FastMulRFS: Statistically consistent polynomial time species tree estimation under gene duplication*

Erin K. Molloy[1[0000−0001−5553−3312]] and Tandy Warnow[1[0000−0001−7717−3514]]

University of Illinois at Urbana-Champaign, Urbana IL, USA
{emolloy2,warnow}@illinois.edu

Abstract. Species tree estimation is a basic part of much biological research, but can be challenging because of gene duplication and loss (GDL), which results in genes that can appear more than once in a given genome. To construct species trees from gene families, biologists typically pick one copy of each gene in each species (based on estimations of orthology), but orthology detection is not yet reliably accurate, and incorrect orthology inferences can result in incorrect species trees. Restricting datasets to single-copy genes is another common solution, but this reduces the amount of available data, which is undesirable (and even genes that appear to be single-copy may have evolved with duplication and loss). Thus, all common approaches in phylogenomics reduce available data and are error-prone, and methods that do not discard data and have high accuracy on large heterogeneous datasets are needed. Here, we present FastMulRFS, a method for estimating species trees from multrees. We prove that FastMulRFS is polynomial time and statistically consistent under a generic model of gene duplication and loss provided that only duplications occur or only losses occur. We present the results of an extensive simulation study where gene tree heterogeneity is due to gene duplication and loss, incomplete lineage sorting,and gene tree estimation error, and show that FastMulRFS matches the accuracy of MulRF (which tries to solve the same optimization problem) and has better accuracy than ASTRAL-multi (which is statistically consistent under GDL), while being much faster than both methods.

Keywords: species tree · gene duplication and loss · statistical consistency · MulRF · FastRFS.

* Supported by the University of Illinois at Urbana-Champaign

1 Introduction

The inference of species trees from multiple loci is a critical component of many biological projects, with multiple applications including biodiversity, adapation of species to environments, etc. [12]. Most species tree estimation methods are designed for orthologous genes (i.e., genes related through speciation events only and not through duplication events) [16]. Because orthology prediction is quite challenging [37,1,22], and mistakes in orthology prediction can result in incorrect species trees, multi-copy genes are typically excluded from species tree estimation [41]. Methods that can infer species trees from gene families are of increasing interest, as this would enable phylogenetic signal to be extracted from multi-copy genes while avoiding the challenges of orthology prediction.

Several different methods have been proposed to infer species trees from multi-copy genes. Some methods promise outstanding accuracy, but are computationally too intensive to use on anything beyond fairly small datasets (e.g., perhaps 20 or so species); PHYLDOG, a Bayesian method that co-estimates gene trees and species trees is perhaps the most well known method of this sort [4]. Gene tree parsimony (GTP) methods operate by first computing gene family trees (which may contain multiple copies of some species), and then seek a species tree that implies the minimum number of gene duplication and loss events. Examples of GTP methods include iGTP [8], DupTree [40], and DynaDup [3], and several phylogenomic studies have used these methods [33,6]. Since gene tree parsimony is NP-hard, most of these methods operate by using hill-climbing to find good solutions to their criteria. DynaDup, in contrast, uses dynamic programming to find an optimal solution within a constrained search space (this type of approach was, to the best of our knowledge, first proposed in [21] and has since been utilized for other optimization problems including the maximum quartet support supertree (MQSS) problem [5,27] and the Robinson-Foulds supetree (RFS) problem [39]). Although GTP methods can be computationally intensive, they are still more scalable than PHYLDOG and similar methods.

Other fast approaches include supertree methods that have been adapted to work with gene family trees, called "mul-trees" in brief, as they can have multiple copies of each species. The supertree method that is most well known for mul-trees is MulRF [11], which attempts to find an optimal solution to the NP-hard Robinson-Foulds Supertree problem for mul-trees (RFS-multree). Although MulRF does not explicitly account for gene duplication and losses, it has been shown to produce more accurate species trees than DupTree and iGTP on datasets simulated under challenging model conditions, including gene duplication and loss (GDL), incomplete lineage sorting (ILS), horizontal gene transfer, and gene tree estimation error (GTEE) [9].

In a recent study, Legried *et al.* [24] showed that ASTRAL-multi [28], a recent extension of ASTRAL [27] (a method that was developed for the problem of species tree estimation in the presence of ILS) is statistically consistent under a standard stochastic model of GDL. They also compared ASTRAL-multi to MulRF and DupTree, and found that it was more accurate than DupTree but

not quite as accurate as MulRF, given estimated gene trees where true gene tree heterogeneity was due to GDL.

Here, we examine the theoretical properties of the RFS-multree problem, which MulRF tries to solve using hill-climbing. We show that an exact solution to the RFS-multree problem is provably statistically consistent under GDL when there are only gene duplications or only gene losses. We also present FastMulRFS, a polynomial time algorithm that uses dynamic programming to solve the RFS-multree problem exactly within a constrained search space, and prove that FastMulRFS is statistically consistent under the same conditions (i.e., duplication-only and loss-only scenarios). We present the results of an extensive experimental study evaluating FastMulRFS in comparison to ASTRAL-multi, MulRF, and DupTree, and show that FastMulRFS and MulRF are tied for being the most accurate, but that FastMulRFS is much faster than MulRF. Our study allows for multiple sources of gene tree heterogeneity, including GDL, ILS, and GTEE, demonstrating that FastMulRFS has substantial robustness. Thus, FastMulRFS is a new and very fast method for species tree estimation that does not require reliable orthology detection, outperforms the leading alternative methods (even under conditions with FastMulRFS is not guaranteed to be statistically consistent), and can be used on large datasets.

2 The RFS-multree problem and FastMulRFS

We define the Robinson-Foulds Supertree problem for mul-trees (RFS-multree) and the FastMulRFS algorithm that solves the problem exactly within a constrained search space. Later, we will prove that FastMulRFS is statistically consistent under a generic model of gene duplication or gene loss.

We begin with some definitions and terminology. A *phylogenetic tree* T is an unrooted tree with leaves labeled by a set S. Let $L(T)$ denote the leaf set of T, and let $\phi : L(T) \to S$ denote the mapping from leaves to labels. Then, T is defined by the triplet (t, ϕ, S) where t is its unrooted tree topology. If each leaf in T is assigned a unique label, then we say that T is *singly-labeled*. If two or more leaves in T are assigned the same label, then we say that T is *multi-labeled* (equivalently, T is a *mul-tree*). Deleting an edge e but not its endpoints from T produces two subtrees T_A and T_B that define two label sets: $A = \{\phi(l) : l \in L(T_A)\}$ and $B = \{\phi(l) : l \in L(T_B)\}$. If no label appears on both sides of e, then A and B are disjoint sets, and the edge e induces a bipartition on the label set of T (i.e., it splits the leaf labels into two disjoint sets). However, if some label appears on both sides of e then A and B are not disjoint and so by definition the edge e does *not* induce a bipartition.

A key distinction between singly-labeled trees and mul-trees is that every edge in a singly-labeled tree induces a bipartition but edges in a mul-tree can fail to induce bipartitions because some leaf label may appear on both sides due to gene duplication. The Robinson-Foulds (RF) distance [30] between two singly-labeled trees has a simple definition as the bipartition distance (i.e., number of bipartitions in one but not in both trees), so that the RF distance between T

and T', both singly-labeled trees on the same set of leaves, is

$$RF(T,T') = |C(T) \triangle C(T')| = |C(T) \setminus C(T')| + |C(T') \setminus C(T)|,$$

where $C(t)$ denotes the set of bipartitions of S defined by the edges of t. However, when one or both trees is a mul-tree, then the RF distance is the edit distance under contraction-and-refinement operations (see Appendix A). Furthermore, when the pair of trees are mul-trees, computing the RF distance is NP-complete [10]. However, Chaudhary *et al.* [10] proved that the RF distance between a mul-tree and a singly-labeled tree can be computed in polynomial time as follows: (1) extend T with respect to M, denoted $Ext(T,M)$ (see Fig. 1), (2) relabel the leaves of M and $Ext(T,M)$ in a "mutually consistent fashion" so that both trees are now singly-labeled, and (3) compute the RF distance between the relabeled versions of M and $Ext(T,M)$ using bipartition distances (see Appendix A for a formal description of this algorithm). Chaudhary *et al.* [10] then proposed the Robinson-Foulds Supertree for mul-trees (RFS-multree) problem:

Definition 1. *The input is a set $\mathcal{P}$ of mul-trees with leaves labeled by elements of the set S, and the output is a binary (i.e., fully resolved) tree T bijectively labeled by S that minimizes*

$$\sum_{M \in \mathcal{P}} RF(Ext(T,M), M). \tag{1}$$

Any tree that minimizes this score is called an **RFS-multree supertree** *for $\mathcal{P}$.*

We simplify the RFS-multree problem by providing an alternative proof that the RF distance between between a mul-tree M and a singly-labeled tree T can be computed in polynomial time (Lemma 4 in Appendix A). We summarize the intuition behind Lemma 4 in Figure 1, which leads easily to Theorem 1.

Definition 2. *Given a mul-tree $M \in \mathcal{P}$, we first collapse edges that do not induce bipartitions (because some label appears on both sides of the edge), and denote this tree by $\mathcal{X}(M)$. We then delete all but one leaf with each label, and denote the result by $M_X = \mathcal{R}(\mathcal{X}(M))$. We define $\mathcal{P}_X := \{M_X : M \in \mathcal{P}\}$.*

Theorem 1. *Let T be a singly-labeled, binary tree, and let $\mathcal{P}$ be a set of mul-trees. Then, T is an RFS-multree supertree for $\mathcal{P}$ if and only if T is a Robinson-Foulds (RF) supertree for $\mathcal{P}_X$. Equivalently, T is an RFS-multree supertree for $\mathcal{P}$ if and only if T is a binary tree that maximizes $\sum_{t \in \mathcal{P}} |C(T) \cap C(t)|$, where we define $C(t)$ to be the set of bipartitions induced by edges in t. (Note that when t is a mul-tree, some edges will not induce bipartitions.)*

A consequence of Theorem 1 is that any heuristic for the RFS problem can be used for the RFS-multree problem simply by transforming the input mul-trees (i.e., computing $\mathcal{P}_X$) and then running the RFS heuristic on $\mathcal{P}_X$. In this study, we explore the impact of using FastRFS [39] in this two-phase approach,

which we refer to as **FastMulRFS**. The input to FastRFS is a profile of singly-labeled trees $\mathcal{T}$, each on a (possibly proper) subset of S and a set Σ of allowed bipartitions on S, and FastRFS provably returns a (binary) supertree T that minimizes the total RF distance to the trees in $\mathcal{T}$ subject to $C(T) \subseteq \Sigma$. FastRFS uses dynamic programming to solve the constrained search problem (a technique that was originally proposed for tree estimation in [21]) in $O(|\Sigma|^2 nk)$ time, where $n = |S|$ and $k = |\mathcal{T}|$. Hence, the size of Σ impacts the running time for FastRFS. Furthermore, how Σ is defined affects the accuracy of FastRFS, and hence also of FastMulRFS. However, we will show that we can define Σ from the input so that FastMulRFS is polynomial time and statistically consistent under GDL, provided that the evolution is duplication-only or loss-only (see Appendix B for details on the running time).

