The Dissertation Committee for Siavash Mir arabbaygi
certifies that this is the approved version of the following dissertation:

Chapter 5 (ASTRAL) of Novel scalable approaches for multiple sequence alignment and phylogenomic reconstruction

Committee:

Keshav Pingali, Supervisor

Tandy Warnow, Co-Supervisor

David Hillis

Bonnie Berger

Joydeep Ghosh

Ray Mooney

Chapter 5 (ASTRAL) of Novel scalable approaches for multiple sequence alignment and phylogenomic reconstruction

by

Siavash Mir arabbaygi, B.S.; M. APPL S.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2015

Chapter 5 (ASTRAL) of
Novel scalable approaches for multiple sequence alignment and phylogenomic reconstruction

Publication No. ____________________

Siavash Mir arabbaygi, Ph.D.
The University of Texas at Austin, 2015

Supervisors: Keshav Pingali
Tandy Warnow

This is chapter 5 taken from the PhD dissertation of Siavash Mirarab. This chapter describes the ASTRAL algorithm in detail and gives results of our experimental evaluations of ASTRAL.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Please refer to the main dissertation for chapters other than Chapter 5.

Chapter 5

ASTRAL[1]

In the previous chapter, we described how a species tree can be estimated from a set of gene trees using either a traditional two step pipeline, or the statistical binning pipeline. Regardless of which pipeline is used, the final step requires a technique that produces a species tree given a collection of input (estimated) gene trees. Such a method is called a summary method, and a desirable attribute of the summary method is to be statistically consistent under the MSC model (see background Section). Many statistically consistent methods have been developed through the years, and of these methods (e.g., MP-EST [3], which we used in the previous chapter) are now in widespread use. However, existing methods are too computationally intensive for use with

[1] Parts of this chapter have appeared in the following papers:

1. Siavash Mirarab, Rezwana Reaz, Md. Shamsuzzoha Bayzid, Théo Zimmermann, M Shel Swenson, and Tandy Warnow. ASTRAL: Genome-Scale Coalescent-Based Species Tree. *Bioinformatics*, 30(17):i541–i548, 2014
2. S. Mirarab and T. Warnow. ASTRAL-II: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 2015

In all three cases, SM and his supervisor, TM, designed the method, designed the studies, and wrote the papers (with comments from others), and SM implemented the methods. SM, MSB, and TZM ran experiments for (1) and SM ran all experiments for (2). MSS and RR worked on earlier versions of ASTRAL algorithmic ideas, and contributed to writing.

genome-scale analyses of large number of species or have poor accuracy under some realistic conditions, as we will show.

Some of these challenges were faced by the thousands plant transcriptomes (1KP) project [4]. The 1KP project has gathered sequences from across the genomes of a large number of plant species (103 plants in the initial phase and more than 1,100 in the ongoing second phase). The goal of the project was to estimate the species tree using various methods, including those that take gene tree incongruence due to incomplete lineage sorting into account. As we will show, these attempts had limited success, mostly due to limitations of existing summary methods.

In this chapter, we introduce a new summary method called ASTRAL (Accurate Species Tree Reconstruction ALgorithm). ASTRAL uses dynamic programming to solve a likely NP-hard optimization problem. ASTRAL can solve the optimization problem exactly in exponential time (doable only for up to 18 species), but more importantly, it can heuristically solve the problem in polynomial time by constraining the search space through a set of allowed bipartitions in the species tree (the constrained version of the problem is solved exactly). As we will show, ASTRAL is statistically consistent, even when run under the "constrained" mode. The constrained version can run on very large datasets, and has outstanding accuracy – improving upon various leading statistically consistent summary methods. ASTRAL is often more accurate than concatenation using maximum likelihood, except when ILS levels are low or there are too few gene trees.

We introduce two versions of ASTRAL: ASTRAL-I and ASTRAL-II. The second version is a direct improvement upon ASTRAL-I, with substantial advantages: ASTRAL-II is faster, can analyze much larger datasets (up to 1000 species and 1000 genes), and has substantially better accuracy under some conditions. ASTRAL-I's running time is $O(n^2k|\mathcal{X}|^2)$, and ASTRAL-II's running time is $O(nk|\mathcal{X}|^2)$, where n is the number of species, k is the number of loci, and $\mathcal{X}$ is the set of allowed bipartitions for the search space. ASTRAL is available in open source at `https://github.com/smirarab/ASTRAL/`.

In the rest of this chapter, we first motivate the development of a new summary method using simulation studies and some observations from the 1KP project. We then give the algorithmic details of ASTRAL-I and ASTRAL-II in Section 5.2 and discuss theoretical properties of both versions of ASTRAL. We then present a simulation study evaluating ASTRAL-I in Section 5.3 and a completely different simulation evaluating ASTRAL-II in Section 5.4. We then evaluate the use of ASTRAL on real biological data (Section 5.5) and finish by discussing results and pointing to directions for future research.

5.1 Motivation

Despite the availability of coalescent-based methods, many biological datasets are too large for the available methods. For example, MP-EST, easily scales to very large number of gene trees but cannot be used on datasets with large number of species due to computational reasons and degradation of accuracy (see [5], but we will show more results supporting this in our results section). BUCKy-pop [6], a method that tries to take into account gene tree uncertainty, is more computationally intensive and cannot run on datasets of moderate size. However, BUCKy tends to have very good accuracy where it can run, and can work with unrooted gene trees [7]. MP-EST has also been shown to have good accuracy under some conditions, but requires rooted gene trees [3]. A new distance-based method called NJst [8] can also handle unrooted gene trees, but NJst is new and its accuracy has not been tested extensively on various datasets.

We were motivated to develop a new summary method by difficulties we were facing on a biological data analysis. The 1KP project [4] gathered sequence data across 103 plant species, with plans to go to more than 1,100 species in the next phase [2]. Our attempts to run MP-EST on this dataset had limited success. The pilot dataset that included 103 species was analyzed to extract 856 genes. We had difficulty in rooting many of these gene trees, since the common ancestor is believed to have existed close to a billion years ago, and our set of outgroups were missing from many of the genes. We built a restricted set of 669 gene trees that could be putatively rooted using outgroups. We attempted to analyze these 669 gene trees using MP-EST.

MP-EST took between 4 to 8 days to finish 5 random runs on each bootstrap replicate of this dataset. The results produced, however, were not

[2] see `http://www.onekp.com/samples/list.php` for the list of species

consistent among the 5 runs, and in some cases had log likelihoods scores that were many times larger than log likelihoods obtained from other runs (e.g., -69150891 in one run and -19540149 in a second run). These differences in log likelihood are not expected and show that the method is failing to search the tree space well in at least some of the random runs (this might be related to the fact that MP-EST uses only NNI moves). The species trees produced using MP-EST had low support, sometimes for easy-to-recover uncontroversial clades that we had recovered with 100% support using concatenation, and even simple statistically inconsistent summary methods such as MRP [9]. The shortcomings of MP-EST on the 1KP dataset could be the result of a combination of factors: rooting is challenging on this dataset, all gene trees are incomplete (are missing some species) and in some gene trees a large number of species are absent, and finally, the number of species being analyzed here is more than all the previous analyses that had tested or used MP-EST (typically below 50 species).

Beyond these challenges, it is possible that optimization scores other than pseudo-likelihood score optimized by MP-EST could simply correlate better with species tree accuracy. For example, a recent paper showed that a simple non-parametric quartet-based way of scoring species trees can predict species tree topological accuracy better than the pseudo-likelihood parametric score used by MP-EST [5]. Similarly, in a recent paper, we have shown that a simple statistically inconsistent method called MRL [11] outperforms MP-EST on large parts of the parameter space (see Fig. 5.1), suggesting that better

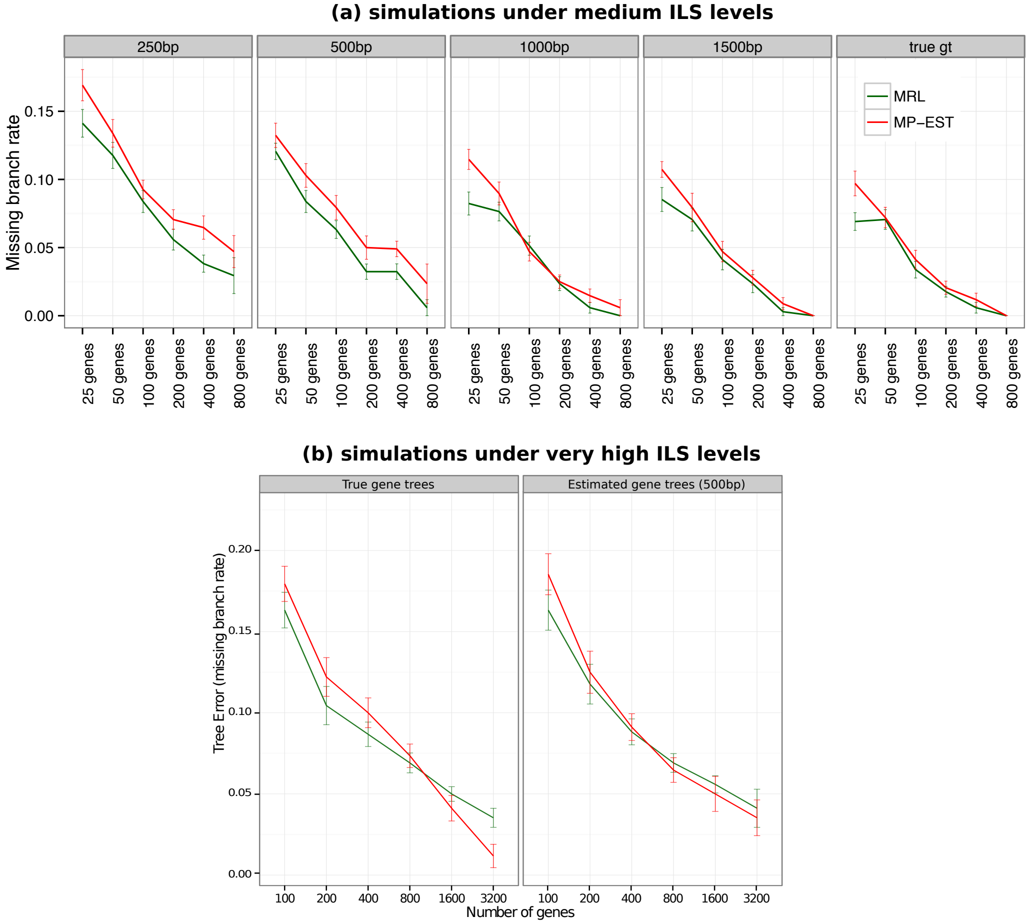


Figure 5.1: **Comparison of MP-EST and MRL on a simulated mammalian dataset**. Species tree error is depicted for a simulated mammalian dataset reported in [10]. Simulation procedures are further described in Section 5.3.1.1. (a) We fix the level of ILS to medium and vary the number of genes and the gene alignment length, which controls gene tree estimation error. (b) We fix the level of ILS to very high, and vary the number of genes. We compare accuracy of MRL and MP-EST. On many conditions MRL has better accuracy; MP-EST, which has theoretical guarantees of statistical consistency, is better than MRL on these data only when levels of ILS are very high and very large number of genes are available.

statistically consistent methods can be developed.

An accurate analysis of the 1KP dataset required a new method that could handle unrooted gene trees, could handle large number of species, and was robust to missing data. More generally, even the best coalescent-based summary methods have not been reliably more accurate than concatenation [12, 13], and analyses of biological datasets have in some cases resulted in species trees that were less well resolved and biologically feasible than concatenation [14, 15]. Hence, the choice between coalescent-based estimation and concatenation is highly controversial [16]. Improved accuracy and scalability for summary methods can help resolving this long-standing debate about the relative accuracy of concatenation and summary methods.

5.2 ASTRAL

Designing a statistically consistent summary method is complicated by the possibility that the most likely gene tree can be different from the species tree (the so-called anomaly zone [17]). However, it has been proved [18–20] that

Theorem 5.2.1. *There are no anomalous rooted 3-taxon species trees and no anomalous unrooted 4-taxon species trees.*

The complete proofs are given in [18] for rooted trees and [19, 20] for unrooted trees. Here, we provide a sketch for the rooted species tree on 3-taxa. Let's consider the case of the 3-taxon tree on human, chimp, and gorilla,

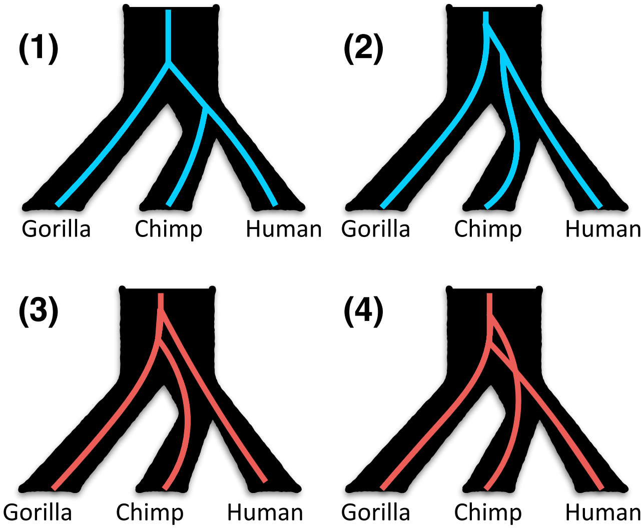


Figure 5.2: **Rooted gene trees and the species tree for 3 taxa**. Four coalescence scenarios can be imagine. (1) The two lineages from sister species chimp and human coalesce in their first ancestral population. The gene tree and the species tree will always be congruent under this scenario. (2-4) Lineages from chimp and human do not coalesce in the ancestral population and go further back into the common ancestor of all three populations. All three scenarios are equiprobable. Blue (1-2): concordance between species tree and gene tree. Red (3-4): discordance.

shown in Figure 5.2. There are three possible gene tree topologies (putting human with chimp or with gorilla, or putting chimp and gorilla together). The lineages from human and chimp have a *non-zero* probability p of coalescing in their most recent common ancestor (scenario 1); gene trees produced by this scenario will agree with the specie tree. If the two lineage fail to coalesce and go further back in time to the previous population, we have three lineages (human, chimp, gorilla) and the first coalescence event is equally likely to be between any pair of lineages (scenarios 2 – 4); thus, the three gene tree topologies are equiprobable in this case, and each topology has a probability of $\frac{1-p}{3}$. The probability of observing the species tree topology among the gene trees, therefore, is $p + \frac{1-p}{3} = \frac{1}{3} + \frac{2p}{3}$, which is strictly greater than $\frac{1}{3}$. Thus, the species tree topology has a higher probability than the two alternative trees. A similar argument can be made for 4-taxon unrooted species trees [18].

The fact that rooted 3-taxon and unrooted 4-taxon species trees do not have anomaly zones underlies the design of some summary methods and their proofs of statistical consistency. These methods decompose the gene trees into triplets or quartets of taxa (for the rooted or the unrooted case, respectively), find the species tree on the triplets or quartets, and then combine the triplet or quartet species trees. ASTRAL uses similar ideas in its design.

While some methods in the literature, such as MP-EST, use rooted triplets of taxa to speed up these analyses, we use unrooted quartet trees in ASTRAL. Rooting gene trees can be challenging, as it typically requires the use of an outgroup, but the given limited data in each gene, the position of the outgroup can be easily misconstructed [16]. For this reason, we believe that by using unrooted input gene trees, ASTRAL finds applicability for more datasets. As we will show, good running time can be achieved even with quartet trees, and ASTRAL has excellent accuracy.

We first start by giving some definitions and describing the notation. We next describe the first version of ASTRAL, and then describe how ASTRAL-II has improved upon ASTRAL-I.

5.2.1 Definitions and notations

We use the following notation throughout the rest of this chapter:

$\mathcal{S}$: a set of n species

$\mathcal{G} = \{t_1, \ldots, t_k\}$: a set of k binary unrooted gene trees leaf-labelled by $\mathcal{S}$.

r: an arbitrary set of four species $\{a, b, c, d\} \subset \mathcal{S}$.

$\mathcal{Q}$: the set of all $\binom{n}{4}$ quartets of taxa selected from $\mathcal{S}$

q: an unrooted tree topology on quartet r. We use $ab|cd$ to indicate that a and b are sisters. Three topologies are possible: $ab|cd$, $ac|bd$, and $ad|bc$.

$t|r$: the quartet tree topology obtained by restricting tree t to the four species of r. When $q = t|r$, we say that t *agrees* or is compatible with q.

$Q(t)$: the set of quartet trees induced by tree t; thus, $Q(t) = \bigcup_{r \in \mathcal{Q}} \{t|r\}$

$w_{\mathcal{G}}(q)$: the number of trees in $\mathcal{G}$ that agree with q.

$\mathcal{X}$: a set of bipartitions (see background Section **??**) on leaf-set $\mathcal{S}$; all bipartitions in $\mathcal{X}$ are complete (include all taxa in $\mathcal{S}$). Each subset of $\mathcal{S}$ is called a *cluster*, and a bipartition defines two clusters. Since bipartitions in $\mathcal{X}$ are complete, we can represent $\mathcal{X}$ as a set of clusters instead of bipartitions, and when we do so, we refer to it as $\mathcal{X}'$.

For any quartet of taxa, the quartet tree topology that has higher $w_{\mathcal{G}}$ than the two alternative topologies is called the *dominant* topology (breaking ties arbitrarily).

5.2.2 ASTRAL-I

5.2.2.1 Optimization problem

Given a set $\mathcal{G}$ of k binary input gene trees on n taxa, there is a multi-set of $k\binom{n}{4}$ quartet trees induced by trees in $\mathcal{G}$. We define the Weighted Quartet (WQ) score of a tree t with respect to $\mathcal{G}$ to be the number of quartet trees from this multi-set that t also induces. Thus,

$$WQ_{\mathcal{G}}(t) = \sum_{1}^{k} |Q(t) \bigcap Q(t_i)| \tag{5.1}$$

An equivalent definition is

$$WQ_{\mathcal{G}}(t) = \sum_{r \in \mathcal{Q}} w_{\mathcal{G}}(t|r) = \sum_{q \in Q(t)} w_{\mathcal{G}}(q)$$

.

We now define an optimization problem for maximizing WQ.

Weighted Quartet Consensus (WQC) problem:

- Input: a set $\mathcal{G}$ of unrooted gene trees
- Output: the tree topology $\hat{T}$ on $\mathcal{S}$ that maximizes $WQ_{\mathcal{G}}$; i.e., return $\hat{T}$ such that $WQ_{\mathcal{G}}(\hat{T}) \geq WQ_{\mathcal{G}}(T')$ for $T' \neq \hat{T}$.

The WQC optimization problem, also called the quartet consensus [21] or Maximum Quartet Support Species Tree (MQSST) [1] problem, is a specific case of the general weighted quartet problem (where $w(q)$ is defined arbitrarily and not with respect to $\mathcal{G}$), which is an NP-hard [22] problem. The complexity of WQC has not been established. If the input trees are allowed to have missing data, then they could all include four leaves; in this case, WQC would be NP-hard [22]. When all the gene trees are restricted to be complete (i.e., contain all the species), the complexity of WQC is an open problem to our knowledge, but we suspect it is also NP-hard.

To be able to cope with the computational complexity of this likely NP-hard problem, we introduce a constrained version of WQC.

Constrained Weighted Quartet Consensus (CWQC) problem:

- Input: a set $\mathcal{G}$ of unrooted gene trees, and a set $\mathcal{X}$ of bipartitions on $\mathcal{S}$.
- Output: the tree topology $\hat{T}$ on species set $\mathcal{S}$ that maximizes $WQ_{\mathcal{G}}$ and all its bipartitions are in $\mathcal{X}$ (equivalently, all its clusters are in $\mathcal{X}'$).

CWQC is a generalization of WQC; setting $\mathcal{X}'$ in CWQC to the power set (set of all possible subsets) of $\mathcal{S}$ would solve WQC. As we show in Theorem 5.2.8, CWQC can be solved in time polynomial in the size of $\mathcal{X}'$, k, and n, and ASTRAL uses a dynamic programing algorithm to solve the problem. An exact solution to the constrained problem gives a heuristic solution to the unconstrained problem. Therefore, we refer to a solution to the constrained problem as the heuristic version of ASTRAL, and a solution to the unconstrained version as the exact version. Various settings of $\mathcal{X}$ would give different heuristics, and would each correspond to a specific constraint on the search space.

A natural way to define $\mathcal{X}$ is using the input gene trees and adding all their bipartitions to the set. The motivation for setting $\mathcal{X}$ in this manner is that we hope each bipartition in the species tree would appear in at least one of the gene trees. This definition of $\mathcal{X}$ is used by default in ASTRAL-I, but we allow the user to add extra bipartitions to this set if desired (in, ASTRAL-II, we expand this set automatically). Besides the intuitive reasons for setting $\mathcal{X}$ to bipartitions in the gene trees, this definition enables us to prove theoretical guarantees of statistical consistency.

Theorem 5.2.2. *An exact solution to CWQC problem is a statistically consistent estimator of the species tree topology under the MSC model when true gene trees are used as input, as long as $\mathcal{X}$ includes at least all bipartitions from all the input gene trees, but perhaps also more bipartitions.*

Proof. Let T be the true species tree. As stated in Theorem 5.2.1, unrooted quartet trees do not have anomaly zones [20]. Therefore, as the number of gene trees increases, with probability that approaches 1, each quartet topology induced by the species tree will appear more frequently in $\mathcal{G}$ than either of the two alternative topologies. Therefore, for every quartet of taxa r and every possible tree T', with probability that approaches 1 as we increase the number of genes, $w_{\mathcal{G}}(T|r) \geq w_{\mathcal{G}}(T'|r)$. By extension, if $\mathcal{Q}$ is the set of all possible quartets of taxa, then

$$\sum_{r \in \mathcal{Q}} w_{\mathcal{G}}(T|r) \geq \sum_{r \in \mathcal{Q}} w_{\mathcal{G}}(T'|r)$$

and thus:

$$WQ_{\mathcal{G}}(T) \geq WQ_{\mathcal{G}}(T')$$

Thus, the optimization criterion in WQC attains its maximum value with the true species tree with probability that approaches 1. The assumption of having a binary species tree ensures that the dominant quartet tree has a frequency that is strictly higher than two alternatives, and therefore, in the limit, the optimization problem has a *unique* maximum value (note that $Q(T_1) = Q(T_2)$ iff $T_1 = T_2$ [23]). Thus, an exact solution to the WQC problem is statistically consistent.

The species tree topology has a non-zero probability of being observed among gene trees. Therefore, as the number of gene trees increases, with probability converging to 1, at least one of the gene trees will be topologically identical to the species tree T. Therefore, in the limit, the set $\mathcal{X}$ will contain all the bipartitions from T with probability approaching 1. Thus, a solution to CWQC is also statistically consistent as long as $\mathcal{X}$ includes all bipartitions from all gene trees. Note also that $\mathcal{X}$ may contain all the bipartitions from T even without having T among its gene trees, but we invoked the probability of observing T in $\mathcal{X}$ for ease of proof. □

We note that CWQC takes into account the relative frequency of all three alternative quartet topologies for all quartets of taxa, and weights them accordingly. Thus, if the dominant quartet topology is much more frequent than the alternatives, trees that don't induce the dominant topology are penalized, but if the three alternative quartet topologies all have frequencies close to 1/3, that quartet will contribute little to the optimization problem. This approach is in contrast to some other quartet-based methods such as the population tree from BUCKy [6] that first try to find the dominant quartet topologies and then summarize them. Estimation of the dominant quartet tree is susceptible to error (due to insufficient gene sampling and estimation error) and the CWQC accounts for this.

The WQC optimization problem could be expressed as finding a *median tree*, where instead of finding a species tree that maximizes the total number of quartet trees that it satisfies, we would seek a fully binary species tree that has a minimum total distance to the input gene trees, where the distance is the number of gene tree quartet trees that it *violates*. Then, Theorem 5.2.2 asserts that the median tree (under this definition) is a statistically consistent estimator of the species tree.

5.2.2.2 Dynamic programming

ASTRAL uses a dynamic programming (DP) approach to solve the CWQC optimization problem. Moreover, the fact that weights of quartet trees are defined according to their frequency in the gene trees and not arbitrarily enables us to optimize the WQ score without explicitly enumerating the set of all possible quartet trees. Thus, we solve CWQC problem without ever explicitly calculating the $3\binom{n}{4}$ values of the $w_{\mathcal{G}}$ function.

