ASTRAL-III: Increased Scalability and Impacts of Contracting Low Support Branches

Chao Zhang, Erfan Sayyari, and Siavash Mirarab$^{(\boxtimes)}$

Department of Electrical and Computer Engineering,
University of California at San Diego, San Diego, USA
{chz069,esayyari,smirarab}@ucsd.edu

Abstract. Discordances between species trees and gene trees can complicate phylogenetics reconstruction. ASTRAL is a leading method for inferring species trees given gene trees while accounting for incomplete lineage sorting. It finds the tree that shares the maximum number of quartets with input trees, drawing bipartitions from a predefined set of bipartitions X. In this paper, we introduce ASTRAL-III, which substantially improves on ASTRAL-II in terms of running time by handling polytomies more efficiently, exploiting similarities between gene trees, and trimming unnecessary parts of the search space. The asymptotic running time in the presence of polytomies is reduced from $O(n^3k|X|^{1.726})$ for n species and k genes to $O(D|X|^{1.726})$ where $D = O(nk)$ is the sum of degrees of all *unique* nodes in input trees. ASTRAL-III enables us to test whether contracting low support branches in gene trees improves the accuracy by reducing noise. In extensive simulations and on real data, we show that removing branches with *very* low support improves accuracy while overly aggressive filtering is harmful.

Keywords: Phylogenomics · Incomplete lineage sorting. ASTRAL

1 Introduction

Reconstructing species phylogenies from a collection of input trees each inferred from a different part of the genome is becoming the standard practice in phylogenomics (e.g., [1–5]). This two-step approach stands in contrast to concatenation [6], where all of the sequences are combined into a supermatrix and analyzed in one maximum likelihood analysis. The two-step approach promises to effectively account for discordances between gene trees and the species tree [7] (but see recent literature for ongoing debates [8–11]) and is more efficient than statistical co-estimation of gene trees and the species tree [12] or site-based estimation of the species tree [13]. Among several causes of gene tree discordance [14], incomplete lineage sorting (ILS) is believed to be ubiquitous [15] and is extensively studied. ILS is typically modeled by the multi-species coalescent model (MSCM) [16,17], where branches of the species tree represent populations, and lineages are allowed to coalesce inside each branch; lineages that fail to coalesce at the root of each branch are moved to the parent branch.

J. Meidanis and L. Nakhleh (Eds.): RECOMB CG 2017, LNBI 10562, pp. 53–75, 2017.
DOI: 10.1007/978-3-319-67979-2_4

Several methods are proposed to infer a species tree from a collection of input trees (even though these trees need not be inferred from functional genes, following the conventions of the field, we will call them “gene trees”). Examples of summary methods include MP-EST [18], NJst [19], DISTIQUE [20], and STAR [21], which only use the topology of the input gene trees, and GLASS [22] and STEAC [21], which also uses the input branch lengths. While most methods need rooted gene trees as input, NJst and DISTIQUE can take unrooted input. These methods are all proved statistically consistent under the MSCM when the input gene trees are error-free, but no summary method is proved consistent when input trees are inferred from sequence data [23].

One of the statistically consistent methods under the MSCM is ASTRAL [24], which takes as input a collection of unrooted gene tree topologies and produces an unrooted species tree. ASTRAL uses dynamic programming to find the tree that shares the maximum number of induced quartet topologies with the collection of input gene trees. Since this problem is NP-Hard [25], ASTRAL solves a constrained version of the problem exactly, where the search space is limited to a predefined set of bipartitions X. In ASTRAL-I, the set X is the collection of all bipartitions in input gene trees. Showing that this space is not always large enough, ASTRAL-II [26] uses several heuristics to further augment the search space. Using the fact that for unrooted quartet trees the species tree always matches the most likely gene tree [27], ASTRAL is proved statistically consistent, even when solving the constrained problem, and its accuracy has been established in simulations [20,24,26,28,29]. ASTRAL-II has running time $O(nk|X|^2)$ for n species and k binary genes. Finally, ASTRAL has the ability to compute branch lengths in coalescent units [14] and a measure of branch support called local posterior probability [30]. Perhaps most importantly, ASTRAL and ASTRAL-II have been adopted by the community as one of the main methods of performing phylogenomics, and many biological analyses have adopted them.

ASTRAL-II has several shortcomings, some of which we address here by introducing ASTRAL-III. While ASTRAL-II can analyze datasets of 1,000 species and 1,000 genes on average in a day, ASTRAL-II has trouble scaling to many tens of thousands of input trees. Datasets with more than ten thousands genomic loci are already available (e.g., [3]) and with the increase in genome sequencing, more will be available in future. Moreover, being able to handle large numbers of input trees enables using multiple trees per locus (e.g., a Bayesian sample) as input to ASTRAL. The limited scalability of ASTRAL with k is because of a $\Theta(nk)$ factor in the running time that corresponds to scoring a potential node in the species tree against all nodes of the input gene trees. This computation does not exploit similarities between gene trees, a shortcoming that we fix in ASTRAL-III. Moreover, while ASTRAL-II can handle polytomies in input gene trees, in the presence of polytomies of maximum degree d_m, its running time inflates to $O(d_m^3k|X|^2) = O(n^3k|X|^2)$, which quickly becomes prohibitive for input trees with polytomies of large degrees. ASTRAL-III uses a mathematical trick to enable scoring of gene tree polytomies in time similar to binary nodes.