3 Species Tree Estimation using FastMulRFS

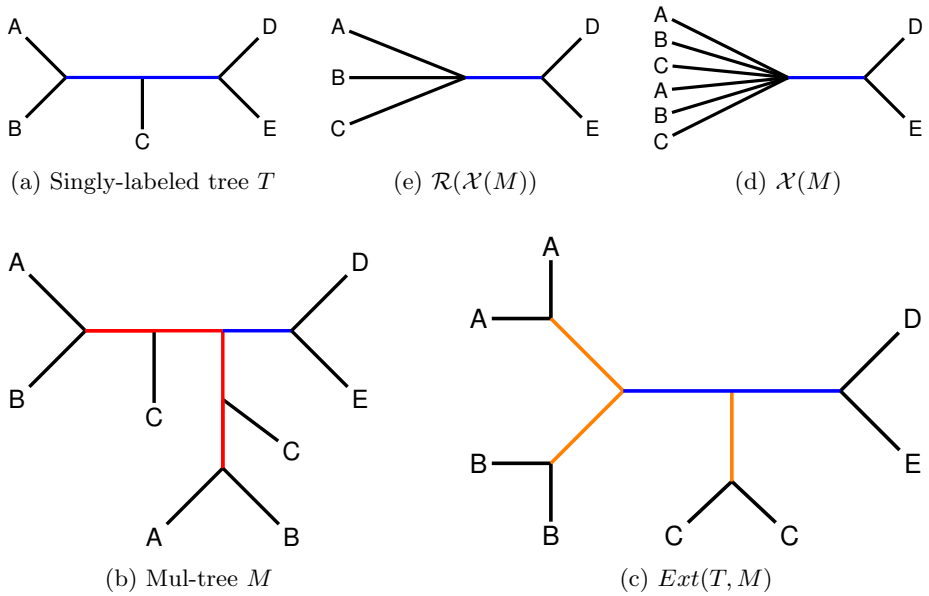
We now show that the optimal RFS-multree supertree is a statistically consistent estimate of the species tree under a generic duplication-only model and a generic loss-only model of gene evolution. We will also show that FastMulRFS is a statistically consistent species tree method under these models that runs in polynomial time. We conclude by discussing the impact of allowing both gene duplication and gene loss events on FastMulRFS.

Generic duplication-only and loss-only models: The only requirement that we have for our generic duplication-only model is that the probability of duplicating on an edge is strictly less than 1 for every gene, but the probabilities of duplication can depend on the gene and on the edge. Similarly, the only requirement that we have for our loss-only model is that the probability of gene loss is strictly less than 1 for every combination of gene and edge, but these probabilities do not need to be the same across genes or edges. Thus, our generic models are similar to the *no common mechanism* model of sequence evolution proposed in [38], and more general than the stochastic GDL models in [2,29]. Note that under this model, there are no other sources of discord between the gene tree and species tree (e.g., there is no ILS or HGT). We also assume that there is no gene tree estimation error or missing data, which are common assumptions in proofs of statistical consistency for species tree estimation methods that operate by combining gene trees [25,27,13,31,34].

The following lemma is key to understanding the RFS-multree problem:

Lemma 1. *Let T^* be the (unrooted) true species tree, and let $\mathcal{P}$ be the set of (unrooted) true gene trees that evolved within the rooted species tree under a stochastic duplication-only model of gene evolution. Then every bipartition in every M_X (see Definition 2) is a bipartition in the T^*.*

Proof. Let M be a gene family tree, and let $e \in E(M)$. We will show that e is collapsed in producing $\mathcal{X}(M)$ if and only if e lies below at least one duplication node in the rooted version of M. Hence, the (unrooted) singly-labeled tree $M_X = \mathcal{R}(\mathcal{X}(M))$ will only retain the edges in M that have no duplication nodes above them in M. To see why, consider any edge e that has no duplication node above



(a) Singly-labeled tree T (e) $\mathcal{R}(\mathcal{X}(M))$ (d) $\mathcal{X}(M)$

(b) Mul-tree M (c) $Ext(T, M)$

Fig. 1: **Reduction of the RFS-multree problem to the RFS problem.** The subfigures (a) through (e) describe the basic algorithmic approach for computing the RF distance between a singly-labeled tree and a mul-tree in the context of the RFS-multree problem. Suppose that T (shown in (a)) is a candidate singly-labeled, binary supertree for a set $\mathcal{P}$ of mul-trees and that M (shown in (b)) is one of the mul-trees in $\mathcal{P}$. In order to compute the RF distance between T and M, we extend T with respect to M, producing $Ext(T, M)$, shown in (c). Note that $Ext(T, M)$ has the same non-trivial edges (shown in blue) and the same trivial edges (shown in orange) as T, and for every leaf label (i.e., species), it has the same number of leaves with that label as mul-tree M. These trivial edges in $Ext(T, M)$ exist in *any possible* singly-labeled, binary tree on S; thus, these edges do not impact the solution to the RFS-multree problem. Similarly, mul-tree M has edges (shown in red) that will be incompatible with an extended version of *any possible* singly-labeled, binary tree on S; thus, these edges also do not impact the solution to the RFS-multree problem. Notably, an edge is incompatible with every possible singly-labeled supertree if and only if it fails to induce a bipartition (i.e., deleting an edge e splits the leaf set into two non-disjoint label sets). Thus, we collapse all edges in M that fail to induce a bipartition, producing $\mathcal{X}(M)$, as shown in subfigure (d). Furthermore, because all leaves with the same label are now on the same side of *every* bipartition in $\mathcal{X}(M)$, we can delete all but one leaf with each label, producing $\mathcal{R}(\mathcal{X}(M))$, as shown in subfigure (e). The resulting tree is a non-binary, singly-labeled tree on label set S, so we can compute the RF distance between $\mathcal{R}(\mathcal{X}(M))$ and T when searching for the solution to the RFS-multree problem. These observations are formally stated and then proven in Lemma 4 (Appendix A). It follows directly from Lemma 4 that the optimal RFS-multree supertree for $\mathcal{P}$ is the optimal Robinson-Foulds supertree for $\mathcal{P}_X = \{\mathcal{R}(\mathcal{X}(M)) : M \in \mathcal{P}\}$, as summarized in Theorem 1. A consequence of this result is that any heuristic for the RF Supertree problem can be used as a heuristic for the RFS-multree problem, simply by transforming the input mul-trees. In Appendix B, we provide an algorithm for computing the set $\mathcal{P}_X$ from $\mathcal{P}$ in $O(mnk)$ time, where $|\mathcal{P}| = k$, n is the number of leaf labels (i.e., species), and m is the largest number of leaves in any mul-tree in $\mathcal{P}$.

it in the rooted gene family tree: no species appears on both sides of e and hence e will not be collapsed. Conversely, if e is collapsed, then there must be at least one species on both sides of e, and so e must be below at least one duplication node in the true rooted gene family tree. Finally, consider a bipartition defined by an edge that is not collapsed, and hence has no duplication nodes above it. This bipartition appears in the true species tree, since the only events that cause the gene family tree to differ from the species tree are duplications. □

A bipartition π on label set $R \subseteq S$ is said to be *compatible* with the true species tree T^* if there exists $\pi' \in C(T)$ such that π' is identical to π when restricted to R.

Lemma 2. *Let $\mathcal{P}$ be a set of true gene trees that evolved within the rooted species tree under a stochastic loss-only model of gene evolution. Then every bipartition in any M_X with $M \in \mathcal{P}$ is* compatible *with the true species tree.*

Proof. When gene loss is the only source of discord between the gene family tree and the species tree, each gene family tree is identical to the species tree when restricted to the same leaf set (i.e., deleting all leaves that are not in M and suppressing internal nodes of degree 2). □

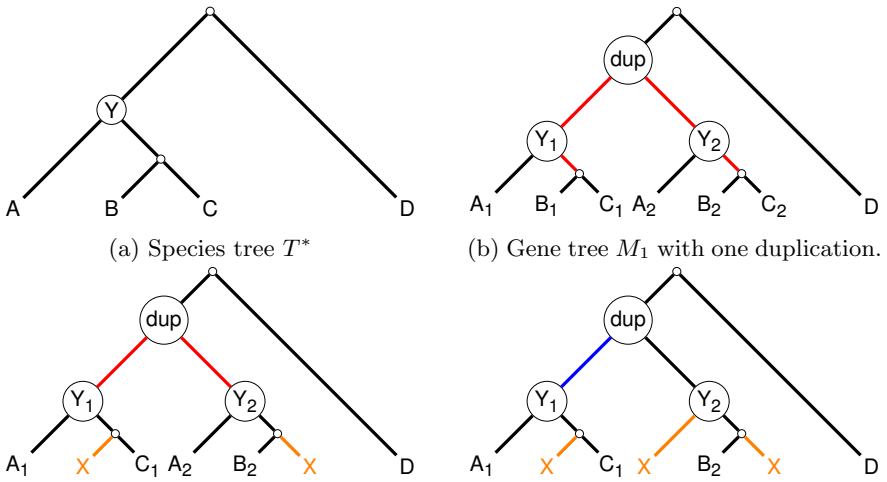
Lemma 3. *The true species tree T^* is an RF Supertree for any set $\mathcal{P}_X$ of singly-labeled trees derived from $\mathcal{P}$, for the duplication-only and loss-only models.*

Proof. It is easy to see that a fully resolved tree minimizes the RF distance to $\mathcal{P}_X$ if and only if it maximizes the number of shared bipartitions. By Lemma 1 (or Lemma 2), every bipartition in any tree in $\mathcal{P}_X$ appears in the true species tree, and so the true species tree optimizes the number of shared bipartitions. It follows that T^* is an optimal RF Supertree for $\mathcal{P}_X$. □

Recall that FastMulRFS solves the RFS-multree supertree problem exactly within a constrained search space defined by a set Σ of allowed bipartitions, which it computes from the input. The *default* technique of how Σ is constructed uses any bipartition that appears in any M_X (i.e., $\Sigma = C(\mathcal{P}_X)$). Note that $|C(\mathcal{P}_X)| \leq (n-3)k$, where $n = |S|$ and $k = |\mathcal{P}|$. Hence, under this default setting, FastMulRFS is polynomial time. We now show that this default technique for defining Σ suffices for statistical consistency.