For a given unrooted binary tree t and four leaves $r = \{a, b, c, d\}$ in the tree, the induced subtree of t connecting the four leaves will have exactly two nodes x and y with degree three (Fig. 5.3). We say that the quartet tree $q = ab|cd$ on four taxa r is associated (or mapped) to a pair of nodes $\{x, y\}$

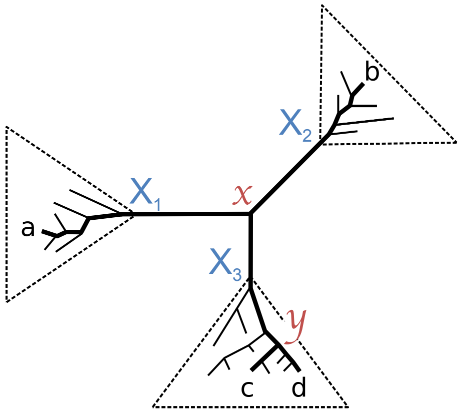


Figure 5.3: **Mapping a quartet tree to a tripartition**. Each node x in an unrooted tree defines a tripartition $(X_1|X_2|X_3)$ of the set of taxa and a tripartition defines a node. Each induced quartet tree $q = ab|cd$ maps to two nodes (x and y here). Node x is where the paths from a to c (or d) and b to c (or d) first join. Similarly, node y is where the paths from c to a and d to a first join.

in an unrooted binary tree t when q is compatible with t and x and y are the only two nodes that have a degree of three in $t|r$. We say that q is mapped to x from its ab side when a and b are on two different edges pending from x (similarly y is associated with the cd side of q).

Deleting x from a tree t separates it into three parts, X_1, X_2, and X_3, as shown in Figure 5.3; this is called a "tripartition", and is denoted $(X_1|X_2|X_3)$. Internal nodes of an unrooted tree and tripartitions are equivalent and we use them interchangeably. We call each part of a tripartition a "side" of the corresponding node.

For an internal node x, we can easily count the number of quartets that are associated with it. Recall that by definition, a quartet mapped to x

has two of its leaves pending from two different edges of x. Thus, to count the number of quartets mapped to x, we simply need to pick one of the three partitions of x (say X_1), and pick two leaves from it, and then pick one leaf from each of the remaining partitions, and do this for all ways of picking the first partition. Thus,

Corollary 5.2.3. *The number of quartet trees mapped to $x = (X_1|X_2|X_3)$, is*
$$F(x_1, x_2, x_3) = \binom{x_1}{2} x_2 x_3 + x_1 \binom{x_2}{2} x_3 + x_1 x_2 \binom{x_3}{2} = \frac{x_1 x_2 x_3 (x_1 + x_2 + x_3 - 3)}{2}$$
where x_1, x_2, and x_3 give the sizes of X_1, X_2, and X_3, respectively.

Recall that $q = ab|cd$ is mapped to x from the ab side when a and b belong to two different sides of x. Now, for two given tripartitions, x and y, we can derive how many quartets are mapped to both x and y *from the same side of the quartet.*

Lemma 5.2.4. *Let $x = (X_1|X_2|X_3)$ and $y = (Y_1|Y_2|Y_3)$ be two tripartitions on the same set of leaves $\mathcal{S}$. Let $\mathbf{C}$ be a 3×3 matrix with $\mathbf{C}_{ij} = |X_i \cap Y_j|$ for $i, j \in \{1, 2, 3\}$. The number of quartet trees mapped to both x and y from the same side of the quartet tree is:*

$$H(x, y) = H(\mathbf{C}) = \sum_{(a,b,c) \in G_3} F(\mathbf{C}_{1a}, \mathbf{C}_{2b}, \mathbf{C}_{3c}) \tag{5.2}$$

where G_3 gives the set of all permutations of $\{1, 2, 3\}$.

Proof. There are six bijections between the three parts of x and y. Take w.l.o.g. one of those bijections $(X_1 \to Y_1, X_2 \to Y_2, X_3 \to Y_3)$. If we find the intersection between all three partitions paired with each other, we get a tripartition $z = (X_1 \cap Y_1, X_2 \cap Y_2, X_3 \cap Y_3)$ on a subset of $\mathcal{S}$. We can use the equation from Corollary 5.2.3 to count the number of quartet trees mapped to z. This is the term inside the sum in Equation 5.2 and note that we are summing over all possible bijections. The quartet trees mapped to z are clearly mapped also to both x and y. Moreover, any quartet tree mapped to z maps to x and y on its same exact side (the side that belonged to two sides of z). Furthermore, a quartet tree that maps to both x and y but from different sides won't be counted because z will not include it. To see this, consider $x = (a|b|cd)$ and $y = (ab|c|d)$; the quartet tree $ab|cd$ is mapped to both x and y, but is mapped from the ab side to x and from the cd side to y. All six ways of calculating z using bijections between partitions of x and y will have at least one empty part, and thus, H will be zero here. Therefore, H counts only quartets that are mapped to both x and y form their same side. We now need to show that all such quartet trees are counted exactly once.

Take any quartet tree $q = ab|cd$ that is mapped to both x and y w.l.o.g. from the ab side. By definition, a and b belong to two sides of x and w.l.o.g. let $a \in X_1$, $b \in X_2$, and $c, d \in X_3$ and similarly, w.l.o.g. let $a \in Y_1$, $b \in Y_2$, and $c, d \in Y_3$. The bijection that produces $z = (Z_1 = X_1 \cap Y_1, Z_2 = X_2 \cap Y_2, Z_3 = X_3 \cap Y_3)$ has $a \in Z_1$, $b \in Z_2$, and $cd \in Z_3$; therefore F applied to this bijection will count q. Tripartitions z produced by all five remaining bijections will miss one of the four taxa, and therefore will not count q. The lemma follows. □

We now count the number of quartet trees that a tripartition x shares with a collection of input trees. Let

$$s_{\mathcal{G}}(x) = \sum_{t \in \mathcal{G}} |Q(t) \cap Q(x)|$$

where $Q(x)$ is the set of quartet trees mapped to x. Then,

Lemma 5.2.5. *For a tripartition x and a set of unrooted binary trees $\mathcal{G}$,*

$$s_{\mathcal{G}}(x) = \sum_{t \in \mathcal{G}} \sum_{y \in \mathcal{N}(t)} H(x, y) \tag{5.3}$$

where $\mathcal{N}(t)$ is the set of internal nodes in t and $H(x, y)$ is given in Equation 5.2.

Proof. The proof follows from the fact that by Lemma 5.2.4, each $H(x, y)$ term counts all quartet trees that are mapped to x and y if and only if they are mapped from the same side. Each quartet tree q in a gene tree t that is mapped to x will therefore be counted, and will be counted only once: when y is the node in the gene tree that has q mapped to it, and has q mapped to it from the same side as x. □

We now present a major result.

Theorem 5.2.6. *The $WQ_{\mathcal{G}}$ score of a species tree $\hat{T}$ can be computed as*

$$WQ_{\mathcal{G}}(\hat{T}) = \frac{1}{2} \sum_{x \in \mathcal{N}(\hat{T})} s_{\mathcal{G}}(x) \tag{5.4}$$

Proof. Recall that $WQ_{\mathcal{G}}$ score defined in Equation 5.1 counts the number of quartet trees induced both by the species tree and the set of gene trees. Each quartet tree in the species tree maps to two of its internal nodes. Thus, if we simply count the number of quartet trees in all gene trees that are mapped to any internal nodes of $\hat{T}$ and sum up these values, we will count each quartet tree shared between the species tree and the gene trees exactly twice. The $s_{\mathcal{G}}(x)$ term, by Lemma 5.2.5, counts exactly this quantity for a given node. Thus, we just need to sum $s_{\mathcal{G}}(x)$ values for all the internal nodes of $\hat{T}$, and divide the sum by two. The theorem follows. □

The ability to score a tripartition of the species tree in isolation from other tripartitions using the $s_{\mathcal{G}}(x)$ function allows us to use dynamic programming to maximize the $WQ_{\mathcal{G}}$ score. The dynamic programming starts from the set $\mathcal{S}$ and recursively divides it into smaller subsets, each time finding the division that maximizes the $WQ_{\mathcal{G}}$ score. Backtracking defines the subtree that maximizes the score and at the top level returns the tree that maximizes $WQ_{\mathcal{G}}$.

Recall that $\mathcal{X}'$ is the set of clusters from bipartitions in $\mathcal{X}$ (i.e., $A \in \mathcal{X}'$ iff the bipartition $(A|\mathcal{S} - A) \in \mathcal{X}$). We compute $V(A)$, which gives the score for an optimal subtree on $A \subset \mathcal{S}$, using the following dynamic programming.

ASTRAL DP algorithm:

- $|A| = 1$: $V(A) = 0$
- $A = \mathcal{S}$: $V(A) = V(A - \{a\})$ for an arbitrary $a \in \mathcal{S}$
- otherwise:

$$V(A) = \max_{A', A-A' \in \mathcal{X}'} \{V(A') + V(A - A') + \frac{1}{2} s_{\mathcal{G}}((A'|A - A'|\mathcal{S} - A))\} \quad (5.5)$$

Note that $s_{\mathcal{G}}$ is defined in Equation 5.3 and $(A'|A - A'|\mathcal{S} - A)$ defines a tripartition, which can be scored using $s_{\mathcal{G}}$.

The recursion in the dynamic programming finds a way of dividing each set A into A' and $A - A'$ (each of which must be in $\mathcal{X}'$) such that the number of quartets satisfied by an optimal rooted tree on A' and $A - A'$, in addition to those satisfied by the tripartition $(A'|A - A'|\mathcal{S} - A)$, is maximized. The boundary cases are singleton clusters; for these, we set $V(A) = 0$. Also note that for $A = \mathcal{S}$, the tripartition $(A'|A - A'|\mathcal{S} - A)$ will have an empty set in its third part, regardless of the choice of A'; therefore $s_{\mathcal{G}}(A'|A - A'|\mathcal{S} - A)$ will be zero for $A = \mathcal{S}$. Since any trivial bipartitions (where one side has only one taxon) has to be in the final species tree, setting A' to any arbitrarily chosen leaf at the top level would work. Each division of A to two parts creates two new bipartitions in the species tree: $(A'|\mathcal{S} - A')$ and $(A - A'|\mathcal{S} - (A - A'))$; note that both of these bipartitions are restricted to those found in the set $\mathcal{X}$.

Theorem 5.2.7. *The ASTRAL DP algorithm finds an optimal solution to the CWQC optimization problem.*

Proof. Let tree $\hat{T}$ be the tree obtained by backtracking the sequence of set divisions in the DP algorithm. The $V(\mathcal{S})$ score computed by the DP algorithm equals the right hand side of Equation 5.4 and by Theorem 5.2.6, it equals $WQ_{\mathcal{G}}(\hat{T})$ (i.e. the optimization score of the tree). To see this, note that the recursive formula simply produces the sum of $s_{\mathcal{G}}$ scores for all the internal nodes of $\hat{T}$. We therefore need to only show the dynamic programming maximizes $V(\mathcal{S})$. For each A, the dynamic programming recursively finds the maximum possible V among all resolutions of A in addition to the score for the node resulting from that resolution; thus, by induction on A, the dynamic programming maximizes V. The theorem follows. $\square$

5.2.2.3 Running time analysis

The score $s_{\mathcal{G}}(x)$ needs to be calculated for each tripartition of taxa visited in the dynamic programming. In ASTRAL-I, to compute $s_{\mathcal{G}}(x)$, we simply follow Equation 5.4. Thus, we sum over $O(nk)$ input gene tree nodes, and, for each node, we first calculate $\mathbf{C}$ and then compute $H(\mathbf{C})$ using Equation 5.2. We represent subsets of taxa as bitsets, which results in $O(n)$ running time for calculating $\mathbf{C}$; therefore, calculating each $s_{\mathcal{G}}(x)$ requires $O(n^2k)$ (we improve this in ASTRAL-II, as we will show). Note that our dynamic programming algorithm draws its clusters from the set $\mathcal{X}'$. Not all pairs of clusters in $\mathcal{X}$ can be put together, but for simplicity we assume they can; with this assumption, there are $O(|\mathcal{X}|^2)$ tripartitions that need to be scored. Thus,

Theorem 5.2.8. *ASTRAL-I runs in $O(n^2|\mathcal{X}|^2k)$ time, where n is the number of species and k is the number of gene trees.*

Note that this is a conservative running time analysis. The number of tripartitions scored is certainly lower than $|\mathcal{X}|^2$, and likely can be bounded with a lower exponent. Also, we do not need to calculate the score multiple times for the tripartitions that appear in multiple gene trees; we can compute the score once and simply multiply it by the number of times it appears. In practice, ASTRAL-I is really fast, as we will show.

We close by noting that our dynamic programming (DP) approach is similar to the algorithm used in [24] for constructing species trees from sets of gene trees, minimizing the total number of duplications and losses, and subsequently used to construct species trees minimizing deep coalescence [25]. We also note that Bryant and Steel give a dynamic programming for solving the general constrained weighted quartet problem (where weights are defined arbitrarily and not by the gene trees) [26]. Their dynamic programming also runs in polynomial time (with a n^4 term) and solves a constrained version of the problem where the bipartitions in the final tree are restricted to those coming from an input constraint set (analogous to $\mathcal{X}$). In our algorithm, we assume weights are the frequencies in the gene trees, and therefore, we can solve the problem without ever listing all $3\binom{n}{4}$ quartet topologies and their weights. Thus, we are able to achieve polynomial time running time with a lower exponent than n^4.

5.2.3 ASTRAL-II

We now describe how ASTRAL-II improves upon the older version. ASTRAL-II has three new features:

1. ASTRAL-II uses a faster algorithm to compute $s_{\mathcal{G}}(x)$.

2. ASTRAL-II searches a larger space by expanding the set $\mathcal{X}$ using heuristics.

3. ASTRAL-II can handle polytomies in its input gene trees.

Algorithm 5.1 - Weight calculation. Input is a gene tree set $\mathcal{G}$ and a tripartition $w = (X|Y|Z)$. Each part (e.g., X) is a bitset indexed by the species (thus, $X[i]$ is 1 if leaf i is in X and otherwise is 0). $H(\mathbf{C})$ is defined as in Eq. 5.2. Function WEIGHT computes $s_{\mathcal{G}}(x)$ defined in Eq. 5.3.

```
function WEIGHT($g, w = (X|Y|Z)$)
    for $t \in \mathcal{G}$ do
        $w \leftarrow 0$
        $S \leftarrow$ empty stack
        for $u \in postOrder(t)$ do
            if $u$ is a leaf then
                $(x, y, z) \leftarrow (X[u], Y[u], Z[u])$
            else
                $(\mathbf{C}_{11}, \mathbf{C}_{12}, \mathbf{C}_{13}) \leftarrow$ pull from $S$
                $(\mathbf{C}_{21}, \mathbf{C}_{22}, \mathbf{C}_{23}) \leftarrow$ pull from $S$
                $(x, y, z) \leftarrow (\mathbf{C}_{11} + \mathbf{C}_{21}, \mathbf{C}_{12} + \mathbf{C}_{22}, \mathbf{C}_{13} + \mathbf{C}_{23})$
                $(\mathbf{C}_{31}, \mathbf{C}_{32}, \mathbf{C}_{33}) \leftarrow (|X| - x, |Y| - y, |Z| - z)$
                $w \leftarrow w + H(\mathbf{C})$
            push $(x, y, z)$ to $S$
```

We motivate and discuss each feature in turn.

5.2.3.1 Running time improvement

Recall that ASTRAL-I computes $s_{\mathcal{G}}$ in $O(n^2k)$ time for each tripartition, by going over all $O(nk)$ input gene tree nodes, and, for each node, calculating H using Equation 5.2 in $O(n)$. In ASTRAL-II, instead of looking at all tripartitions in input gene trees, we use a post-order traverse of all gene trees (rooted arbitrarily) to calculate the score using Algorithm 5.1.

To score the input tripartition $w = (X|Y|Z)$, we traverse all the nodes of all gene trees. For each traversal node u, we compute a tuple (x, y, z), which gives the number of leaves under u that are shared with X, Y, and Z.

To do this for leaves, we simply need to find which side of w includes that leaf, which can be done in $O(1)$ if the tripartition is represented as three bitsets. For internal nodes, we can calculate (x, y, z) by simply summing up the same quantities already calculated for the two children of u, which also takes $O(1)$. The tuples from the two children of u in addition to $(|X| - x, |Y| - y, |Z| - z)$ give all the element of the 3×3 matrix $\mathbf{C}$ that gives the size of the intersection between all three sides of u and all three sides of w. Given $\mathbf{C}$, we simply need to calculate $H(\mathbf{C})$, which also takes $O(1)$. Thus, each inner-loop takes $O(1)$ and therefore, calculating $s_{\mathcal{G}}(w)$ for one tripartition requires $O(nk)$ running time. Thus,

Theorem 5.2.9. *ASTRAL-II runs in $O(nk|\mathcal{X}|^2)$ time, where n is the number of species, and k is the number of gene trees.*

5.2.3.2 Additions to $\mathcal{X}$

Theorem 5.2.2 established that the default way of setting the set $\mathcal{X}$ is statistically consistent. However, for a limited number of genes, as we will show in our results section, it is possible and sometimes likely that some of the bipartitions in the species tree do not appear in any of the gene trees. In ASTRAL-II, to account for this, we use a host of heuristic strategies to add extra bipartitions to the default set $\mathcal{X}$.

Similarity Matrix: For each pair of species a and b, we define

$$Q(\{a, b\}) = \{(ab|cd) : c, d \in \mathcal{S} - \{a, b\}\}$$

We now define a similarly measure between a pair of species:

$$s(a,b) = \sum_{t \in \mathcal{G}} |Q(\{a,b\}) \cap Q(t)|$$

Thus, the similarity between the two taxa is the number of quartet trees induced by gene trees where the pair appear on the same side of the quartet. This similarity matrix can be calculated using Algorithm 5.2. This algorithm traverses all nodes of all input gene trees (rooted arbitrarily), and for each node u, we look at all pairs of leaves chosen each from one of the children of u. For each such pair, we add $\binom{o}{2}$ to their similarity score, where o is the number of leaves *outside* the subtree below u. This will process each pair of nodes in each of the input k genes exactly once and would therefore require $O(n^2k)$ computations. The final score can be normalized by $|Q(\{a,b\})|$, the total number of quartet trees that include a and b on the same side. When input gene trees are complete, this normalization is not necessary and is not shown in Algorithm 5.2.

Once the similarity matrix is computed, we calculate an UPGMA tree and add all its bipartitions to the set $\mathcal{X}$. The UPGMA algorithm starts from

Algorithm 5.2 - Computing similarity matrix. *leafCount* gives the number of leaves under a node and is easily precomputed.

```
function GETSIMILARITY($\mathcal{G}$)
    $S \leftarrow Zeros(n \times n)$
    for $g \in \mathcal{G}$ and $u \in postOrder(g)$ do
        for $l \in Left(u)$ do
            for $r \in Right(u)$ do
                $S[l,r] = s[r,l] = s[r,l] + \binom{n-leafCount(u)}{2}$
```

Algorithm 5.3 Additions to $\mathcal{X}$ using greedy consensus. See descriptions of functions in Table 5.1. Constants are by default set to $THS = \{0, \frac{1}{100}, \frac{1}{50}, \frac{1}{20}, \frac{1}{10}, \frac{1}{4}, \frac{1}{3}\}$; $ITERS = 10$; $RWD = 2$; and $FRQ = LTH = \frac{1}{100}$.

```
function addByGreedy($\mathcal{G}, S$)
    for $t \in THS$ do
        $gc \leftarrow greedy(\mathcal{G}, t, False)$
        for $p \in polytomies(gc)$ do
            $updateX(upgma(S, start = clusters(p)))$
            $c \leftarrow 0$
            $itercount \leftarrow ITERS$
            while $c < itercount$ do
                $c \leftarrow c + 1$
                $sample \leftarrow randSample(p)$
                $gr \leftarrow greedy(\mathcal{G}|sample, 0, True)$
                if $updateX(resolve(p, gr)) \geq FRQ$ then
                    $itercount \leftarrow itercount + RWD$
                $updateX(resolve(p, upgma(S|sample)))$
                if $t \leq LTH$ and $c < ITERS$ then
                    for $s \in sample$ do
                        $ld \leftarrow pectinate(sortBy(S, sample, s)$
                        $updateX(resolve(p, ld))$
```

n singleton clusters, one per taxa, and in each step, combines the two clusters with the highest similarity. The similarity of two clusters is the average similarity between all pairs of leaves chosen each from one of the two clusters.

Greedy: The greedy consensus of a set of trees is obtained by starting from a star tree and adding bipartitions from input trees in the decreasing order of their frequency if they don't conflict with previous bipartitions. This process ends when no remaining bipartition has frequency above a given threshold, or when the tree is fully resolved. We use greedy consensus of gene trees to

Table 5.1: **Functions used in Algorithm 5.3**.

Function	**Description**
$polytomies(t)$	For a given unrooted tree t, return all nodes with degree $d > 3$.
$greedy(\mathcal{G}, t, b)$	Finds bipartitions in all input trees in $\mathcal{G}$ and for each bipartitions notes its frequency. Sorts bipartitions by the descending order of frequency (with arbitrary tiebreakers) and discards those with frequency below t. Starts with a fully unresolved tree (i.e., the star tree), and adds bipartitions one at a time according to the order; if a bipartition conflicts with the tree, ignores it. At the end, if b is true, any remaining polytomies in the tree are randomly resolved. The branches (i.e., bipartitions) in the resulting tree are labelled by their bipartition frequency (i.e., their frequency in trees in $\mathcal{G}$).
$updateX(t)$	Adds all bipartition from t to the set $\mathcal{X}$ and notes which bipartitions are new. When edges in t have a frequency label (e.g., labels generated by the *greedy* function), *updateX* returns the maximum label of any *new* bipartition added to $\mathcal{X}$.
$clusters(p)$	An unrooted node p with degree d divides taxa into d subsets (Fig. 5.4). This function returns the partitions defined by p.
$upgma(S, C)$	Runs UPGMA using similarity matrix S on n taxa. By default, starts from n singleton clusters, one per taxa, and in each step, combines the two clusters with highest similarity. The similarity of two clusters is the average similarity between all pairs of leaves chosen each from one of the two clusters. When a set of clusters C is given, instead of starting with n singletons, starts by C.
$randSample(p)$	Selects a random leaf from each partition around node p.
$resolve(p, t)$	The input p is a node in an unrooted tree with leaf set L, and t is an unrooted tree on $L' \subset L$ such that L' contains exactly one leaf from each partition defined by p. Note that the tree t will be compatible with the tree that includes p. Every bipartition in t defines a further resolution of p. This function resolves p according to t and returns the results.
$pectinate(O)$	Given an ordered list of taxa O, it returns a pectinate tree based on O; e.g., $pectinate(a, d, e, c, b) = (a, (d, (e, (c, b))))$.
$sortBy(S, l, x)$	Sorts a list of taxa l based on their decreasing similarity to x and according to the similarity matrix S.

compute and add further bipartitions to $\mathcal{X}$, using Algorithm 5.3.

Algorithm 5.3 estimates the greedy consensus of the gene trees with various thresholds (THS). For each polytomy in each of these greedy consensus trees, it resolves the polytomy in multiple ways and adds bipartitions implied by those resolutions to the set $\mathcal{X}$ (if they don't already exist).

1. We resolve the polytomy by applying UPGMA to the similarity matrix; however, unlike the normal UPGMA algorithm that starts from singleton clusters, here, we start from clusters defined by each side of the polytomy.

2. We sample one leaf from each side of the polytomy randomly, and use the greedy consensus of the gene trees restricted to this subsample to find a resolution of the polytomy (randomly resolving remaining polytomies). We repeat this process at least 10 times, but if the subsampled greedy consensus trees include new bipartitions that are sufficiently frequent ($\geq 1\%$), we do more rounds of random sampling (we increase the number of iterations by two).

3. For each random subsample around a polytomy, we also resolve it by calculating an UPGMA tree on the similarity matrix restricted to the set of subsampled species.

4. For the two first greedy threshold values in THS and only for the first 10 random subsamples, we also use a third strategy that can potentially add a larger number of bipartitions: for each subsampled taxon a, we

resolve the polytomy as a pectinate tree (see Table 5.1) by sorting the remaining taxa according to their similarity with a (in decreasing order).

Gene tree polytomies: When gene trees include polytomies, we also add new bipartitions to $\mathcal{X}$. We first compute the greedy consensus of the input gene trees with threshold 0, and if the greedy consensus has polytomies, we resolve them using UPGMA; we repeat this process twice to account for random tie-breakers in the greedy consensus estimation. Then, for each gene tree polytomy, we use the two resolved greedy consensus trees to infer a resolution of the polytomy, and we add the implied bipartitions to $\mathcal{X}$.

Incomplete gene trees: The optimization problem used in ASTRAL can easily handle incomplete gene trees; i.e., gene trees where some of the leaves are not present. If $m < n$ quartets are present in a gene, it would contribute $\binom{m}{4}$ quartets to the WQ score defined in Equation 5.1. It is easy to show that if patterns of missing data are unbiased, the exact version of ASTRAL remains statistically consistent under gene trees that are incomplete. The challenging part of handling inputs with missing data is ensuring that the set $\mathcal{X}$ will include usable bipartitions.