The ability to handle large polytomies in input gene trees is important for two reasons. On the one hand, some of the conditions that are conducive to ILS, namely shallow trees with many short branches, are also likely to produce gene sequence data that are identical between two species. A sensible gene tree (e.g., those produced by FastTree [31]) would leave the relationship between identical sequences unresolved (tools such as RAxML that output a random resolution take care to warn the user about such input data). On the other hand, all summary methods, including ASTRAL, are sensitive to gene tree estimation error [26,32–36]. One way of dealing with gene tree error, previously studied in the context of minimizing deep coalescence [37], is to contract low support branches in gene trees and use these unresolved trees as input to the summary method. While earlier studies found no evidence that this approach helps ASTRAL when the support is judged by SH-like FastTree support [26], no study has tested this approach with bootstrap support values. We will for the first time evaluate the effectiveness of contracting low support branches and show that conservative filtering of very low support branches does, in fact, help the accuracy. We note that the main competitors to ASTRAL, namely NJst [19] and its fast implementation, ASTRID [38], are not able to handle polytomies in input gene trees. ASTRAL-III makes it efficient to use unresolved gene trees as input to the species tree. Empirically, we observe that ASTRAL-III improves the running time compared to ASTRAL-II by a factor of 3X-4X for binary trees with large numbers of genes. Moreover, ASTRAL-III finishes on a dataset of 5,000 species and 500 genes in 18–30 h (24 on average). The ASTRAL-III software is publicly available at https://github.com/smirarab/ASTRAL.

2 Background and Notation

2.1 Notations and Definitions

We denote the set of species by L and let $n = |L|$. Let G be the set of k input gene trees. The set of quartet trees induced by any tree t is denoted by $Q(t)$. We refer to any subset of L as a cluster and refer to clusters with cardinality one as singletons. We define a partition as a set of clusters that are pairwise mutually exclusive (note that we abuse the term here, as the union of all clusters in a partition need not give the complete set). A bipartition (tripartition) is a partition with cardinality two (three); a partition with cardinality at least four corresponds to a polytomy and is referred to as a polytomy in this paper. Let X (the constraint bipartition set) be a set of clusters such that for each $A \in X$, we also have $L - A \in X$. We use Y to represent the set of all tripartitions examined in the ASTRAL dynamic programming:

$$Y = \{(A', A - A', L - A) | A' \subset A, A \in X, A' \in X, A - A' \in X\}.$$

We use $N(g)$ to represent the set of partitions correspondent to internal nodes in the gene tree g. We use E to denote the set of unique partitions and the number of times they appear in G:

$$E = \{(M, \sum_{g \in G} |N(g) \cap \{M\}|) | M \in N(g), g \in G\} \tag{1}$$

and we define D as the sum of the cardinalities of unique partitions in gene trees:

$$D = \sum_{(M,c) \in E} |M|. \tag{2}$$

Finally, we use $[d]$ to represent the set $\{1, 2 \ldots, d\}$.

2.2 Background on ASTRAL-I and ASTRAL-II

The problem addressed by ASTRAL is:

Given: a set G of input gene trees
Find: find the species tree t that maximizes $\sum_{g \in G} |Q(g) \cap Q(t)|$.

Lanford and Scornavacca recently proved this problem is NP-hard [25]. ASTRAL solves a constrained version of this problem where a set of clusters X restricts bipartitions that the output species tree may include (note $\forall A \in X : L - A \in X$).

To solve the constrained version, ASTRAL uses a dynamic programming method with the following recursive relation to obtain the optimal tree.

$$V(A) = \max_{(A'|A-A'|L-A) \in Y} V(A') + V(A - A') + w(A'|A - A'|L - A)$$

where the function $w(T)$ scores each tripartition $T = (A|B|C)$ against each node in each input gene tree. Let partition $M = (M_1|M_2| \ldots |M_d)$ represent an internal node of degree d in a gene tree. The overall contribution of T to the score of any species tree that includes T is:

$$w(T) = \sum_{g \in G} \sum_{M \in N(g)} \frac{1}{2} QI(T, M) \tag{3}$$

where, defining $a_i = |A \cap M_i|$, $b_i = |B \cap M_i|$, and $c_i = |C \cap M_i|$, we have:

$$QI((A|B|C), M) = \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \frac{a_i + b_j + c_k - 3}{2} a_i b_j c_k. \tag{4}$$

As previously proved [24], $QI(T, M)$ computes twice the number of quartet trees that are going to be shared between any two trees if one includes only T and the other includes only M. ASTRAL-II requires $\Theta(d^3)$ time for computing $QI(.)$, making the overall running time $O(n^3 k|Y|)$ with polytomies of unbounded degrees or $O(nk|Y|)$ in the absence of polytomies.

3 ASTRAL-III Algorithmic Improvements

Noting trivially that $|Y| < |X|^2$, the previously published running time analysis of ASTRAL-II was $O(nk|X|^2)$ for binary gene trees and $O(n^3 k|X|^2)$ for input trees with polytomies. A recent result by Kane and Tao [39] (motivated by the question raised in analyzing the ASTRAL algorithm) indicates that $|Y| \leq |X|^{3/log_3(27/4)}$. This result immediately gives a better upper bound:

Corollary 1. *ASTRAL-II runs in $O(nk|X|^{1.726})$ and $O(n^3k|X|^{1.726})$, respectively, with and without polytomies in gene trees.*

ASTRAL-III further improves this running time using three new features:

1. A new way of handling polytomies is introduced to reduce the running time for scoring a gene tree to $O(n)$, instead of $O(n^3)$, in the presence of polytomies, which reduces the total running time to $O(nk|X|^{1.726})$ irrespective of the gene tree resolution.
2. A polytree is used to represent gene trees, and this enables an algorithm that reduce the overall running time from $O(nk|X|^{1.726})$ to $O(D|X|^{1.726})$.
3. An A*-like algorithm is used to trim parts of the dynamic programming DAG.