Theorem 2. *Under a generic duplication-only or loss-only model, the true species tree is the unique RFS-multree supertree for $\mathcal{P}$ with probability going to 1, as the number of gene trees in $\mathcal{P}$ goes to infinity; and default FastMulRFS is a statistically consistent method for estimating species trees under these generic models.*

Proof. T^* is an RF supertree for $\mathcal{P}_X$ (Lemma 3), and an RF supertree for $\mathcal{P}_X$ is an RFS-multree supertree for $\mathcal{P}$ (Theorem 1). Under a stochastic duplication-only (or loss-only) model of gene evolution, every bipartition in T^* has positive probability of appearing in some gene family tree (e.g., since both duplication and loss have probability strictly less than 1, T^* has strictly positive probability



(a) Species tree T^*

(b) Gene tree M_1 with one duplication.

(c) Gene tree M_2 with one duplication and two losses.

(d) Gene tree M_3 with one duplication and three losses.

Fig. 2: **Impact of gene duplications and losses on species tree estimation using RFS-multree methods.** Subfigure (a) shows a species tree T^* and subfigures (b) through (d) show three gene family trees that evolved within the species tree. Subfigure (b) shows gene family tree M_1 with a duplication event in species Y (i.e., the most recent common ancestor of species A, B, and C). Note that all edges in M_1 below the duplication node (shown in red) fail to induce bipartitions and so will be contracted, and will therefore not impact the solution space for the RFS-multree criterion. Subfigure (c) shows gene tree M_2 with a duplication event in species Y followed by the first copy of the gene being lost from species B and the second copy of the gene being lost from species C. Because one of the species that evolved from Y retains both copies of the gene, the non-trivial edges in M_2 below the duplication node fail to induce bipartitions, and so these edges also do not impact the solution space for RFS-multree. Subfigure (d) shows gene family tree M_3 with a duplication event in species Y followed by the first copy of the gene being lost from species B and the second copy of the gene being lost from both species A and C. None of the species that evolved from Y retain both copies of the gene, so all edges below the duplication node induce bipartitions *and hence will not be contracted*; we refer to this situation as "adversarial gene duplication and loss," because it produces bipartitions in the singly-labeled trees in $\mathcal{P}_X$ that conflict with the species tree (shown in blue). Such a scenario leads to the possibility that the true species tree may not be an optimal solution to the RFS-multree problem.

of appearing as a gene family tree). Then, by Lemma 1 (or Lemma 2), as the number of genes goes to infinity with probability going to 1, the set of bipartitions $C(\mathcal{P}_X)$ will equal $C(T^*)$, and thus T^* will be the unique RF supertree for $\mathcal{P}_X$. Furthermore, as the number of genes increases, Σ (as constructed by the default setting within FastMulRFS) will converge to $C(T^*)$ with probability converging to 1, and so FastMulRFS will return T^* with probability going to 1. □

Adversarial gene duplication and loss: We now show a (probably rare) condition where evolution with both gene duplication and loss can challenge RFS-multree methods. Consider a mul-tree M for a gene duplicated in an ancestral species Y that is the least common ancestor for a set Z of extant (i.e., present-day) species; for example, $Z = \{A, B, C\}$ in Figure 2. If even one species in Z retains both copies of the duplicated gene, then every edge below the duplication node will contracted, as described in Lemma 1 (Fig. 2c). Otherwise, the true gene family tree may have bipartitions that do not appear in the true species tree (e.g., Fig. 2d). We refer to this scenario as "adversarial gene duplication and loss." Although adversarial GDL will create conditions that make true gene family trees have bipartitions that do not appear in the species tree, these events are likely to be relatively rare, and may not substantially impact the accuracy of FastMulRFS. On the other hand, this scenario relates more broadly to challenges with orthology detection, specifically the case where a gene appears to be single-copy for a particular set of species but is actually multi-copy.

4 Experimental Study

We evaluated ASTRAL-multi, DupTree, FastMulRFS, MulRF, and STAG [17] on simulated datasets with 100 species, using GDL and ILS levels based on a 16-species fungal dataset [29][1]. We computed trees on the fungal dataset (see Appendix E). FastMulRFS produced the same tree as most other species tree methods, differing in small ways from the tree returned by ASTRAL-multi; other phylogenies reported for this dataset show similar variability [7].

We generated a collection of 100-species simulated datasets, where the easiest model condition was based on GDL and ILS rates estimated for the 16 fungi species (see [29,14]), we then increased the GDL and ILS rates to make more challenging model conditions. Overall, we examined six basic model conditions (each with ten replicate datasets) with three rates of gene duplication/loss and two levels of ILS. We evaluated methods given both true and estimated gene family trees, using RAxML [35] under the GTRGAMMA model to estimate trees on the true alignments, with sequence lengths varied to produce four different levels of GTEE. All simulated datasets had 1000 genes, and we estimated species trees on the first 25, 50, 100, and 500 gene family trees. This created $2 \times 3 \times 5 \times 4 = 120$ model conditions, each with 10 replicate datasets, for a total of

[1] FastMulRFS and all datasets necessary to reproduce the study have been deposited to the Illinois Data Bank https://doi.org/10.13012/B2IDB-5721322_V1 (link will be active on November 11, 2019).

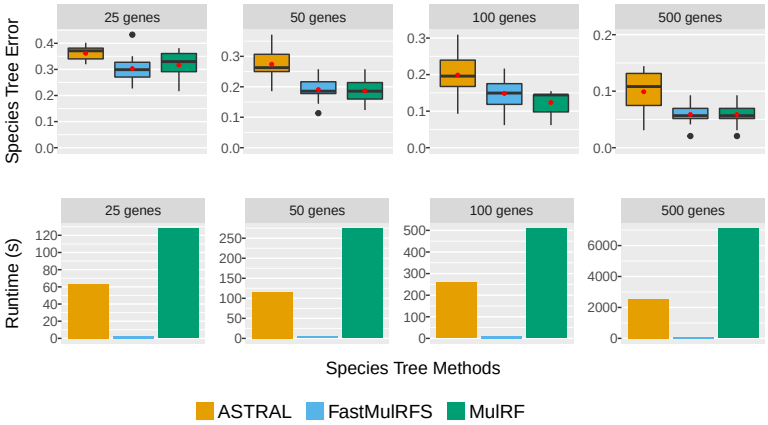


Fig. 3: Species tree error (i.e., normalized RF distance) and running time (seconds) are shown for FastMulRFS, MulRF, and ASTRAL-multi under the hardest model conditions with 100 species. All datasets have moderate GTEE (52%), high GDL (D/L rate: 5×10^7), and low/moderate ILS (12%).. Red dots (first row) and bars (second row) are means for 10 replicate datasets.

1200 datasets. STAG failed to run on 830/1200 datasets, and DupTree had the worst accuracy of all tested methods. Hence, we focus on comparing MulRF, FastMulRFS, and ASTRAL-multi. The fastest method was FastMulRFS, MulRF was the slowest, and ASTRAL-multi was intermediate. We present results for MulRF, FastMulRFS, and ASTRAL-multi under the most challenging model conditions (i.e., the highest levels of ILS and GDL) in Figure 3; results on all datasets and details about the simulation study are in Appendices C and D.

FastMulRFS vs. MulRF: Both methods try to solve RFS-multree but use different approaches; they were essentially tied for topological accuracy across all tested conditions, but FastMulRFS was dramatically faster (Tables 3 and 4). In addition, FastMulRFS nearly always returned trees with better RFS-multree scores than MulRF (Appendix D).

FastMuLRFS vs. ASTRAL-multi: FastMulRFS had better accuracy than ASTRAL-multi across all model conditions tested, but the biggest improvement was under the harder model conditions with high GTEE (Table 3). The running times for ASTRAL-multi and FastMulRFS increased with the number of genes, but FastMulRFS was always much faster. For example, FastMulRFS never exceeded 23 seconds for any model condition with up to 100 genes, but ASTRAL-multi took more than 7 minutes on some such conditions. Furthermore, on the 500-gene model conditions, FastMulRFS typically completed in 1–2 minutes (and always in under 5 minutes), but ASTRAL-multi used between 10 minutes and 1.2 hours (Table 4).

5 Conclusions

There are three main contributions of this paper. First, we showed a simple property about the Robinson-Foulds Supertree problem for mul-trees (RFS-multree) that establishes that an exact solution to the RFS-multree is a statistically consistent method for estimating species trees under generic duplication-only and loss-only gene family evolution models. Second, we presented FastMulRFS, a polynomial time algorithm to find an exact solution to the RFS-multree problem within a constrained search space, and we proved that the default version is statistically consistent under generic duplication-only or loss-only models. Thus, FastMulRFS is the *second* of *only two* methods proven to be statistically consistent under scenarios with gene duplication and/or loss (ASTRAL-multi, which was proven consistent under a parametric model of GDL in [24], is the other). Third, we showed that FastMulRFS maintains high accuracy even under conditions where both duplication and loss occur, where moderate incomplete lineage sorting (ILS) is present, where there is substantial gene tree estimation error (GTEE), and for 25 to 500 genes. Under all tested conditions, FastMulRFS was more accurate and much faster than ASTRAL-multi.

