2	0	42157.3	0.4707	0.2867
50.H	50	0.2	5	42064.1	0.4693	0.2865
50.I	50	0.2	10	41524.8	0.4695	0.2786
100.A	100	0.02	0	32030.3	0.0918	0.0641
100.B	100	0.02	5	31938.0	0.0912	0.0599
100.C	100	0.02	10	32022.7	0.0916	0.0632
100.D	100	0.06	0	35872.1	0.2322	0.1623
100.E	100	0.06	5	35987.1	0.2320	0.1648
100.F	100	0.06	10	35892.0	0.2316	0.1641
100.G	100	0.2	0	49321.9	0.4872	0.3919
100.H	100	0.2	5	49745.7	0.4890	0.3956
100.I	100	0.2	10	49507.5	0.4896	0.3936

The assembled contigs served as the sequence s_q for query taxon q. The contigs in s_q were aligned to the reference MSA A_{REF} using MAFFT version 7.305 with an "-addfragments" option. (6) Phylogenetic placement of the query taxon q into the backbone tree T was performed in an identical manner as step 4 in the genomic experiment procedure. (7) The leave-one-out procedure was repeated for all other taxa in turn.

2.3 Empirical dataset analyses

Dataset from study of Treangen and Rocha [40]. We utilized genomic sequence data from the study of Treangen and Rocha [40], which examined the contribution of HGT to protein family expansion in eight groups of prokaryotes. We focused on two genera of bacteria – *Neisseria* and *Helicobacter* – where Treangen and Rocha [40]'s reported relative genomic contributions of HGT – 89% versus 97%, respectively (cf. Figure 2 in [40]) – enables differential placement experiments. Table 2 lists summary statistics for the empirical datasets.

As with the simulation study, the empirical study's experimental procedure consisted of multiple steps. (1) Open reading frames (ORFs) were predicted using Prodigal version 2.6.3 [20]. (2) USEARCH version 11.0.667 [10] was then used to align ORFs in each genome against 400 reference genes which were curated and used in PhyloPhlAn [2]. (3) A subset of 50 orthologous genes were randomly selected as the basis for the multi-locus dataset. Unaligned gene sequences for each locus were aligned using MAFFT version 7.305 with the "–auto" setting, and MSAs were concatenated across loci to obtain the reference alignment. (4) Using the reference alignment as input, RAxML was used to perform phylogenetic MLE under the GTR+Γ model and obtain a reference tree. The reference tree was then midpoint rooted. (5) Similar to the simulation experiments, a leave-one-out approach was used to perform phylogenetic placement of each taxon in turn: the query taxon was pruned from the reference tree to obtain a backbone tree, and APPLES version 2.0.5 with default settings was used to perform phylogenetic placement of the query taxon into the backbone tree.

Augmented Neisseria datasets. We also augmented the *Neisseria* dataset with synthetic reticulation events and performed leave-one-out comparative analysis of two datasets. The original or "control" dataset corresponded to the empirical *Neisseria* dataset (see steps 1 through 3 above). The control dataset was then augmented with simulated reticulation events to obtain the "augmented" dataset. Data augmentation utilized the following procedure. Beginning with the control dataset, a reference tree was obtained using step 4 above. Then, 10 random reticulations were added to the reference tree using the same approach as in the simulation study, resulting in a species network model. We used ms to simulate local coalescent histories and gene trees for 10 loci under the species network model. INDELible was then used to simulate gene sequence evolution along each gene tree, resulting in a set of gene sequences and true MSAs for each gene. The species phylogeny and gene tree simulations utilized the same procedures as in the simulation study. Finally, the simulated multi-locus unaligned sequences were appended to the empirical multi-locus unaligned sequences, and similarly for the aligned sequence data. The resulting dataset is referred to as the augmented dataset.

A companion pair of metagenomic datasets – control and augmented – was also used to perform leave-one-out comparative analysis. Each metagenomic dataset was obtained using the corresponding genomic dataset (i.e., a control metagenomic dataset was obtained using the control genomic dataset, and similarly for augmented datasets). Metagenomic NGS data simulation for a query taxon, metagenome assembly, and query taxon placement procedures followed steps 4 through 7 in the simulation study's metagenomic data experiments.

2.4 Performance assessments

Topological error assessments. Topological comparisons of phylogenetic trees were based on the Robinson-Foulds distance. For two phylogenetic trees T_a and T_b with respective bipartition sets $\mathcal{B}(T_a)$ and $\mathcal{B}(T_b)$, the Robinson-Foulds distance $\delta(T_a, T_b)$ is the size of the symmetric difference $|\mathcal{B}(T_a) - \mathcal{B}(T_b)| + |\mathcal{B}(T_b) - \mathcal{B}(T_a)|$. The normalized Robinson-Foulds (nRF) distance is obtained by dividing absolute Robinson-Foulds distance divided by its maximum, which is $2(n-3)$.

Topological comparisons of phylogenetic networks utilized Nakhleh [29]'s distance for comparing a pair of phylogenetic network topologies. For a pair of phylogenetic networks χ_a and χ_b, the distance calculation corresponds to the number of rooted sub-networks that appear in χ_a but not χ_b or vice versa. We used PhyloNet [39] to calculate topological distances between phylogenetic networks.

To assess the topological accuracy of phylogenetic placement in our study, we adapted the tree-based placement error calculations used by [4] and [3]. We refer to the adapted calculation as network delta error (NDE). Let $\mathcal{N}$ denote the model network and $\mathcal{N}_q$ is the model network with query taxon q deleted (i.e., with q's leaf edge contracted). Following the above notation, the phylogenetic placement problem under study concerns the placement of a query taxon q into a backbone tree T, resulting in placement tree P_q. The absolute NDE is defined as $\Delta(\mathcal{N}, P_q) - \Delta(\mathcal{N}_q, T)$. Relative topological error was assessed using normalized NDE, where the above absolute NDE calculation is normalized by a baseline NDE that reflects a null hypothesis where the noise-to-signal ratio is saturated. The baseline NDE was empirically estimated by repeating the absolute NDE calculation's placement procedure for a query taxon q, but replacing q's original sequence with a sequence of the same length that was chosen uniformly at random (UAR).

Normalized NDE was also used to assess topological accuracy of our new network placement method, where the backbone tree T and placement tree P_q were replaced with a backbone network and placement network.

Phylogenetic placement support. In the empirical study, we conducted phylogenetic bootstrap analyses to assess reproducibility of phylogenetic placement (i.e., estimated placement of a query taxon q into a backbone tree T using an input MSA A, resulting in a placement tree P_q). The standard bootstrap method was used to resample 100 bootstrap replicates from the MSA A. Then, to obtain a bootstrap tree on each bootstrap replicate, RAxML version 8.2.12 was used to perform maximum likelihood estimation under the GTR+Γ substitution model. The resulting set of bootstrap trees β were then used to calculate phylogenetic support for the placement tree P_q, where the support for an edge e in P_q is the proportion of

Table 2: *Summary statistics for empirical datasets.* Genomic sequence data were obtained from Treangen and Rocha [40]'s study of HGT in eight groups of prokaryotes. Each dataset consisted of 8 taxa from one of two genera – either *Neisseria* or *Helicobacter* – where the latter exhibited relatively higher genomic contributions of HGT compared to the former ("Contribution of HGT"), based on the findings of Treangen and Rocha [40]. Average unaligned genome sequence length in kb ("Avg genome length"), and the reference MSA's length in kb ("Reference MSA length") and average normalized Hamming distance ("ANHD") are also listed.

Clade	Contribution of HGT (%) Contribution of HGT (%)	Avg genome length (kb)	Reference MSA length (kb)	ANHD
Neisseria	89	2201.7	80.1	0.159
Helicobacter	97	1621.8	77.7	0.216

bootstrap trees β that display e; support for the placement of q is based on the support for the edges incident on q's leaf edge.

2.5 NetPlacer, a new phylogenetic network placement algorithm

As an alternative to phylogenetic tree placement, we introduce NetPlacer – a new computational framework for phylogenetic network placement of genomes and metagenomes. The current version of NetPlacer is version zero. A high-level flowchart diagram of NetPlacer is provided in Supplementary Figure S1.