When an input gene tree has missing data, at least one of its two parts (but possibly both parts) would not be in the complete gene tree, and therefore the inclusion of that part in $\mathcal{X}'$ is unlikely to be helpful (recall that $\mathcal{X}'$ is the set of all parts from all bipartitions in $\mathcal{X}$). When dealing with incomplete gene trees, we need to complete their bipartitions before adding them to $\mathcal{X}$. In ASTRAL-II, we use a heuristic approach to complete incomplete gene trees, and add bipartitions from the completed gene trees to $\mathcal{X}$. Note that this does not affect the scoring function, and only impacts the search space.

We use the similarity matrix computed in Algorithm 5.2 for adding missing taxa into incomplete trees. To ensure that the similarity matrix is not affected by arbitrary patterns of missing data in the gene trees, we need to also normalize the similarity values. As noted before, the normalization factor for each pair of leaves can simply be the number of quartets in all input gene trees that include the two taxa:

$$m(a,b) = \sum_{1}^{k} \binom{n_i - 2}{2} I_i(a,b)$$

where n_i is the number of leaves in gene tree g_i and $I_i(a,b) = 1$ if $\{a,b\} \subset g_i$ and otherwise $I_i(a,b) = 0$.

Given the similarity matrix, we add each missing taxon to each gene tree using an application of the four point condition [27]. When a distance matrix d is defined based on pairwise distances of leaves of a binary tree (i.e., with strictly positive branch lengths), for any quartet of taxa r, if the tree induces the quartet topology $q = ab|cd$, we have:

$$d(a,b) + d(c,d) < d(a,d) + d(b,c) = d(a,c) + d(b,d)$$

This inequality is called the four point condition.

We assume our similarity matrix (which can be converted to a distance matrix) uniquely defines a tree (i.e., is additive [28]). If all incomplete gene trees were identical topologically, our distance matrix would become additive as the number of genes increased. In the presence of discordance no such guarantees can be made, but we use this matrix anyway as a heuristic and note that our algorithm can be used with any similarity (or distance) matrix. We use Algorithm 5.4 to add missing leaves to the incomplete trees.

Algorithm 5.4 -Completing incomplete gene trees. Adds missing taxon m to tree t using similarity matrix S according to the four point condition. $arbLeaf(x)$ choses an arbitrary leaf under node x (by default, the left-most child). $addChild(x, y)$ adds y as a child of x.

```
function PLACE($t, S, m$)
    $closest \leftarrow \text{argmin}_{i \neq m} S[i, m]$
    $reroot(t, closest)$
    $u \leftarrow child(closest)$
    while $true$ do
        if $isLeaf(u)$ then
            $n \leftarrow Parent(u)$
            break
        $(l, r) \leftarrow (left(u), right(u))$
        $(lc, rc) \leftarrow (arbLeaf(lc), arbLeaf(rc))$
        $betterSide \leftarrow fourPoint(S, m, closest, lc, rc)$
        if $betterSide = closest$ then
            break
        else if $betterSide = lc$ then
            $u \leftarrow l$
        else if $betterSide = rc$ then
            $u \leftarrow r$
    $addAsChild(u, m)$
function FOURPOINT($S, m, a, b, c$)
    $as \leftarrow S[m, a] + S[c, b] - (S[m, c] + S[a, b])$
    $bs \leftarrow S[m, b] + S[a, c] - (S[m, a] + S[b, c])$
    $cs \leftarrow S[m, c] + S[b, a] - (S[m, b] + S[c, a])$
    $max \leftarrow \max(as, bs, cs)$
    return $c$ if $max = c$ else $b$ if $max = b$ else $a$
```

This heuristic algorithm first finds the taxon that has the highest similarity to the missing taxon m; it then roots the tree at this closest species c and traverses the nodes of the tree from root to the leaves. At each traversal point u, it decides whether it should move further down to the left (l) or the right (r) child of the current node u (we are assuming binary input genes, but extensions are straight forward), or if it should place the taxon at the branch above the current node. It arbitrarily chooses two leaves lc and rc under l and r (by default we choose the left most leaf). It places the taxon at the current branch iff m is closer to c than it is to either lc or rc according to the four point condition. If m is closer to one of the two arbitrarily chosen nodes, say lc, it choses that child of u, say l, as the next traversal nodes. Note that for each taxon x and any other three taxa, we can answer which of the three is closer to x by examining the four point conditions for all three possible topologies and finding which four point condition is closer to holding true (i.e., has a lower residual).

5.2.3.3 Multifurcating input gene trees

Although true gene trees are assumed to be binary, estimated gene trees can include polytomies. For example, some ML programs such as FastTree produce polytomies when several leaves have identical sequences. In maximum parsimony estimation of gene trees, if there are multiple trees with equal scores, a consensus of the trees is typically used, which can also result in polytomies. Most importantly, when bootstrapping (or other approach for obtaining branch support) are used, one can collapse low support branches in the gene trees, with the hope that impacts of gene tree estimation error are reduced [25, 29].

Extending ASTRAL to inputs that include polytomies requires solving the weighted quartet tree problem when each node of the input defines not a tripartition, but a multi-partition of the set of taxa. We start by a basic observation: every *resolved* quartet tree induced by a gene tree maps to two nodes in the gene tree *regardless* of whether the gene tree is binary or not (Fig. 5.4). In other words, induced quartet trees that map to only one node of the gene tree are *unresolved.*

When maximizing the quartet support, these unresolved gene tree quartet trees are inconsequential and need to be ignored. Now, consider a polytomy of degree d, which divides the set of taxa into d parts. There are $\binom{d}{3}$ ways to select three parts around the polytomy, and each of these defines a tripartition. Any selection of two taxa from one part of this tripartition and one taxon from each of the remaining two parts induces a resolved quartet tree, and each resolved quartet tree maps to exactly two nodes in our multifurcating tree. Thus, all the algorithmic assumptions of ASTRAL remain intact, as long as for each degree d node in an input gene tree, we treat it as a collection of $\binom{d}{3}$ tripartitions. Thus, to score a species tree tripartition $x = (X_1|X_2|X_3)$ with respect to a gene tree multi-partition $y = Y_1|\dots|Y_d$, we let $\mathbf{C}_{ij} = |X_i \cap Y_j|$ for all $i \in \{1, 2, 3\}$ and $j \in \{1, \dots, d\}$, and we generalize Equation 5.2 to:

$$H(x, y) = H(\mathbf{C}) = \sum_{(a,b,c) \in P_3} F(\mathbf{C}_{1a}, \mathbf{C}_{2b}, \mathbf{C}_{3c}) \tag{5.6}$$

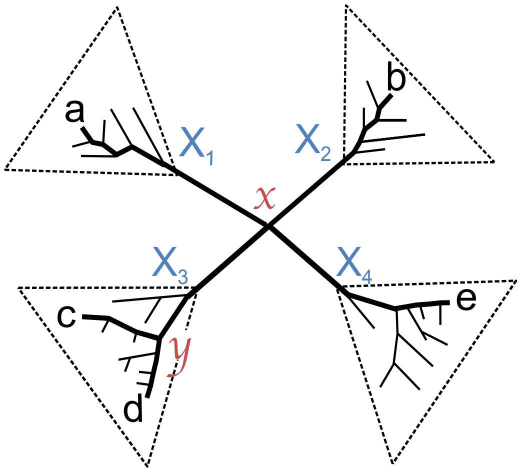


Figure 5.4: **Multipartitions in unrooted gene trees**. A polytomy divides the set of taxa into more than three parts (here, $d = 4$). A quartet tree mapped to two nodes (e.g., $ab|cd$) is a resolved quartet topology and needs to be counted towards WQ. A quartet tree mapped to only one node (e.g., $ab|ce$) is an unresolved quartet, and does not contribute to WQ; these need to be ignored. By treating the polytomy as a collection of $\binom{d}{3}$ tripartitions (in this case, $X_1|X_2|X_3$, $X_1|X_2|X_4$, $X_1|X_3|X_4$, and $X_2|X_3|X_4$), we ensure that all resolved quartet trees are counted and all unresolved quartet trees are left out. For example, here, $ab|ce$ would not be counted in our collection of $\binom{d}{3}$ tripartitions since each of its taxa are on a different part.

where P_3 is the set of all ordered subsets of size 3 from $\{1, \dots, d\}$.

Extending Algorithm 5.1 to compute Equation 5.6 is straightforward. The leaves are treated the same. For internal nodes, instead of popping two values from the stack, $d - 1$ values are popped and are summed to calculate the tuple for the traversal node. All $\binom{d}{3}$ ways of choosing three subsets around that polytomy are then iterated over and H values are summed.

In the presence of polytomies, the running time analysis can change because analyzing each polytomy requires time cubic in its degree and the degree can increase with n. It is not hard to see that the worst case is when all gene trees have a polytomy with $d = \frac{n}{2}$ and each side of each polytomy has two leaves; in this case, Algorithm 5.1 would require $\binom{\frac{n}{2}}{3}$ calculations, which requires $O(n^3)$ running time; thus, the running time of ASTRAL-II is $O(n^3k|\mathcal{X}|^2)$ instead of $O(nk|\mathcal{X}|^2)$ in presence of polytomies.

5.3 Evaluation of ASTRAL-I on simulated data

5.3.1 Experimental setup

We evaluate ASTRAL-I on a collection of simulated datasets. Our simulation procedure is similar to what was used in Chapter 4. Simulated data are generated under the GTR+MSM model by first simulating gene trees down a species tree according to MSM and then simulating sequence data down each gene tree according to GTR. Gene trees are then estimated form the sequence data, and species trees are estimated from the gene trees using various summary methods. We also run concatenation under maximum likelihood (CA-ML) on the sequence data. The accuracy of the estimated species tree is evaluated against the model true species tree using the Robinson-Foulds (RF) [30] rate; because all species trees estimated here are completely bifurcating, this is the same as the missing branch rate (proportion of internal edges in the model tree missing in the estimated tree).

5.3.1.1 Datasets

100-taxon simulated datasets. These data were generated by Yang and Warnow [7]; we briefly describe the simulation process and direct the reader to the original publication [7] for details. The 100-taxon model species tree was created by a birth-death process, and 25 genes were evolved within the species tree under the MSC, producing ultrametric gene trees. Nucleotide sequences with 1000 sites were evolved down each gene tree under a process with GTR+Γ substitutions as well as insertions and deletions, using ROSE [31]. True alignments were used to generate estimated gene trees using RAxML.

37-taxon "mammalian" simulated datasets. We use the same mammalian simulated dataset used for evaluating statistical binning; Chapter 4 gives details of the simulation procedure, which we summarize here.

We simulated this collection of datasets based on a 37-taxon mammalian dataset with 447 genes studied in [32]. First, we used MP-EST to estimate a species tree on the biological dataset from [32], and used it as a model species tree, with branch lengths in coalescent units. We evolved gene trees down the model tree under the MSC model using Dendropy [33], and then rescaled the gene trees to deviate from the molecular clock and produce branch length patterns observed in the biological dataset. We then evolved sequences with 500 and 1000 sites down each gene tree under the GTR model of site evolution, using GTR parameters estimated on the biological dataset. This produces the "default" model condition that has the amount of ILS estimated for this dataset by MP-EST. We varied this protocol by scaling the model species tree branch lengths up (2X and 5X) or down (0.2X and 0.5X) to modify the amount of ILS; longer branch lengths reduces ILS, and shorter branch lengths increases ILS. The default model tree conditions (including the number of genes, sequence length distribution, and amount of ILS) were set to produce a dataset called the "mixed condition" that most resembled the biological dataset.

The average bootstrap support (BS) in the biological data was 71%, and so we generated sequence lengths that produced estimated gene trees with BS values bracketing that value – 500bp alignments produced estimated gene trees with 63% average bootstrap support and 1000bp alignments produced estimated gene trees with 79% BS. The "mixed dataset" of 400 genes was produced using 200 genes with 63% BS and 200 genes with 79% BS, and had average BS of 71% - like the biological data.

We vary ILS levels, the number of genes, and sequence length. We go up to 3,200 genes for the most challenging conditions with 0.2X branch lengths (thus, very high ILS). For each model condition (specified by the ILS level, the number of genes, and the sequence length), we created 20 replicates, except for the 1600- and 3200-gene model conditions where we created 10 and five replicates respectively. We used RAxML to estimate gene trees on the simulated sequence alignments, and we generated 200 ML bootstrap replicates for the mixed dataset.

5.3.1.2 Methods

We compare ASTRAL-I with MP-EST [3], BUCKy-pop (the population tree from BUCKy [6]), MRP (a supertree method [34]), the Greedy Consensus, and CA-ML computed by RAxML. Of these six methods, three are statistically consistent summary methods, two are inconsistent summary methods, and CA-ML is also inconsistent Note that BUCKy takes into account gene tree uncertainty and other methods don't [35].

For 100-taxon datasets and the mixed mammalian datasets, we ran summary methods using three different procedures: using maximum likelihood gene trees as input (bestML), using all bootstrap replicates of all genes as input (All BS), and using the site-only multi-locus bootstrapping (MLBS) procedure [36], described in Chapter 4. For MLBS, we used the greedy consensus of 200 replicate species trees, each computed on an input consisting of one bootstrap replicate tree per gene. BUCKy-pop takes as input distributions of gene trees, and its authors intended a Bayesian distribution to be the input; following results from Yang and Warnow [7], we approximate the distribution using bootstrap gene trees which are less computational intensive to generate and have resulted in the same accuracy as Bayesian trees in some analyses [7]; thus, BUCKy-pop is run with a procedure analogous to All BS. In subsequent analyses, where we study the impact of various model parameters, we only study the bestML approach. Exact commands and versions used are given in Appendix A.1.1.

5.3.2 Simulation results

5.3.2.1 Results on mammalian simulated datasets

We address the following three research questions on the mammalian simulated dataset, in three separate experiments.

RQ1: Given a choice of the gene tree input type (bestML, MLBS, or All BS), which of the six methods produces the best accuracy under the default mixed condition?

RQ2: How is relative performance of methods affected by the number of genes, levels of ILS, and gene tree error?

RQ3: How do summary methods compare under the highest levels of ILS if the number of genes is allowed to increase?

We now describe the results obtained for each question and finish by discussing the running time of ASTRAL-I in comparison to other methods.

RQ1: Figure 5.5 shows results on the mixed mammalian dataset, comparing all six methods and three types of inputs to summary methods (bestML, MLBS, and All BS). For MRP, MP-EST and ASTRAL-I, using bestML input trees produced more accurate species trees than using bootstrap replicates, either as one input (All BS) or using MLBS. The purpose of using bootstrap replicates is to take gene tree uncertainty (resulting from insufficient sequence length, for example) into account; the fact bestML gene trees had the best accuracy indicates that for this model condition, using bootstrapping does not alleviate the gene tree estimation problem. However, it is possible that other model conditions or other ways of addressing gene tree uncertainty might show some advantage over the bestML approach. For example, we have found in other studies that with few genes, the accuracy of the MLBS approach tends to be higher than the bestML approach, but as the number of genes increases, bestML becomes better [10]. Nevertheless, in this study we are not seeing any improvements form the use bootstrapped gene trees. Therefore, we use bestML input trees in the remaining experiments in this chapter (see [10] for more comparisons of using bestML or bootstrapped gene trees).

For the mixed model condition and using bestML trees, ASTRAL-I is the most accurate of these methods, MP-EST the next most accurate, followed by the other summary methods, and finally by CA-ML. ASTRAL-I with any of the three sets of inputs is also more accurate than BUCKy-pop; however, differences between ASTRAL-I on All BS and BUCKy-pop are relatively small.

RQ2: We now explore variants of the basic mammalian simulation, exploring the impact of changes to the number of genes, gene sequence length, and the ILS level (by scaling the species tree branch lengths) on the absolute and relative performance of various methods using bestML input. We first fix the ILS to the default 1X and vary both the number of genes and the sequence length. We then fix the number of genes to 200, and sequence length to 500bp, and vary the amount of ILS, in both cases also showing results on

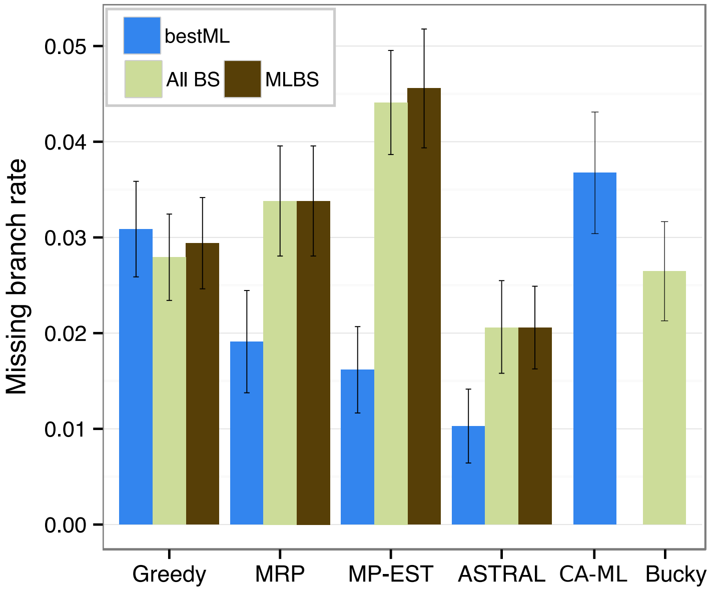


Figure 5.5: **Species tree estimation error on the default mixed mammalian datasets.**. This dataset has 200 genes with 500bp and 200 genes with 1000bp, which results in 71% mean BS. We show the missing branch rates for estimated species trees computed using summary methods (MRP, MP-EST, greedy, BUCKy-pop, and ASTRAL-I) as well as concatenation using RAxML. Results are shown for running summary methods on maximum likelihood gene trees (bestML) and on the set of all bootstrap replicates from all genes (All BS), as well as the greedy consensus of running summary methods on individual bootstrap replicates from all genes (MLBS). CA-ML is run on the true alignment. Average and standard error shown based on 20 replicates.

true (simulated) gene trees. Figure 5.6 shows results for this experiment for all these model conditions. General trends as we changed parameters were as expected: all summary methods gave improved accuracy as the sequence length in each gene increased from 500bp to 1000bp; using true gene trees gave the best results; species tree error rates generally reduced as the number of genes increased; and species tree error rates increased as ILS levels increased.

ASTRAL-I was commonly more accurate than all the other summary methods we studied. ASTRAL-I was never outperformed by other summary methods; however, for a few cases, ASTRAL-I and one or more summary methods had identical accuracy. For example, on 800 true gene trees from default ILS levels, all summary methods (except for Greedy) produced the true species tree. We performed an ANOVA test comparing the species tree accuracy differences between ASTRAL and MP-EST, with the amount of ILS, number of genes, and the sequence length as independent variables. ASTRAL was significantly better than MP-EST ($p < 10^{-5}$) and the relative accuracy of ASTRAL and MP-EST depended only on the amount of ILS ($p = 0.008$), but not the number of genes ($p = 0.8$) or gene sequence length ($p = 0.3$).

Comparison of ASTRAL-I and CA-ML was interesting. ASTRAL-I was more accurate than CA-ML in general ($p < 10^{-5}$ according to an ANOVA test); however, the relative performance depended significantly on the level of ILS ($p < 10^{-5}$). With reduced ILS, CA-ML had better accuracy than all summary methods, including ASTRAL, but as the level of ILS increased, ASTRAL became more accurate.

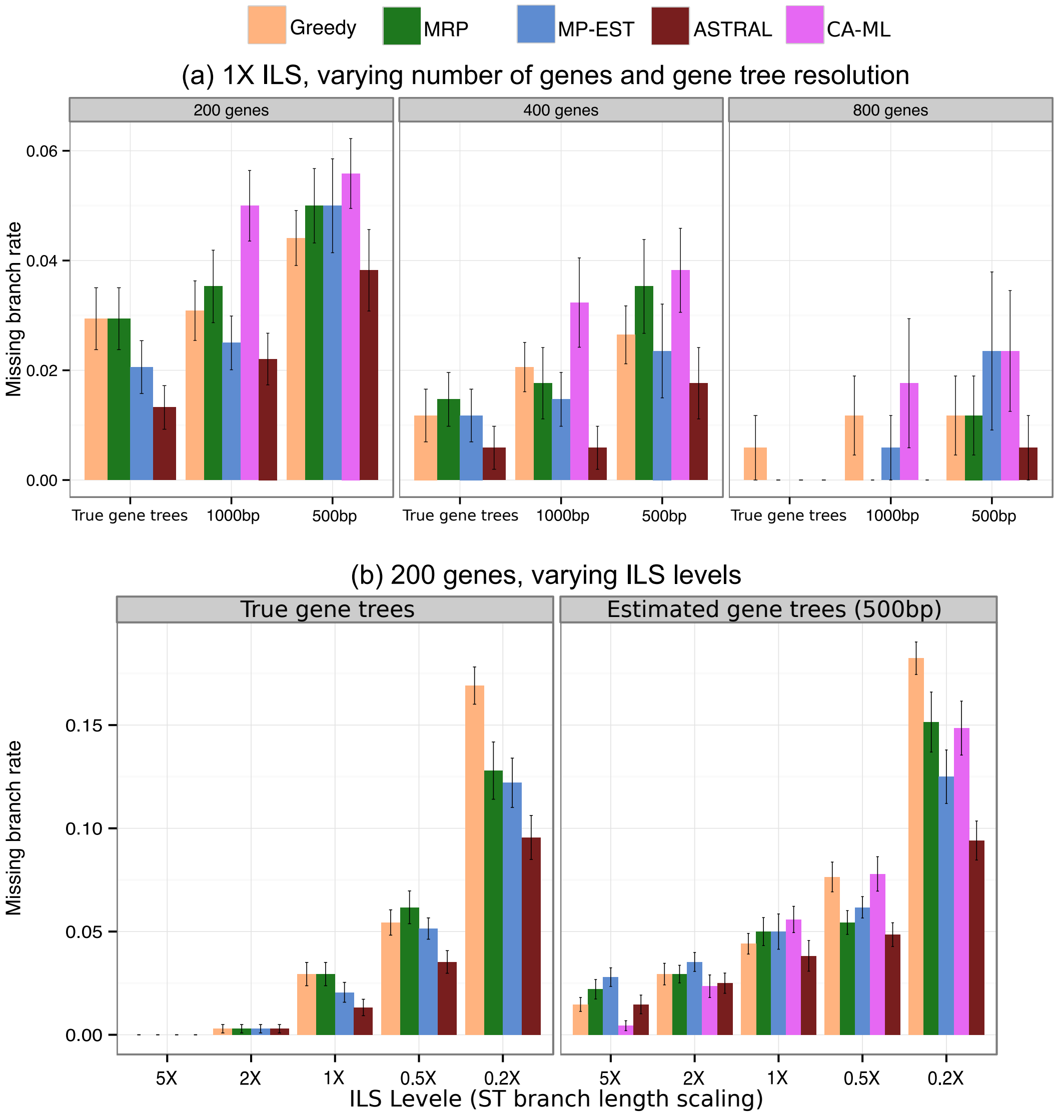


Figure 5.6: **Species tree estimation error on the simulated mammalian datasets, varying simulation parameters**. We show the missing branch rates for estimated species trees computed using summary methods (MRP, MP-EST, greedy, and ASTRAL-I) as well as CA-ML. Summary methods are run on RAxML bestML gene trees and true gene trees, and CA-ML is run using RAxML. (a) Default levels of ILS, varying the number of genes and gene tree resolution; (b) 200 genes, varying the amount of ILS from very low (5X species tree branch lengths) to very high (0.2X species tree branch lengths).

RQ3: For the most challenging ILS level, where with 200 genes the error was still high for all methods including ASTRAL-I, we asked whether increasing the number of genes reduces the error, as expected by the statistical consistency of ASTRAL-I. Figure 5.7 shows results for the case where fix the ILS level to 0.2X (very high) and increase the number of genes up to 3,200. As we increase the number of genes, the error reduces for all summary methods, except for the greedy consensus. With 3,200 gene trees, ASTRAL-I has 0.5% error, with true gene trees, and only 1.5% error with estimated trees. Thus, even with the most challenging ILS scenarios, with increased number of genes, high accuracy can be obtained. MP-EST also has reduced error with increased number of genes, but is always less accurate than ASTRAL-I. For example, the error of MP-EST with 1600 true gene trees is 4.1%, which is exactly the same as the error of ASTRAL-I with 800 genes, but with 1,600 true gene trees, ASTRAL-I has 2.0% error.

Running time. We examine running times under moderate ILS, gene sequences of length 500bp, and with 400 and 800 genes and with bestML input trees (except for BUCKy-pop). BUCKy-pop strictly runs in serial, using a Bayesian MCMC technique, which can take a long time and substantial memory to reach convergence. On the 37-taxon mammalian simulated datasets, BUCKy-pop ran to completion for datasets with up to 400 genes (where it took approximately 5 hours), but failed to complete (due to memory issues) on the 800-gene dataset.