This study suggests several directions for future work. For example, although FastMulRFS was consistently at least as accurate as ASTRAL-multi, we did not explore a large range of model conditions. In particular, the simulation conditions we explored did not have very large numbers of copies of species in the gene family trees (e.g., on average about 1 copy per species per gene, and a maximum of just under 10). These simulation conditions were based on a yeast dataset, and so are likely to be typical of some types of biological data, but not all. Hence, our study did not evaluate conditions where there are much larger numbers of duplication events, and so the relative accuracy of ASTRAL-multi, FastMulRFS, and DupTree under other conditions cannot be predicted with high confidence. Future studies should evaluate these methods under a much wider range of model conditions, to enable biologists to select methods with the best expected accuracy for their particular data.

Another direction for future work is to evaluate the statistical consistency of FastMulRFS under parametric GDL models, where both gene duplication and gene loss can occur. Our proofs establish consistency under generic models of gene family evolution with duplication-only and loss-only conditions, but leave open the possibility of statistical consistency under other conditions. The very good accuracy of FastMulRFS under parametric GDL models suggests the possibility that it may be statistically consistent, but this needs to be established with a proof. Similarly, DupTree and more generally gene tree parsimony approaches should be evaluated for statistical consistency under GDL models. Even though DupTree was not as accurate as ASTRAL-multi or FastMulRFS, it may still be statistically consistent under parametric GDL models, and this should be evaluated.

In summary, the recent advances in development of statistically consistent methods for species tree estimation under GDL models is exciting, and the good performance of these methods under a range of model conditions suggests that

novel combinations and ideas may lead to even better methods that provide improved accuracy and scalability.

6 Funding

This study was supported in part by NSF grants CCF-1535977 and 1513629 (to TW) and by the Ira and Debra Cohen Graduate Fellowship in Computer Science (to EKM). This study was performed on the Illinois Campus Cluster and the Blue Waters supercomputer, computing resources that are operated and financially supported by UIUC in conjunction with the National Center for Supercomputing Applications. Blue Waters is supported by the NSF (grants OCI-0725070 and ACI-1238993) and the state of Illinois.

References

1. Altenhoff, A.M., Glover, N.M., Dessimoz, C.: Inferring orthology and paralogy. In: Evolutionary Genomics, pp. 149–175. Springer (2019)
2. Arvestad, L., Lagergren, J., Sennblad, B.: The Gene Evolution Model and Computing Its Associated Probabilities. Journal of the ACM **56**(2), 7:1–7:44 (2009). https://doi.org/10.1145/1502793.1502796
3. Bayzid, M.S., Warnow, T.: Gene tree parsimony for incomplete gene trees: addressing true biological loss. Algorithms for Molecular Biology **13**(1), 1 (2018). https://doi.org/10.1186/s13015-017-0120-1
4. Boussau, B., Szöllősi, G.J., Duret, L., Gouy, M., Tannier, E., Daubin, V.: Genome-scale coestimation of species and gene trees. Genome Research **23**(2), 323–330 (2013). https://doi.org/10.1101/gr.141978.112
5. Bryant, D., Steel, M.: Constructing Optimal Trees from Quartets. Journal of Algorithms **38**(1), 237–259 (2001). https://doi.org/10.1006/jagm.2000.1133
6. Burleigh, J.G., Bansal, M.S., Eulenstein, O., Hartmann, S., Wehe, A., Vision, T.J.: Genome-Scale Phylogenetics: Inferring the Plant Tree of Life from 18,896 Gene Trees. Systematic Biology **60**(2), 117–125 (12 2010). https://doi.org/10.1093/sysbio/syq072
7. Butler, G., Rasmussen, M.D., Lin, M.F., Santos, M.A.S., Sakthikumar, S., Munro, C.A., Rheinbay, E., Grabherr, M., Forche, A., Reedy, J.L., Agrafioti, I., Arnaud, M.B., Bates, S., Brown, A.J.P., Brunke, S., Costanzo, M.C., Fitzpatrick, D.A., de Groot, P.W.J., Harris, D., Hoyer, L.L., Hube, B., Klis, F.M., Kodira, C., Lennard, N., Logue, M.E., Martin, R., Neiman, A.M., Nikolaou, E., Quail, M.A., Quinn, J., Santos, M.C., Schmitzberger, F.F., Sherlock, G., Shah, P., Silverstein, K.A.T., Skrzypek, M.S., Soll, D., Staggs, R., Stansfield, I., Stumpf, M.P.H., Sudbery, P.E., Srikantha, T., Zeng, Q., Berman, J., Berriman, M., Heitman, J., Gow, N.A.R., Lorenz, M.C., Birren, B.W., Kellis, M., Cuomo, C.A.: Evolution of pathogenicity and sexual reproduction in eight candida genomes. Nature **459**(7247), 657–662 (2009). https://doi.org/10.1038/nature08064
8. Chaudhary, R., Bansal, M.S., Wehe, A., Fernández-Baca, D., Eulenstein, O.: iGTP: a software package for large-scale gene tree parsimony analysis. BMC Bioinformatics **11**(1), 574 (2010)

9. Chaudhary, R., Boussau, B., Burleigh, J.G., Fernández-Baca, D.: Assessing Approaches for Inferring Species Trees from Multi-Copy Genes. Systematic Biology **64**(2), 325–339 (2014). https://doi.org/10.1093/sysbio/syu128
10. Chaudhary, R., Burleigh, J.G., Fernández-Baca, D.: Inferring Species Trees from Incongruent Multi-Copy Gene Trees Using the Robinson-Foulds Distance. Algorithms for Molecular Biology **8**, 28 (2013). https://doi.org/10.1186/1748-7188-8-28
11. Chaudhary, R., Fernández-Baca, D., Burleigh, J.G.: MulRF: a software package for phylogenetic analysis using multi-copy gene trees. Bioinformatics **31**(3), 432–433 (2014). https://doi.org/10.1093/bioinformatics/btu648
12. Cracraft, J., Donoghue, M., Dragoo, J., Hillis, D., Yates, T. (eds.): Assembling the Tree of Life: Harnessing Life's History to Benefit Science and Society. National Science Foundation (2002), available at http://ucjeps.berkeley.edu/tol.pdf
13. Daskalakis, C., Roch, S.: Species trees from gene trees despite a high rate of lateral genetic transfer: A tight bound (extended abstract). In: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1621–1630 (2016). https://doi.org/10.1137/1.9781611974331.ch110
14. Du, P., Hahn, M.W., Nakhleh, L.: Species Tree Inference under the Multispecies Coalescent on Data with Paralogs is Accurate. bioRxiv (2019). https://doi.org/10.1101/498378
15. Edgar, R.C.: MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic acids research **32**(5), 1792–1797 (2004). https://doi.org/10.1093/nar/gkh340
16. Eisen, J.A.: Phylogenomics: improving functional predictions for uncharacterized genes by evolutionary analysis. Genome research **8**(3), 163–167 (1998)
17. Emms, D., Kelly, S.: STAG: Species Tree Inference from All Genes. bioRxiv (2018). https://doi.org/10.1101/267914
18. Fletcher, W., Yang, Z.: INDELible: A Flexible Simulator of Biological Sequence Evolution. Molecular Biology and Evolution **26**(8), 1879–1888 (2009). https://doi.org/10.1093/molbev/msp098
19. Ganapathy, G., Goodson, B., Jansen, R., Le, H.s., Ramachandran, V., Warnow, T.: Pattern Identification in Biogeography. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **3**(4), 334–346 (2006). https://doi.org/10.1109/TCBB.2006.57
20. Guindon, S., Dufayard, J.F., Lefort, V., Anisimova, M., Hordijk, W., Gascuel, O.: New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. Systematic biology **59**(3), 307–321 (2010). https://doi.org/10.1093/sysbio/syq010
21. Hallett, M.T., Lagergren, J.: New Algorithms for the Duplication-loss Model. In: Proceedings of the Fourth Annual International Conference on Computational Molecular Biology. pp. 138–146. RECOMB '00, ACM, New York, NY, USA (2000). https://doi.org/10.1145/332306.332359
22. Lafond, M., Meghdari Miardan, M., Sankoff, D.: Accurate prediction of orthologs in the presence of divergence after duplication. Bioinformatics **34**(13), i366–i375 (2018)
23. Lefort, V., Desper, R., Gascuel, O.: FastME 2.0: A Comprehensive, Accurate, and Fast Distance-Based Phylogeny Inference Program. Molecular Biology and Evolution **32**(10), 2798–2800 (2015). https://doi.org/10.1093/molbev/msv150
24. Legried, B., Molloy, E.K., Warnow, T., Roch, S.: Polynomial-time statistical estimation of species trees under gene duplication and loss. bioRxiv (2019). https://doi.org/10.1101/821439