NetPlacer utilizes a summary-based approach, where gene trees are used as input to "summarize" multi-locus sequence data. NetPlacer is thus used as part of a multi-stage computational pipeline, where gene trees are estimated in an upstream stage and then used as input to downstream phylogenetic placement. Summary-based placement offers the potential for improved scalability relative to sequence-based placement, but requires simplifying assumptions concerning gene tree estimation accuracy.

NetPlacer performs statistical optimization of placements under the multi-species network coalescent (MSNC) model [27, 44]. Whereas the multi-species coalescent (MSC) model [14, 22] accounts for genetic drift and lineage coalescence during strictly vertical evolutionary descent, the MSNC model generalizes the MSC model to also account for horizontal evolutionary processes in the form of network reticulations.

Under both the MSC and MSNC models, summary-based phylogenetic MLE requires calculation of model likelihood for a species phylogeny given a set of gene trees. [8] and [44] introduced model likelihood calculations under these respective models, where topological information from the latter is used as input. The calculation is defined as follows. Let $G = \{g_1, g_2, \ldots, g_k\}$ be the set of input gene tree topologies for summary-based inference. Following the definitions in Yu et al. [44], one-shot summary-based inference of a species network maximizes the MSNC model likelihood $\mathcal{L}(\psi, \lambda, \gamma|G) = \prod_{g \in G} \mathcal{P}(g|\psi, \lambda, \gamma)$.

We begin with the definition of the phylogenetic network placement problem that NetPlacer addresses. The problem input consists of: a backbone network $\chi = (\psi, \lambda, \gamma)$ with topology $\psi = (V, E)$ for a set of reference taxa S, a query taxon q, and a set of gene trees G_q and locus alignments A_q for taxa $S \cup \{q\}$. The output is a placement network P_q that is obtained by attaching q's leaf edge to an existing edge in the backbone topology ψ, where the placement optimizes a phylogenetic criterion. NetPlacer's phylogenetic criterion adapts one-shot MSNC likelihood maximization to the network placement problem:

$$\underset{\psi' \in \{P_q : e_q \text{ attaches to } e \in E \text{ and results in } P_q\}}{\arg\max} \; \underset{\lambda', \gamma'}{\arg\max} \, \mathcal{L}(\psi', \lambda', \gamma'|G) \tag{1}$$

We now describe the NetPlacer placement algorithm. Pseudocode for the NetPlacer MLE algorithm is shown in Algorithm 1. To begin, multi-locus data for reference taxa S consists of the following: a set of per-locus MSAs A, a set of estimated gene trees G, and an estimated species network that serves as the backbone network χ. In our experiments, a backbone network χ was estimated using PhyloNet version 3 with default settings to perform summary-based maximum MSNC likelihood optimization. The problem input also includes a de novo assembled metagenome sequence s_q for query taxon. The query taxon's sequence s_q is aligned using locus MSAs A, resulting in augmented locus MSAs A_q; we used MAFFT version 7.305 with default settings for this purpose. The augmented locus MSAs are then used to either perform one-shot gene tree estimation or place q into gene trees G, resulting in augmented gene trees G_q; our experiments used FastTree version 2.1.10 to estimate the former. Finally, maximum likelihood optimization under the MSNC criterion in equation 1 is used to place q into χ. PhyloNet's local optimization heuristics are used to perform the inner optimization of continuous parameters λ' and γ', which includes the MSNC model likelihood calculation as described by Yu et al. [44]. Exhaustive search is used to perform the outer optimization of the network topology ψ'.

2.6 NetPlacer experiments

We conducted additional simulation study experiments to assess NetPlacer's performance. We utilized the previously described metagenomic data simulation procedures (see "Metagenomic dataset simulations" above) to simulate 8-taxon datasets with either zero or one reticulation and 50 loci, where each locus had sequence length of 1 kb. Also, the placement experimental procedures used elsewhere in our simulation study (steps 1 through 7 in second paragraph under "Phylogenetic tree placement methods" above) were used in our network placement experiments, where the loci used for phylogenetic placement were restricted to the three longest contigs in an assembled metagenome. Model condition parameter settings and summary statistics for simulated datasets are shown

Algorithm 1: *Pseudocode for NetPlacer algorithm.*

Data: Backbone phylogenetic network $\chi = (\psi, \lambda, \gamma)$, set of MSAs A_{REF} and set of gene trees G_{REF} for reference taxa, query sequence s_q
Result: Placement network P_q
$A_q \leftarrow$ EstimateAugmentedMSAs(A_{REF}, s_q)
$G_q \leftarrow$ EstimateAugmentedGeneTrees(G_{REF}, A_q)
maxLikelihood ←lowest value ;
for *each directed edge* $e \in \psi$ **do**
  $\psi' \leftarrow$ Add an intermediate node(i) in e and attach q as a leaf node to create the edge (i, q) ;
  $\lambda', \gamma', \mathcal{L}(\psi', \lambda', \gamma' | G_q) \leftarrow$ CalGTProb(ψ', G_q) ;
  /* calculated using expression 1 and PhyloNet's CalGTProb implementation */
  $\chi' \leftarrow (\psi', \lambda', \gamma')$;
  currL $\leftarrow \mathcal{L}(\psi', \lambda', \gamma' | G_q)$;
  if *currL* > *maxLikelihood* **then**
    $\chi_p \leftarrow \chi'$;
    $maxL \leftarrow currL$;
  end
end
return χ_p ;

in Table 3. NetPlacer performance was assessed based on topological error (using the normalized NDE calculation described above), computational runtime, and peak main memory usage.

2.7 Computing facilities

Experiments in our study were conducted using the High-Performance Computing Center at Michigan State University (MSU), which is hosted and maintained by the MSU Institute for Cyber-Enabled Research. All experiments were conducted on the amd-20 cluster which is comprised of compute nodes with 2.595 GHz AMD EPYC 7H12 processors and 0.5, 1, or 2 TB RAM per compute node.

2.8 Data availability

The datasets and scripts used in our study are available under open copyleft licenses at https://gitlab.msu.edu/liulab/impact-of-non-tree-like-evolution-on-phylogenetic-placement.

3 RESULTS

3.1 Simulation study

Performance evaluation of tree placement methods. Figure 1 and Supplementary Table S2 show the impact of reticulations on the topological error of tree placement using genomes. We first consider the 50-taxon simulations with the lowest mutation rate $h = 0.02$. For the simulation condition with 0 reticulations, evolution is strictly tree-like. It is precisely on the 0-reticulation simulation conditions that we observed the highest placement accuracies throughout our study. Consistent with the simulation studies of Balaban et al. [4] and Balaban et al. [3], normalized delta error averaged 6.5%, which is far from saturation. As the number of reticulations increases from 0 to 5, normalized topological error increased by multiple factors – over half an order of magnitude, on average. Then, as the number of reticulations doubled again from 5 to 10, normalized delta error topped 50% on average. On 50-taxon simulation conditions with higher mutation rates, a similar pattern was observed. Increasing evolutionary divergence was associated with relatively small increases in observed topological error, compared to the effect of increasing numbers of reticulations.

A companion set of experiments involved tree placement of metagenomes (Figure 1 and Supplementary Table S2). On the 50-taxon simulation condition with the lowest mutation rate $h = 0.02$ and 0 reticulations, normalized topological error of metagenome placements increased dramatically compared to genome placements – amounting to an increase of around an order of magnitude, on average. As the evolutionary simulations became more non-tree-like – moving from 0 to 5 to 10 reticulations – we consistently observed concomitant increases in normalized topological error of metagenome placements, which mirrored the experimental findings for genome placements. At the high end of 10 reticulations, normalized delta error became as large as 70% to 75%, which begins to approach error saturation. As in the genome placement experiments, increasing mutation rates – to 0.06 and 0.2 – had a relatively smaller effect on metagenome placement error, compared to the effect of increasing reticulations.

Figure 2 and Supplementary Table S3 show results for tree placement error outcomes on the 100-taxon simulation conditions. Overall, normalized topological error outcomes on 100-taxon simulation conditions were qualitatively similar to 50-taxon conditions. Across different data types (genomic vs. metagenomic) and mutation rates, we observed the smallest placement error on 0-reticulation conditions, and increasing reticulations consistently resulted in increased placement error. The impact of increasing reticulations tended to be larger than those observed for mutation rate and dataset size in terms of number of taxa. Finally, metagenome placement error was multiple factors larger than genome placement error, and the relative influence of other experimental factors became more difficult to discern as metagenome placement error approached saturation – with maximum normalized delta error of 85% or so.