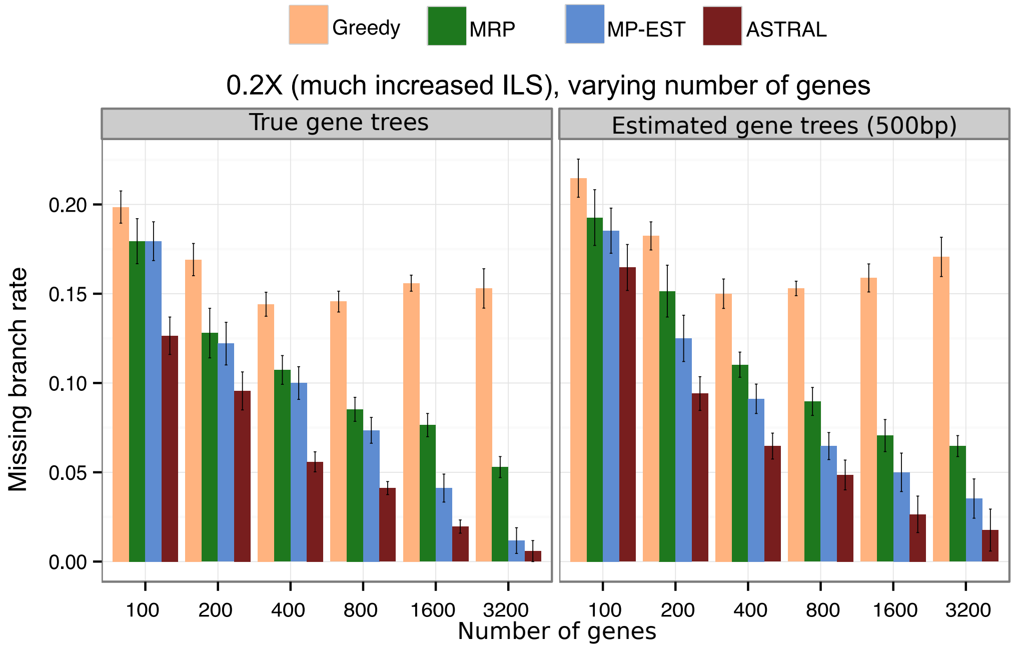


Figure 5.7: **Species tree estimation error on the simulated mammalian datasets with highest level of ILS**. We show the missing branch rates for estimated species trees computed using summary methods (MRP, MP-EST, greedy, and ASTRAL-I) run on RAxML bestML gene trees and true gene trees. ILS levels are fixed to 0.2X (very high) and the number of genes is increased to 3200.

MP-EST completed relatively quickly - about 100 minutes - for both the 400-gene and 800-gene datasets. We ran MP-EST with 10 random starting points, so this time could be reduced by using just one starting point, but with a potential decrease in accuracy.

ASTRAL-I completed in 3.3 seconds on the 400-gene dataset, and in 5.3 seconds on the 800-gene dataset. Thus, ASTRAL-I is dramatically faster than the other methods, and able to run on these moderately large datasets in extremely short time frames. However, BUCKy is used with 200 bootstrapped gene trees for each gene, and outputs support values. Running ASTRAL-I and MP-EST using MLBS to obtains support values would increase their running times if run in serial, but ASTRAL-I would still be much faster than BUCKy (e.g., 11 minutes on the 400-gene dataset rather than 5 hours). In addition, parallelizing MLBS is trivial since each bootstrap replicate is independent.

Finally, Figure 5.8 shows how the running time of ASTRAL-I is impacted by the number of genes and the level of ILS. The running time of ASTRAL-I increases as the level of ILS is increased, because the set $\mathcal{X}$ is populated with more bipartitions when gene trees have high levels of ILS. As the number of genes are increased, the number of unique bipartitions in input gene trees increases, which increases the time required to calculate the score function w, and also the size of the set $\mathcal{X}$ is likely to increase. Thus, both factors impact the running time, but even under the most challenging conditions (3200 genes of 0.2X ILS level), ASTRAL-I finished in about two hours on the mammalian dataset.

5.3.2.2 100-taxon dataset

We evaluated the feasibility of using ASTRAL-I on datasets with large numbers of taxa using the 100-taxon simulated datasets, with 25 genes and 10 replicates. Because there is no single outgroup, the estimated trees are not rooted, and so we could not use MP-EST. ASTRAL-I had no difficulty analyzing these data (completing in under one second). ASTRAL-I had average

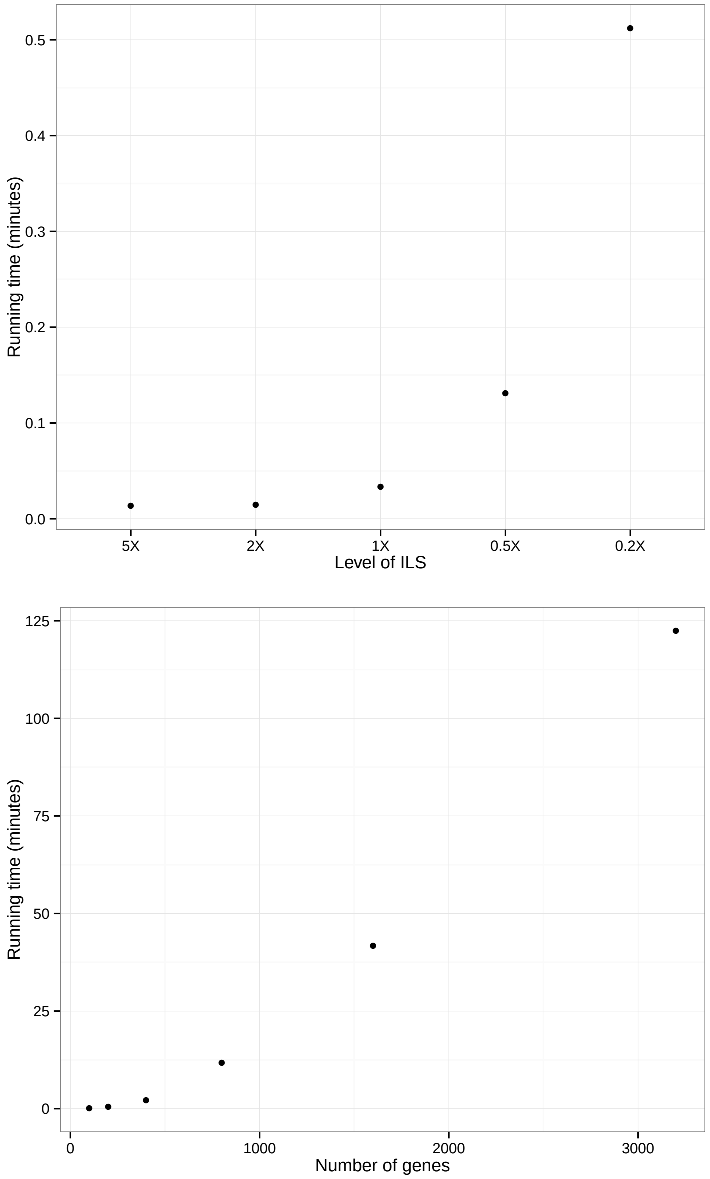


Figure 5.8: **Running time of ASTRAL**. We show the running time for default ASTRAL on the mammals simulated datasets with (top) varying levels of ILS with 200 genes of 500bp resolution, and (bottom) varying number of true gene trees with much increased ILS level (0.2X).

Table 5.2: **Results on 100-taxon dataset**. Average FN rates (over 20 replicates) of different methods on the 100-taxon 25-gene simulated datasets. The dataset does not have an outgroup, and therefore, we could not run MP-EST on it. Gene trees and CA-ML are estimated using RAxML.

Method	bestML	All BS
CA-ML	0.057	
ASTRAL	0.061	**0.052**
Greedy	0.064	0.056
MRP	0.064	0.055

missing branch rate of 6.1%, better than MRP and Greedy (6.4%), but not as good as CA-ML (5.7%); differences are not statistically significant ($p > 0.1$; paired Wilcoxon test).

5.3.3 Summary of results

In our study, ASTRAL-I was more accurate than MP-EST and BUCKy-pop, two leading coalescent-based methods, and improved or matched the accuracy of concatenation under maximum likelihood under many conditions, except when the amount of ILS was very low, where concatenation was more accurate. This study also showed that concatenation could be more accurate than coalescent-based estimation, provided that the amount of ILS is low enough. However, the best coalescent-based methods can be more accurate than concatenation under biologically realistic conditions.

Using bootstrap replicate gene trees instead of best ML gene trees did not improve species tree estimation accuracy on the simulated mixed mammalian dataset – and in fact made species tree estimations less accurate for MRP, MP-EST and ASTRAL-I. Similar results have been observed by others when taking gene tree estimation error into account [37]. This suggests the possibility that the topological error in bootstrap gene trees is large enough to offset any improvement in species tree estimation obtained by taking gene tree uncertainty into account. However, it is possible that an improvement might be obtained under other conditions, or that using a sample of gene trees estimated by a Bayesian MCMC analysis might be better suited to coalescent-based species tree estimation methods than maximum likelihood bootstrap trees, as suggested by [12] (although see [7]).

5.4 Evaluation of ASTRAL-II on simulated data

Our experiments on ASTRAL-I were all using relatively small datasets; we had either few species and large numbers of genes, or moderately large numbers of species and few gene trees. Here, we report the result of a more extensive simulation study that shows under certain conditions ASTRAL-I can have reduced accuracy because of the restrictions imposed by the default setting of the set $\mathcal{X}$. We show that ASTRAL-II addresses these problems, and we demonstrate that ASTRAL-II can run on datasets with up to 1000 genes and 1000 species in about a day.

5.4.1 Experimental setup

5.4.1.1 Dataset

We used SimPhy [38] to simulate species trees and gene trees under MSC and to generate gene trees in mutation units, and then used Indelible [39] to simulate nucleotide sequences down the gene trees according to GTR with varying length and model parameters. We estimated gene trees on these simulated gene alignments, which we then used as input to ASTRAL-I, ASTRAL-II, NJst [8], and MP-EST, in addition to concatenation.

We used SimPhy to simulate species trees according to the Yule process, characterized by the number of taxa, maximum tree length, and the speciation rate (this combination defines a model condition). We simulated 11 model conditions, which we divide into two datasets, with one model condition appearing in both datasets.

Dataset I: In 6 model conditions (forming Dataset I), we fixed the number of taxa to 200 and varied tree length (500K, 2M, and 10M generations), and speciation rates (1e-6, and 1e-7 per generation). The tree length impacts the amount of ILS, with lower length resulting in shorter branches, and therefore higher levels of ILS (Fig. 5.9). Speciation rate impacts whether speciation events tend to happen close to the tips (1e-06) or close to the base (1e-07). Different tree shapes (i.e., combinations of tree length and speciation rate) produce different levels of ILS starting from relatively low and going up to very high. The 10M/1e-06 condition had 0% to 20% distance between true gene trees and the species tree, measured by the RF distance, whereas 500K length (with 1e-06 or 1e-07 rate) had between 60% and 80% RF distance (Fig. 5.9). Thus, the 500K length has the highest ILS levels and 10M has the lowest, and 2M is in between.

Dataset II In six model conditions (forming Dataset II), we fixed the tree shape to 2M/1e-06 (medium ILS levels) and set the number of taxa to 10, 50, 100, 200, 500, and 1000. The amount of ILS only slightly increased as we increased the number of species (Fig. 5.9). Note that the model condition with 200 taxa and the 2M/1e-6 tree shape appears in both datasets.

For each model condition, we simulated 50 species trees, forming 50 replicates. On each species tree, 1000 gene trees were simulated according to the MSC model with the population size fixed to 200,000 (a reasonable value for vertebrates). SimPhy uses various rate parameters and rate heterogeneity modifiers to convert gene tree branch lengths to mutation units, introducing deviations from molecular clock and rate heterogeneity between genes. Parameters for these simulations are given in Appendix A.2.1.

We simulated indel-free gene alignments using Indelible [39] under the GTR+Γ model. First, for each replicate, two parameters, μ and σ, were drawn uniformly from $(5.7, 7.3)$ and $(0, 0.3)$ respectively. Then, the sequence length for each gene in that replicate was drawn from a log-normal distribution with μ and σ parameters (thus, average sequence length is uniformly distributed between 300bp and 1500bp). GTR+Γ parameters were drawn from a Dirich-

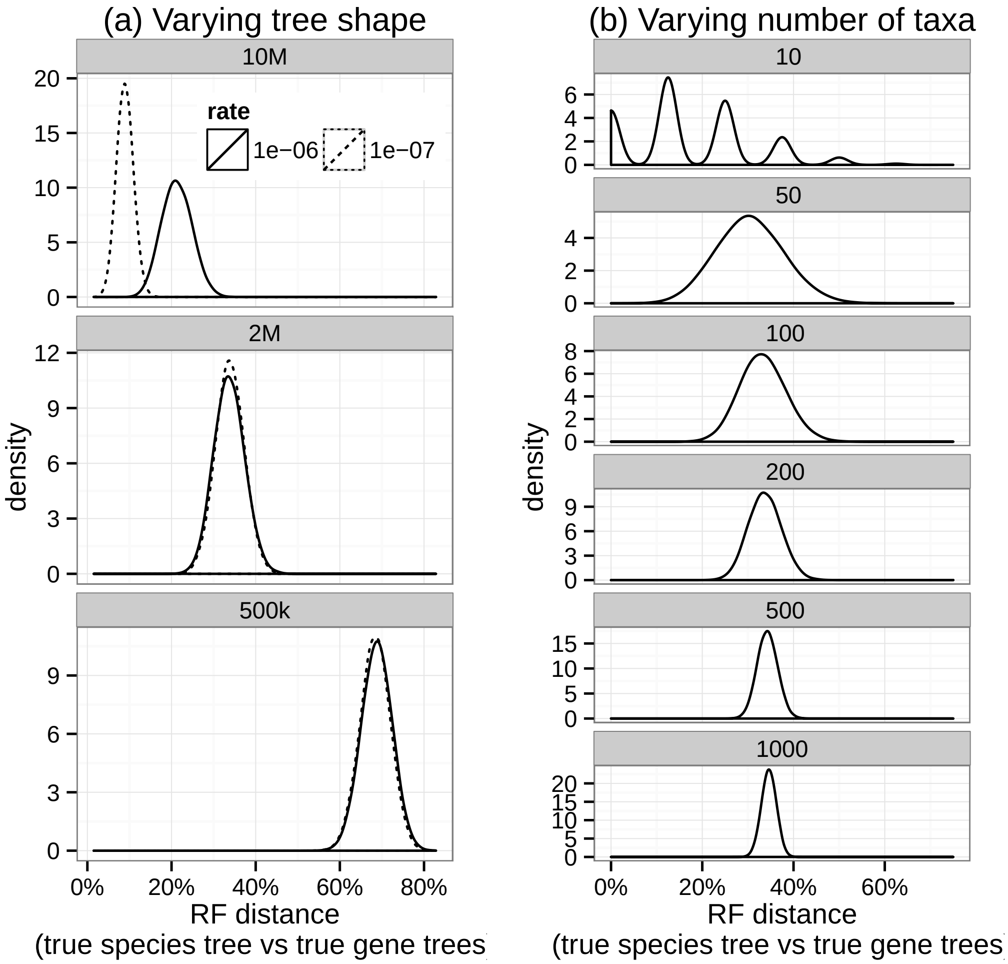


Figure 5.9: **ILS levels in ASTRAL-II simulation data**. RF distance between the true species tree and the true gene trees (50 replicates of 1000 genes) for (a) Dataset I and (b) Dataset II. Tree height directly affects the amount of true discordance; the speciation rate affects true gene tree discordance only with 10M tree length. The number of taxa has a modest effect on the amount of ILS.

let(36,26,28,32) distribution; we estimated the Dirichlet parameters from a collection of biological datasets using ML (see Appendix A.2.2 for details).

5.4.1.2 Methods

Gene tree estimation: Previous studies [40] have shown that FastTree-II [41] is generally as accurate at estimating the tree topology as more extensive ML heuristics such as RAxML [42], while being much faster. In our simulation studies, we used FastTree to estimate the 550,000 gene trees ranging from 10 to 1000 species. Our estimated gene trees had wide-ranging levels of gene tree estimation error (see Figure 5.10). The tree error was impacted by tree shape parameters; as expected, more ILS and deeper speciation lead to higher levels of gene tree error. Moreover, average gene tree estimation error varied across replicates, and gene tree error varied considerably among the 1000 genes in each replicate (Fig. 5.10). The number of taxa had only a small impact on gene tree estimation error.

FastTree outputs polytomies when sequence alignments cannot distinguish between competing tree resolutions. We removed any gene tree where more than 50% of the internal nodes were polytomies because they would not add much new information but would increase the running time of ASTRAL (and would be randomly resolved for other methods). This pruning left fewer than 500 genes for 9 out of 550 replicates in some model conditions: 200-taxon/500K/1e-06 (3 replicates), 50-taxon (3 replicates), 100-taxon (2 replicates), and 10-taxon (1 replicate). We removed these 9 replicates.

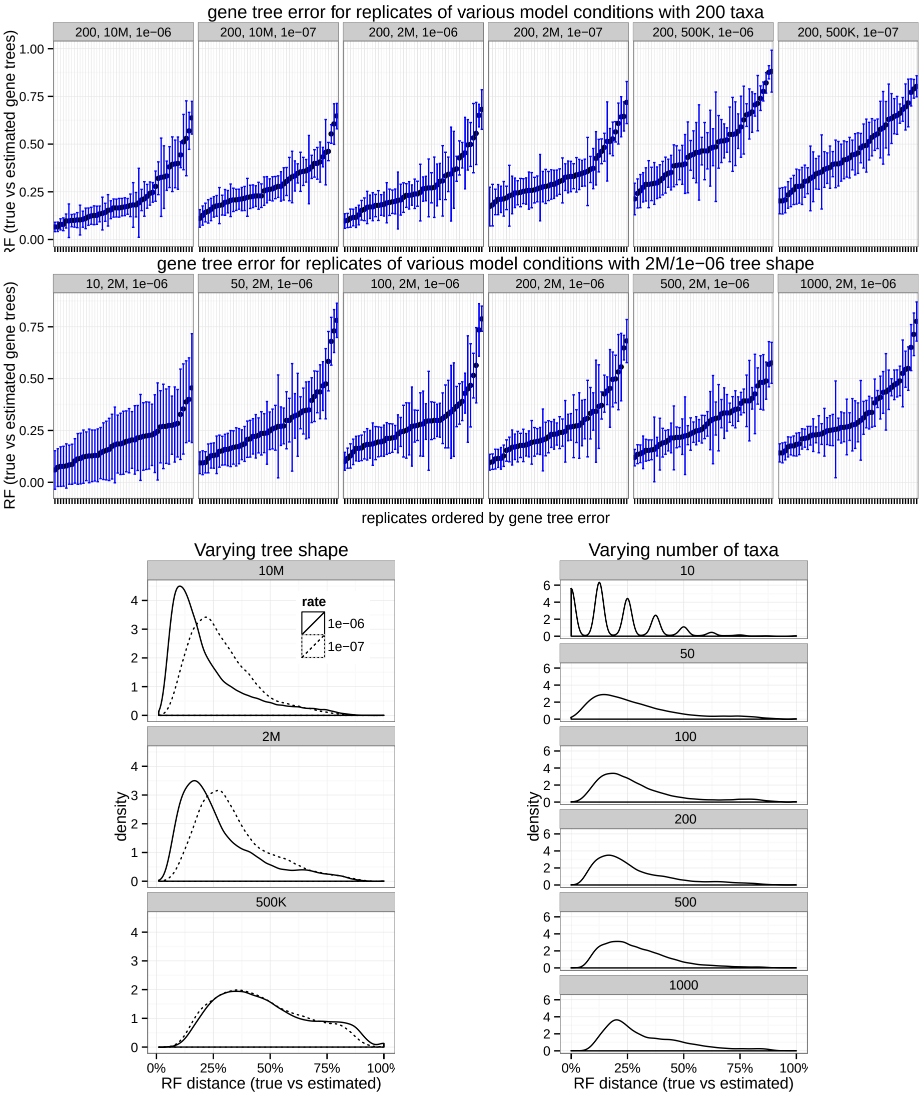


Figure 5.10: **Gene tree estimation error in simulated ASTRAL-II datasets**. Many parameters (e.g. alignment length, gene tree length, and substitution rates) were varied in a heterogeneous way to simulate 50 replicates per model condition with varying gene tree estimation error. Top: each box (box title: number of taxa, height, rate) shows averages and standard deviations of gene tree estimation error (across 1000 genes) for each replicate. Note wide variations in gene tree error across and within replicates. Bottom: both tree height and rate (left) affect gene tree estimation error; more ILS and deeper speciation result in higher error rates. With fixed tree shape (2M, 1e-06), changing the number of taxa (right) has little impact on the gene tree estimation error.

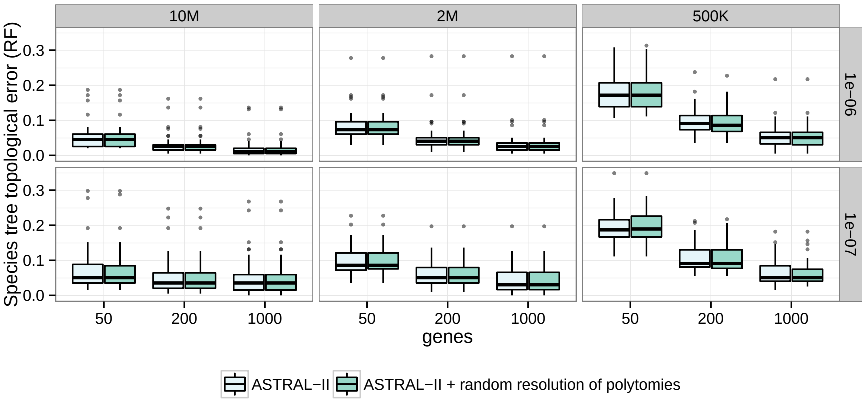


Figure 5.11: **Impact of polytomies**. Comparison of ASTRAL-II run on estimated gene trees with polytomies output by FastTree and with random resolutions of polytomies. Results are shown for dataset-I.

Species tree methods: We compared ASTRAL-I only to ASTRAL-II, and after establishing the improvements obtained in ASTRAL-II, we focused on the new version and compared it to MP-EST, NJst and CA-ML run using FastTree. We ran all methods given a maximum of 4 days of running time and 24GB of memory. MP-EST only finished for datasets with at most 100 taxa within time limits. Because of its running time, we ran MP-EST once (one random seed number) for each analysis. NJst, ASTRAL-I and MP-EST could not handle polytomies; therefore, we randomly resolved polytomies in inputs of these methods. We also ran ASTRAL-II on gene trees with randomly resolved polytomies and observed no differences with ASTRAL-II run on gene trees with polytomies (Fig. 5.11). Thus, differences between ASTRAL-II and other methods were not due to the random resolutions of polytomies.

5.4.1.3 Evaluation criteria

We evaluate methods in terms of species tree error and we also evaluate running time for coalescent-based methods. Species tree error is measured using the standard normalized RF distance. Running time of summary methods gives the wall clock running time and is measured on a heterogeneous Condor cluster at the University of Texas, Computer Science department.

5.4.2 Simulation results

We start by comparing ASTRAL-II with ASTRAL-I in terms of accuracy and running time (RQ1). We next focus on ASTRAL-II and compare it to other coalescent-based methods (RQ2) and then compare it to CA-ML (RQ3). This question leads us to a more in depth analysis of the effects of gene tree estimation error on the accuracy of various methods (RQ4). Finally, we evaluate the impact of collapsing low support branches in input gene trees on the accuracy of ASTRAL-II (RQ5).

5.4.2.1 RQ1: ASTRAL-I versus ASTRAL-II

Search space: ASTRAL-II adds extra bipartitions to the search space, which allows it to explore a larger search space; this tends to increase the accuracy of ASTRAL-II over ASTRAL-I. In our simulations, the extent of the improvement depended on the model condition. Table 5.3 shows the improvements obtained by ASTRAL-II compared to ASTRAL-I, and Figures 5.12 and 5.13 compare the two methods in terms of accuracy for Datasets I and II. In

Dataset I, with the lowest level of ILS or with the medium ILS level and recent speciation, ASTRAL-I and ASTRAL-II both had extremely low error (Fig. 5.12) and no substantial improvements were detected by the addition of extra bipartitions (Table 5.3). With 2M length and deep speciation, ASTRAL-II improved upon ASTRAL-I substantially, with improvements ranging from 3.5% with 1000 genes to 10.1% with 50 genes. Most dramatic differences were observed on the high ILS conditions, where ASTRAL-I performed extremely poorly, but ASTRAL-II reduced the error by about 40% (Table 5.3). Results on Dataset II showed that the effect of adding extra bipartitions also depended on the number of taxa in expected ways (Table 5.3): ASTRAL-I was as accurate as ASTRAL-II for up to 200 taxa, but with 500 taxa or more, ASTRAL-II had a substantial advantage (as large as 9%). As expected, the advantage of ASTRAL-II was larger with few genes and reduced with more genes.