25. Liu, L., Yu, L., Edwards, S.V.: A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. BMC Evolutionary Biology **10**(1), 302 (2010). https://doi.org/10.1186/1471-2148-10-302
26. Mallo, D., Martins, L.D.O., Posada, D.: SimPhy: phylogenomic simulation of gene, locus, and species trees. Systematic Biology **65**(2), 334–344 (2016). https://doi.org/10.1093/sysbio/syv082
27. Mirarab, S., Reaz, R., Bayzid, M.S., Zimmermann, T., Swenson, M.S., Warnow, T.: ASTRAL: genome-scale coalescent-based species tree estimation. Bioinformatics **30**(17), i541–i548 (2014). https://doi.org/10.1093/bioinformatics/btu462
28. Rabiee, M., Sayyari, E., Mirarab, S.: Multi-allele species reconstruction using ASTRAL. Molecular Phylogenetics and Evolution **130**, 286–296 (2019). https://doi.org/10.1016/j.ympev.2018.10.033
29. Rasmussen, M.D., Kellis, M.: Unified modeling of gene duplication, loss, and coalescence using a locus tree. Genome Research **22**(4), 755–765 (2012). https://doi.org/10.1101/gr.123901.111
30. Robinson, D.F., Foulds, L.R.: Comparison of Phylogenetic Trees. Mathematical Biosciences **53**(1-2), 131–147 (1981). https://doi.org/10.1016/0025-5564(81)90043-2
31. Roch, S., Nute, M., Warnow, T.: Long-branch attraction in species tree estimation: Inconsistency of partitioned likelihood and topology-based summary methods. Systematic Biology **68**(2), 281–297 (2018). https://doi.org/10.1093/sysbio/syy061
32. Ronquist, F., Huelsenbeck, J.P.: MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics **19**(12), 1572–1574 (2003)
33. Sanderson, M.J., McMahon, M.M.: Inferring angiosperm phylogeny from EST data with widespread gene duplication. BMC evolutionary biology **7**(1), S3 (2007)
34. Shekhar, S., Roch, S., Mirarab, S.: Species tree estimation using ASTRAL: how many genes are enough? IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **15**(5), 1738–1747 (2018)
35. Stamatakis, A.: RAxML Version 8: A tool for Phylogenetic Analysis and Post-Analysis of Large Phylogenies. Bioinformatics **30**(9) (2014). https://doi.org/10.1093/bioinformatics/btu033
36. Sukumaran, J., Holder, M.T.: Dendropy: a python library for phylogenetic computing. Bioinformatics **26**(12), 1569–1571 (2010)
37. The Quest for Orthologs Consortium, Sousa da Silva, A.W., Boeckmann, B., Dessimoz, C., Sonnhammer, E.L., Robinson-Rechavi, M., Martin, M., Thomas, P.D., Gabald ón, T.: Big data and other challenges in the quest for orthologs. Bioinformatics **30**(21), 2993–2998 (2014). https://doi.org/10.1093/bioinformatics/btu492
38. Tuffley, C., Steel, M.: Links between maximum likelihood and maximum parsimony under a simple model of site substitution. Bulletin of mathematical biology **59**(3), 581–607 (1997)
39. Vachaspati, P., Warnow, T.: FastRFS: fast and accurate Robinson-Foulds Supertrees using constrained exact optimization. Bioinformatics **33**(5), 631–639 (2016). https://doi.org/10.1093/bioinformatics/btw600
40. Wehe, A., Bansal, M.S., Burleigh, J.G., Eulenstein, O.: DupTree: a program for large-scale phylogenetic analyses using gene tree parsimony. Bioinformatics **24**(13), 1540–1541 (2008). https://doi.org/10.1093/bioinformatics/btn230
41. Wickett, N.J., Mirarab, S., Nguyen, N., Warnow, T., et al.: Phylotranscriptomic analysis of the origin and early diversification of land plants. Proceedings of the National Academy of Sciences of the USA **111**(45), E4859–E4868 (2014). https://doi.org/10.1073/pnas.1323926111

42. Zhang, C., Rabiee, M., Sayyari, E., Mirarab, S.: ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. BMC bioinformatics **19**(6), 153 (2018)

A Reduction of MulRF Distance to RF Distance

A.1 Notation and Definitions

A *phylogenetic tree* T is an unrooted tree topology t with leaves labeled by a set S. Let $L(T)$ denote the leaf set of T, and let $\phi : L(T) \to S$ denote the mapping from leaves to labels. Then, T is defined by the triplet (t, ϕ, S). If ϕ is a bijection, then each leaf has a unique label, and thus, we say that T is *singly-labeled*. If two or more leaves have the same label, we say that T is *multi-labeled*. Multi-labeled trees are also referred to as *mul-trees*.

Let $V(T)$ and $E(T)$ denote the vertex set and the edge set of a T, respectively. Deleting an edge e but not its endpoints from T produces two rooted subtrees T_A and T_B, splitting the leaf labels in two sets: $A = \{\phi(l) : l \in L(T_A)\}$ and $B = \{\phi(l) : l \in L(T_B)\}$. If $A \cap B = \emptyset$, then edge e induces a bipartition $\pi(e) = A|B$. We denote the set of bipartitions in T as $C(T) = \{\pi(e) : e \in E(T)\}$.

An *contraction* operation corresponds to deleting an edge (u, v) but not its endpoints from T and then identifying u and v; in this case, we say that edge (u, v) was contracted. We say that tree T is a contraction of T' if it is obtained from tree T' by a sequence of edge contractions. A *refinement* operation is the reverse of an edge contraction, and we say that T' is a refinement of T if and only if T is a contraction of T'. A *restriction* operation corresponds to deleting all leaves with labels in the set $Z \subset S$ and suppressing internal nodes of degree 2; in this case, we say that T was restricted to Z.

A vertex in an unrooted tree with degree greater than three is referred as a *polytomy*. If at least one vertex in $V(T)$ is a polytomy, then we say that T is *unresolved*; otherwise, we say that T is *fully resolved*, as no refinements are possible.

Definition 3 (Robinson-Foulds Distance [30]). *The Robinson-Foulds (RF) distance between two phylogenetic trees T_1 and T_2 with leaves labeled by the set S is the minimum number of contraction and refinement operations required to transform T_1 into T_2.*

Then, the RF distance between two singly-labeled trees T_1 and T_2 on the same leaf set can be computed by:

$$RF(T_1, T_2) = |C(T_1) \setminus C(T_2)| + |C(T_2) \setminus C(T_1)| \tag{2}$$

By convention, if two phylogenetic trees T_1 and T_2 are on different label sets S_1 and S_2, respectively, then T_1 and T_2 are restricted to their shared label set $S_1 \cap S_2$ prior to computing their RF distance. In order to compute the RF distance for two mul-trees, we require the following two additional definitions from [10,19].

Definition 4 (Full Differentiation). *We say that $T' = (t, \phi', S')$ is a full differentiation of mul-tree $T = (t, \phi, S)$ if $\phi' : L(T) \to S'$ is a bijection. In other words, T' is a singly-labeled version of T.*

Definition 5 (Mutually Consistent Full Differentiations). *Let T_1 and T_2 be mul-trees. Let $T_1' = (t_1, \phi_1', S')$ and $T_2' = (t_2, \phi_2', S')$ be a full differentiation of $T_1 = (t_1, \phi_1, S)$ and $T_2 = (t_2, \phi_2, S)$, respectively. Let $R_1(s) \in S'$ be the set of labels given to the leaves in T_1' that are labeled s in T_1 (i.e., $R_1(s) = \{\phi_1'(v) : \phi_1(v) = s\}$). Similarly, let $R_2(s) \in S'$ be the set of labels given to the leaves in T_2' that are labeled s in T_2 (i.e., $R_1(s) = \{\phi_1'(v) : \phi_1(v) = s\}$). We say that T_1' and T_2' are mutually consistent full differentiations (MCFDs) of T_1 and T_2 if $R_1(s) = R_2(s)$ for all $s \in S$.*

A.2 Computation of RF distances

[19] showed that if T_1 and T_2 are both mul-trees, then their RF distance can be computed as follows.

$$MulRF(T_1, T_2) := \min\{RF(T_1', T_2') : T_1', T_2' \text{ are MCFDs of } T_1, T_2\} \quad (3)$$

Equation 3 implies an exponential-time algorithm for computing the RF distance between two mul-trees [19], and this problem was later shown to be NP-complete by [10]. In addition, [10] introduced a special case of this problem, where one of the two mul-trees has the property: every leaf with the same label is grouped together into polytomy that is separated by an edge from the remainder of the tree. Importantly, a mul-tree with this property can be viewed as an extended version of a singly-labeled tree.

Definition 6 (Extended Version). *Let $T = (t, \phi_T, S)$ be a singly-labeled tree, and let $M = (m, \phi_M, S)$ be a mul-tree. Let $k(s)$ be the number of leaves with label s in M. The extended version of T relative to M, denoted $Ext(T, M)$, is created by attaching $k(s)$ new leaves to the leaf labeled s in T, assigning label s to each of these new leaves, and repeating this process for all $s \in S$.*

[10] showed that the RF distance between a mul-tree M and an extended version of a singly-labeled tree T relative to M can be computed in polynomial time. Here, we provide an alternative, although related, proof that further simplifies this problem.

Consider a fully resolved, singly-labeled T and a mul-tree M, both on the same label set S. In this section, we refer to elements of the set S as species or species labels. Let M' denote a version of M with leaves labeled bijectively by a set S' (i.e., M' is a full differentiation of M). Then, given a function $f : S' \to S$ and a bipartition $A|B \in C(M')$, we can apply f to every element of A and B, denoted $f(A)$ and $f(B)$, respectively. We now present two transformations that can be applied to M' by using the function f; we described how to apply these transformations directly to mul-tree M (rather than its full differentiation M') in the main text (Definition 2).

Definition 7 (Contracted Version). *The contracted version of M', denoted $\mathcal{X}(M')$, is created by contracting every edge that induces a bipartition with at least one species label on both sides of the split. Mathematically, contract edge e if and only if it fails to induce a bipartition (i.e., contract edge e if $\pi(e) = A|B$ such that $f(A) \cap f(B) \neq \emptyset$).*

Definition 8 (Reduced Version). *If all leaves with the same species label are on the same side of every bipartition, then they can be represented by a single leaf. More formally, let $\phi : L(M') \to S'$ be the labeling function for M'. Then, if, for all $A|B \in C(M')$, $(f(A) \cap \{s\}) \cap (f(B) \cap \{s\}) = \emptyset$, then delete all but one of the leaves in the set $\{f(\phi(l)) = s : l \in L(M')\}$ (suppressing internal vertices of degree 2) and assign species label s to the only remaining leaf; repeat for all $s \in S$. We call the resulting tree the reduced version of M', denoted $\mathcal{R}(M')$.*

It is easy to see that $\mathcal{R}(\mathcal{X}(M'))$ is singly-labeled tree that is isomorphic to $\mathcal{R}(\mathcal{X}(M))$ (Fig. 1). By definition, after applying the function $\mathcal{X}$ to either M' or M, leaves with species label s will be on the same side of every bipartition; thus they can be replaced by a single leaf with species label s by applying the function $\mathcal{R}$. This observation holds for all $s \in S$.

Lemma 4. *Let T be a singly-labeled, fully resolved tree on leaf set S, let M be a mul-tree on leaf set S, let E be the extended version of T for M, and let E' and M' be MCFDs of E and M, respectively. Then,*

$$RF(E', M') = RF(T, M_X) + K \tag{4}$$

where K is constant for all possible T and $M_X = \mathcal{R}(\mathcal{X}(M))$.