In addition to these running time improvements, ASTRAL-III changes parameters of heuristics described in ASTRAL-II [26] to expand the size of set X for gene trees that include polytomies. We next describe each improvement in detail.

3.1 Efficient Handling of Polytomies

Recall that ASTRAL-II uses Eq. 4 to score a tripartition against a polytomy of size d in $\Theta(d^3)$ time. We now show this can be improved.

Lemma 1. *Let $QI(T, M)$ be twice the number of quartet tree topologies shared between an unrooted tree that only includes a node corresponding to the tripartition $T = (A|B|C)$ and another tree that includes only a node corresponding to a partition $M = (M_1|M_2|\dots|M_d)$ of degree d; then, $QI(T, M)$ can be computed in time $\Theta(d)$.*

Proof. In $\Theta(d)$ time, we can compute:

$$S_a = \sum_{i\in[d]} a_i \quad \text{and} \quad S_{a,b} = \sum_{i\in[d]} a_i b_i \tag{5}$$

where $a_i = |A \cap M_i|$ and $b_i = |B \cap M_i|$; ditto for S_b, S_c, $S_{a,c}$ and $S_{b,c}$. Equation 4, as proved before [26], computes twice the number of quartet tree topologies shared between an unrooted tree with internal node T and another tree with one internal node M. Equation 4 can be rewritten using these intermediate sums as:

$$\begin{aligned} QI((A|B|C), M) = &\sum_{i\in[d]} \binom{a_i}{2}((S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i) \\ &+ \sum_{i\in[d]} \binom{b_i}{2}((S_a - a_i)(S_c - c_i) - S_{a,c} + a_i c_i) \\ &+ \sum_{i\in[d]} \binom{c_i}{2}((S_a - a_i)(S_b - b_i) - S_{a,b} + a_i b_i) \end{aligned} \tag{6}$$

(the derivation is given in the Appendix A). Computing Eq. 6 instead of Eq. 4 clearly reduces the running time to $\Theta(d)$ instead of $\Theta(d^3)$. □

ASTRAL needs to score each of the $|Y|$ tripartitions considered in the dynamic programming against each internal node of each input gene tree. The sum of degrees of k trees on n leaves is $O(nk)$ (since that sum can never exceed the number of bipartitions in gene trees) and thus:

Corollary 2. *Scoring a tripartition (i.e., computing w) can be done in $O(nk)$.*

3.2 Gene Trees as a Polytree

ASTRAL-II scores each dynamic programming tripartition against each individual node of each gene tree. However, nodes that are repeated in several genes need not be recomputed. Recalling the definitions of E and D (Eqs. 1 and 2),

Lemma 2. *The score of a tripartition $T = (A|B|C)$ against all gene trees (i.e., the $w(T)$ score) can be computed in $\Theta(D)$.*

Proof. In ASTRAL-III, we keep track of nodes that appear in multiple trees. This enables us to reduce the total calculation by using multiplicities:

$$w(T) = \sum_{(M,c)\in E} c \times QI(T, M). \tag{7}$$

We achieve this in two steps. In the first step, for each distinct gene tree cluster W, we compute the cardinality of the intersection of W and sets A, B, and C once using a depth first search with memoization. Let $children(W)$ denote the set of children of W in an arbitrarily chosen tree $g \in G$ containing W. Then, we have the following recursive relation:

$$|W \cap A| = \sum_{Z\in children(W)} |Z \cap A| \tag{8}$$

(ditto for $|W \cap B|$ and $|W \cap C|$). All such intersection values can be computed in a post-order traversal of a polytree. In this polytree, all unique clusters in the gene trees are represented as vertices and parent-child relations are represented as edges; note that when a cluster has different children in two different input trees, we arbitrary choose one set of children and ignore the others. The polytree will include no more than D edges; thus, the time complexity of traversing this polytree and computing Eq. 8 for all nodes is $\Theta(D)$. Once all intersections are computed, in the second step, we simply compute the sum in Eq. 7. Each $QI(.)$ computation requires $\Theta(d)$ by Lemma 1. Recalling that $D = \sum_{(M,c)\in E} |M|$, it is clear that computing Eq. 7 requires $\Theta(D)$. Therefore, both steps can be performed in $\Theta(D)$. □

Theorem 1. *The ASTRAL-III running time is $O(D|X|^{1.726})$ for both binary and unresolved gene trees.*

Proof. By results of Kane and Tao [39], the size of the set Y is $O(|X|^{1.726})$, and for each element in Y, by Lemma 2, we require $O(D)$ to compute the weights, regardless of the presence or absence of polytomies. The running time of ASTRAL is dominated by computing the weights [26]. Thus, the overall running time is $O(D|Y|) = O(D|X|^{1.726})$.

3.3 Trimming of the Dynamic Programming

Our last feature does not improve theoretical running time but can result in some improvements in the experimental running time. Our main insight is that $U(A) = \frac{w(A|A|L)}{2} - \frac{w(A|A|A)}{3}$ is an upperbound of $V(A)$ (see the Appendix A for proof). Since $V(A) \leq U(A)$, for any $(A'|A-A'|L-A') \in Y$ and $(A''|A-A''|L-A'') \in Y$, we no longer need to recursively compute $V(A'')$ and $V(A - A'')$ when:

$$\begin{aligned} U(A'') + U(A - A'') + w(A''|A - A''|L - A) \leq \\ V(A') + V(A - A') + w(A'|A - A'|L - A) \end{aligned} \tag{9}$$

Thus, in ASTRAL-III we trim the DAG of the memoized recursive dynamic programming when this calculation indicates that a path has no chance of improving the final score. To heuristically improve the efficiency of this approach, we order all $(A'|A-A'|L-A) \in Y$ according to $U(A')+U(A-A')+w(A'|A-A'|L-A)$.