Performance evaluation of NetPlacer, a new network placement method. Topological accuracy assessments are shown in Figure 3. For strictly tree-like simulations (i.e., 0 reticulations), network placement returned normalized delta error of around 27%, on average. On non-tree-like simulation with a single reticulation, NetPlacer returned average normalized delta error of around 45%.

NetPlacer's computational runtime and main memory usage are shown in Table 4. On tree-like simulations, NetPlacer's runtime amounted to a few minutes per placement on average. In comparison, NetPlacer's per-placement runtime increased from a few minutes to half an hour, on average – an increase of around half an order of magnitude. Main memory usage increased by 30% as well, but was under 1 GiB on average – well within the scope of modern personal computers.

3.2 Empirical study of tree placement methods

Our empirical study included reproducibility assessments using genomic sequence data from the study of Treangen and Rocha [40]. HGT was the driving factor for 97% of Helicobacter protein family expansions, versus 89% in Neisseria, indicating a differential role

Table 3: *Model conditions and summary statistics for the NetPlacer experiments in the simulation study.* Each model condition included 10 experimental replicates. True gene tree discordance was assessed using nRF distance, and the average discordance is reported across a model condition ($n = 10$). For each model condition, gene tree estimation error was assessed using using average nRF distance between an estimated gene tree and true gene tree, and the average error is reported across a model condition ($n = 10$).

Model condition	# Taxa	Mutation rate	#Reticulations	Migration rate	Seq lengths	ANHD	True gene tree discordance	Gene tree estimation error
8.A	8	0.2	0	5.0	50 kb	0.6434	0.20	0.26
8.B	8	0.2	1	5.0	50 kb	0.6435	0.28	0.27

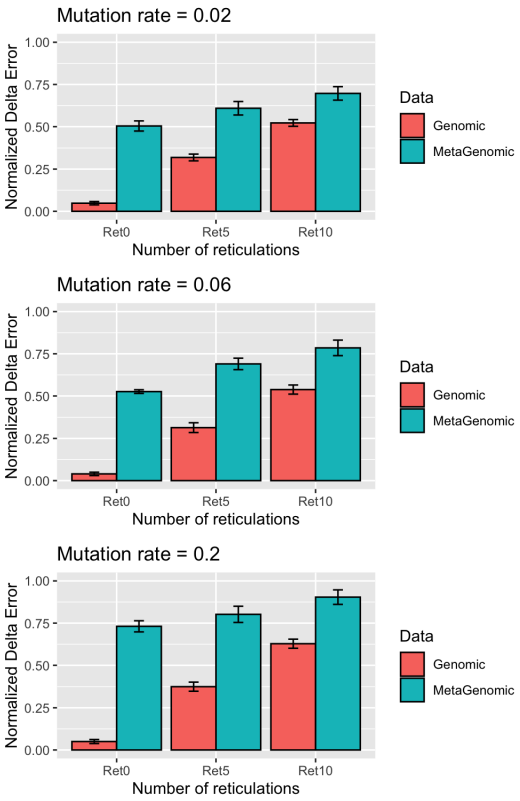

Figure 1: *Phylogenetic tree placement error in the 50-taxon simulation experiments.* Results are reported for genomic and metagenomic data simulations with a mutation rate of either 0.02, 0.06 or 0.2 and either 0, 5 or 10 reticulations. APPLES was used to perform phylogenetic tree placement. Phylogenetic placement error was assessed using normalized delta error (NDE). Average and standard error bars are shown for each model condition ($n = 10$).

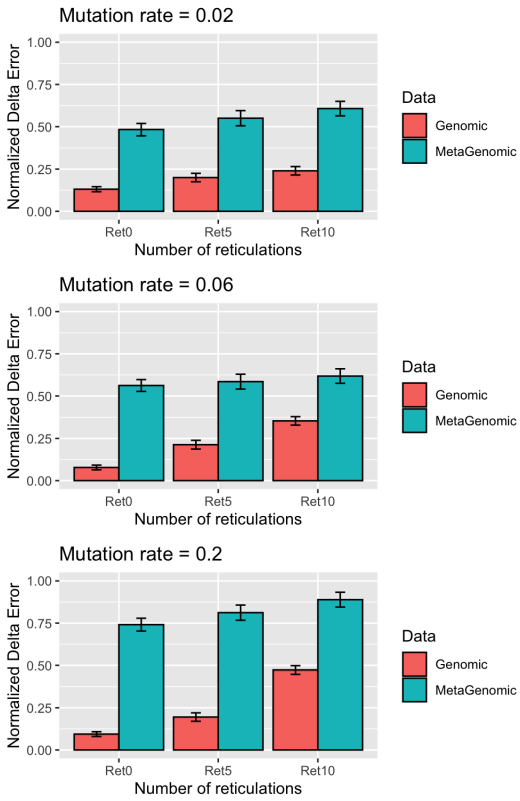

Figure 2: *Phylogenetic tree placement error in the 100-taxon simulation experiments.* Figure description and layout are otherwise identical to Figure 1.

Table 4: *NetPlacer experiment results: runtime and main memory usage.* Each simulation condition included 8 taxa and either 0 or 1 reticulation. Runtime and peak main memory utilization for a single query placement are reported as an average for each model condition ($n = 10$).

# Reticulations	Run time (Minutes)	Memory usage (MB)
0	6.17	658.98
1	33.31	858.53

of HGT in genome evolution within the two clades (cf. Figure 2 in [40]).

One set of experiments used bootstrap resampling to evaluate phylogenetic placement support for query taxa in Helicobacter versus Neisseria. Consistent with Treangen and Rocha [40]'s relative findings of HGT in Neisseria and Helicobacter – less for the former versus the latter – we find that reproducibility of tree placement for Neisseria genomes exceeds that of Helicobacter genomes – ∼ 85%

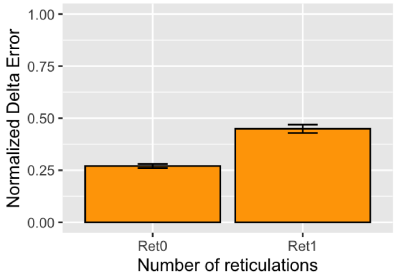

Figure 3: *NetPlacer experiment results: phylogenetic placement error.* The NetPlacer method was used to perform phylogenetic network placement of aligned metagenomes. Phylogenetic placement error was assessed using normalized delta error. Each simulation condition included 8 taxa and either 0 or 1 reticulation. Average and standard error bars are shown for each model condition ($n = 10$).

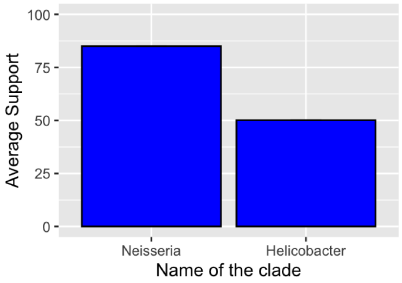

Figure 4: *Empirical study: bootstrap analysis results.* Phylogenetic bootstrap support was calculated for placement trees in the empirical study. Each bootstrap analysis utilized 100 bootstrap replicates. Each clade (i.e., Neisseria and Helicobacter) has 8 taxa.

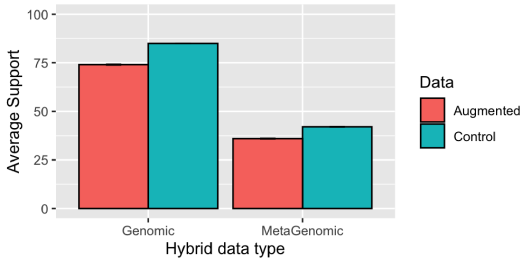

Figure 5: *Hybrid study: bootstrap analysis results.* The hybrid genomic and metagenomic datasets were obtained using augmentation of the empirical study datasets (see Methods section for details). Phylogenetic bootstrap support was calculated for placement trees using 100 bootstrap replicates.

for the former versus ~ 50% for the latter, as measured using phylogenetic bootstrap support for query taxon placement (Figure 4). We note a key distinction with respect to the rest of the study: the analyses utilized reproducibility assessments, rather than direct accuracy assessments, and our comparative findings are based on differential HGT reported by [40] in two clades under study. The choice is a practical one due to the lack of explicit ground truth.