Proof. Let $X \in C(M')$ denote the set of bipartions in M' that are guaranteed to be missing from $C(T')$ for all possible T, that is, $A|B \in X$ if and only if $|f(A) \cap f(B)| \neq 0$. Let $R \in C(M')$ denote the set of bipartitions that are guaranteed to be in $C(T')$ for all possible T, that is, $A|B \in R$ if and only if $|f(A)| = 1$ or $|f(B)| = 1$. Then,

$$\begin{aligned} RF(E', M') :=& |C(E') \setminus C(M')| + |C(E') \setminus C(M')| \\ =& 2|S| - 3 + |E(M')| - |L(M')| - 2|C(E') \cap C(M')| \end{aligned}$$

where

$$\begin{aligned} |C(E') \cap C(M')| &= |C(E') \cap (C(M') \setminus X)| \\ &= |C(E') \cap C(\mathcal{X}(M'))| \\ &= |(C(E') \setminus R) \cap (C(\mathcal{X}(M')) \setminus R)| + |R| \\ &= |C(\mathcal{R}(E')) \cap C(\mathcal{R}(\mathcal{X}(M')))| + |R| \\ &= |C(T) \cap C(M_X)| + |R| \end{aligned}$$

$|S|$, $|E(M)|$, $|L(M)|$ are independent of T, and $|R|$ is the same for all T. □

From Lemma 4, it easily follows that $MulRF(E, M)$ can be computed in polynomial time, as computing the RF distance does not depend on the MCFDs of E and M. While our proof is related to the proof of Theorem 3 in [10], we additionally show that the RF distance between M and T can be computed (up to a constant factor that does not depend on T) by simply transforming M into a singly-labeled tree on the set S.

B Algorithm for Preprocessing Mul-trees

The algorithm below is for contracting the tree in $O(np)$ space and $O(np)$ time. The reduced version of T can clearly be computed in $O(p)$ time.

Algorithm 1: Contracted Version of Mul-tree $\mathcal{X}(T)$

Input : A mul-tree $T = (t, \phi, S)$ with $|L(T)| = p$ and $|S| = n$
Output: The contracted version of $\mathcal{X}(T)$

```
Root T on an arbitrary edge
L ← [0]_{2p×n}; R ← [0]_{2p×n}; U ← [0]_{2p×n}
// Create Down Profiles
foreach v ∈ Post-order traversal of the vertices in T do
    if v is a leaf then
        s ← φ(l); D[v][s] ← 1;
    else
        l ← leftchild(v); r ← rightchild(v)
        foreach s ∈ S do
            D[v][s] ← D[l][s] ∨ D[r][s]
// Create Up Profile
l ← leftchild(root); r ← rightchild(root)
foreach s ∈ S do
    U[l][s] ← D[r][s]
    U[r][s] ← D[l][s]
foreach v ∈ Pre-order traversal of the internal vertices in T minus the
  root and the root's children do
    p ← parent(v); pl ← leftchild(v)
    if v == pl then
        x ← rightchild(p)
    else
        x ← pl
    foreach s ∈ S do
        U[v][s] ← U[p][s] ∨ D[x][s]
// Contract Edges
foreach v ∈ Post-order traversal of the internal vertices in T do
    foreach s ∈ S do
        if D[v][s] ∧ U[v][s] then
            Contract the edge (v, p)
            Break
return T
```

C Data Generation and Software Commands

All datasets and scripts used in this study are available on the Illinois Data Bank: https://doi.org/10.13012/B2IDB-5721322_V1 (link will be active on November 11, 2019).

C.1 SimPhy Simulation

Here we describe the simulation of gene trees from a species tree under a model of GDL. Our protocol is based on the simulation study by [14] that uses a biological dataset (16 species of fungi) from [29]. We simulated gene trees with different rates of GDL and ILS based on the species tree estimated by Rasmussen and Kellis [29] (download: http://compbio.mit.edu/dlcoal/pub/config/fungi.stree), which had estimated branch lengths (in millions of years). We multiplied each branch length by 10^7 (i.e., assuming 10 generations per year) to get a tree height of 1,800,000,337.5 years (as suggested by [14]). We then found (through visualization of 16-taxon species trees) that speciation rates of 1.8×10^{-10} and 1.8×10^{-8} corresponded to deep and recent speciation, respectively; we used the intermediate value of 1.8×10^{-9}.
Finally, we ran SimPhy [26] Version 1.0.2 with the command:

```
simphy-1.0.2-mac64 -rs 10 -rl F:$ngen -rg 1 -sb F:$sprt -st $trln \
    -sl F:$ntax -si F:1 -sp F:$psiz -su F:$murt -so F:1 -lb F:$dlrt \
    -ld F:lb -hg LN:1.5,1 -o <output directory> -ot 0 -om 1 -od 1 \
    -op 1 -oc 1 -ol 1 -v 3 -cs 293745 &> <log file>
```

where `$ntax` is the number of taxa (100), `$ngen` is the number of genes (1000), `$trln` is the tree length (1,800,000,337.5 years), `$sprt` is the speciation rate (1.8×10^{-9} events per year), `$psiz` is the effective population size (either 1×10^7 or 5×10^7), `$murt` is the mutation rate (0.0000000004 mutations per year) and `$dlrt` is the duplication and loss rate (either 1×10^{-10}, 2×10^{-10}, or 5×10^{-10} events per year). Note that the tree-wide effective population size (`-sp`), the tree-wide substitution rate (`-su`), the duplication rate (`-lb`), and the loss rate (`-ld`) are the same parameters used by [14]; these parameters (with GDL rate of 1×10^{-10} and effective population size of 1×10^7) are similar to those estimated from the biological dataset by [29].

Unlike in the simulation performed by [14], we did not enable gene conversion, and we allowed gene trees to deviate from a molecular clock by using gene-by-lineage-specific rate heterogeneity modifiers (`-hg`), meaning that for each gene tree, a gamma distribution was defined for each gene tree by drawing α from a log-normal distribution with a location of 1.5 and a scale of 1 (same parameters as used in [42]), and then each branch length in a gene tree is multiplied by a value drawn the gamma distribution corresponding to that gene tree.

SimPhy simulates gene trees from a species tree under the unified model of GDL and ILS proposed by [29]. This simulation procedure has two steps: first, a collection of locus trees are simulated from the species tree, and second, a gene tree is simulated for each of the locus trees. Gene trees can differ from the species

tree due to GDL as well as ILS, whereas gene trees differ from the locus trees due to ILS *only*. We quantified the level of ILS by computing the normalized RF distance between each true locus tree and its respective true gene tree (which are on the same leaf set), averaging this value across all 1000 locus/gene trees. The average locus-to-gene tree discord (AD) across the 10 replicate datasets is 2% and 12% for datasets with an effective population size of 1×10^7 and 12% for 5×10^7, respectively. We also quantified the level of GDL by examining the number of leaves and the number of species per gene tree. All gene trees had approximately 100 leaves, as the duplication and loss rates are equal. As the duplication/loss rate increases, the number of species per gene tree decreases, and thus, even though locus/gene trees have the same number of leaves on average, these leaves are labeled by fewer species. For duplication/loss rates of 1×10^{-10}, 2×10^{-10}, and 5×10^{-10} the average number of species per gene tree was 85, 74, and 53 species. More information about the number of copies per species is reported in Table 1.

Table 1: For each dataset, we computed the mean (± standard deviation), the minimum, and the maximum number of copies of each species across all gene tree. We then averaged these values across all species, and then across all ten replicate datasets in order to report a single value per model condition.

GDL rate	ILS (AD%)	Mean ± Standard deviation	Minimum	Maximum
1×10^{-10}	2%	1.0 ±0.6	0	4.4
1×10^{-10}	12%	1.0 ±0.6	0	4.3
2×10^{-10}	2%	1.0 ± 0.9	0	6.0
2×10^{-10}	12%	1.0 ± 0.9	0	5.9
5×10^{-10}	2%	1.0 ± 1.3	0	9.6
5×10^{-10}	12%	1.0 ± 1.4	0	9.6

C.2 INDELible Simulation

Here we describe the simulation of multiple sequence alignments for each model gene tree produced by SimPhy under the GTR+Γ model of evolution. Again, our protocol is also based on the fungal dataset from [29] (http://compbio.mit.edu/dlcoal/pub/data/real-fungi.tar.gz), which included a multiple sequence alignment estimated using MUSCLE [15] and a maximum likelihood tree estimated using PhyML [20] for each of the 5,351 genes. We estimated GTR+Γ model parameters for each of the PhyML gene trees by running RAxML Version 8.2.12 [35] with the following command:

```
raxmlHPC-SSE3 -m GTRGAMMA -f e -t <PhyML gene tree file> \
    -s <MUSCLE alignment file> -n <output name>
```

We then fit distributions to the GTR+Γ model parameters estimated from alignments with at least 500 distinct alignment patterns and at most 25% gaps.

Table 2: For the fungal dataset, we computed the mean (± standard deviation), the minimum, and the maximum number of copies of each species across 5,351 gene trees. We also computed the number of gene trees with more than 1, 2, 5, 10, and 20 copies of a species (i.e., > 1 indicates the number of gene trees out of 5,351 with more than 1 copy of the species.)

Species	Mean ± Std	Minimum	Maximum	>1	>2	>5	>10	>20
A. gossypii	0.85 ± 0.58	0	13	267	41	4	1	0
C. alibicans	1.04 ± 0.65	0	7	596	145	6	0	0
C. glabrata	0.93 ± 0.81	0	27	589	104	9	4	1
C. guilliermondii	0.99 ± 0.70	0	11	587	143	7	1	0
C. lusitaniae	0.95 ± 0.62	0	10	523	101	2	0	0
C. parapsilosis	1.00 ± 0.73	0	12	588	148	15	2	0
C. tropicalis	1.04 ± 0.73	0	8	655	174	17	0	0
D. hansenii	1.02 ± 0.65	0	7	588	138	10	0	0
K. latics	0.89 ± 0.63	0	15	336	55	10	2	0
K. waltii	0.88 ± 0.71	0	18	315	57	12	4	0
L. elongisporus	0.98 ± 0.66	0	9	576	115	7	0	0
S. bayanus	0.94 ± 0.81	0	23	684	117	9	2	1
S. castellii	1.01 ± 0.91	0	25	826	148	17	6	1
S. cerevisiae	1.03 ± 1.09	0	42	752	146	19	5	3
S. mikatae	0.92 ± 0.76	0	18	629	114	11	2	0
S. paradoxus	0.95 ± 0.83	0	25	690	125	13	2	1

Finally, we drew GTR+Γ model parameters from these distributions and then simulated a multiple sequence alignment (1000 base pairs) using INDELible Version 1.03 for each of the gene trees [18]. Note that GTR base frequences (A, C, G, T) were drawn from Dirchlet(113.48869, 69.02545, 78.66144, 99.83793), GTR substitution rates (AC, AG, AT, CG, CT, GT) were drawn from Dirchlet(12.776722, 20.869581, 5.647810 9.863668, 30.679899, 3.199725), and α was drawn from LogNormal(-0.470703916, 0.348667224), where the first parameter is the meanlog and the second parameter is the sdlog.