The improvements obtained by ASTRAL-II are due to additions to the search space. We therefore asked whether the heuristic approaches used to add bipartitions to set $\mathcal{X}$ are sufficient, or improvements could be obtained by further expanding $\mathcal{X}$. To answer this question, we tested the impact of adding all the bipartitions from the species tree to the set $\mathcal{X}$, and compared ASTRAL-II with and without these extra bipartitions (see Figs. 5.12 and 5.13). We saw no significant differences between ASTRAL-II with and without these potentially new bipartitions (p=0.77 according to a two-way ANOVA test), indicating that the accuracy of ASTRAL-II is very unlikely to be improved further by expanding the search space.

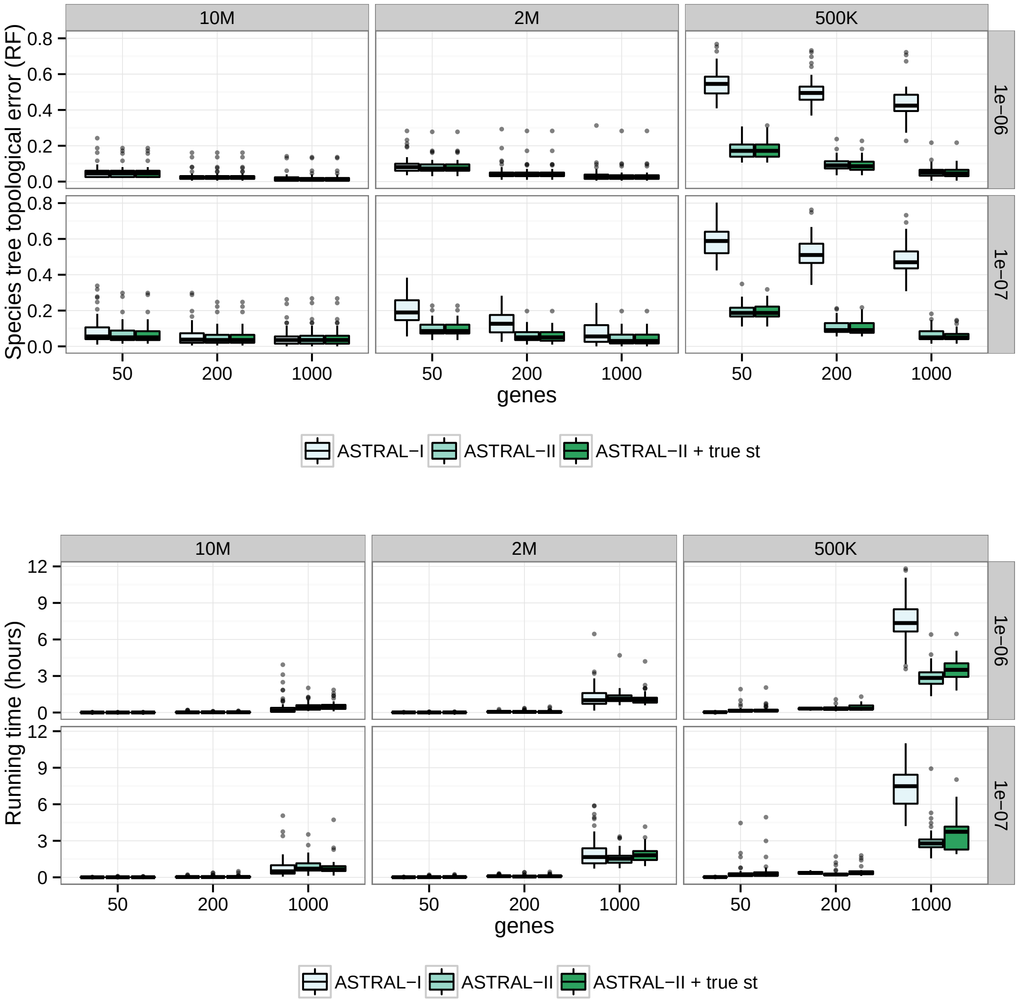


Figure 5.12: **Comparison of ASTRAL-I and ASTRAL-II on Dataset-I**. Species tree error (top) and running times (bottom) are shown. "ASTRAL-II + true st" shows the case where the true species tree is added to the search space; this is included to approximate an ideal solution (e.g. exact) where the set $\mathcal{X}$ includes all bipartitions that lead to the optimal score.

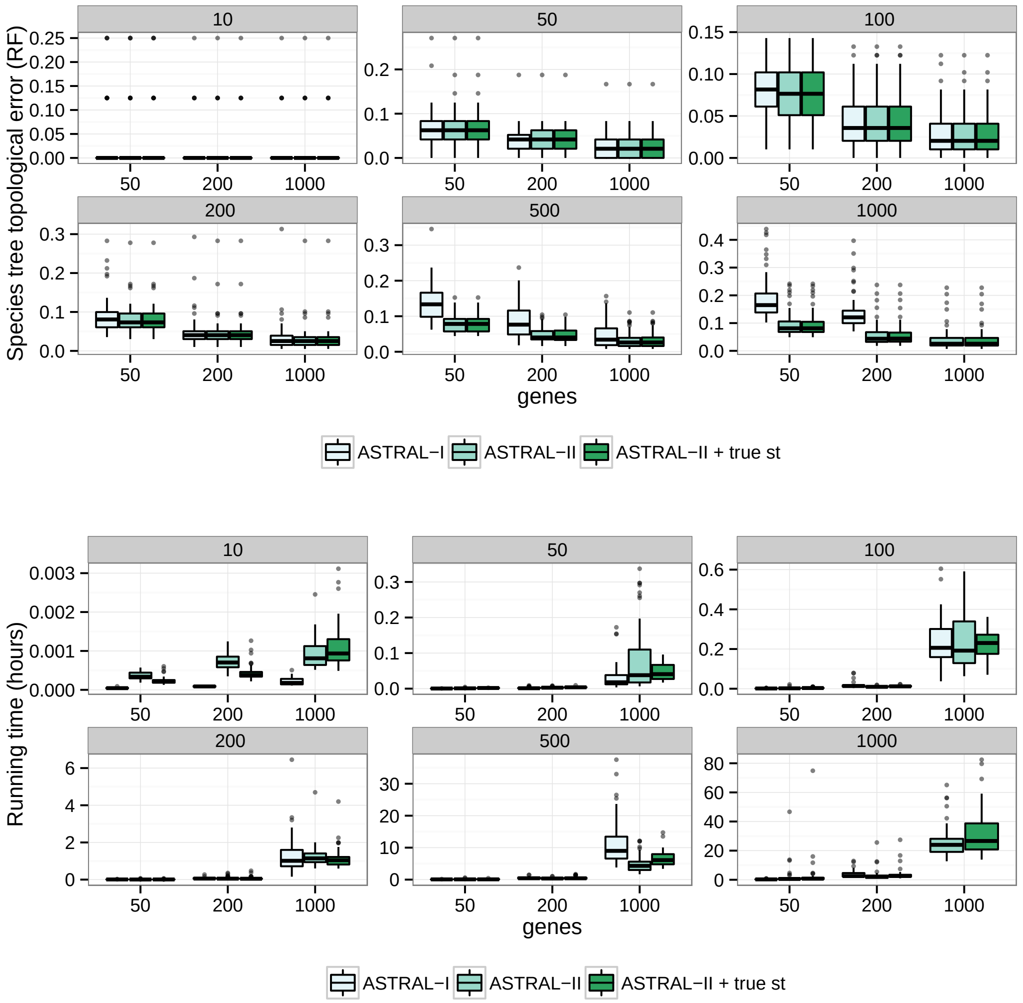


Figure 5.13: **Comparison of ASTRAL-I and ASTRAL-II on Dataset-II**. Species tree error (top) and running times (bottom) are shown. "ASTRAL-II + true st" shows the case where the true species tree is added to the search space; this is included to approximate an ideal solution (e.g. exact) where the set $\mathcal{X}$ includes all bipartitions that lead to the optimal score.

Table 5.3: **Reductions in species tree error obtained by ASTRAL-II compared to ASTRAL-I**. We report results using the difference in RF percentage; values above 0.0% indicate ASTRAL-II is more accurate.

Dataset I [200 taxa, varying tree shape (columns) and number of genes (rows)]

	10e-6 (recent)			10e-7 (deep)		
	10M	2M	500K	10M	2M	500K
50	0.2±0.2	0.7±0.3	37.9±1.0	1.7±0.6	10.1±0.9	38.7±0.9
200	0.0±0.1	0.2±0.1	41.0±1.1	0.7±0.3	7.4±0.7	41.4±1.0
1000	0.0±0.0	0.2±0.1	39.2±1.2	0.0±0.0	3.5±0.7	41.4±1.1

Dataset II [2M/1e-6 shape, varying the number of taxa (columns) and genes (rows)]

	10	50	100	200	500	1000
50	0.3±0.3	0.0±0.1	0.3±0.2	0.7±0.3	6.0±0.6	9.3±0.6
200	0.0±0.0	0.0±0.0	0.0±0.0	0.2±0.09	3.9±0.5	8.3±0.5
1000	0.0±0.0	0.1±0.1	0.0±0.0	0.2±0.08	1.7±0.4	

Running time: With 200 taxa and lower levels of ILS, ASTRAL-I and ASTRAL-II had similar running times (Fig. 5.12), but ASTRAL-II was faster with increased ILS (3 versus 7.5 hours of median run time). The improvement in speed is noteworthy, given that ASTRAL-II searches a larger tree space than ASTRAL-I. With small numbers of taxa, the two versions had close running times, but as the number of taxa increased, the running time of ASTRAL-II increased more slowly (Fig. 5.13). For 500 taxa, ASTRAL-II was twice as fast as ASTRAL-I (a median of 5 versus 10 hours), while ASTRAL-I did not complete on 1000 taxa and 1000 genes.

5.4.2.2 RQ2: ASTRAL-II vs. other summary methods

Completion within time constraints: ASTRAL-II completed on all model conditions, MP-EST completed only on datasets with at most 100 taxa, and NJst completed on all model conditions except for the condition with 1000 genes and 1000 taxa.

Dataset I: ASTRAL-II was more accurate than NJst in all model conditions, except 1e-07/500K where the two methods had identical error (Table 5.4, Fig. 5.14). Overall, the differences between ASTRAL-II and NJst were statistically significant ($p < 10^{-5}$), according to a two-way ANOVA test, and the relative performance of the methods was significantly impacted by the speciation rate ($p = 0.026$) but not by the number of genes or tree length. ASTRAL-II was faster than NJst, in some cases by an order of magnitude (Fig. 5.15).

Dataset II: On 10-taxon datasets all methods had high accuracy (Table 5.12). On 50- and 100-taxon datasets, MP-EST was able to finish, but it was the least accurate of all the methods. ASTRAL-II was more accurate than NJst for all conditions except for 50 taxa with 50 genes (Table 5.12); however, differences were generally small when the number of taxa was 200 or less, and more substantial with more taxa. Overall, differences between ASTRAL-II and NJst were significant ($p = 0.0007$) and were significantly impacted by the number of taxa ($p = 0.0004$) but not the number of genes. ASTRAL-II was also faster than NJst, especially with more genes and more taxa (Fig. 5.15).

Table 5.4: **Species tree error on Dataset I of ASTRAL-II analyses**. We show average and standard error of RF percentage. ASTRAL-II is always more accurate than NJst, but CA-ML (using FastTree) is sometimes more accurate than ASTRAL. For each row, the lowest average error and those error values that have an overlapping standard error with the lowest error value are in bold.

rate	height	genes	ASTRAL-II	NJst	CA-ML
1e-06	10M	50	**5.2±0.5**	**5.6±0.6**	**5.4±0.3**
1e-06	10M	200	**3.1±0.4**	**3.4±0.5**	**3.1±0.3**
1e-06	10M	1000	**2.0±0.4**	2.3±0.5	**1.4±0.2**
1e-06	2M	50	**8.4±0.6**	**9.1±0.7**	**9.2±0.4**
1e-06	2M	200	**5.0±0.6**	**5.6±0.6**	**5.5±0.5**
1e-06	2M	1000	**3.4±0.6**	3.9±0.6	**2.8±0.4**
1e-06	500K	50	**17.6±0.7**	20.9±0.7	27.9±0.7
1e-06	500K	200	**9.6±0.5**	11.0±0.5	16.2±0.7
1e-06	500K	1000	**5.3±0.5**	**5.7±0.4**	8.0±0.3
1e-07	10M	50	7.3±0.9	10.2±1.0	**4.0±0.4**
1e-07	10M	200	5.4±0.7	8.2±1.0	**2.2±0.3**
1e-07	10M	1000	5.0±0.8	8.0±1.0	**1.8±0.3**
1e-07	2M	50	**10.2±0.6**	11.7±0.7	**10.3±0.3**
1e-07	2M	200	**6.0±0.5**	7.5±0.7	**5.7±0.3**
1e-07	2M	1000	4.4±0.6	6.0±0.7	**2.8±0.2**
1e-07	500K	50	**19.3±0.7**	22.5±0.6	28.2±0.6
1e-07	500K	200	**10.7±0.6**	**11.4±0.5**	16.1±0.7
1e-07	500K	1000	**6.3±0.5**	**6.3±0.5**	8.0±0.4

Table 5.5: **Species tree error on Dataset II. of ASTRAL-II analyses**. We show average and standard error of RF percentage. Note that ASTRAL-II is always more accurate than MP-EST, and more accurate than NJst under all conditions except one (50 taxa and 50 genes), where NJst is slightly more accurate (7.2% vs. 7.3%). CA-ML (using FastTree) is also less accurate than ASTRAL, except for 100 taxon and 200 or 1000 genes, where the two methods differ in less than 0.5%. For each row, the lowest average error and those error values that have an overlapping standard error with the lowest error value are in bold.

taxa	genes	ASTRAL-II	NJst	CA-ML	MP-EST
10	50	**2.8±1.0**	**2.8±1.0**	3.8±0.9	**2.8±1.0**
10	200	**1.5±0.7**	**1.5±0.7**	**1.8±0.7**	**1.8±0.7**
10	1000	**1.5±0.7**	**1.8±0.7**	**2.1±0.8**	**1.5±0.7**
50	50	**7.3±0.7**	**7.2±0.6**	**7.8±0.6**	13.5±1.7
50	200	**4.2±0.5**	**4.4±0.5**	**4.5±0.4**	9.1±1.5
50	1000	**2.6±0.4**	**2.7±0.5**	**2.7±0.4**	8.2±1.5
100	50	**7.9±0.5**	**8.7±0.5**	9.1±0.4	16.9±1.3
100	200	**4.8±0.5**	**5.1±0.6**	**4.7±0.4**	13.7±1.5
100	1000	**3.0±0.4**	3.9±0.6	**2.5±0.3**	14.1±1.55
200	50	**8.4±0.6**	**9.1±0.7**	**9.2±0.4**	
200	200	**5.0±0.6**	**5.6±0.6**	**5.5±0.5**	
200	1000	**3.4±0.6**	3.9±0.6	**2.8±0.4**	
500	50	**8.0±0.4**	9.7±0.5	9.2±0.3	
500	200	**4.9±0.3**	6.1±0.5	**4.7±0.2**	
500	1000	3.3±0.4	4.7±0.5	**2.3±0.1**	
1000	50	**9.9±0.7**	12.1±0.9	**9.8±0.3**	
1000	200	**6.0±0.7**	7.9±0.9	**5.1±0.2**	
1000	1000	**4.5±0.7**			

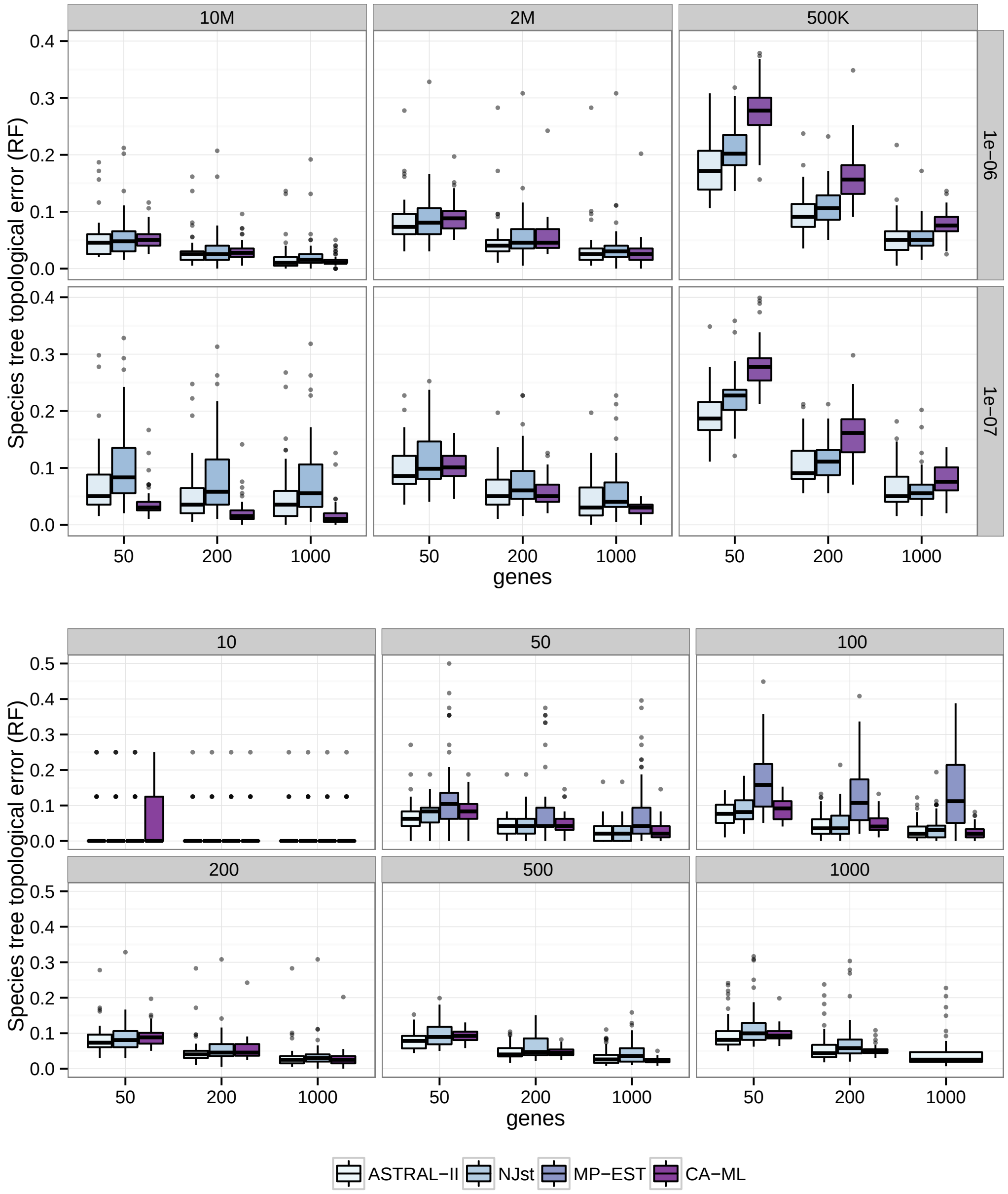


Figure 5.14: **Comparison of methods with respect to species tree topological error on ASTRAL-II simulated data**. Species tree error is shown for Dataset-I (top) and Dataset-II (bottom). ASTRAL-II is always at least as accurate as NJst and MP-EST, but CA-ML (using FastTree) is under some conditions more accurate.

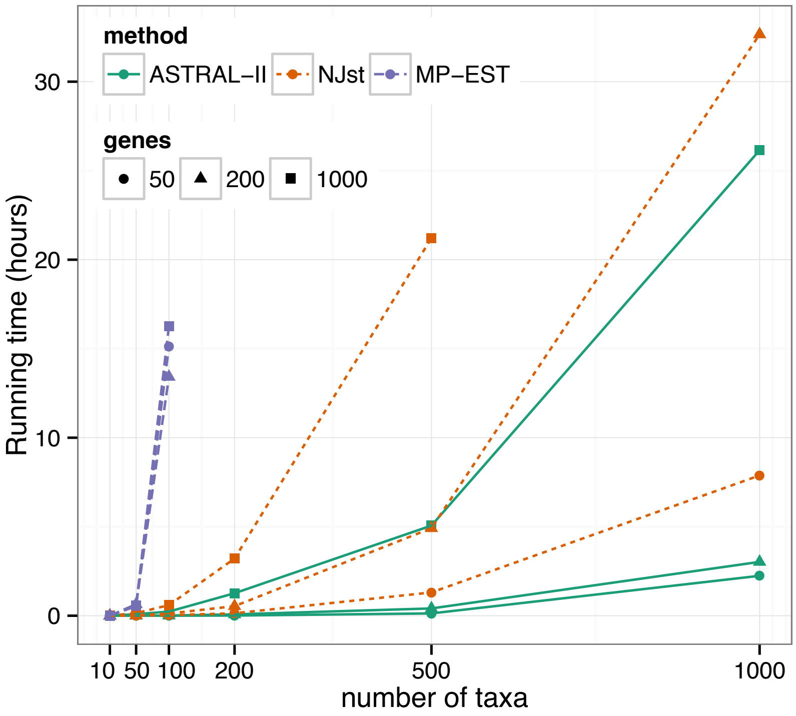


Figure 5.15: **Running time comparison with varying number of taxa and genes on Dataset II**. Average running time is shown for NJst and ASTRAL-II. Note that ASTRAL-II is much faster on large datasets.

For example, on 500 taxa and 1000 genes, ASTRAL-II typically finished in 2 to 10 hours, whereas NJst required 12 to 30 hours. MP-EST was the slowest method, but its running time was not impacted by the number of genes.

5.4.2.3 RQ3: ASTRAL-II vs. CA-ML

Dataset I: Interestingly, the relative accuracy of CA-ML and ASTRAL-II was significantly impacted by tree length ($p < 10^{-5}$), speciation rate ($p =$

0.00004), and the number of genes ($p < 10^{-5}$). With lower levels of ILS (10M and 2M) and recent speciation, CA-ML and ASTRAL-II had close accuracy, but CA-ML tended to be better with more genes and ASTRAL-II was better with fewer genes (Table 5.5, Fig. 5.14). With deep speciation and lower ILS, CA-ML was substantially more accurate than ASTRAL-II, but increasing the number of genes reduced the gap. At the high ILS levels, ASTRAL-II was much more accurate than CA-ML for all number of genes and for both recent and deep speciation.

Dataset II: Overall, differences between ASTRAL-II and CA-ML were not significant ($p = 0.2$), but the relative accuracy seemed to be impacted by the number of genes ($p = 0.06$). Regardless of the number of taxa, which did not impact relative accuracy ($p = 0.2$), CA-ML was slightly more accurate with 1000 genes, and ASTRAL-II was slightly often more accurate otherwise (Table 5.5, Fig. 5.14).

Running time: We ran CA-ML and ASTRAL-II on different platforms, and hence cannot make direct running time comparisons. Nevertheless, we provide our running time numbers to give a general idea. CA-ML using FastTree on 200-taxon model conditions with 1000 genes took roughly two hours, whereas ASTRAL-II took roughly one hour to estimate the species tree, and estimating gene trees also took about 1.5 hours. In general, therefore, the running times of ASTRAL-II and CA-ML are relatively close on this dataset.

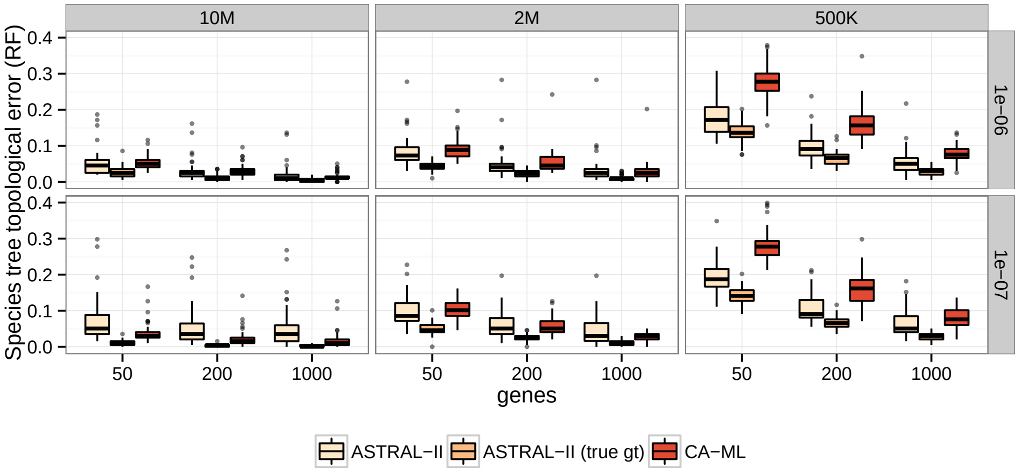


Figure 5.16: **Comparison of ASTRAL-II run on estimated and true gene trees and CA-ML on Dataset I**. The different between ASTRAL-II with true gene tree ("true gt") and ASTRAL-II with estimated gene trees indicates the impact of gene tree error. Note that with true gene trees, ASTRAL has excellent accuracy and is always better than CA-ML (using FastTree).