Another set of experiments evaluated reproducibility using augmented empirical dataset analyses. Two forms of augmentation were used. (1) The first consisted of empirical genomic dataset augmentation with simulated HGT events, where the Neisseria-estimated phylogeny was augmented with simulation of additional reticulations. We will refer to original empirical dataset as "control" and simulation-augmented dataset as "augmented". (2) The second consisted of companion metagenomic datasets, where control or augmented genomic datasets were used to perform metagenomic data simulations and analysis. The latter followed the procedures used for simulating metagenomic data in the simulation study. Reproducibility of the original empirical estimate serves as a "control" baseline. Artificial reticulations are then added using a simulated data augmentation procedure, resulting in a hybrid dataset. Dataset augmentation with simulated reticulation events has the expected effect of reducing tree placement reproducibility. We saw a reduction of about 10% on genomic data (Figure 5). A smaller reduction was seen on metagenomic data, as compared to the genomic data analyses. We attribute the finding to lower overall reproducibility due to the added complexity of metagenomic data processing and analysis, where performance assessment comparisons tend to become more muted as error approaches the saturation point. The finding is consistent with the genomic versus metagenomic data comparisons in simulation study.

4 DISCUSSION

Throughout our performance study, we observed a strong impact of reticulate evolution on topological accuracy and/or repeatability of phylogenetic tree placement. The finding was consistently observed across the model conditions in our study, which spanned a range of dataset sizes in terms of number of taxa, evolutionary divergence, and complexity of model phylogeny in terms of number of reticulations. Consistent outcomes were also observed in the empirical study. We interpret the finding to be primarily due to the violation of the simplifying assumption of tree-like evolution that is made by state-of-the-art phylogenetic placement methods. As the number of reticulations increases, the model violation grows stronger and so too did topological error of tree-based placements in our simulation study experiments. The impact of increasing numbers of reticulations on phylogenetic placement outstrips that of other factors such as evolutionary divergence and dataset size. And yet the amount of reticulations in our experiments and analyses is expected to be an underestimate for most microbial genomic and metagenomic studies. Depending on the group(s) under study, the gap may amount to multiple orders of magnitude.

The simulation study experiments yielded a few comparisons worth noting. We observed an important difference between the genome placement and metagenome placement experiments: increasing reticulations tended to yield smaller absolute increases in normalized delta error in the latter versus the former. We attribute this difference to the higher placement error observed in the latter versus the former. Performance comparisons at or near error saturation are especially problematic, where metagenome assembly and

processing error becomes so large as to swamp downstream phylogenetic signal for phylogenetic placement and other subsequent computational tasks. Another difference concerned the 50- and 100-taxon experiments. On comparable pairs of model conditions that differed only in terms of the number of taxa, the 0-reticulation conditions yielded tree placement errors that were somewhat higher on the latter compared to the former; however, the reverse was true on the 5- and 10-reticulation model conditions. One contributing factor is the slightly elevated ANHD of the latter versus former, which is as expected under the simulation models and procedures (i.e., increasing dataset size in terms of number of taxa also increases the sum of branch lengths in the model phylogeny). Slightly greater sequence divergence may increase the noise to signal ratio in the zero-reticulation simulation experiments. The non-tree-like simulations (with 5 or 10 reticulations) add extra complicating factors of model mis-specification and varying model complexity.

Using 8-taxon simulations with either no reticulations or a single reticulation, we studied the performance of NetPlacer, our new method for phylogenetic network placement of aligned genomes and metagenomes. NetPlacer's placement error was somewhat higher for datasets with non-tree-like evolutionary histories, as compared to those with strictly tree-like histories. While both are far from error saturation, our study's other findings suggest that simulations with more reticulations would see further elevation of NetPlacer's placement error. One factor is worth noting: the NetPlacer experiments added a layer of complexity that is not present in the rest of our study: estimated gene tree error. This additional factor likely contributed to more challenging placements. We surmise that more accurate gene trees may yield more accurate summary-based phylogenetic placements.

But a bigger concern with network placement is computational scalability. As with other state-of-the-art statistical methods for estimating phylogenetic networks from genomic and multi-locus sequence data, scalability on non-trivially sized datasets is a major challenge. The experimental outcomes clearly demonstrate the tradeoff at hand. A more complex model is a better fit for the data and can bring topological accuracy improvements, but comes at the significant cost of greatly increased computational runtime requirements. The tradeoff motivates the need for scalability enhancements as part of future research. This is a primary reason why our new method is referred to as NetPlacer version 0. The version number reflects a proof-of-concept status. Later versions require new algorithmic techniques to enhance scalability by multiple orders of magnitude (and see below for relevant future research directions).

A brief aside: we caution that it is difficult to make direct comparisons between tree placement methods and network placement methods. Differences in model complexity (i.e., a tree versus a network with one or more reticulations) greatly complicate head-to-head evaluation. Similar situations arise in other phylogenetic contexts (e.g., comparison of non-binary trees versus binary trees). Another key difference between these method classes is worth noting as well. The tree placement methods under study use a concatenation approach, whereas NetPlacer uses multi-locus statistical analysis that directly accounts for local gene tree discordance.

5 CONCLUSIONS

In summary, the impact of non-tree-like evolution on tree placement accuracy of genomes and metagenomes was confirmed and quantified using *in silico* simulations and empirical data analyses. We also introduced a new phylogenetic network placement method: NetPlacer version 0. We evaluated NetPlacer's performance using simulated benchmarking datasets, and we found that relaxing the simplifying assumption of tree-like evolution came at a cost – namely, computational overhead.

We conclude with some thoughts on future research directions. In our opinion, the foremost need concerns new network placement method development. NetPlacer version 0 provides an initial proof of concept, but scalability-enhancing algorithmic techniques are clearly needed. Particularly salient is one of our past contributions to phylogenetic inference and learning using large-scale biomolecular sequence datasets: FastNet, a phylogenetic divide-and-conquer algorithm for fast and accurate species network reconstruction [16]. Placement of query taxa into "sub"-networks inferred on subproblems – as represented by FastNet's subproblem decomposition graph – may prove more tractable than placement into the full dataset, which is larger and more divergent than any individual subproblem. Also, phylogenetic network placement using multi-locus sequence data that integrates over the distribution of all gene tree placements under a maximum likelihood or other statistical criterion would provide an alternative to NetPlacer's summary-based approach. As above, the primary anticipated challenge is scalability. One possible solution would be to adapt Bryant et al. [7]'s dynamic programming calculation to this task.

ACKNOWLEDGMENTS

The authors would like to thank two anonymous reviewers for their detailed and constructive feedback. This research has been supported in part by the National Science Foundation (2144121, 2214038, 1737898 to KJL). All computational experiments and analyses were performed on the MSU High Performance Computing Center, which is part of the MSU Institute for Cyber-Enabled Research.

REFERENCES

[1] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. 1990. Basic local alignment search tool. *Journal of Molecular Biology* 215, 3 (1990), 403–410.

[2] Francesco Asnicar, Andrew Maltez Thomas, Francesco Beghini, Claudia Mengoni, Serena Manara, Paolo Manghi, Qiyun Zhu, Mattia Bolzan, Fabio Cumbo, Uyen May, et al. 2020. Precise phylogenetic analysis of microbial isolates and genomes from metagenomes using PhyloPhlAn 3.0. *Nature Communications* 11, 1 (2020), 1–10.

[3] Metin Balaban, Yueyu Jiang, Daniel Roush, Qiyun Zhu, and Siavash Mirarab. 2022. Fast and accurate distance-based phylogenetic placement using divide and conquer. *Molecular Ecology Resources* 22, 3 (2022), 1213–1227.

[4] Metin Balaban, Shahab Sarmashghi, and Siavash Mirarab. 2020. APPLES: scalable distance-based phylogenetic placement with or without alignments. *Systematic Biology* 69, 3 (2020), 566–578.

[5] Pierre Barbera, Alexey M Kozlov, Lucas Czech, Benoit Morel, Diego Darriba, Tomáš Flouri, and Alexandros Stamatakis. 2019. EPA-ng: massively parallel evolutionary placement of genetic sequences. *Systematic Biology* 68, 2 (2019), 365–369.