C.3 Gene Tree Estimation

On gene trees with 4 or more species, we estimated gene trees using RAxML Version 8.2.12 with the command:

```
raxmlHPC-SSE3 -m GTRGAMMA -p <random seed> -n <output name> \
    -s <alignment file>
```

Sequences were truncated to the first 25, 50, 100, and 250 nucleotides to produce datasets with varying levels of gene tree estimation error (GTEE). GTEE was measured by the normalized Robinson-Foulds (RF) distance between the true and the estimated gene family trees. Sequence lengths of 25, 50, 100, and 250 resulted in average GTEE of 67%, 52%, 35%, and 19%, respectively.

C.4 Species Tree Estimation Commands

We estimated species trees for each data set using either the first 25, 50, 100, or 500 gene trees; recall that gene trees were either true or had one of four

GTEE levels. Gene trees were formatted for each species tree method with custom Python scripts using Dendropy [36].

ASTRAL Version 5.6.3 was run with the command:

```
java -Xms2000M -Xmx20000M -jar astral.5.6.3.jar -i <gene tree file> \
    -a <species to gene name map file> -o <output file> &> <log file>
```

DupTree (download: http://genome.cs.iastate.edu/CBL/DupTree/linux-i386.tar.gz) was run with the command:

```
./duptree -i <gene tree file> -o <output file> &> <log file>
```

MulRF Version 2.1 was run with the command:

```
./MulRFSupertreeLin -i <gene tree file> \
    -o <output file> &> <log file>
```

FastRFS Version 1.0 (Initial Release) was run with the command:

```
./FastRFS -i <gene tree file> -o <output file> &> <log file>
```

FastMulRFS requires ASTRAL to generate the constrained search space; we used ASTRAL Version 5.6.3. Gene trees were pre-processed before running FastMulRFS as described in Appendix B using a custom Python script. STAG (download: https://github.com/davidemms/STAG) was run with the command:

```
python stag.py <species to gene name map file> \
    <gene tree folder> &> <log file>
```

STAG requires FastME [23] as a dependency; we used FastME Version 2.1.5. Importantly, STAG only uses gene trees that include at least one copy of every species, so STAG failed on many datasets, and is excluded from the figures on simulated datasets.

D Results on Simulated Datasets

We compared FastMulRFS and MulRF with respect to RFS-multree criterion scores. Out of the 1200 datasets analyzed:

- FastMulRFS was worse than MulRF on 67 datasets
- FastMulRFS was equal to MulRF on 907 datasets
- FastMulRFS was better than MulRF on 226 datasets

This comparison shows that MulRF was actually quite good at finding good scores, but that when the two methods produced trees with different scores, then FastMulRFS dominated MulRF. We also compare methods in terms of species tree accuracy (Table 3) and running time (Table 4).

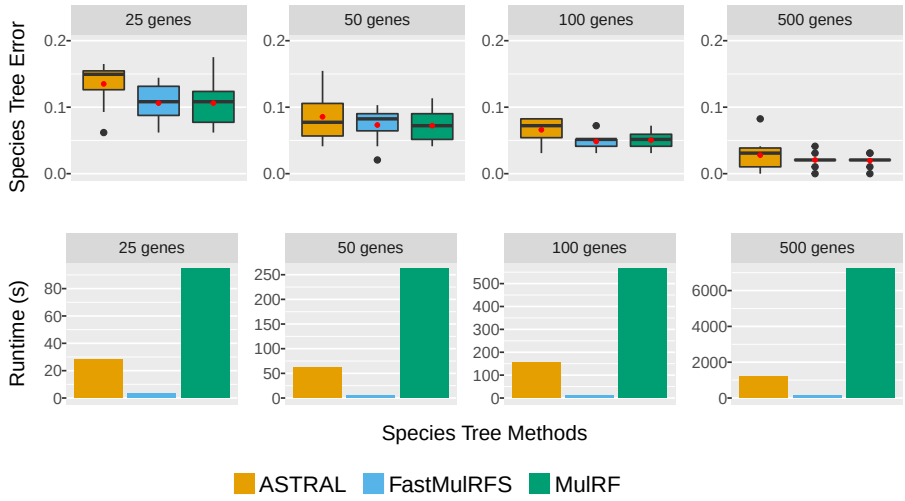


Fig. 4: Species tree error (i.e., normalized RF distance) and running time (seconds) are shown for three methods (FastMulRFS, MulRF, and ASTRAL-multi) under the easiest model condition with 100 species. All data sets have substantial GTEE (52%), low GDL (D/L rate: 1×10^7), and very low ILS (2%). Red dots (first row) and bars (second row) are means for 10 replicate datasets.

Table 3: Species tree error (i.e., normalized RF distance between the true and estimated species trees) averaged across 10 replicate datasets for each of the model conditions on 100-species simulated datasets for ASTRAL-multi / FastMulRFS / MulRF. ILS level is measured using the AD (average normalized RF distance) between true locus trees and true gene trees.

GTEE	25 genes	50 genes	100 genes	500 genes
ILS: 2% AD, D/L Rate: 1E-10				
67%	0.26 / 0.22 / 0.23	0.19 / 0.12 / 0.13	0.17 / 0.08 / 0.09	0.10 / 0.04 / 0.05
52%	0.14 / 0.11 / 0.11	0.09 / 0.07 / 0.07	0.07 / 0.05 / 0.05	0.03 / 0.02 / 0.02
35%	0.07 / 0.07 / 0.06	0.05 / 0.05 / 0.05	0.04 / 0.03 / 0.03	0.02 / 0.02 / 0.02
19%	0.04 / 0.04 / 0.04	0.03 / 0.03 / 0.02	0.02 / 0.02 / 0.02	0.01 / 0.01 / 0.01
0%	0.01 / 0.01 / 0.01	0.01 / 0.00 / 0.00	0.00 / 0.00 / 0.00	0.00 / 0.00 / 0.00
ILS: 2% AD, D/L Rate: 2E-10				
67%	0.33 / 0.30 / 0.36	0.27 / 0.21 / 0.21	0.19 / 0.14 / 0.16	0.12 / 0.06 / 0.07
52%	0.19 / 0.16 / 0.16	0.15 / 0.10 / 0.10	0.11 / 0.08 / 0.08	0.05 / 0.04 / 0.04
35%	0.11 / 0.10 / 0.09	0.08 / 0.05 / 0.05	0.05 / 0.04 / 0.04	0.03 / 0.02 / 0.02
19%	0.07 / 0.05 / 0.05	0.04 / 0.03 / 0.03	0.03 / 0.03 / 0.03	0.01 / 0.01 / 0.01
0%	0.01 / 0.01 / 0.01	0.01 / 0.00 / 0.00	0.01 / 0.00 / 0.00	0.01 / 0.00 / 0.00
ILS: 2% AD, D/L Rate: 5E-10				
67%	0.48 / 0.42 / 0.48	0.37 / 0.31 / 0.33	0.32 / 0.21 / 0.22	0.19 / 0.08 / 0.07
52%	0.33 / 0.28 / 0.29	0.23 / 0.18 / 0.17	0.18 / 0.10 / 0.10	0.09 / 0.05 / 0.04
35%	0.19 / 0.18 / 0.16	0.14 / 0.12 / 0.09	0.11 / 0.07 / 0.06	0.05 / 0.03 / 0.02
19%	0.11 / 0.11 / 0.08	0.09 / 0.07 / 0.05	0.07 / 0.04 / 0.03	0.02 / 0.02 / 0.02
0%	0.04 / 0.01 / 0.01	0.04 / 0.01 / 0.01	0.02 / 0.01 / 0.01	0.00 / 0.00 / 0.00
ILS: 12% AD, D/L Rate: 1E-10				
67%	0.32 / 0.27 / 0.27	0.24 / 0.17 / 0.17	0.18 / 0.12 / 0.14	0.11 / 0.06 / 0.06
52%	0.19 / 0.15 / 0.15	0.13 / 0.12 / 0.11	0.12 / 0.07 / 0.08	0.05 / 0.04 / 0.04
35%	0.11 / 0.09 / 0.09	0.07 / 0.06 / 0.06	0.05 / 0.05 / 0.05	0.03 / 0.02 / 0.02
19%	0.07 / 0.06 / 0.06	0.05 / 0.04 / 0.04	0.03 / 0.03 / 0.03	0.02 / 0.02 / 0.01
0%	0.04 / 0.03 / 0.04	0.02 / 0.02 / 0.02	0.01 / 0.02 / 0.02	0.01 / 0.01 / 0.01
ILS: 12% AD, D/L Rate: 2E-10				
67%	0.35 / 0.30 / 0.34	0.26 / 0.22 / 0.23	0.20 / 0.14 / 0.15	0.11 / 0.08 / 0.07
52%	0.19 / 0.16 / 0.16	0.15 / 0.12 / 0.12	0.12 / 0.08 / 0.07	0.06 / 0.03 / 0.03
35%	0.12 / 0.10 / 0.10	0.09 / 0.09 / 0.08	0.08 / 0.06 / 0.05	0.03 / 0.02 / 0.02
19%	0.07 / 0.06 / 0.06	0.05 / 0.05 / 0.05	0.05 / 0.04 / 0.03	0.02 / 0.02 / 0.02
0%	0.04 / 0.03 / 0.03	0.03 / 0.03 / 0.02	0.02 / 0.01 / 0.01	0.01 / 0.01 / 0.01
ILS: 12% AD, D/L Rate: 5E-10				
67%	0.47 / 0.45 / 0.50	0.41 / 0.32 / 0.33	0.31 / 0.22 / 0.22	0.17 / 0.11 / 0.09
52%	0.36 / 0.30 / 0.32	0.27 / 0.19 / 0.19	0.20 / 0.15 / 0.12	0.10 / 0.06 / 0.06
35%	0.24 / 0.18 / 0.18	0.17 / 0.11 / 0.10	0.13 / 0.09 / 0.08	0.06 / 0.03 / 0.03
19%	0.18 / 0.13 / 0.11	0.13 / 0.09 / 0.08	0.10 / 0.07 / 0.06	0.04 / 0.03 / 0.03
0%	0.12 / 0.05 / 0.05	0.08 / 0.05 / 0.04	0.06 / 0.03 / 0.03	0.02 / 0.01 / 0.01

Table 4: Running time (in seconds) averaged across 10 replicate datasets on 100-species simulated datasets for each of the model conditions for ASTRAL-multi / FastMulRFS / MulRF. ILS level is measured using the AD (average normalized RF distance) between true locus trees and true gene trees.