5.4.2.4 RQ4: Effect of gene tree error

In RQ3, we observed that under some conditions, CA-ML was more accurate than ASTRAL-II, a pattern that we attribute to high levels of gene tree error present in our simulations. When true (simulated) gene trees are used instead of the estimated gene trees, the accuracy of ASTRAL-II is outstanding, regardless of the model condition (see Fig. 5.16) and ASTRAL-II is always more accurate than CA-ML. Thus, the fact that CA-ML is occasionally more accurate than ASTRAL-II under lower levels of ILS is related to estimation error in the input provided to ASTRAL-II.

In our ASTRAL-II and NJst analyses, gene tree error had a positive correlation with species tree error (Fig. 5.17), with correlation coefficients that were similar for ASTRAL-II and NJst. The error of CA-ML also correlated with gene tree error (obviously the relationship is indirect as factors such as short alignments impact both CA-ML and gene tree error), but the correlation was weaker than the correlation observed for coalescent-based methods (Fig 5.18). Interestingly, the correlation between gene tree estimation error and species tree error was typically higher with fewer genes.

To further investigate the impact of the gene tree error, we divided replicates of each model condition into three categories: average gene tree estimation error below 0.25 is labelled low, between 0.25 and 0.4 is labelled medium, and above 0.4 is labelled high. We plotted the species tree error within each of these categories (see Figs. 5.19 and 5.20). The relative performance of ASTRAL-II and NJst is typically unchanged across various categories of gene tree error, but increasing gene tree error tends to increases in the magnitude of the difference between ASTRAL-II and NJst. Furthermore, MP-EST seemed to be more sensitive to gene tree error than either NJst or ASTRAL-II (Fig. 5.20).

The relative performance of ASTRAL-II and CA-ML depended on gene tree error. For those model conditions where CA-ML was generally more accurate than ASTRAL-II (e.g., 2M/1e-07), ASTRAL-II tended to outperform CA-ML on the replicates with low gene tree estimation error (Fig. 5.19). Consistent with this observation, we noted that ASTRAL-II was impacted by gene

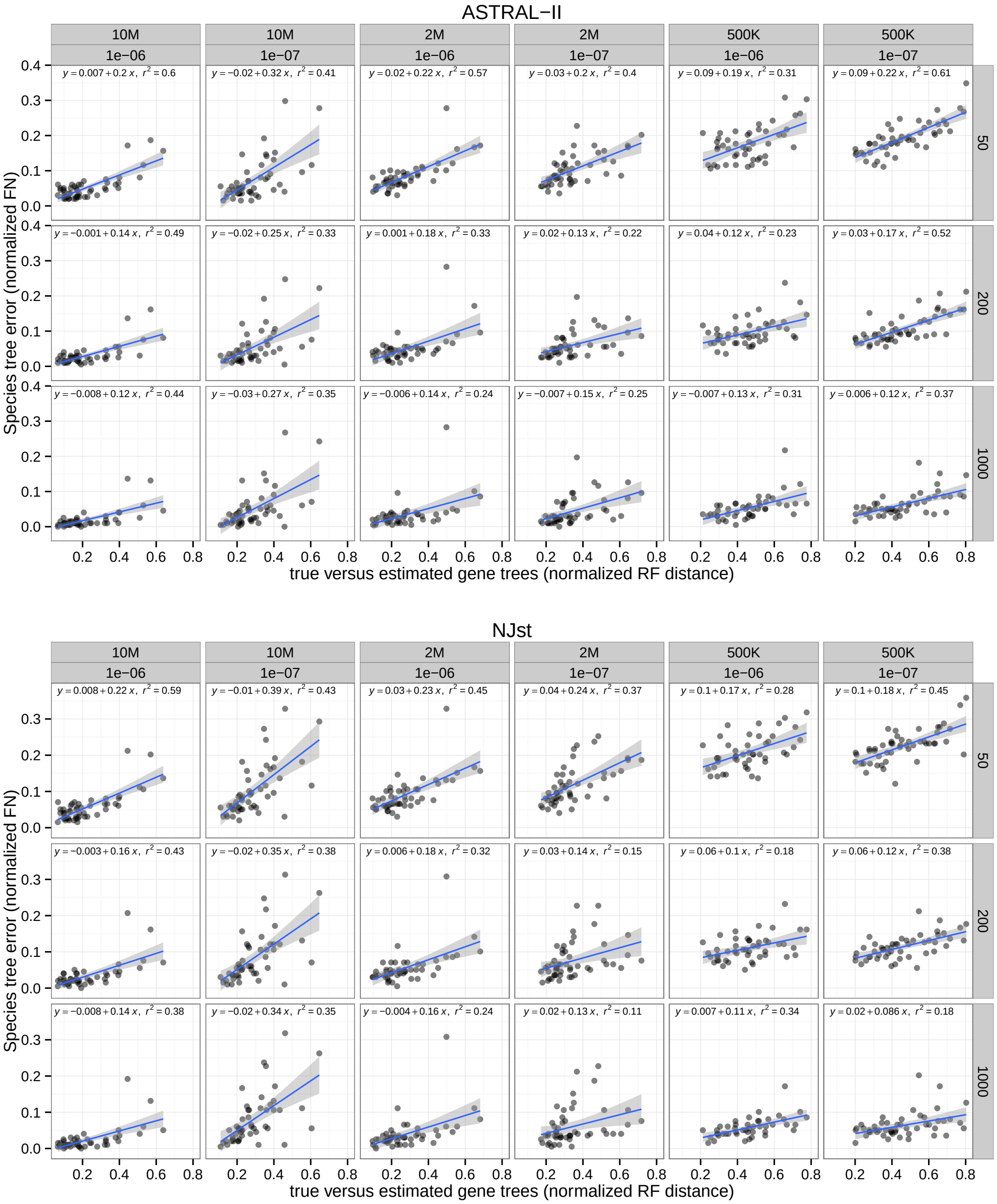


Figure 5.17: **Correlation between gene tree estimation error and species tree error for ASTRAL and NJst on Dataset-I**. Gene tree and species tree error correlate well, and the correlation is stronger for fewer genes and *lower* levels of ILS. Varying tree shapes are shown in columns and numbers of genes are showed in rows.

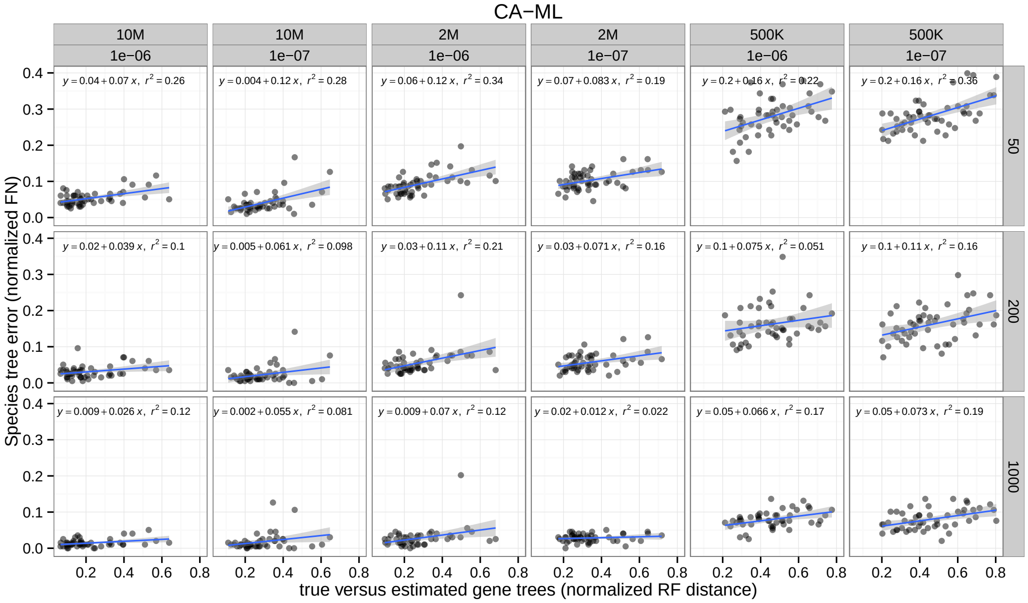


Figure 5.18: **Correlation between gene tree estimation error and species tree error for CA-ML on Dataset-I**. A correlation between gene tree error (controlled by parameters such as alignment length that also affect concatenation) and species tree error is detectable for concatenation, but is smaller compared to NJst and ASTRAL.

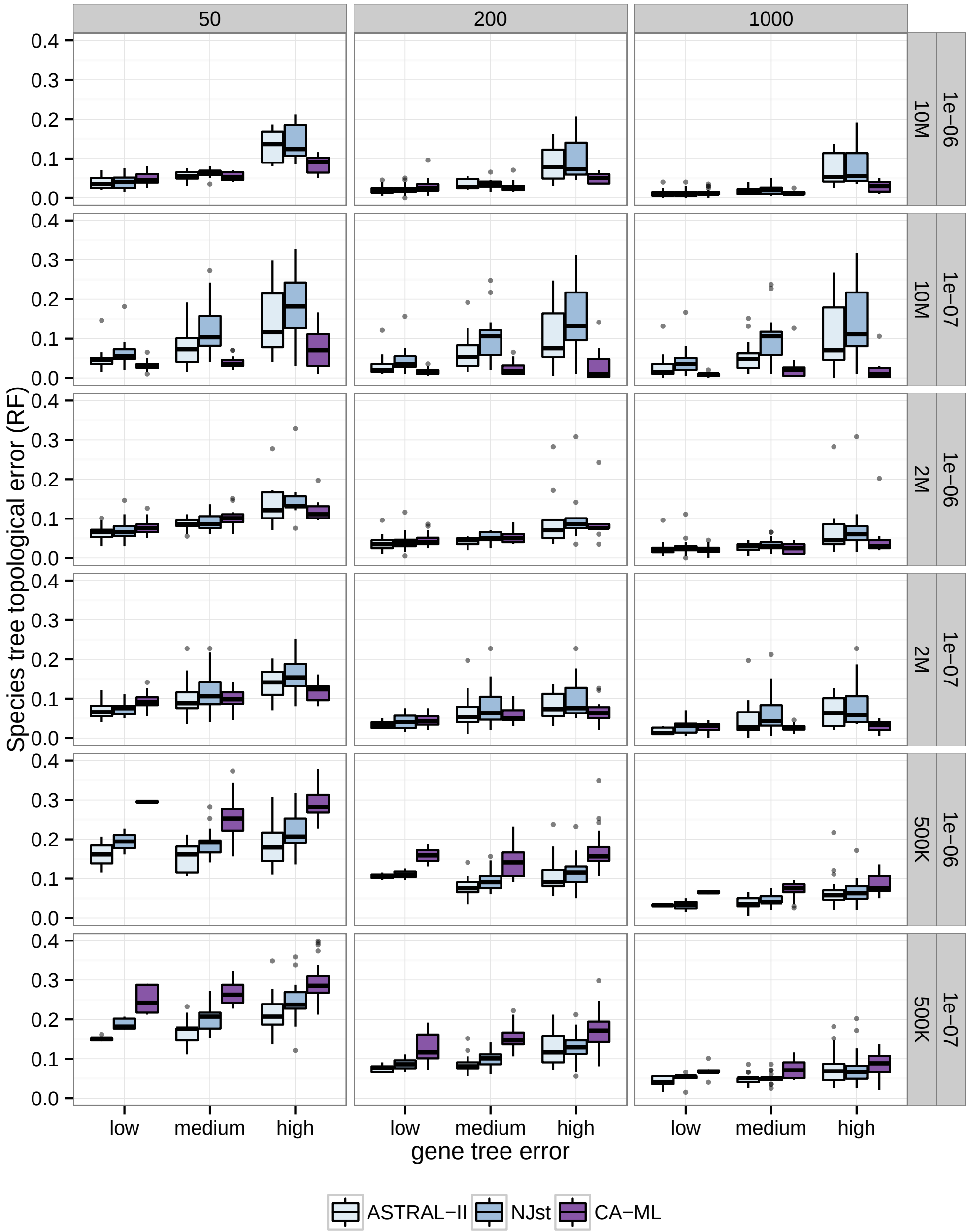


Figure 5.19: **Comparison of species tree error on Dataset-I, divided into three categories of gene tree estimation error**. Results are shown for 200 taxa and varying tree shapes (rows), and varying number of genes (columns), divided into three categories of gene tree estimation error: low, medium, and high.

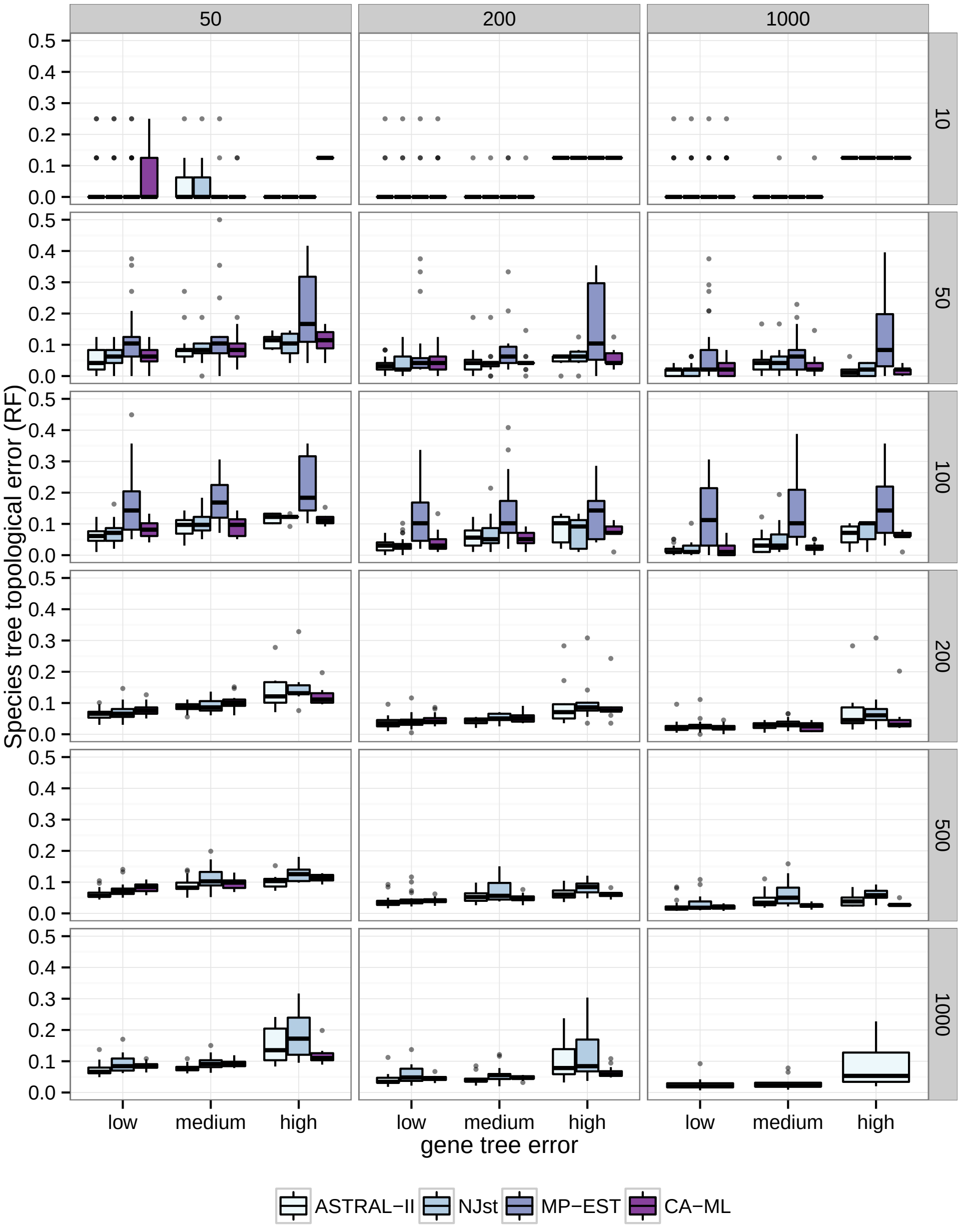


Figure 5.20: **Comparison of species tree error on Dataset-II, divided into three categories of gene tree estimation error**. Results are shown for varying number of taxa (rows), and varying number of genes (columns), divided into three categories of gene tree estimation error: low, medium, and high.

tree error more than CA-ML (Fig. 5.19).

5.4.2.5 RQ5: Collapsing low support branches

ASTRAL-II can handle inputs with polytomies. In this study, because of the prohibitive costs of applying bootstrapping to datasets of this size, we have not done bootstrapping on our genes to get reliable measures of support. However, we do get local SH-like branch support [43] from FastTree-II. Using these SH-like support values, we collapsed low support branches (10%, 33%, and 50%) and ran ASTRAL-II on the resulting unresolved gene trees. We measured the impact of contracting low support branches on the species RF rate. The median delta RF (error before collapsing minus error after collapsing) is typically zero (Fig. 5.21), never above zero, but in a few cases below zero (signifying that accuracy was improved in those few cases). However, these differences are not statistically significant ($p = 0.36$). Since this analysis was performed using SH-like branch support values instead of bootstrap support values (or other ways of estimating support values), it's hard to generalize and make conclusions about the use of other measures of support. Further studies are therefore needed for understanding the effect of collapsing low support branches in other situations.

5.4.3 Summary of results

Our wide-ranging simulation results show that ASTRAL-II, unlike the other methods we studied, can analyze datasets with up to 1000 taxa and

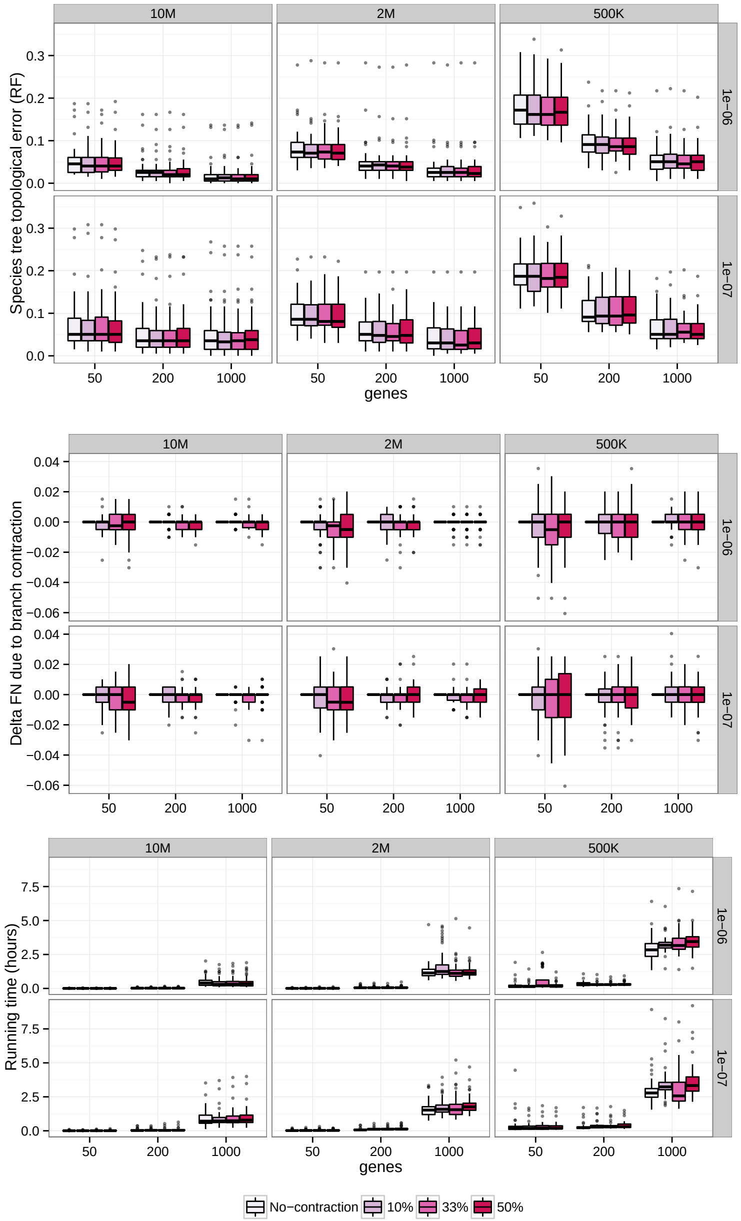


Figure 5.21: **Effect of contracting low support branches on ASTRAL-II**. Gene tree branches with FastTree SH-like local support below 10%, 33%, and 50% were contracted before running ASTRAL-II. Species tree error (top), change in species tree accuracy (middle) and running times (bottom) are shown. Delta FN (middle) shows changes in error compared to using binary trees, and so Delta FN < 0 indicates collapsing low branches improved accuracy.

1000 genes within reasonable running times. The next most computationally feasible method we explored was NJst, but ASTRAL-II was faster and more accurate than NJst. ASTRAL-II was also much more accurate than MP-EST, especially with larger numbers of species, but MP-EST was much slower and could not run on datasets with more than 100 species. Finally, ASTRAL-II improved upon ASTRAL-I in terms of both accuracy and running time. ASTRAL-II was more accurate than CA-ML, except when gene tree estimation error was high and ILS levels sufficiently low.

5.5 Biological Results

5.5.1 Datasets and methods

We analyzed five biological datasets:

- The 1KP dataset from [4], containing 103 plant species and 424 genes.
- The land plant dataset from [44], containing 32 species and 184 genes.
- The angiosperm dataset from [45] containing 42 angiosperm species and 4 outgroups with 310 genes.
- The mammalian dataset from [32], containing 37 species and 447 genes.
- The amniota dataset from [46], containing 16 species and 248 genes.

On these datasets, we compare ASTRAL-II, MP-EST, and concatenation using RAxML (CA-ML). We use gene trees that we estimated for the 1KP project; for the amniota and land plant datasets, gene trees were available from the respective publications. For the mammalian and the angiosperm datasets, we re-estimated gene trees from the gene alignments that were available. We used RAxML under the GTR+Γ model with 200 replicates of bootstrapping and 10 rounds of ML. We used the MLBS procedure [36] to obtain BS values (see Chapter 4).

In our analysis of the mammalian dataset, we found 21 genes with mislabelled sequences (easily confused taxon names, subsequently confirmed by the authors of [32]). We removed all those and two outliers genes from the dataset, and re-analyzed the reduced dataset. We used the MLBS procedure with 100 replicates, with both site and gene resampling, in order to be consistent with [32]. We re-estimated the gene trees using RAxML on the gene sequence alignments produced by [32].

On the amniota dataset, since the number of taxa is small, we ran the exact version of ASTRAL; in other cases, we ran ASTRAL-II.

5.5.2 Results

5.5.2.1 1KP dataset

As we noted earlier, analyzing 1KP dataset was one of our motivations for designing a new summary method. This dataset was very challenging for existing summary methods; it had 103 species, which is larger than what most methods are designed for and tested on. Also, since the 103 taxa span close to a billion years of evolution, rooting gene trees was challenging; finally, no single gene tree was complete, and some gene trees had substantial levels of missing data (note that this also affects the ability to root gene trees) As we noted before, the other summary methods were not able to produce reliable species trees on this dataset.

There are several interesting questions about plant evolution that this dataset can help answering, but three stand out.

Sister to land plants: The sister species to a clade including all the land plants remains unresolved. Two sets of streptophyte algae, Charales, and Coleochaetales, share complex characteristics with land plants (e.g., oogamous sexual reproduction and parental retention of the egg), which traditionally lead to the belief that Charlales, or Charales+Coleochaetales are sister to land plants. However, previous molecular analyses have inferred many different possible sister clades, including the following four major hypotheses: Zygnematales [47–49], Coleochaetales [50], Zygnematales + Coleochaetales [51], and Charales [52].

Bryophytes: Mosses, liverworts, and hornworts (collectively called bryophytes) are plants that separated out from other land plants early in the evolution of land plants. All various possible hypothesis of branching order involving these groups has been proposed in the literature and many have been supported by various data [53–55].