4 Experimental Setup

Using simulation studies and on real data, we study two research questions:

RQ1: Can contracting low support branches improve the accuracy of ASTRAL?
RQ2: How does ASTRAL-III running time compare to ASTRAL-II for large polytomies and many gene trees?

Note that addressing RQ1 in a scalable fashion is made possible only through the running time improvements of ASTRAL-III.

4.1 Datasets

Avian Biological Dataset: Neoaves have gone through a rapid radiation, and therefore, have extremely high levels of ILS [3]. A dataset of 48 whole-genomes was used to resolve this rapid radiation [3]. MP-EST run on 14,446 gene trees (exons, introns, and UCEs) produced a tree that conflicted with strong evidence from the literature and other analyses on the dataset (e.g., the Passerimorphae/-Falcons/Seriemas grade was not recovered). This motivated the development of the statistical binning method to reduce the impacts of gene tree error [32,33]. MP-EST run on binned gene trees produced results that were largely congruent with the concatenation and other analyses. Here, we test if simply contracting low support branches of gene trees produces a tree that is congruent with other analyses and the literature. This analysis is made possible because ASTRAL-III can handle datasets with a large number of polytomies and large k efficiently.

Simulated Avian-Like Dataset: We use a simulated dataset that was previously used in the statistical binning paper [32] to emulate the biological avian dataset. Since estimating the true branch lengths in coalescent units are hard, three versions of this dataset are available: 1X is the default version, whereas

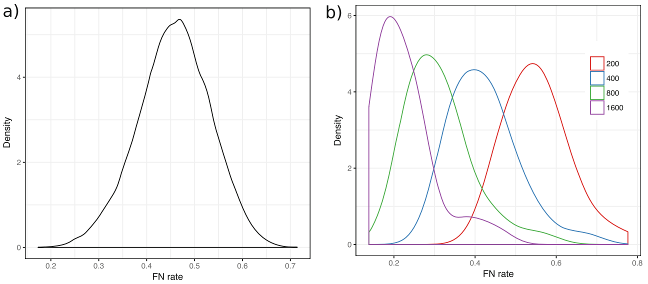


Fig. 1. Properties of the S100 dataset. (a) The density plot of the amount of true gene discordance measured by the FN rate between the true species tree and the true gene trees. (b) The density plot of gene tree estimation error measured by FN rate between true gene trees and estimated gene trees for different set of sequence lengths.

0.5X divides each branch length in half (increasing ILS) and 2X multiplies them by 2 (reducing ILS). The amount of true discordance (ILS), measured by the average RF distances between true species tree and true gene trees, is moderate at 0.35 for 2X, high at 0.47 for 1X, and very high at 0.59 for 0.5X. Moreover, to study the impact of gene tree estimation error, sequence lengths were varied to create four conditions: very high error with 250 bp alignments (0.67 RF distance between true gene trees and estimated gene trees), high error with 500 bp (0.54 RF), medium error with 1000 bp (0.39 RF) and moderate error with 1500 bp (0.30 RF). We use 1000 gene trees, and 20 replicates per condition. Gene trees are estimated using RAxML [40] with 200 replicates of bootstrapping.

SimPhy-Homogeneous (S100): We simulated 50 replicates of a 101-taxon dataset using SimPhy [41] under the MSCM, where each replicate has a different species tree. In order to generate the species trees, we used the birth-only process with birth rate 10^{-7}, fixed haploid effective population size of 400K, and the number of generations sampled from a log-normal distribution with mean 2.5 M. For each replicate, 1000 true gene trees are simulated under the MSCM (the exact simulation commands are given in Appendix B and parameters are shown in Table 2). The amount of ILS, measured by the false-negative (FN) rate between true species trees and true gene trees, mostly ranged between 0.3 and 0.6 with an average of 0.46 (Fig. 1). We use Indelible [42] to simulate the nucleotide sequences along the gene trees using the GTR evolutionary model [43] with 4 different fixed sequence lengths: 1600, 800, 400, and 200 bp (Table 2). We then use FastTree2 [31] to estimate both ML and 100 bootstrapped gene trees under the GTR model for each gene of each replicate ($>$ 2000200 runs in total). Gene tree estimation error, measured by the FN rate between the true gene trees and the estimated gene trees, depended on the sequence length as shown in Fig. 1 (0.55, 0.42, 0.31, and 0.23 on average for 200 bp, 400 bp, 800 bp, and 1600 bp, respectively). We sample 1000, 500, 200, or 50 genes to generate datasets with varying numbers of gene trees.

4.2 Methods and Evaluation

We compare ASTRAL-III (version 5.2.5) to ASTRAL-II (version 4.11.1) in terms of running time. To address RQ1, we draw the bootstrap support values on the ML gene trees using the newick utility package [44]. We then contract branches with bootstrap support up to a threshold (0, 3, 5, 7, 10, 20, 33, 50, and 75%,) using the newick utility and use these contracted gene trees as input to ASTRAL. Together with the original set, this creates 10 different ways to run ASTRAL.

To measure the accuracy of estimated species trees, we use False Negative (FN) rate. Note that in all our species tree comparisons, FN rate is equivalent to normalized Robinson-Foulds (RF) [45] metric, since the ASTRAL species trees are fully resolved. All running times are measured on a cluster with servers with Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50 GHz; each run was assigned to a single process, sharing cache and memory with other jobs.