GTEE	25 genes	50 genes	100 genes	500 genes
ILS: 2% AD, D/L Rate: 1E-10				
67%	42 / 4 / 113	102 / 8 / 263	245 / 21 / 614	2148 / 296 / 7089
52%	28 / 3 / 95	61 / 6 / 262	156 / 12 / 565	1181 / 143 / 7225
35%	22 / 3 / 89	51 / 9 / 211	121 / 8 / 623	756 / 72 / 6921
19%	19 / 2 / 86	37 / 3 / 226	86 / 7 / 596	641 / 35 / 6727
0%	20 / 2 / 99	46 / 8 / 228	106 / 4 / 582	769 / 17 / 6431
ILS: 2% AD, D/L Rate: 2E-10				
67%	52 / 4 / 91	123 / 7 / 273	306 / 23 / 580	2514 / 199 / 6937
52%	36 / 5 / 99	76 / 5 / 280	213 / 10 / 577	1357 / 119 / 7225
35%	24 / 3 / 92	49 / 11 / 254	173 / 8 / 636	930 / 66 / 6987
19%	26 / 2 / 105	47 / 10 / 237	159 / 6 / 613	978 / 44 / 6852
0%	26 / 2 / 103	55 / 2 / 197	157 / 12 / 597	829 / 21 / 6827
ILS: 2% AD, D/L Rate: 5E-10				
67%	86 / 4 / 102	232 / 7 / 232	418 / 14 / 585	4495 / 133 / 6884
52%	49 / 4 / 114	128 / 5 / 691	258 / 10 / 692	2615 / 83 / 6793
35%	40 / 2 / 112	79 / 12 / 265	195 / 8 / 565	1856 / 60 / 6635
19%	32 / 8 / 111	84 / 3 / 241	177 / 6 / 546	1288 / 44 / 6645
0%	36 / 5 / 110	83 / 11 / 215	153 / 12 / 542	1308 / 25 / 6608
ILS: 12% AD, D/L Rate: 1E-10				
67%	43 / 5 / 112	93 / 8 / 239	214 / 20 / 680	2107 / 274 / 6882
52%	29 / 4 / 111	63 / 6 / 573	133 / 14 / 603	1044 / 132 / 7597
35%	21 / 3 / 105	55 / 5 / 479	125 / 14 / 643	726 / 69 / 7351
19%	22 / 2 / 77	49 / 5 / 232	111 / 11 / 610	673 / 42 / 6976
0%	21 / 2 / 83	44 / 4 / 318	102 / 6 / 572	704 / 29 / 6485
ILS: 12% AD, D/L Rate: 2E-10				
67%	57 / 4 / 109	130 / 7 / 510	302 / 24 / 694	3098 / 206 / 7582
52%	32 / 10 / 107	74 / 14 / 383	193 / 12 / 583	1767 / 120 / 7085
35%	28 / 10 / 103	62 / 13 / 223	166 / 12 / 672	1052 / 61 / 6896
19%	22 / 2 / 106	53 / 11 / 238	182 / 11 / 541	836 / 40 / 7189
0%	23 / 2 / 104	47 / 5 / 279	121 / 8 / 586	808 / 32 / 6669
ILS: 12% AD, D/L Rate: 5E-10				
67%	105 / 4 / 109	205 / 7 / 431	469 / 14 / 599	4368 / 124 / 6889
52%	63 / 3 / 128	115 / 5 / 274	258 / 10 / 509	2572 / 80 / 7107
35%	46 / 3 / 112	101 / 4 / 204	185 / 16 / 647	1896 / 69 / 6745
19%	37 / 2 / 112	82 / 7 / 261	193 / 13 / 638	1392 / 44 / 6786
0%	37 / 7 / 95	73 / 3 / 251	173 / 5 / 544	1402 / 77 / 6746

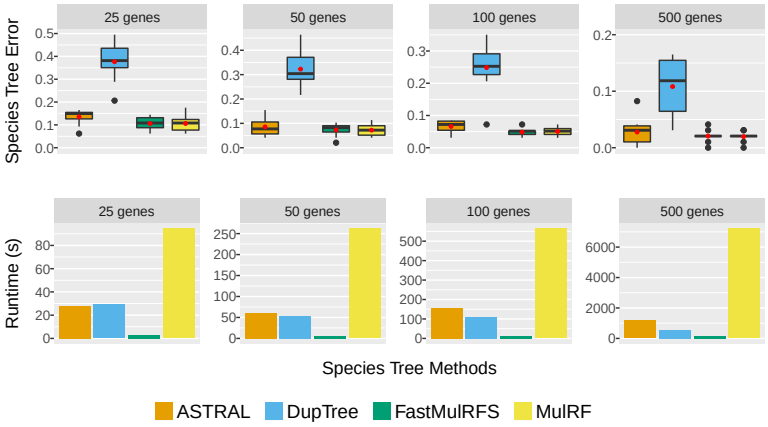


Fig. 5: Species tree error (i.e., normalized RF distance) and running time (seconds) are shown for FastMulRFS, MulRF, ASTRAL-multi, and DupTree under the easier model conditions, each with 100 species. The model conditions have substantial GTEE (52%), low GDL (D/L rate: 1×10^7), and very low ILS (2%). Red dots (first row of each subfigure) and bars (second row of each subfigure) are means for 10 replicate datasets.

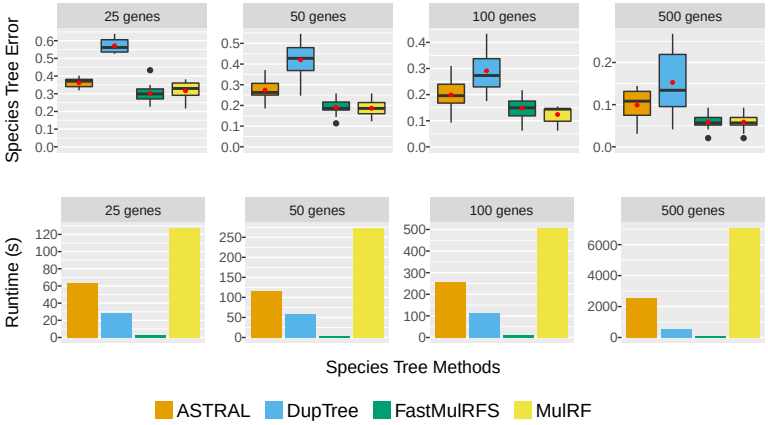
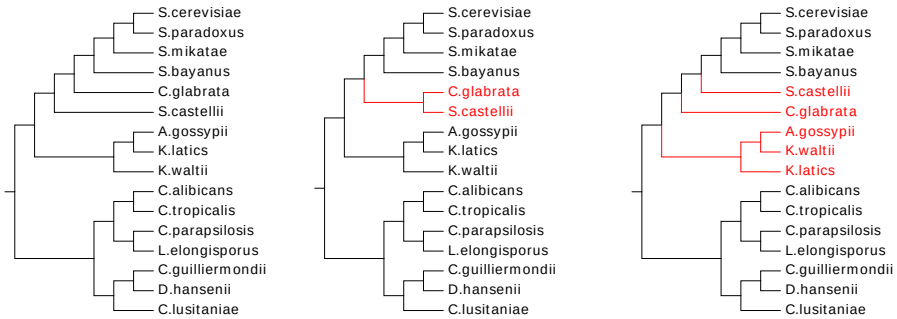


Fig. 6: Species tree error (i.e., normalized RF distance) and running time (seconds) are shown for FastMulRFS, MulRF, ASTRAL-multi, and DupTree, under the harder model conditions, each with 100 species. The model conditions have substantial GTEE (52%), high GDL (D/L rate: 5×10^7), and moderate ILS (12%). Red dots (first row of each subfigure) and bars (second row of each subfigure) are means for 10 replicate datasets.

E Biological Dataset Analysis



Concatenation analysis ASTRAL-multi Other Methods

Fig. 7: Species trees estimated on 16-taxon fungi dataset with 5,351 estimated gene trees from [14]. "Other methods" refers to the tree found by MulRF, DupTree, STAG, and FastMulRFS. The differences between the trees are fairly small, and the true placement of these taxa is not established. DupTree and FastMulRFS were the fastest (both completed in less than a minute), ASTRAL-multi estimated a species tree in 18 minutes, STAG completed in 39 minutes, and MulRF completed in 40 minutes.

We analyzed a fungal dataset with 16 species and 5,351 nucleotide gene family trees from [14]. The concatenation tree is one of the trees obtained in [7], and was computed using MrBayes [32] on a concatenated amino acid alignments of putatively orthologous sequences, constrained to enforce the out-grouping of *S. castellii* with respect to *S. cerevisiae* and *C. glabrata*.

The other trees they report differed with respect to this group (i.e., not all returned this as a clade) and differed in the placement of *K. waltii*. The trees that we show for ASTRAL-multi, FastMulRFS, DupTree, MulRF, and STAG, are all similar to the peptide MrBayes tree, and the differences are minor (given the variability in the trees found in [7] and the use of a topological constraint in the MrBayes analysis). See the Supplementary Information, Section 5 from [7] for additional discussion about the phylogenomic analyses they performed.