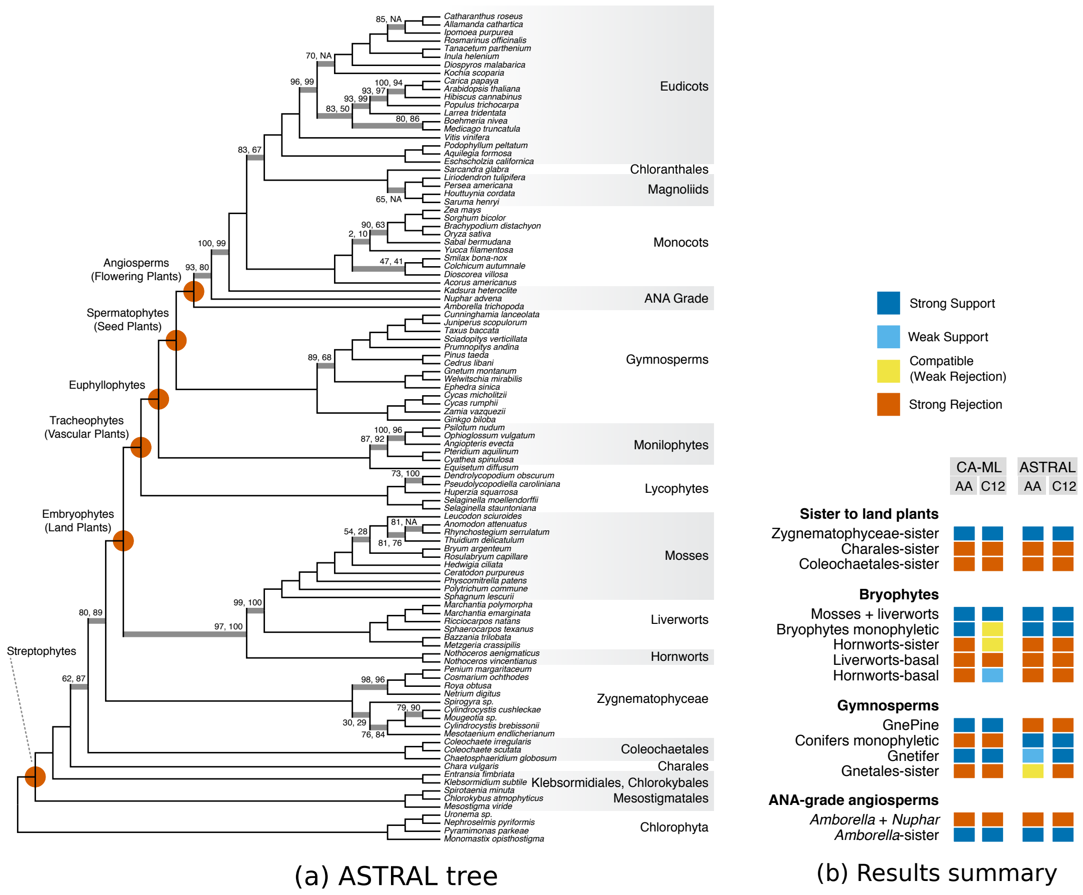
Gnetales: The position of Gnetales within a monophyletic gymnosperm clade is also unresolved, with various hypotheses recovered in the literature [56–58].

Angiosperms: The earliest branch that diverged from the remaining flowing plants (angiosperms) has been the subjective of debate. Amborella and Nymphaeales (water lilies), have been identified as earliest branches of the tree [59, 60]; however, it is not clear whether Amborella [60, 61] or a clade containing Nymphaeales+Amborella [62, 63] should be placed as sister to all other extant angiosperm lineages.

In its initial phase, the 1KP project gathered entire transcriptomes of 103 different plant species, and from those gathered a set of 852 single-copy putatively orthologous genes [4]. As part of the 1KP project, we estimated gene trees on all 852 genes, and then analyzed them in various ways, including various ways of filtering data. An important filtering was to remove fragmentary data from gene alignments. Fragments can reduce alignment accuracy [64], and can also result in poorly estimated gene trees. After removing sequences that were more than 66% gaps, and removing genes that were missing more than 50% of the sequence data, we obtained a dataset that included 424 gene trees (close to half of gene trees had less than half of the species and these were removed). We estimated gene trees based on amino acid sequences and also on DNA sequences with 3rd codon position removed (to avoid effects of GC bias [4, 65]). We report results on these two sets of 424 gene trees, and refer the reader to [4] for other analyses on the complete dataset. As mentioned in Section 5.1, our attempts at running MP-EST on this dataset had limited success.

Figure 5.22 shows the ASTRAL tree on 1KP and summarizes the differences between CA-ML and ASTRAL trees. Both concatenation and ASTRAL recover Zygnematales as sister to land plants, with high support. Similarly, the sister to flowering plants is recovered to be Amborella with high support, regardless of the dataset used or whether ASTRAL or CA-ML was used.

The relationships among Bryophytes and Gymnosperms are less consistent. In all analyses, mosses and liverworts were sister groups. However, in the CA-ML analysis of DNA sequences, hornworts were recovered with low support as the sister to all remaining land plants (a clade containing mosses, liverworts, and all the other land plants) whereas in both ASTRAL analyses and the CA-ML analysis of the AA data, hornworts were sister to mosses + liverworts, and this clade was at the base of land plants. The correct relationship is not known, but the fact that ASTRAL and concatenation recover different relationships is important, especially given short branch lengths at the base of land plants. Similarly, within Gymnosperms, the exact relationships recovered depend on the method used. ASTRAL analyses both recover Conifers as a monophyletic clade and Gnetales as the base of Gymnosperms, a topology previously recovered in other analyses [66]. However, CA-ML analyses put Gnetales as sister to pines, breaking the monophyly of Conifers (this topology was also previously observed [56]).



(a) ASTRAL tree

(b) Results summary

Figure 5.22: **Summary of results of 1KP dataset**. (a) ASTRAL results on the 1KP dataset (ASTRAL-I and ASTRAL-II produced identical results); the DNA tree is shown and the support values are shown for both DNA and AA astral analyses. Branches without designation have 100% support in both analyses. NA means a branch was missing from the AA analysis. (b) Summary of results. Rows show hypotheses of plant evolution for four parts of the tree. Columns show two ASTRAL and two CA-ML analyses (using RAxML). Colors indicate whether a hypothesis was supported, or rejected and whether support or rejection had support that was at least 75%.

5.5.2.2 Land plant dataset

The question of greatest interest on this dataset is the sister group to land plants. As noted before, our recent 1KP analysis recovered Zygnematales as sister to Land plants with high confidence using both ASTRAL and concatenation. Zhong *et al.* used MP-EST to analyze their data, and inferred Zygnematales as the sister with 64% BS [44]. A re-analysis of the same data using STAR was performed by Springer and Gatesy [16], who obtained Zygnematales + Coleochaetales with 44% BS.

We analyzed this dataset using ASTRAL-II and obtained a tree that generally has high BS on most branches (i.e., with the exception of four branches, all branches have support at least 86%, and most have 100% support). However, one edge had very low support (only 18%). After collapsing the single branch with very low support, we obtained a tree (see Fig. 5.23) in which the Charales + Land plants hypothesis is rejected with moderately high support (86%); however, it is not determined whether Zygnematales, Coleochaetales, or Zygnematales + Coleochaetales are the sister group to Land plants (the branch that distinguishes between these three hypotheses is the one with 18% support). Thus, ASTRAL's analysis of this dataset can be seen as suggesting that this dataset is insufficient to completely resolve the sister relationship to Land plants. However, the most interesting question is whether Charales are sister to Land plants, and the ASTRAL tree rejects that hypothesis with 86% support. The ASTRAL results, therefore, are consistent between the Zhong *et al.* dataset and 1KP dataset.

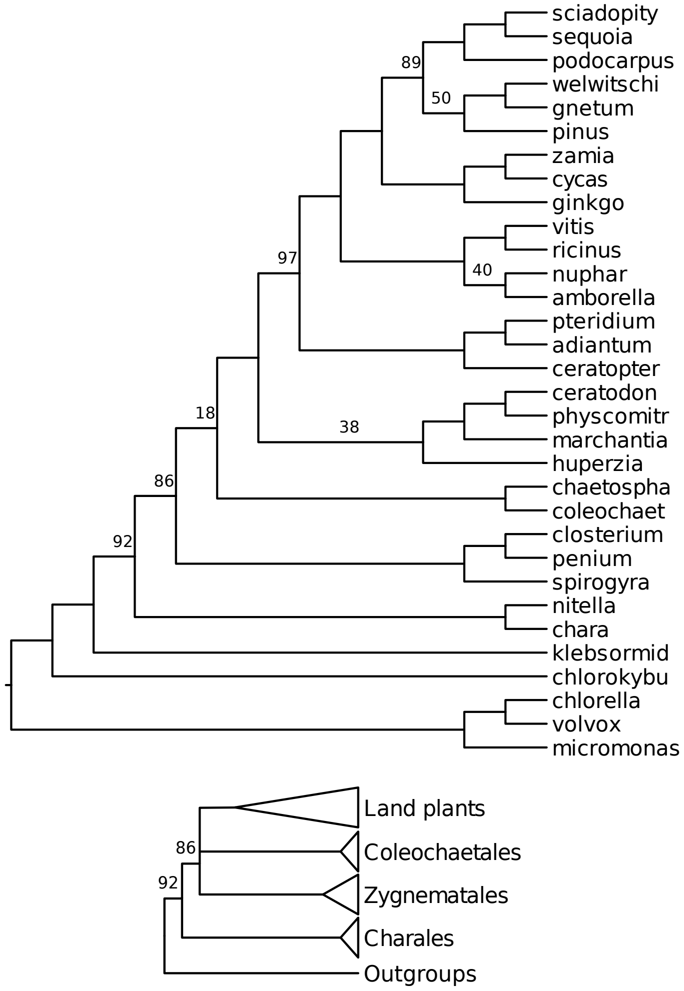


Figure 5.23: **ASTRAL tree on the Zhong *et al.* land plant dataset**. We analyzed a plant dataset with 32 species and 184 genes from [44]. Bootstrap support values were obtained using the multi-locus bootstrapping procedure with 100 replicates; values not shown indicate 100% support. ASTRAL-II tree (with bootstrap support values) is shown on top, and we show a cartoon version of the tree below. The cartoon version only shows the relationship between the 5 groups – Land plants, Coleochaetales, Zygnematales, Charales, and the outgroups, after collapsing the branch with bootstrap support of 18%. Note that there are three possible sister groups to Land plants: Coleochaetales, Zygnematales, or the two together (Zygnematales+Coleochaetales); however, Charlaes is strongly rejected as the sister group to Land plants.

5.5.2.3 Angiosperms

The evolution of angiosperms, and the placement of *Amborella trichopoda Baill.*, is one of the challenging questions in Land plant evolution. One hypothesis recovered in some recent molecular studies and all of our 1KP analyses is that *Amborella trichopoda* is sister to the rest of angiosperms, followed by Nymphaeales (e.g., see [4, 67–69]). A competing hypothesis is that Amborella is sister to Nymphaeales and this whole group is sister to other angiosperms [63, 69]. Xi *et al.* [45] have examined this question using a collection of 310 genes sampled from 42 angiosperms and 4 outgroups. They observed that concatenation using maximum likelihood (CA-ML) produced the first hypothesis and MP-EST produced the second hypothesis, and they argued that these differences are due to the fact that CA-ML does not model ILS, whereas MP-EST does.

We ran MP-EST and ASTRAL on the gene tees that we re-estimated on this dataset, and we obtained two different species trees (Fig. 5.24). Reproducing results by Xi *et al.*, MP-EST recovered the sister relationship of Amborella and Nymphaeales with 100% support. However, ASTRAL, just like CA-ML (using RAxML), recovers Amborella as sister to other angiosperms, with 75% support. While the exact position of Amborella is debated, our analysis shows that the differences between CA-ML and MP-EST results cannot be simply attributed to the fact that CA-ML does not consider ILS.

There are several possible reasons for the differences between the ASTRAL and MP-EST on this dataset, including the possibility that rooting

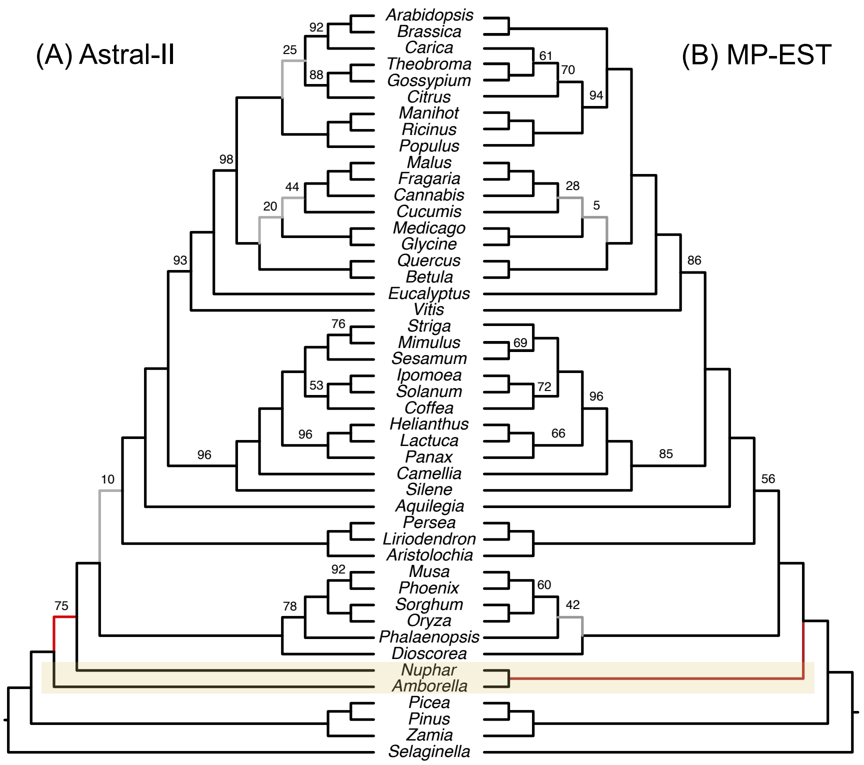


Figure 5.24: **Comparison of species trees computed on the angiosperm dataset**. MP-EST and ASTRAL-II differ in the placement of Amborella; the concatenation tree agrees with ASTRAL-II.

gene trees (required by MP-EST but not by ASTRAL-II) by *Selaginella* can be problematic for some genes, or that the impact of the gene tree estimation error is different for the two methods. We also note that ASTRAL-II is a non-parametric method that does not estimate branch lengths, and it is possible that non-parametric methods are less sensitive to gene tree estimation error than parametric methods (like MP-EST).

Our reanalysis of this dataset and our results on the 1KP dataset taken together point to more support for the hypothesis that Amborella is sister to the remaining flowering plants.

5.5.2.4 Mammalian

On the mammalian dataset, two of the questions of greatest interest were the placement of bats (Chiroptera) and tree shrew (Scandentia), where their MP-EST analysis differed from the concatenated analyses they performed. We recomputed the MP-EST tree, obtaining a tree topologically identical to the MP-EST tree reported in [32], but with lower bootstrap for the placement of Scandentia (62% in our analysis). CA-ML analyses of the full and reduced datasets using RAxML were topologically identical and had similar branch support. Thus, the CA-ML and MP-EST trees on the reduced dataset still differed in the placement of both Scandentia and Chiroptera.

We compare ASTRAL to MP-EST in Figure 5.25. Both ASTRAL and MP-EST trees placed Chiroptera as the sister to all other Laurasiatheria except Eulipotyphyla, while CA-ML placed Chiroptera as the sister to Cetar-

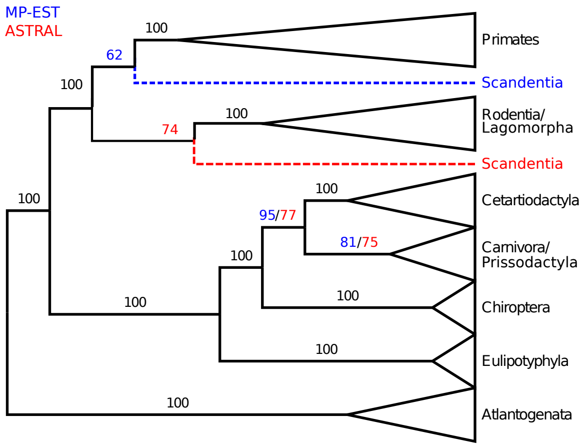


Figure 5.25: **Analysis of the Song et al. mammals dataset using ASTRAL and MP-EST**. We show the result of applying ASTRAL and MP-EST to 424 gene trees on 37-taxon mammalian species. MP-EST is based on rooted gene trees; ASTRAL is based on unrooted gene trees, and then rooted at the branch leading to the outgroup. Branch support values in black are for both methods, those in red are for ASTRAL, and values in blue are for MP-EST.

tiodactyla. The ASTRAL tree placed Scandentia as sister to Glires with 74% support, and thus agrees with the CA-ML tree but differs from the MP-EST tree. Thus, the differences between CA-ML and MP-EST cannot simply be attributed to use of a coalescent-based method, as Song *et al.* conjectured, since ASTRAL, which is also coalescent-based, recovers the same relationship as MP-EST.

5.5.2.5 Amniota dataset

Chiari *et al.* [46] assembled a dataset of Amniota to resolve the position of turtles relative to birds and crocodiles. Most recent studies favor an Archosaurus hypotheses that unites birds and crocodiles as sister groups [70]. The MP-EST analyses by [46] resolved this relationship differently when AA and DNA gene trees were used; thus, AA had 99% support for the Archosaurus clade, but DNA rejected Archosaurus with 90% support. We analyzed the same dataset using the exact version of ASTRAL and found that both AA and DNA recover Archosaurus; however, while ASTRAL on AA gene trees recovered Archosaurus with 100% support, ASTRAL on DNA gene trees had only 55% support for Archosaurus.

5.6 Discussions and future work

This study introduced ASTRAL, a method for estimating species trees from unrooted gene trees. We introduced two versions of ASTRAL, and proved that both versions are statistically consistent under the MSC model, but our second version, ASTRAL-II, has lowered running time and better empirical performance. Our simulation and biological results show that upcoming multi-gene datasets with large numbers of species can be accurately analyzed using ASTRAL-II. For example, we are currently analyzing the next of 1KP dataset that includes 400 genes, but more than 1,100 species.

Our biological analyses suggest that interestingly, some of the observed discrepancies between existing coalescent-based analyses and concatenation in previous studies [16] might be the result of the choice of coalescent-based method. Therefore, improved coalescent-based analyses might not only help to identify alternate relationships, but might also confirm prior hypotheses produced using concatenation.

An interesting observation was that in our simulations, concatenation was under certain conditions more accurate than ASTRAL and other summary methods. These results suggest that CA-ML should not be rejected, even though it is not statistically consistent. Conversely, proofs of consistency of standard summary methods assume gene trees estimated without error [71], and this assumption limits the relevance of consistency results in practice.

Our analyses also highlighted a problem that we addressed in Chapter 4: gene tree estimation error can affect the species tree, and that the accuracy of summary methods is depended on the accuracy of gene trees. This results in an interesting question: can the statistical binning approach also improve the accuracy of ASTRAL? Our preliminary results suggest that the answer is yes. We analyzed the avian simulated dataset presented in the previous chapter and observed that 1) ASTRAL-II has better accuracy than MP-EST on this dataset, and 2) binning used with ASTRAL-II further improved its accuracy for many model conditions (see results in Fig. 5.26 and see [72] for more). We also noted some interesting cases (e.g., the 1000bp model condition in Fig. 5.26) where ASTRAL, unlike MP-EST, did not improve using binning, but with or without binning ASTRAL had better accuracy than MP-EST. Nevertheless, our results make it clear that the use of all summary methods, including ASTRAL should be with the understanding that gene tree error can impact their results, and that practitioners need to make an effort to obtain the best gene trees possible using their data. The requirement to use

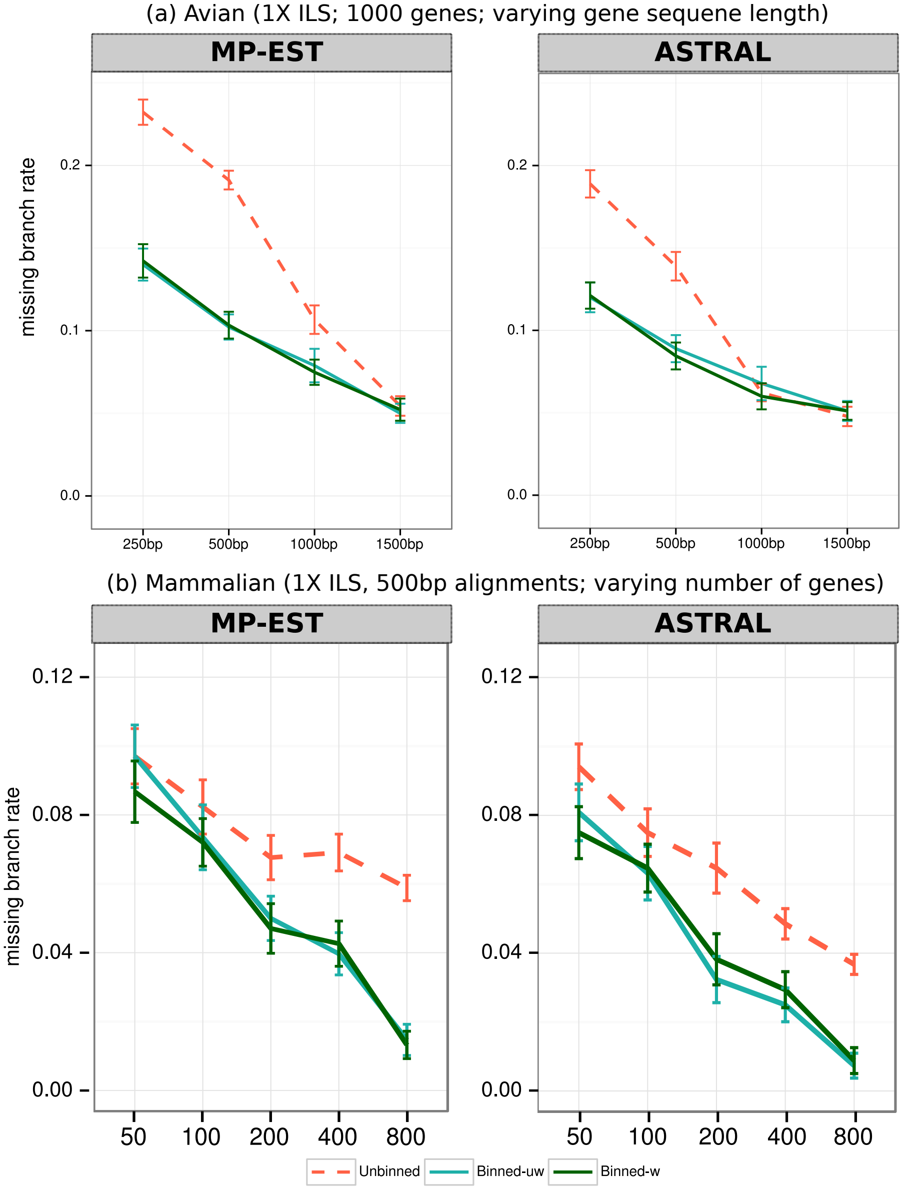


Figure 5.26: **Impact of binning on ASTRAL**. We compare weighted and unweighted statistical binning when run using MP-EST or ASTRAL-II as the summary method on simulated (a) avian and (b) mammalian datasets ($\mathcal{S}$ = 50% for avian and $\mathcal{S}$ = 75% for mammalian). ASTRAL, just like MP-EST, is improved in terms of accuracy when used with binned supergene trees. Also note that ASTRAL has lower error than MP-EST with or without binning, except with the longest sequences.

recombination-free regions complicates this pursuit as recombination-free "c-genes" can be very short, especially as the number of taxa increases [73]. Future work is needed to study the impact of using shorter gene sequence alignments, and conversely the presence of recombination events within genes.

Several limitations in ASTRAL need to be addressed in future work.

Comparison to other types of methods: While we compared ASTRAL to simple summary methods, future studies need to compare ASTRAL-II to boosting approaches (e.g., [5, 74]) that enable slower coalescent-based methods to scale to large datasets. Also, the running time of NJst and other simple distance-based methods that we didn't analyze here (e.g., STAR [75] and GLASS [76]) might be improved if better implementation of them is produced. Finally, a comparison to co-estimation methods under conditions where those methods can run (e.g., small numbers of species and genes) would also be interesting.

Missing data: We presented algorithms for handling incomplete gene trees. However, we have not rigorously studied the effect of incomplete gene trees on the accuracy of ASTRAL. A more comprehensive study needs to test the accuracy of ASTRAL in the presence of incomplete gene trees. These studies would be most interesting if they also include cases where missing data are not randomly distributed throughout the tree (e.g., basal taxa could be missing more often). While the optimization problem of ASTRAL is likely sufficient even when there are missing taxa, whether our current construction of set $\mathcal{X}$ from a set of incomplete genes is sufficient remains to be tested.

Multiple individuals: In studies where closely related species are analyzed, it is believed that sampling more than one individual per species can help in resolving the relationships [77, 78]. The optimization problem in ASTRAL can be easily extended to cases where multiple individuals are sampled from each species. Once again, computing the set $\mathcal{X}$ requires more care when multiple individuals are present, and future algorithmic developments are needed to obtain good accuracy on such datasets.

Further running time improvements: Further improvements to the running time of ASTRAL can be potentially obtained. For example, currently, in our traversal of gene trees, we do not exploit similarities between gene trees. If two gene trees are identical, we can traverse only one of them and simply count the resulting score twice. Taking this idea one step further would allow us to find commonalities between gene trees, and to exploit those commonalities to reduce the computational burden.