5 Results

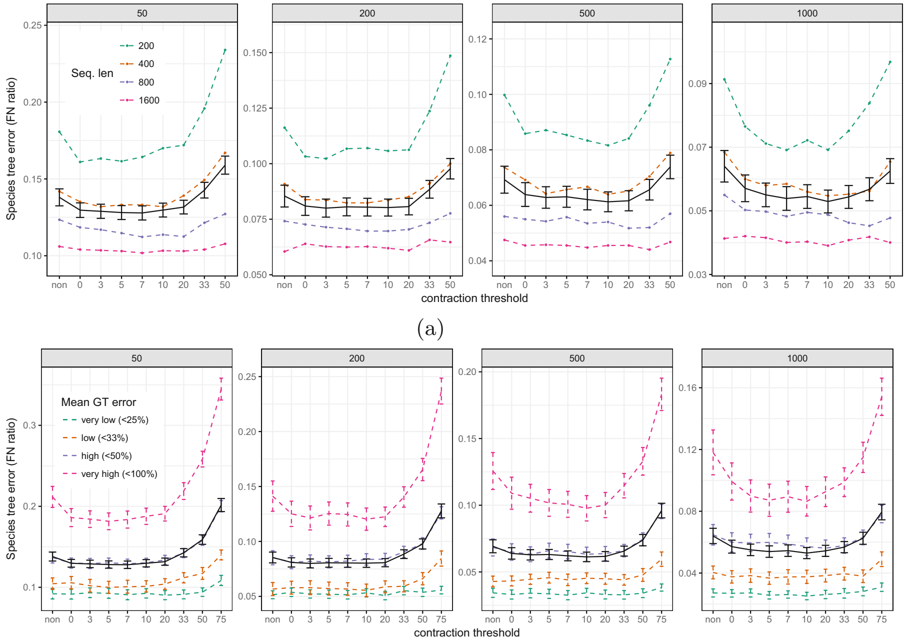
5.1 Impact of Contracting Low Support Branches on Accuracy

We investigate the impact of contracting branches with low support (RQ1) on our two simulated datasets (avian and S100) and on the real avian dataset.

S100: On this dataset, contracting *very* low support branches in most cases improves the accuracy (Table 1 and Fig. 2); however, the excessive removal of

Table 1. Species tree error (FN ratio) for all model conditions of the S100 dataset, with true gene trees (*true*), no filtering (*non*), and all filtering thresholds (*columns*).

Genes	Alignment	true	non	0	3	5	7	10	20	33	50	75
50	200 bp	7.0	18.0	**16.1**	16.5	16.3	16.5	16.9	17.3	19.9	23.5	31.6
50	400 bp		14.2	13.5	13.3	13.3	13.4	**13.2**	14.1	15.0	16.6	21.0
50	800 bp		12.3	11.9	11.7	11.6	11.3	11.4	**11.2**	12.2	12.9	15.9
50	1600 bp		10.6	10.4	10.4	10.3	**10.1**	10.3	10.4	10.5	10.8	12.4
200	200 bp	3.7	11.6	10.5	**10.3**	10.8	10.6	10.8	10.7	12.5	15.3	21.5
200	400 bp		9.2	8.4	**8.3**	**8.3**	8.3	8.4	8.6	9.2	10.1	13.6
200	800 bp		7.4	7.3	7.2	7.1	**7.0**	**7.0**	7.1	7.4	7.7	9.1
200	1600 bp		6.1	6.4	6.3	6.3	6.3	6.2	**6.1**	6.6	6.5	7.4
500	200 bp	2.4	9.9	8.8	8.8	8.7	**8.5**	**8.5**	8.6	9.8	11.8	16.7
500	400 bp		7.3	7.1	6.6	6.6	6.7	**6.5**	6.6	7.0	8.0	10.8
500	800 bp		5.6	5.5	5.5	5.6	5.4	5.4	**5.3**	**5.3**	5.7	6.6
500	1600 bp		4.8	4.5	4.6	4.6	4.5	4.6	4.6	**4.4**	4.8	5.3
1000	200 bp	1.5	9.1	8.0	7.6	7.3	7.3	**7.1**	7.6	8.6	10.2	13.6
1000	400 bp		6.9	6.0	5.8	5.9	5.7	**5.6**	**5.6**	5.8	6.8	8.5
1000	800 bp		5.5	5.0	5.0	4.9	5.0	4.9	4.7	**4.6**	4.8	5.8
1000	1600 bp		4.1	4.2	4.1	4.0	4.0	**3.9**	4.1	4.1	4.1	4.5



(a)

(b)

Fig. 2. The error in species trees estimated by ASTRAL-III on the S100 dataset given $k = 50$, 200, 500, or 1000 genes (*boxes*) and with full FastTree gene trees (*non*) or trees with branches with $\leq \{0, 3, 5, 7, 10, 20, 33, 50\}\%$ support contracted (*x-axis*). Average FN error and standard error bars are shown for all 50 replicates with the four alignment lengths combined (*black solid line*); average FN error broken down by alignment length (a) or gene tree error (b) is also shown (*dashed colored lines*). In (b) we divide the replicates based on their average gene tree error (normalized RF) into four categories: $[0, \frac{1}{4}]$,$(\frac{1}{4}, \frac{1}{3}]$,$(\frac{1}{3}, \frac{1}{2}]$,$(\frac{1}{2}, 1]$. (Color figure online)

branches with high, moderate or occasionally even low support degrades the accuracy. The threshold where contracting starts to become detrimental depends on the condition, especially the number of gene trees and the alignment length.