[6] Holly M Bik, Dorota L Porazinska, Simon Creer, J Gregory Caporaso, Rob Knight, and W Kelley Thomas. 2012. Sequencing our way towards understanding global eukaryotic biodiversity. *Trends in Ecology & Evolution* 27, 4 (2012), 233–243.

[7] David Bryant, Remco Bouckaert, Joseph Felsenstein, Noah A Rosenberg, and Arindam RoyChoudhury. 2012. Inferring species trees directly from biallelic

genetic markers: bypassing gene trees in a full coalescent analysis. *Molecular Biology and Evolution* 29, 8 (2012), 1917–1932.
[8] James H Degnan and Laura A Salter. 2005. Gene tree distributions under the coalescent process. *Evolution* 59, 1 (2005), 24–37.
[9] Casey W Dunn, Felipe Zapata, Catriona Munro, Stefan Siebert, and Andreas Hejnol. 2018. Pairwise comparisons across species are problematic when analyzing functional genomic data. *Proceedings of the National Academy of Sciences* 115, 3 (2018), E409–E417.
[10] Robert Edgar. 2010. *Usearch*. Technical Report. Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States).
[11] Joseph Felsenstein. 1985. Phylogenies and the comparative method. *The American Naturalist* 125, 1 (1985), 1–15.
[12] William Fletcher and Ziheng Yang. 2009. INDELible: a flexible simulator of biological sequence evolution. *Molecular Biology and Evolution* 26, 8 (2009), 1879–1888.
[13] Adrian Fritz, Peter Hofmann, Stephan Majda, Eik Dahms, Johannes Dröge, Jessika Fiedler, Till R Lesker, Peter Belmann, Matthew Z DeMaere, Aaron E Darling, et al. 2019. CAMISIM: simulating metagenomes and microbial communities. *Microbiome* 7, 1 (2019), 1–12.
[14] Jotun Hein, Mikkel Schierup, and Carsten Wiuf. 2004. *Gene Genealogies, Variation and Evolution: a Primer in Coalescent Theory*. Oxford University Press, USA.
[15] Hussein A Hejase and Kevin J Liu. 2016. A scalability study of phylogenetic network inference methods using empirical datasets and simulations involving a single reticulation. *BMC Bioinformatics* 17, 1 (2016), 1–12.
[16] Hussein A Hejase, Natalie VandePol, Gregory M Bonito, and Kevin J Liu. 2018. FastNet: fast and accurate statistical inference of phylogenetic networks using large-scale genomic sequence data. In *Comparative Genomics: 16th International Conference, RECOMB-CG 2018, Magog-Orford, QC, Canada, October 9-12, 2018, Proceedings 16*. Springer, 242–259.
[17] Cody E Hinchliff, Stephen A Smith, James F Allman, J Gordon Burleigh, Ruchi Chaudhary, Lyndon M Coghill, Keith A Crandall, Jiabin Deng, Bryan T Drew, Romina Gazis, et al. 2015. Synthesis of phylogeny and taxonomy into a comprehensive tree of life. *Proceedings of the National Academy of Sciences* 112, 41 (2015), 12764–12769.
[18] Weichun Huang, Leping Li, Jason R Myers, and Gabor T Marth. 2012. ART: a next-generation sequencing read simulator. *Bioinformatics* 28, 4 (2012), 593–594.
[19] Richard R Hudson. 2002. ms a program for generating samples under neutral models. *Bioinformatics* 18, 2 (2002), 337–338.
[20] Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer, and Loren J Hauser. 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* 11, 1 (2010), 1–11.
[21] Kazutaka Katoh and Daron M Standley. 2013. MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Molecular Biology and Evolution* 30, 4 (2013), 772–780.
[22] John Frank Charles Kingman. 1982. The coalescent. *Stochastic Processes and Their Applications* 13, 3 (1982), 235–248.
[23] Vincent Lefort, Richard Desper, and Olivier Gascuel. 2015. FastME 2.0: a comprehensive, accurate, and fast distance-based phylogeny inference program. *Molecular Biology and Evolution* 32, 10 (2015), 2798–2800.
[24] Kevin Liu, Tandy J Warnow, Mark T Holder, Serita M Nelesen, Jiaye Yu, Alexandros P Stamatakis, and C Randal Linder. 2012. SATe-II: very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. *Systematic Biology* 61, 1 (2012), 90.
[25] James Mallet, Nora Besansky, and Matthew W Hahn. 2016. How reticulated are species? *BioEssays* 38, 2 (2016), 140–149.
[26] Frederick A Matsen, Robin B Kodner, and E Armbrust. 2010. pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics* 11, 1 (2010), 1–16.
[27] Chen Meng and Laura Salter Kubatko. 2009. Detecting hybrid speciation in the presence of incomplete lineage sorting using gene tree incongruence: a model. *Theoretical Population Biology* 75, 1 (2009), 35–45.
[28] Siavash Mirarab, Nam Nguyen, and Tandy Warnow. 2012. SEPP: SATé-enabled phylogenetic placement. In *Biocomputing 2012*. World Scientific, 247–258.
[29] Luay Nakhleh. 2009. A metric on the space of reduced phylogenetic networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 7, 2 (2009), 218–222.
[30] Luay Nakhleh, Bernard ME Moret, Usman Roshan, Katherine St. John, Jerry Sun, and Tandy Warnow. 2001. The accuracy of fast phylogenetic methods for large datasets. In *Biocomputing 2002*. World Scientific, 211–222.
[31] Nam-phuong Nguyen, Siavash Mirarab, Bo Liu, Mihai Pop, and Tandy Warnow. 2014. TIPP: taxonomic identification and phylogenetic profiling. *Bioinformatics* 30, 24 (2014), 3548–3555.
[32] Sergey Nurk, Dmitry Meleshko, Anton Korobeynikov, and Pavel A Pevzner. 2017. metaSPAdes: a new versatile metagenomic assembler. *Genome Research* 27, 5 (2017), 824–834.
[33] Howard Ochman, Jeffrey G Lawrence, and Eduardo A Groisman. 2000. Lateral gene transfer and the nature of bacterial innovation. *Nature* 405, 6784 (2000), 299–304.
[34] F. Rodriguez, J.L. Oliver, A. Marin, and J.R. Medina. 1990. The general stochastic model of nucleotide substitution. *Journal of Theoretical Biology* 142 (1990), 485–501.
[35] Luna L Sánchez-Reyes, Martha Kandziora, and Emily Jane McTavish. 2021. Physcraper: a Python package for continually updated phylogenetic trees using the Open Tree of Life. *BMC Bioinformatics* 22, 1 (2021), 1–13.
[36] Michael J Sanderson. 2003. r8s: inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock. *Bioinformatics* 19, 2 (2003), 301–302.
[37] Esther Singer, Bill Andreopoulos, Robert M Bowers, Janey Lee, Shweta Deshpande, Jennifer Chiniquy, Doina Ciobanu, Hans-Peter Klenk, Matthew Zane, Christopher Daum, et al. 2016. Next generation sequencing data of a defined microbial mock community. *Scientific Data* 3, 1 (2016), 1–8.
[38] Alexandros Stamatakis. 2014. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics* 30, 9 (2014), 1312–1313.
[39] Cuong Than, Derek Ruths, and Luay Nakhleh. 2008. PhyloNet: a software package for analyzing and reconstructing reticulate evolutionary relationships. *BMC Bioinformatics* 9 (2008), 1–16.
[40] Todd J Treangen and Eduardo PC Rocha. 2011. Horizontal transfer, not duplication, drives the expansion of protein families in prokaryotes. *PLoS Genetics* 7, 1 (2011), e1001284.
[41] Susannah Green Tringe and Edward M Rubin. 2005. Metagenomics: DNA sequencing of environmental samples. *Nature Reviews Genetics* 6, 11 (2005), 805–814.
[42] Tandy Warnow. 2013. Large-scale multiple sequence alignment and phylogeny estimation. *Models and Algorithms for Genome Evolution* (2013), 85–146.
[43] Derrick E Wood and Steven L Salzberg. 2014. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biology* 15, 3 (2014), 1–12.
[44] Yun Yu, James H Degnan, and Luay Nakhleh. 2012. The probability of a gene tree topology within a phylogenetic network with applications to hybridization detection. *PLoS Genetics* 8, 4 (2012), e1002660.