Appendices

Appendix A

Commands

A.1 ASTRAL

A.1.1 ASTRAL-I analyses

A.1.1.1 Gene tree estimation

RAxML version 7.3.5 [79] was used to estimate gene trees. The following command was used for estimating the best ML trees.

```
 raxmlHPC-SSE3 -m GTRGAMMA -s [input_file] -n [a_name]
-N 20 -p [random_seed_number]
```

The following command was used for bootstrapping.

```
 raxmlHPC-SSE3 -m GTRGAMMA -s [input_file] -n [a_name] -N 200
-p [random_seed_number] -b [random_seed_number]
```

A.1.1.2 ASTRAL

We ran version 3.1.1 of ASTRAL (corresponding to the github commit `fb21c0ce6140e9e238575356bc174c88c6cfc597` from March 6th on `https://github.com/smirarab/ASTRAL` with the following command:

```
java -jar astra_3.1.1.jar -wq -in [input_tree]
```

Where the exact version of ASRAL was used, we ran it with the following command:

```
java -jar astra_3.1.1.jar -wq -in [input_tree] -xt
```

To add new bipartitions to $\mathcal{X}$, we used it with the following command:

```
java -jar astra_3.1.1.jar -wq -in [input_tree] -ex [extra_trees]
```

A.1.1.3 BUCKy-population

We ran BUCKy with the default settings, except for the number of generations that we changed from 100K to one million. The following command was used to run BUCKy.

```
bucky -n <numberOfGenerations> -o <outputFileRoot> <inputFiles>
```

A.1.1.4 MRP and MRL

MRP trees are built using a custom Java program available at https://github.com/smirarab/mrpmatrix. The following command was used to create the MRP matrix.

```
java -jar mrp.jar [input_file] [output_file] NEXUS
```

We used the default heuristic in PAUP* (v. 4. 0b10) [80] for maximum parsimony. This heuristic first generates an initial tree through random sequence addition and then uses Tree Bisection and Reconnection (TBR) moves to reach a local optimum. This process is repeated 1000 times, and the most parsimonious tree is returned. When multiple trees have the same maximum parsimony score, the greedy consensus of those trees is returned. The following shows the PAUP* commands used.

```
 begin paup;
set criterion=parsimony maxtrees=1000
increase=no;
hsearch start=stepwise addseq=random
nreps=100 swap=tbr;
filter best=yes;
savetrees file = <treeFile> replace=yes
format=altnex;
contree all/ strict=yes
treefile = <strictConsensusTreeFile>
replace=yes;
tcontree all/ majrule=yes strict=no
treefile = <majorityConsensusTreeFile>
```

```
replace=yes;
contree all/ majrule=yes strict=no
le50=yes
treefile = <greedyConsensusTreeFile>
replace=yes;
log stop;
quit; end;
```

MRL stands for "Matrix Representation with Likelihood", and is the supertree method obtained by running two-state symmetric maximum likelihood on the MRP matrix [11]. We computed maximum likelihood trees on the same MRP matrix using RAxML under the two-state maximum likelihood model, to obtain MRL (matrix representation with likelihood) trees.

A.1.1.5 Concatenation

We used RAxML version 7.3.5 to create the parsimony starting trees:

```
 raxmlHPC-SSE3 -y -s supermatrix.phylip -m GTRGAMMA
-n [a_name] -p [random_seed_number] -s [alignment]
```

We then used RAxML-light version 1.0.6 with the following command to search for the ML tree.

```
 raxmlLight-PTHREADS -T 4 -s supermatrix.phylip -m GTRGAMMA -n name
-t [parsimony_tree] -s [alignment]
```

A.2 ASTRAL-II

A.2.1 SimPhy parameters

We used the following parameters in our simulation using SimPhy. The scripts for the simulation are given at http://www.cs.utexas.edu/users/phylo/software/astral/.

A.2.2 Indelible parameters

We used a perl script available also at http://www.cs.utexas.edu/users/phylo/software/astral/ to draw parameters for the Indelible simulations. For

Table A.1: **Parameters used in SimPhy simulations**.

Arg.	Description	Value	Notes
RS	number of replicates	50	
RL	number of loci	1000	
RG	number of genes	1	no duplications
ST	maximum tree length	500K, 2M, or 10M	
SI	number of individuals per species	1	
SL	number of leaves	10,50,100,200,500, or 1000	
SB	birth rates	0.000001, 0.0000001	
P	global population sizes	200000	
HS	Species-specific branch rate heterogeneity modifiers	Log normal (1.5,1)	
HL	Locus-specific rate heterogeneity modifiers	Log normal (1.2,1)	
HG	Gene-tree-branch-specific rate heterogeneity modifiers	Log normal (1.4,1)	
U	Global substitution rate	Exponential (10000000)	
SO	Outgroup branch length relative to half the tree length	1	
CS	Random number generator seed	293745	

each replicate, some hyperparameters are first drawn and these hyperparameters affect how the actual parameters are drawn for each gene in that replicate.

Gene Length: The alignments lengths are drawn from log normal distributions for genes of each replicate. For each replicate, a hyperparameter controls the two model parameters of the log normal distribution. The log mean is drawn uniformly between 5.7 and 7.3, which correspond to 300 sites to 1500 sites. Thus, the average alignment length for each replicate is a random value between 300 and 1500. The log standard deviation for the log normal distribution is also drawn uniformly between 0.0 and 0.3.

Base frequencies: We used a Dirichlet(36,26,28,32) to draw the base frequencies for A, C, G, and T. These values were calculated using maximum likelihood estimation form a collection of three large scale multi-locus datasets: 1KP dataset, Song et al Mammalian dataset, and Avian phylogenomics dataset. The base values used for this maximum likelihood estimation and the corresponding scripts are available at `http://www.cs.utexas.edu/~phylo/software/astral/`.

Substitution matrices: As with base frequencies, GTR matrices were drawn from a Dirichlet(16,3,5,5,6,15) and these parameters were also estimated using maximum likelihood from our empirical data.

Rates-across-sites shape parameter: α was drawn from an exponential distribution with rate 1.2, with values below 0.1 discarded. Like rates and base frequencies, these values were estimated from real data.

Bibliography

[1] Siavash Mirarab, Rezwana Reaz, Md. Shamsuzzoha Bayzid, Théo Zimmermann, M Shel Swenson, and Tandy Warnow. ASTRAL: Genome-Scale Coalescent-Based Species Tree. *Bioinformatics*, 30(17):i541–i548, 2014.

[2] S. Mirarab and T. Warnow. ASTRAL-II: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 2015.

[3] Liang Liu, Lili Yu, and Scott V Edwards. A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC Evolutionary Biology*, 10(1):302, 2010.

[4] Norman J. Wickett, Siavash Mirarab, Nam Nguyen, Tandy Warnow, Eric Carpenter, Naim Matasci, Saravanaraj Ayyampalayam, Michael S. Barker, J. Gordon Burleigh, Matthew A. Gitzendanner, Brad R. Ruhfel, Eric Wafula, Joshua P. Der, Sean W. Graham, Sarah Mathews, Michael Melkonian, Douglas E. Soltis, Pamela S. Soltis, Nicholas W. Miles, Carl J. Rothfels, Lisa Pokorny, A. Jonathan Shaw, Lisa DeGironimo, Dennis W. Stevenson, Barbara Surek, Juan Carlos Villarreal, Béatrice Roure, Hervé Philippe, Claude W. dePamphilis, Tao Chen, Michael K. Deyholos, Regina S. Baucom, Toni M. Kutchan, Megan M. Augustin, Jun Wang, Yong Zhang, Zhijian Tian, Zhixiang Yan, Xiaolei Wu, Xiao Sun, Gane Ka-Shu Wong, and James Leebens-

Mack. Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences of the United States of America*, 111(45):E4859–E4868, 2014.

[5] Md Shamsuzzoha Bayzid, Tyler Hunt, and Tandy Warnow. Disk covering methods improve phylogenomic analyses. *BMC Genomics*, 15(Suppl 6):S7, 2014.

[6] Bret Larget, Satish K Kotha, Colin N Dewey, and Cécile Ané. BUCKy: Gene tree/species tree reconciliation with the Bayesian concordance analysis. *Bioinformatics*, 26(22):2910–2911, 2010.

[7] Jimmy Yang and Tandy Warnow. Fast and accurate methods for phylogenomic analyses. *BMC Bioinformatics*, 12(Suppl 9):S4, 2011.

[8] Liang Liu and Lili Yu. Estimating species trees from unrooted gene trees. *Systematic Biology*, 60:661–667, 2011.

[9] Yuancheng Wang and James H. Degnan. Performance of Matrix Representation with Parsimony for Inferring species from gene trees. *Statistical Applications in Genetics and Molecular Biology*, 10(1):1–39, 2011.

[10] Siavash Mirarab, Md. Shamsuzzoha Bayzid, and Tandy Warnow. Evaluating summary methods for multi-locus species tree estimation in the presence of incomplete lineage sorting. *Systematic Biology*, page syu063, 2014.

[11] Nam Nguyen, Siavash Mirarab, and Tandy Warnow. MRL and SuperFine+MRL: new supertree methods. *Algorithms for Molecular Biology*, 7(1):3, 2012.

[12] Michael DeGiorgio and James H. Degnan. Robustness to divergence time underestimation when inferring species trees from estimated gene trees. *Systematic Biology*, 63(1):66–82, 2014.

[13] Md. Shamsuzzoha Bayzid and Tandy Warnow. Naive binning improves phylogenomic analyses. *Bioinformatics*, 29(18):2277–84, 2013.

[14] Rebecca T Kimball, Ning Wang, Victoria Heimer-McGinn, Carly Ferguson, and Edward L Braun. Identifying localized biases in large datasets: A case study using the avian tree of life. *Molecular Phylogenetics and Evolution*, 69:1021–1032, 2013.

[15] John E McCormack, Michael G. Harvey, Brant C. Faircloth, Nicholas G Crawford, Travis C. Glenn, and Robb T. Brumfield. A Phylogeny of Birds Based on Over 1,500 Loci Collected by Target Enrichment and High-Throughput Sequencing. *PLoS ONE*, 8(1):e54848, 2013.

[16] Mark S Springer and John Gatesy. Land plant origins and coalescence confusion. *Trends in Plant Science*, 19(5):267–9, 2014.

[17] James H. Degnan and Noah A. Rosenberg. Discordance of species trees with their most likely gene trees. *PLoS Genetics*, 2(5):e68, 2006.

[18] James H. Degnan and Noah A. Rosenberg. Gene tree discordance, phylogenetic inference and the multispecies coalescent. *Trends in Ecology and Evolution*, 24(6):332–340, 2009.

[19] Elizabeth S. Allman, James H. Degnan, and John A Rhodes. Identifying the rooted species tree from the distribution of unrooted gene trees under the coalescent. *Journal of Mathematical Biology*, 62:833–862, 2011.

[20] James H. Degnan. Anomalous unrooted gene trees. *Systematic Biology*, 62:574–590, 2013.

[21] Tao Jiang, Paul Kearney, and Ming Li. A Polynomial Time Approximation Scheme for Inferring Evolutionary Trees from Quartet Topologies and Its Application. *SIAM Journal on Computing*, 30(6):1942–1961, 2001.

[22] Michael Steel. The complexity of reconstructing trees from qualitative characters and subtrees. *Journal of Classification*, 9(1):91–116, 1992.

[23] Peter Buneman. The recovery of trees from measures of dissimilarity. *Mathematics in the archaeological and historical sciences*, pages 387–395, 1971.

[24] Michael T. Hallett and Jens Lagergren. New algorithms for the duplication-loss model. In *Proceedings of the International Conference on Research in Computational Molecular Biology*, pages 138–146. ACM, 2000.

[25] Yun Yu, Tandy Warnow, and Luay Nakhleh. Algorithms for MDC-based multi-locus phylogeny inference: beyond rooted binary gene trees on single alleles. *Journal of Computational Biology*, 18(11):1543–1559, 2011.

[26] David Bryant and Michael Steel. Constructing Optimal Trees from Quartets. *Journal of Algorithms*, 38:237–259, 2001.

[27] Peter Buneman. A note on the metric properties of trees. *Journal of Combinatorial Theory, Series B*, 17(1):48–50, 1974.

[28] Peter Erdos, Michael Steel, László Székely, and Tandy Warnow. A few logs suffice to build (almost) all trees (I). *Random Structures and Algorithms*, 14(2):153–184, 1999.

[29] Md. Shamsuzzoha Bayzid and Tandy Warnow. Estimating optimal species trees from incomplete gene trees under deep coalescence. *Journal of Computational Biology*, 19(6):591–605, 2012.

[30] David F Robinson and Leslie R Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, 1981.

[31] Jen Stoye, Dirk Evers, and Folker Meyer. Rose: generating sequence families. *Bioinformatics*, 14(2):157–163, 1998.

[32] Sen Song, Liang Liu, Scott V Edwards, and Shaoyuan Wu. Resolving conflict in eutherian mammal phylogeny using phylogenomics and the multispecies coalescent model. *Proceedings of the National Academy of Sciences of the United States of America*, 109(37):14942–7, 2012.

[33] Jeet Sukumaran and Mark T Holder. Dendropy: a Python library for phylogenetic computing. *Bioinformatics*, 26(12):1569–71, 2010.

[34] Mark A Ragan. Phylogenetic inference based on matrix representation of trees. *Molecular Phylogenetics and Evolution*, 1(1):53–58, 1992.

[35] Cécile Ané, Bret R Larget, David A Baum, Stacey D Smith, and Antonis Rokas. Bayesian estimation of concordance among gene trees. *Molecular Biology and Evolution*, 24(2):412–426, 2007.

[36] Tae Kun Seo. Calculating bootstrap probabilities of phylogeny using multilocus sequence data. *Molecular Biology and Evolution*, 25(5):960–971, 2008.

[37] L Lacey Knowles, Hayley C. Lanier, Pavel B. Klimov, and Qixin He. Full modeling versus summarizing gene-tree uncertainty: Method choice and species-tree accuracy. *Molecular Phylogenetics and Evolution*, 65(2):501–509, 2012.

[38] Diego Mallo, L de Oliveira Martins, and D Posada. Simphy: Comprehensive simulation of gene, locus and species trees at the genome-wide level., 2015. (In Prep, available at `https://code.google.com/p/simphy-project/`).

[39] William Fletcher and Ziheng Yang. Indelible: A flexible simulator of biological sequence evolution. *Molecular Biology and Evolution*, 26(8):1879–1888, 2009.

[40] Kevin Liu, C. Randal Linder, and Tandy Warnow. RAxML and FastTree: Comparing Two Methods for Large-Scale Maximum Likelihood Phylogeny Estimation. *PLoS ONE*, 6(11), 2011.

[41] Morgan N. Price, P S Dehal, and Adam P. Arkin. FastTree-2 Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 2010.

[42] Alexandros Stamatakis, Andre J Aberer, C Goll, Stephen A Smith, Simon A Berger, and Fernando Izquierdo-Carrasco. RAxML-Light: a tool for computing

terabyte phylogenies. *Bioinformatics*, 28(15):2064–2066, 2012.

[43] Hidetoshi Shimodaira and Masami Hasegawa. Multiple Comparisons of Log-Likelihoods with Applications to Phylogenetic Inference. *Molecular Biology and Evolution*, 16(8), 1999.

[44] Bojian Zhong, Liang Liu, Zhen Yan, and David Penny. Origin of land plants using the multispecies coalescent model. *Trends in Plant Science*, 18(9):492–495, 2013.

[45] Zhenxiang Xi, Liang Liu, Joshua S Rest, and Charles C Davis. Coalescent versus Concatenation Methods and the Placement of Amborella as Sister to Water Lilies. *Systematic Biology*, 63(6):919–932, 2014.

[46] Ylenia Chiari, Vincent Cahais, Nicolas Galtier, and Frédéric Delsuc. Phylogenomic analyses support the position of turtles as the sister group of birds and crocodiles (archosauria). *BMC Biology*, 10(1):65, 2012.

[47] Sabina Wodniok, Henner Brinkmann, Gernot Glöckner, Andrew J Heidel, Hervé Philippe, Michael Melkonian, and Burkhard Becker. Origin of land plants: do conjugating green algae hold the key? *BMC Evolutionary Biology*, 11:104, 2011.

[48] Ruth E. Timme, Tsvetan R. Bachvaroff, and Charles F. Delwiche. Broad phylogenomic sampling and the sister lineage of land plants. *PLoS ONE*, 7(1), 2012.

[49] Brad R Ruhfel, Matthew A. Gitzendanner, Pamela S Soltis, Douglas E Soltis, and J Gordon Burleigh. From algae to angiosperms - inferring the phylogeny of green plants (Viridiplantae) from 360 plastid genomes. *BMC Evolutionary Biology*, 14:23, 2014.

[50] Cédric Finet, Ruth E. Timme, Charles F. Delwiche, and Ferdinand Marlétaz. Multigene phylogeny of the green lineage reveals the origin and diversification of land plants. *Current Biology*, 20(24):2217–2222, 2010.

[51] Monique Turmel, Christian Otis, and Claude Lemieux. The chloroplast genome sequence of Chara vulgaris sheds new light into the closest green algal relatives of land plants. *Molecular Biology and Evolution*, 23(6):1324–1338, 2006.

[52] Kenneth G Karol, Richard M McCourt, Matthew T Cimino, and Charles F Delwiche. The closest living relatives of land plants. *Science*, 294(5550):2351–2353, 2001.

[53] Roberto Ligrone, Jeffrey G. Duckett, and Karen S. Renzaglia. Major transitions in the evolution of early land plants: A bryological perspective. *Annals of Botany*, 109(5):851–871, 2012.

[54] Yin-Long Qiu, Libo Li, Bin Wang, Zhi-Duan Chen, Olena Dombrovska, Jungho Lee, Livija Kent, Rui-Qi Li, Richard W. Jobson, Tory A. Hendry, David W. Taylor, Christopher M. Testa, and Mathew Ambros. A Nonflowering Land Plant Phylogeny Inferred from Nucleotide Sequences of Seven Chloroplast, Mitochondrial, and Nuclear Genes. *International Journal of Plant Sciences*, 168(5):691–708, 2007.

[55] Tomoaki Nishiyama, Paul G. Wolf, Masanori Kugita, Robert B. Sinclair, Mamoru Sugita, Chika Sugiura, Tatsuya Wakasugi, Kyoji Yamada, Koichi Yoshinaga, Kazuo Yamaguchi, Kunihiko Ueda, and Mitsuyasu Hasebe. Chloroplast phylogeny indicates that bryophytes are monophyletic. *Molecular Biology and Evolution*, 21(10):1813–1819, 2004.

[56] L Michelle Bowe and Gwénaële Coat. Phylogeny of seed plants based on all three genomic compartments: Extant gymnosperms are monophyletic and Gnetales' closest relatives are conifers. *Proceedings of the National Academy of Sciences of the United States of America*, 97(8):4092–4097, 2000.

[57] Jose Eduardo De La Torre-bárcena, Sergios Orestis Kolokotronis, Ernest K. Lee, Dennis Wm Stevenson, Eric D. Brenner, Manpreet S. Katari, Gloria M. Coruzzi, and Rob DeSalle. The impact of outgroup choice and missing data on major seed plant phylogenetics using genome-wide EST data. *PLoS ONE*, 4(6):e5764, 2009.

[58] Shu-Miaw Chaw, Andrey Zharkikh, Huang-Mo Sung, Tak-Cheung Lau, and Wen-Hsiung Li. Molecular phylogeny of extant gymnosperms and seed plant evolution: analysis of nuclear 18S rRNA sequences. *Molecular Biology and Evolution*, 14(1):56–68, 1997.

[59] Yin-Long Qiu, Jungho Lee, Fabiana Bernasconi-Quadroni, Douglas E Soltis, Pamela S Soltis, Michael Zanis, Elizabeth A Zimmer, Zhi-Duan Chen, Vincent Savolainen, and Mark W Chase. The earliest angiosperms: evidence from mitochondrial, plastid and nuclear genomes. *Nature*, 402:404–407, 1999.

[60] Pamela S Soltis, Douglas E Soltis, and Mark W. Chase. Angiosperm phylogeny inferred from multiple genes as a tool for comparative biology. *Nature*, 402(6760):402–404, 1999.

[61] Michael J Moore, Charles D Bell, Pamela S Soltis, and Douglas E Soltis. Using plastid genome-scale data to resolve enigmatic relationships among basal angiosperms. *Proceedings of the National Academy of Sciences of the United States of America*, 104(49):19363–19368, 2007.

[62] Yin-Long Qiu, Libo Li, Bin Wang, Jia-Yu Xue, Tory a. Hendry, Rui-Qi Li, Joseph W. Brown, Yang Liu, Geordan T. Hudson, and Zhi-Duan Chen. Angiosperm phylogeny inferred from sequences of four mitochondrial genes. *Journal of Systematics and Evolution*, 48(6):391–425, 2010.

[63] Vadim V Goremykin, Svetlana V Nikiforova, Patrick J Biggs, Bojian Zhong, Peter Delange, William Martin, Stefan Woetzel, Robin A Atherton, Patricia A Mclenachan, and Peter J Lockhart. The Evolutionary Root of Flowering Plants. *Systematic Biology*, 62(1):50–61, 2013.

[64] Nam Nguyen, Siavash Mirarab, Keerthana Kumar, and Tandy Warnow. Ultra-large alignments using phylogeny-aware profiles. *Genome Biology*, 16(1):124, 2015.

[65] Oliver Jeffroy, Henner Brinkmann, Frédéric Delsuc, and Hervé Philippe. Phylogenomics: the beginning of incongruence? *Trends in Genetics*, 22(4):225–231, 2006.

[66] Ernest K. Lee, Angelica Cibrian-Jaramillo, Sergios Orestis Kolokotronis, Manpreet S. Katari, Alexandros Stamatakis, Michael Ott, Joanna C. Chiu, Damon P. Little, Dennis Wm Stevenson, W. Richard McCombie, Robert A. Martienssen, Gloria M. Coruzzi, and Rob DeSalle. A functional phylogenomic view of the seed plants. *PLoS Genetics*, 7(12), 2011.

[67] Yin-Long Qiu, Jungho Lee, Fabiana Bernasconi-Quadroni, Douglas E. Soltis, Pamela S. Soltis, Michael Zanis, Elizabeth A. Zimmer, Zhiduan Chen, Vincent Savolainen, and Mark W. Chase. Phylogeny of Basal Angiosperms: Analyses of Five Genes from Three Genomes. *International Journal of Plant Sciences*, 161:S3–S27, 2000.

[68] Ning Zhang, Liping Zeng, Hongyan Shan, and Hong Ma. Highly conserved low-copy nuclear genes as effective markers for phylogenetic analyses in angiosperms. *New Phytologist*, 195:923–937, 2012.

[69] Bryan T Drew, Brad R Ruhfel, Stephen A Smith, Michael J Moore, Barbara G Briggs, Matthew A Gitzendanner, Pamela S Soltis, and Douglas E Soltis. Another Look at the Root of the Angiosperms Reveals a Familiar Tale. *Systematic Biology*, 63(3):368–382, 2014.

[70] Andrew F Hugall, Ralph Foster, and Michael SY Lee. Calibration choice, rate smoothing, and the pattern of tetrapod diversification according to the long nuclear gene rag-1. *Systematic Biology*, 56(4):543–563, 2007.

[71] Sebastien Roch and Tandy Warnow. On the robustness to gene tree estimation

error (or lack thereof) of coalescent-based species tree methods. *Systematic Biology*, page syv016, 2015.

[72] Md. Shamsuzzoha Bayzid, Siavash Mirarab, Bastien Boussau, and Tandy Warnow. Weighted Statistical Binning: enabling statistically consistent genome-scale phylogenetic analyses. *PLoS ONE*, 10(6):e0129183, 2015.

[73] John P. Gatesy and Mark S. Springer. Phylogenetic analysis at deep timescales: Unreliable gene trees, bypassed hidden support, and the coalescence/concatalescence conundrum. *Molecular Phylogenetics and Evolution*, 80:231–266, 2014.

[74] Théo Zimmermann, Siavash Mirarab, and Tandy Warnow. BBCA: Improving the scalability of *BEAST using random binning. *BMC Genomics*, 15(Suppl 6):S11, 2014.

[75] Liang Liu, Lili Yu, Dennis K Pearl, and Scott V Edwards. Estimating species phylogenies using coalescence times among sequences. *Systematic Biology*, 58(5):468–477, 2009.

[76] Elchanan Mossel and Sebastien Roch. Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(1):166–171, 2010.

[77] Josef Heled and Alexei J Drummond. Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution*, 27:570 – 580, 2010.

[78] Swati Patel, Rebecca T Kimball, and Edward L Braun. Error in phylogenetic estimation for bushes in the tree of life. *Journal of Phylogenetics and Evolutionary Biology*, 1(2):110, 2013.

[79] Alexandros Stamatakis. RAxML-VI-HPC: Maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690, 2006.

[80] David L Swofford. *PAUP*. Phylogenetic Analysis Using Parsimony (*and other methods). Version 4.* Sinauer Associates, 2003.