As the number of genes increases, the optimal threshold for contracting also tends to increase. Combining all model conditions, the error continues to drop until a 10% contracting threshold with 1000 genes, whereas no substantial improvement is observed after contracting branches with 0% support for 50 genes. The alignment length and gene tree error also impact the effect of contraction. For short alignments (200 bp) and 1000 genes, contracting branches with up to 10% support reduces the species tree error by 21% (from 8.9% with no contraction to 7.0%). As alignment length grows (and gene tree error decreases), benefits of gene tree contraction diminish, so that with 1600 bp genes, the reduction in error is merely from 4.1% to 3.7%. While aggressive filtering at 33% or higher sometimes increases the error compared to no filtering, filtering at 10% is either neutral or beneficial on average for all conditions in this dataset.

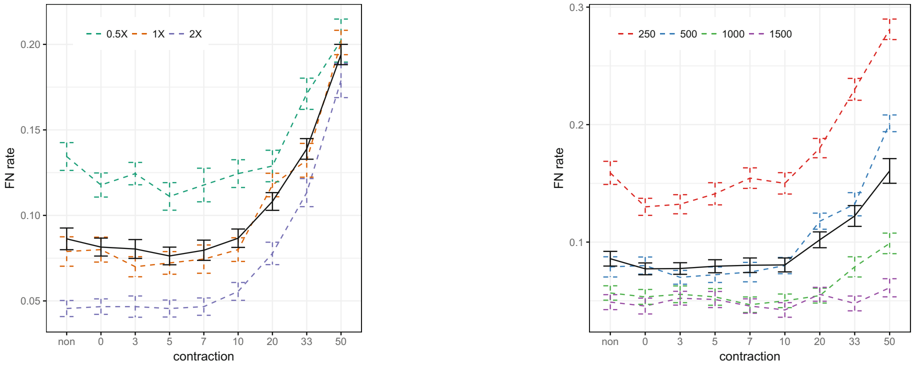


Fig. 3. Species trees error for ASTRAL-III on the avian dataset given $k = 1000$ genes with (*left*) fixed sequence lengths = 500 and varying levels of ILS, or (*right*) fixed ILS (1X) and varying sequence length, in each case both with full FastTree gene trees (*non*) or trees with branches with $\leq \{0, 3, 5, 7, 10, 20, 33, 50\}\%$ support contracted (*x-axis*). Average FN error and standard error bars are shown for all 20 replicates (*black solid line*) and also for each model condition separately (*dashed color lines*). (Color figure online)

Avian-Like Simulations: On the avian dataset, overall, contracting low support branches helps accuracy marginally, but the extent of improvements depends on the model condition (Fig. 3). We first fix the sequence length to 500 bp and allow the amount of ILS to change (e.g., from moderate with 2X to very high with 0.5X). With moderate ILS (2X), we see no improvements as a result of contracting low support branches, perhaps because the average error is below 5% even with no contraction. Going to high and very high ILS, we start to see improvements with contracted gene trees. For example, removing branches of up to 5% support reduces the error from 13% to 11% with 0.5X, and from 8% to 7% for the 1X condition. Just like the S100 dataset, aggressive filtering reduces the accuracy, but here, thresholds of 20% and higher seem to be detrimental. When ILS is fixed to 1X and sequence length is varied (Fig. 3), contracting is helpful mostly with short sequences (e.g., 250 bp). With longer sequences, where gene tree estimation error is low, little or no improvement in accuracy is obtained. The best accuracy is typically observed by contracting at 0–5%. Overall, the gains in accuracy comparing no contraction to contraction at 0% are statistically significant with p-value 0.042 (are close to significant with p-values 0.087 and 0.058 for 3% and 5% thresholds) according to a two-way ANOVA test with the sequence length and ILS levels as extra variables. Irrespective of significance, the improvements are not large on this dataset.

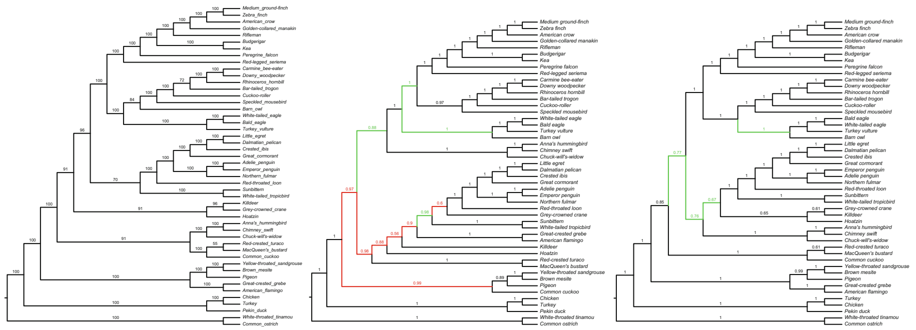


Fig. 4. Avian biological results on 14,446 unbinned gene trees. Species trees are shown for the TENT RAxML concatenation tree [3] (*left*), ASTRAL-III tree with no contraction (*middle*), and with 33% contraction (*right*). ASTRAL-III branches conflicting with the TENT tree are in color; red indicates disagreement with strong evidence [3]. (Color figure online)

Avian Biological Dataset: On the avian dataset with 14 K genes, ASTRAL-III managed to complete with 0%, 33%, 50%, and 75% thresholds in 48 h. Results on the runs that did finish are very interesting. The ASTRAL-III tree with no contraction had 11 and 9 branch differences respectively with the TENT and the binned MP-EST analyses from the original paper [3]. Some of these differences were on strong results (Fig. 4) from the avian dataset (e.g., the Columbea group). After contracting branches below 33% BS, the ASTRAL-III tree had only 4 and 3 branch differences, respectively with TENT and the binned MP-EST trees; these differences were among the branches that were deemed unresolved by Jarvis *et al.* and changed among their analyses as well [3]. ASTRAL-III obtained on collapsed gene trees agreed with all major new findings of Jarvis et al. [3]. For example, at 33% filtering, the novel Columbea group was corroborated, whereas the unresolved tree completely missed this important clade (Fig. 4).