Received 11 June 2023; revised 04 August 2023; accepted

Supplementary Online Materials: Phylogenetic Placement of Aligned Genomes and Metagenomes with Non-tree-like Evolutionary Histories

MD ALAMIN, Michigan State University, USA
KEVIN J. LIU, Michigan State University, USA

ACM Reference Format:
Md Alamin and Kevin J. Liu. 2023. Supplementary Online Materials: Phylogenetic Placement of Aligned Genomes and Metagenomes with Non-tree-like Evolutionary Histories. In *14th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '23), September 3–6, 2023, Houston, TX, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3584371.3612981

S1 SUPPLEMENTARY METHODS

Software commands used in the simulation study. Model trees were sampled using r8s [7] version 1.81 with the following script:

```
begin r8s;
simulate diversemodel=bdback seed=random_number nreps=10
ntaxa=< 50 or 100 >  T=0;
describe tree=0 plot=chrono_description;
end;
```

Local coalescent histories and gene trees were simulated under a model network using ms [2] with the following command:

```
ms  <50 or 100> 100 -T -I 100 <n1 n2 ... nk>
- ej <t> i j -em <tA> i j 5.0 -em <tB> i j 0
```

The "-T" parameter outputs sampled gene trees. The "-I" parameter is followed by the number of structured populations. The list of integers ($n_1 n_2 ... n_k$) represents the number of alleles sampled from each population. One allele per population was sampled in our experiments. The "-ej" parameter specifies a speciation event where all lineages in population i are moved to population j at time t. The first "-em" parameter specifies the start of a migration event at time t_A from population j to population i with migration rate 5.0; the second "-em" parameter specifies that the end of the migration event from population j to population i will occur at time t_B by setting the migration rate to zero. The above example command includes a single reticulation event; more reticulations can be added via additional "-em" parameter options.

MSA estimation was performed using MAFFT [4] version 7.305 using the following command:

```
mafft --auto <sequence file> >
<estimated MSA file>
```

A query sequence was aligned to a reference MSA using the following command:

BCB '23, September 3–6, 2023, Houston, TX, USA

ACM ISBN 979-8-4007-0126-9/23/09…$15.00
https://doi.org/10.1145/3584371.3612981

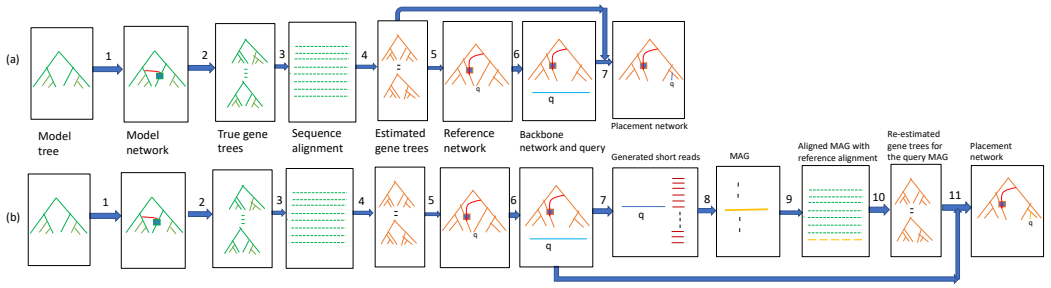

Fig. S1. A flow chart showing the steps of the simulation study for the network based placement for both genomic(a) and metagenomic(b) data. Step 1 and 2 are the same steps followed in the simulation study section to simulate true gene trees. First a random birth-date model tree was generated using r8s and then reticulate nodes were added to generate the model network. Step 3) Sequence evolution was simulated using seq-gen, which takes the gene trees as the input and simulates sequence evolution along each genealogy under Jukes-Cantor substitution model. We simulated 1000 bp per locus. Step 4) FastTree was used to estimate the gene trees. The estimated gene trees were rooted using the outgroup rooting method and the out groups were then discarded. Step 5) PhyloNet [9] was used to infer a network using the rooted estimated gene trees. We call this inferred network as the reference network. Step 6) Leave on out methods were followed as before for the placements. Each query taxon was pruned from the inferred reference network to find the backbone network. Staring from step 7, the procedures were different for (a) genomic and (b) metagenomic data. In (a), the query taxon was replaced in the backbone network to find the placement network. In (b), Illumina short reads were generated in step 7 from the query using CAMISIM. Step 8) MetaSpades was used to find the assembled contigs which were then treated as the query. Step 9) The query was aligned with the reference alignment using MAFFT. Step 10) The locus boundaries of the alignments were used to re-estimate the corresponding the gene trees for the query. Step 11) The query was then placed in the backbone network by the network based placement method using the re-estimated gene trees.

Table S1. *Topological discordance among true gene trees in simulations.* Topological discordance was measured using normalized Robinson-Foulds ("nRF") distance between true gene tree pairs. Average ("Avg") and standard error ("SE") are reported across all experimental replicates in an MSC+IM simulation condition ($n = 10$).

# of taxa	# of retic	nRF Avg	nRF SE
50	0	0.47	0.0003
50	5	0.53	0.0004
50	10	0.59	0.0004
100	0	0.52	0.0002
100	5	0.56	0.0003
100	10	0.60	0.0003

```
mafft --auto --addfragments <query sequence file>
<reference alignment file> > <estimated MSA file>
```

RAxML [8] version 8.2.12 was used to estimate a phylogenetic tree from an MSA file using the following command:

```
raxmlHPC -s <MSA file> -n <unique_id> -p <random number>
```

```
 -m GTRGAMMA
```

Phylogenetic bootstrap support analysis utilized the following two commands. The first command generates 100 bootstrap trees, and the second command calculates phylogenetic support values for an annotation tree using the bootstrap trees.

```
raxmlHPC-PTHREADS-AVX2 -m GTRGAMMA -p <random number>
-b <random number> -# 100 -s <MSA file> -n <unique id>
-T 8

raxmlHPC -m GTRGAMMA -p <random number> -f b
-t <placement tree> -z <bootstrap tree file> -n <unique id>
```

Software commands used in NetPlacer experiments. PhyloNet [9] version 3 was used to infer a reference network from an input set of estimated gene trees. The following sample Nexus file was used to configure the PhyloNet analysis:

```
#NEXUS
BEGIN TREES;
Tree geneTree1 = <gene tree 1 in Newick format>
.
.
Tree geneTree50 = <gene tree 50 in Newick format>
END;
BEGIN PHYLONET;
InferNetwork_ML (geneTree1, geneTree2, . . . . . , geneTree50)
 #num_of_reticulation -pl 8;
END;
```

Option "-pl" indicates the number of processors used.

The CalGTProb command from the PhyloNet software package was used to calculate model likelihood for a network topology given a set of gene trees. The following sample Nexus file was used to perform the calculation:

```
#NEXUS
BEGIN NETWORKS;
Network net = <phylogenetic network topology in extended Newick format>;
END;
BEGIN TREES;
Tree geneTree1 = <gene tree 1 in Newick format>
.
.
Tree geneTree50 = <Gene tree 50 in Newick format>
END;
BEGIN PHYLONET;
CalGTProb net (geneTree1, geneTree2, . . . . . , geneTree50)
 -o -pl 8;
END;
```

The "-o" option was used to estimate network branch lengths and inheritance probabilities.

Software commands used in the empirical study. Prodigal [3] version 2.6.3 was used to find ORFs in input genomes using the following command:

Table S2. Delta error constituents for model conditions with 50 taxa. Delta error has two constituent: A, which denotes the placement tree error and B, which denotes the reference tree error. Both A and B are measured using Nakhleh distance. The average and standard error values are shown ($n = 10$).