5.2 Running Time Improvements

The Impact of the Number of Genes (k): We evaluate the improvement of ASTRAL-III compared to ASTRAL-II on the avian simulated dataset, changing the number of genes from 2^8 to 2^{16}. We allow each replicate run to take up to two days. ASTRAL-II is not able to finish on the dataset with $k = 2^{16}$, while ASTRAL-III finishes on all conditions. ASTRAL-III improves the running time over ASTRAL-II and the extent of the improvement depends on k (Fig. 5). With 1000 genes or more, there is at least a 2.7X improvement. With 2^{13} genes, the largest value where both versions could run, ASTRAL-III finishes on average four times faster than ASTRAL-II (190 versus 756 min). Moreover, fitting a line to the average running time in the log-log scale graph reveals that on this dataset, the running time of ASTRAL-III on average grows as $O(k^{2.04})$, which is better than that of ASTRAL-II at $O(k^{2.27})$, and both are better than the theoretical worst case, which is $O(k^{2.726})$.

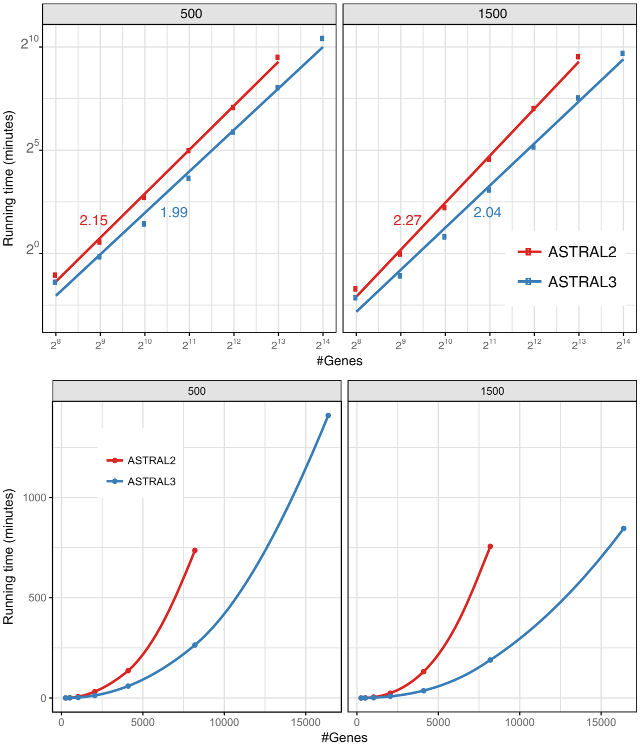


Fig. 5. Average running time of ASTRAL-II versus ASTRAL-III on the avian dataset with 500 bp or 1500 bp alignments with varying numbers of gens (k), shown both in log-log scale (*top*) and normal scale (*bottom*). A line (*top*) or a LOESS curve (*bottom*) is fit to the data points. ASTRAL-II could not finish on 2^{14} genes in the allotted 48 h time. Line slopes are shown for the log-log plot. Averages are over 4 runs.

The Impact of Polytomies: ASTRAL-III has a clear advantage compared to ASTRAL-II with respect to the running time when gene trees include polytomies (Fig. 6). Since ASTRAL-II and ASTRAL-III can potentially weight different numbers of tripartitions, we show the running time per weight calculation (i.e., Eq. 3). As we contract low support branches and hence increase the prevalence of polytomies, the weight calculation time quickly grows for ASTRAL-II, whereas, in ASTRAL-III, the weight calculation time remains flat, or even decreases.

6 Discussion

Comparison to True Gene Trees: Although we observed improvements in the tree accuracy with contracting low support branches, the gap between performance on true gene trees and estimated gene trees remains wide (Table 1). On the S100 dataset, respectively for 50, 200, 500, and 1000 genes, the average error with 1600 bp gene trees were 10.1%, 6.1%, 4.4%, and 3.9% compared

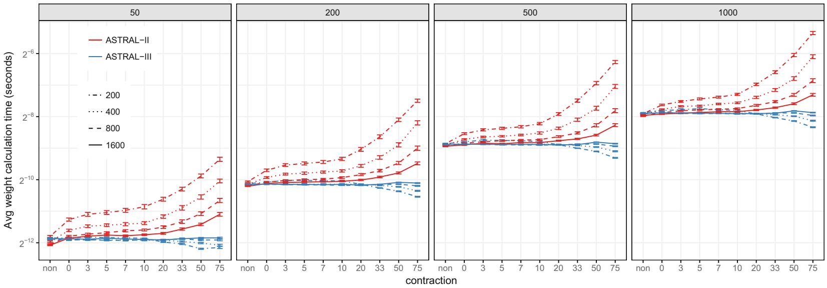


Fig. 6. Average and standard error of running times of ASTRAL-II and ASTRAL-III on the S100 dataset for scoring a single weight (Eq. 3). Running time is in log scale for varying numbers of gene trees (*boxes*) and sequence length (200, 400, 800, and 1600).

to 7.0%, 3.7%, 2.4%, and 1.5% with true gene trees. Thus, while contracting low support branches helps in addressing gene tree error, it is not a panacea. Improved methods of gene tree estimation remain crucial as ever before. Our results also indicate that in the presence of noisy gene trees, increased numbers of genes are needed to achieve high accuracy. For example, on the S100 dataset, with 1000 gene trees of only 200 bp and contracting with a 10% threshold, the species tree error was 7.1%, which matched the accuracy with only 50 true gene trees. The increased data requirement for noisy genes encourages the use of many thousands of gene trees, making scalability gains of ASTRAL-III more relevant.