Data type	Mutation rate	#Reticulations	Average(A)	Standard Error(A)	Average(B)	Standard Error(B)
Genomic	0.02	0	14.1	0.308	13.85	0.308
Genomic	0.02	5	36.174	0.152	35.01	0.169
Genomic	0.02	10	52.088	0.111	50.828	0.117
Genomic	0.06	0	13.962	0.179	13.762	0.189
Genomic	0.06	5	35.704	0.179	34.616	0.181
Genomic	0.06	10	51.044	0.133	49.682	0.140
Genomic	0.2	0	13.206	0.295	12.982	0.295
Genomic	0.2	5	36.062	0.190	34.986	0.204
Genomic	0.2	10	50.83	0.118	49.38	0.129
Metagenomic	0.02	0	16.388	0.148	13.34	0.14
Metagenomic	0.02	5	37.524	0.123	35.6	0.13
Metagenomic	0.02	10	52.31	0.069	50.26	0.08
Metagenomic	0.06	0	15.652	0.167	12.84	0.16
Metagenomic	0.06	5	37.037	0.106	34.82	0.12
Metagenomic	0.06	10	52.358	0.063	50.35	0.07
Metagenomic	0.2	0	15.118	0.153	11.45	0.14
Metagenomic	0.2	5	37.38	0.117	35.11	0.12
Metagenomic	0.2	10	51.17	0.067	49.08	0.07

```
prodigal -i <sequence file> -o <output file>
-a <translated proteins file>
```

USEARCH [1] version 11.0.667 was used to align ORFs against reference genes using the following command:

```
usearch11.0.667_i86linux32 -ublast <translated proteins file>
-db <data base file of the reference genes>
-evalue 1e-40 -top_hit_only -blast6out <output file>
```

S2 SUPPLEMENTARY RESULTS

Figure S2 and Figure S3 compare phylogenetic placement error returned by APPLES and pplacer [5]. In general, we found that APPLES returned better placement error than pplacer in most of the model conditions.

Table S4 and S5 report type I and type II error for phylogenetic placement experiments on 50-taxon simulations. The reported errors are based on Nakhleh's metric [6] for comparing a pair of phylogenetic network topologies, where the symmetric difference used in the metric is split into two parts (i.e., one part corresponding to network structure appearing in one network but not the other, and the other part corresponding to vice versa).

We performed statistical testing to compare placement tree error for the query taxon's original sequence versus placement of a random sequence for the query taxon (i.e., the "null" baseline used in our simulation experiments' normalized error assessments). Table S6 and Table S7 report p-values for genomic and metagenomic data simulations with mutation rate 0.2. The differences in error were statistically significant for all model conditions ($\alpha = 0.05$). We did observe that, as the number of reticulations increased, the p-values increased as well. We also observed that p-values for metagenomic data simulations were always larger than those for genomic data simulations with otherwise equivalent simulation settings.

Table S3. Delta error constituents for model conditions with 100 taxa. Table layout and description are otherwise identical to Supplementary Table S2.

Data type	Mutation rate	#Reticulations	Average(A)	Standard Error(A)	Average(B)	Standard Error(B)
Genomic	0.02	0	28.231	0.255	27.336	0.27
Genomic	0.02	5	54.898	0.175	54.025	0.18
Genomic	0.02	10	75.887	0.177	75.272	0.18
Genomic	0.06	0	26.195	0.23	23.037	0.29
Genomic	0.06	5	53.775	0.22	53.117	0.16
Genomic	0.06	10	74.693	0.15	72.4	0.18
Genomic	0.2	0	23.525	.29	23.037	0.29
Genomic	0.2	5	53.658	0.15	53.117	0.162
Genomic	0.2	10	73.628	0.16	72.4	0.18
Metagenomic	0.02	0	32.038	0.133	28.639	0.13
Metagenomic	0.02	5	55.503	0.12	52.94	0.12
Metagenomic	0.02	10	74.78	0.096	73.007	0.11
Metagenomic	0.06	0	29.15	0.13	28.15	0.14
Metagenomic	0.06	5	55.31	0.11	53.56	0.14
Metagenomic	0.06	10	76.38	0.08	71.805	0.10
Metagenomic	0.2	0	31.30	0.15	28.15	0.14
Metagenomic	0.2	5	56.02	0.13	53.56	0.14
Metagenomic	0.2	10	74.13	0.09	71.81	0.10

Table S4. *Type I and type II error for simulation experiments on 50-taxon genomic datasets.* Average and standard error are reported ($n = 500$).

Mutation rate	# ret	Type I error Avg	Type I error SE	Type II error Avg	Type II error SE
0.02	0	14.1	0.31	15.07	0.31
0.02	5	28.2	0.15	44.17	0.15
0.02	10	37.09	0.11	68.07	0.11
0.06	0	13.962	0.18	14.94	0.18
0.06	5	27.73	0.18	43.70	0.18
0.06	10	36.04	0.13	67.02	0.13
0.2	0	13.21	0.29	14.18	0.29
0.2	5	28.09	0.19	44.06	0.19
0.2	10	35.83	0.12	66.81	0.12

Table S5. *Type I and type II error for simulation experiments on 50-taxon metagenomic datasets.* Average and standard error are reported ($n = 500$).

Mutation rate	# ret	Type I error Avg	Type I error SE	Type II error Avg	Type II error SE
0.02	0	16.72	0.25	17.7	0.25
0.02	5	29.51	0.211	45.49	0.21
0.02	10	37.39	0.12	68.374	0.12
0.06	0	15.71	0.29	16.68	0.29
0.06	5	29.11	0.18	45.09	0.18
0.06	10	37.36	0.10	68.33	0.11
0.2	0	15.27	0.26	16.25	0.26
0.2	5	29.45	0.20	45.43	0.20
0.2	10	36.24	0.12	67.22	0.12

We also performed additional 50-taxon simulation experiments with more reticulations. Figures S4, S5, and S6 show results for these experiments. We can see that, as the number of reticulations increased, phylogenetic placement error increases rapidly and approaches error saturation on

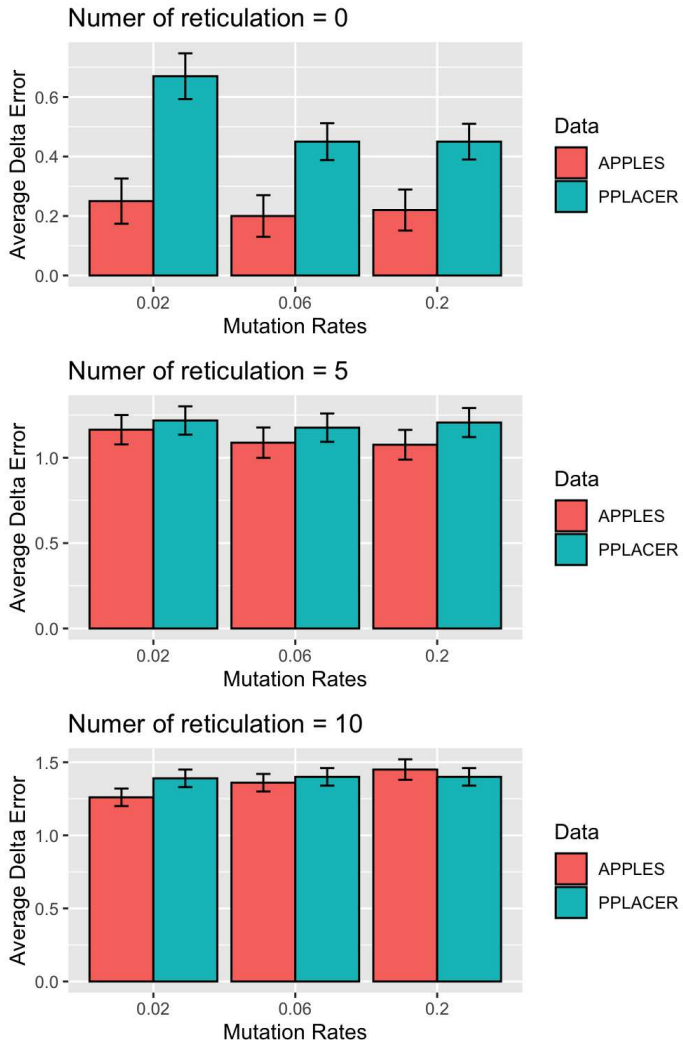

Fig. S2. *Comparison of APPLES and PPLACER on simulated genomic datasets.* Simulated datasets had 50 taxa. The simulations utilized mutation rates of 0.02, 0.06, and 0.2 and included 0, 5 or 10 reticulations. Phylogenetic placement error was assessed using delta error. Averages and standard error bars are shown ($n = 500$).

Table S6. *Genomic data simulations: p-values for statistical tests of placement tree error.* The simulations utilized a mutation rate of 0.2.

Mutation rate	# reticulations	$Avg\ Placement\ Error_{query}$	$Avg\ Placement\ Error_{baselinequery}$	P-value
0.2	0	13.19	17.52	5.3312E-133
0.2	5	36.06	37.86	2.55177E-81
0.2	10	50.84	51.71	4.9398E-53

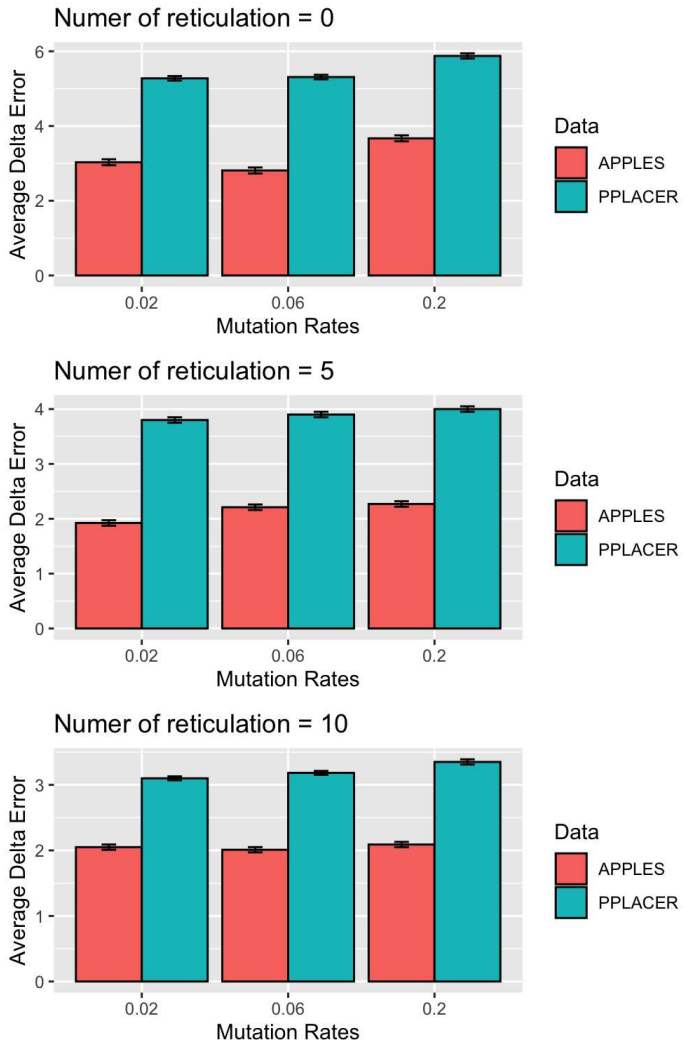

Fig. S3. *Comparison of APPLES and PPLACER on simulated metagenomic datasets.* Averages and standard error bars are shown ($n = 1500$). Figure description and layout are otherwise identical to Supplementary Figure S2.

Table S7. *Metagenomic data simulations: p-values for statistical tests of placement tree error.* The simulations utilized a mutation rate of 0.2.

Mutation rate	# reticulations	$Avg\ Placement\ Error_{query}$	$Avg\ Placement\ Error_{baselinequery}$	P-value
0.2	0	15.31	16.54	2.93672E-13
0.2	5	37.44	37.97	4.44321E-10
0.2	10	51.2	51.43	0.000646284

simulations with a higher number of reticulations. As in the other simulation study experiments, phylogenetic placement error on metagenomic datasets are typically greater than those observed on genomic datasets.

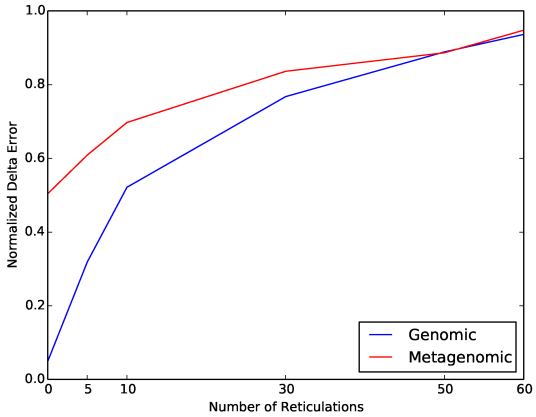

Fig. S4. *Simulation study: phylogenetic placement error on 50-taxon simulations with mutation rate 0.02 and varying numbers of reticulations.* Results are reported for both genomic and metagenomic data simulations with either 0, 5, 10, 30, 50 or 60 reticulations. Normalized delta errors are reported.

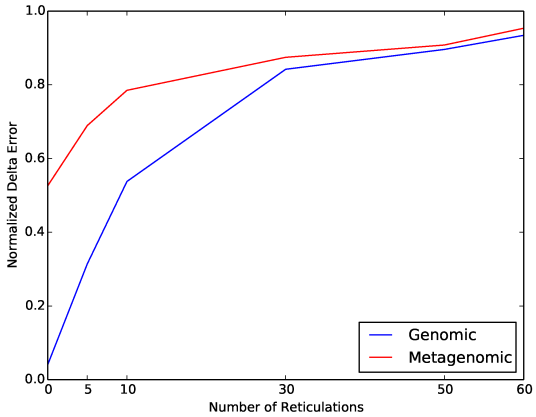

Fig. S5. *Simulation study: phylogenetic placement error on 50-taxon simulations with mutation rate 0.06 and varying numbers of reticulations.* Figure layout and description are otherwise identical to Supplementary Figure S4.

Table S8 includes results for an additional simulation experiment involving the NetPlacer algorithm. The experiment utilizes a mutation rate of 0.02, which is lower than that used in the simulation experiments reported in the main manuscript. The resulting placement errors are slightly lower than the latter, which we attribute to lower evolutionary divergence, and the experiment outcomes generally follow the trends observed in the main manuscript.

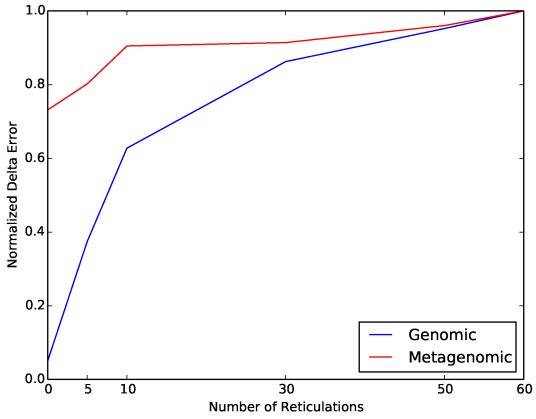

Fig. S6. *Simulation study: phylogenetic placement error on 50-taxon simulations with mutation rate 0.2 and varying numbers of reticulations.* Figure layout and description are otherwise identical to Supplementary Figure S4.

Table S8. *Additional NetPlacer experiments.* The 8-taxon simulated datasets had 50 sampled gene trees per dataset, and were generated with mutation rate 0.02.

Mutation rate	# reticulations	# loci per gene	Normalized Delta Error
0.02	0	1000	0.1972
0.02	1	1000	0.4238

REFERENCES

[1] Robert Edgar. Usearch. Technical report, Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States), 2010.

[2] Richard R Hudson. ms a program for generating samples under neutral models. *Bioinformatics*, 18(2):337–338, 2002.

[3] Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer, and Loren J Hauser. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics*, 11(1):1–11, 2010.

[4] Kazutaka Katoh and Daron M Standley. Mafft multiple sequence alignment software version 7: improvements in performance and usability. *Molecular Biology and Evolution*, 30(4):772–780, 2013.

[5] Frederick A Matsen, Robin B Kodner, and E Armbrust. pplacer: linear time maximum-likelihood and bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics*, 11(1):1–16, 2010.

[6] Luay Nakhleh. A metric on the space of reduced phylogenetic networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(2):218–222, 2009.

[7] Michael J Sanderson. r8s: inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock. *Bioinformatics*, 19(2):301–302, 2003.

[8] Alexandros Stamatakis. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

[9] Cuong Than, Derek Ruths, and Luay Nakhleh. Phylonet: a software package for analyzing and reconstructing reticulate evolutionary relationships. *BMC Bioinformatics*, 9:1–16, 2008.