Arbitrary Resolutions and 0% Filtering: Interestingly, in most datasets, the most substantial improvements were observed when only 0% BS branches were removed, and one can assume that such branches are essentially resolved randomly. As the use of Ultra Conserved Elements [9] continues to gain in popularity, instances where two or more taxa have identical sequences in some genes will become more prevalent. Many tree estimation methods generate binary trees even under such conditions. Removing branches that are arbitrarily resolved make sense and, as our results indicate, will improve accuracy.

Statistical Consistency: While removing branches with low support can improve the accuracy empirically, its theoretical justification is less clear. In principle, branches that have low support are not necessarily expected to have a random distribution among gene trees. Thus, while the empirical results could support the use of (conservative) filtering, it must be understood that the resulting procedure may introduce small biases. Whether ASTRAL remains statistically consistent given contracted gene trees should be studied in future work.

Other Strategies: Beyond removing branches with low bootstrap support, several other strategies could be employed. Our previous results [24] indicate that simply using a concatenation of all bootstrap gene trees as input to ASTRAL

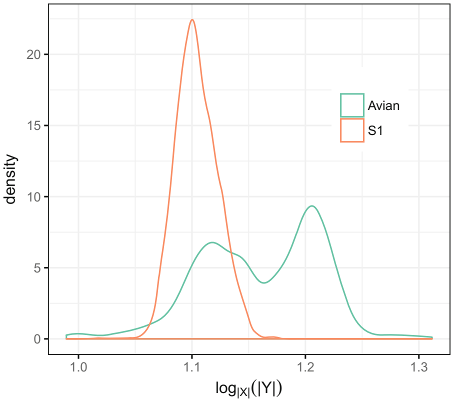


Fig. 7. The density plots of $\log_X |Y|$ across all ASTRAL-III runs reported in this paper. Size of the dynamic programming space Y is never above $|X|^{1.312}$ here.

increases error, perhaps because of the increased noise in bootstrap replicates [30,34]. However, it is possible that a fixed sized sample from a Bayesian estimate of each gene tree distribution would improve accuracy.

$|Y|$: The ASTRAL-III running time analysis of $O(|X|^{1.726})$ is based on the fact that $|Y| \leq |X|^{1.726}$ [39]. However, this upper bound is for specialized formations of the set X. Empirically, as the size of set X increases, the size of $|Y|$ in ASTRAL-III does not increase as fast as the worst-case scenario implies. Across all of our ASTRAL-III runs in this paper, $|Y|$ ranged mostly between $|X|^{1.05}$ and $|X|^{1.25}$, with an average of $|X|^{1.11}$ (Fig. 7). Thus, the average running time of ASTRAL seems closer to $O(D|X|^{1.1})$, though, the exact value depends on the dataset. The size of X is not currently controlled to be a polynomial function of n and k, but such constraints can be imposed in future versions of ASTRAL.

Large n: To assess scalability limits of ASTRAL-III, we tested it on 4 replicates of a dataset with 5,000 species and 500 true gene trees (simulation procedure described in Appendix B). ASTRAL-III took between 18 and 30 h to run on this dataset (24 h on average). We also ran ASTRAL-III on similar datasets with 1,000 and 2,000 species. Average over our four replicates, ASTRAL running time increases as $O(n^{1.9})$. Our attempts to analyze 10K species within 72 h failed. Future work should reproduce these results with more replicates.

Acknowledgments. This work was supported by the NSF grant IIS-1565862 to SM and ES. Computations were performed on the San Diego Supercomputer Center (SDSC) through XSEDE allocations, which is supported by the NSF grant ACI-1053575.

A Derivations

Derivation of Equation 6: First note that:

$$\begin{aligned} QI((A|B|C),M) &= \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \frac{a_i+b_j+c_k-3}{2} a_i b_j c_k \\ &= \sum_{i\in[d]} \binom{a_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} b_j c_k \\ &+ \sum_{i\in[d]} \binom{b_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} a_j c_k \\ &+ \sum_{i\in[d]} \binom{c_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} a_j b_k \ . \end{aligned} \tag{10}$$

Now, we note that:

$$\begin{aligned} &\sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} b_j c_k \\ &= \sum_{j\in[d]-\{i\}} b_j \sum_{k\in[d]-\{i,j\}} c_k \\ &= \sum_{j\in[d]-\{i\}} b_j (S_c - c_i - c_j) \\ &= -b_i(S_c - c_i - c_i) + \sum_{j\in[d]} b_j(S_c - c_i - c_j) \\ &= 2b_i c_i - S_c b_i + S_b S_c - S_b c_i - S_{b,c} \\ &= (S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i \end{aligned} \tag{11}$$

Replacing this (and similar calculations for other terms) in Eq. 10 directly gives us the Eq. 6:

$$\begin{aligned} QI((A|B|C),M) = &\sum_{i\in[d]} \binom{a_i}{2} ((S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i) \\ &+ \sum_{i\in[d]} \binom{b_i}{2} ((S_a - a_i)(S_c - c_i) - S_{a,c} + a_i c_i) \\ &+ \sum_{i\in[d]} \binom{c_i}{2} ((S_a - a_i)(S_b - b_i) - S_{a,b} + a_i b_i) \end{aligned} \tag{12}$$

Derivation of the Upperbound *U(Z)*: In ASTRAL, $V(Z)$ denotes the total contribution to the support of the best rooted tree T_Z on taxon set Z, where each quartet tree in the set of input gene trees contributes 0 if it conflicts with T_Z or only intersects it with one leaf, and otherwise contributes 1 or 2, depending on the number of nodes in T_Z it maps to. Let $U(Z)$ be the sum of max possible support of each quartet tree in the gene trees with respect to any resolution T_Z of set Z, allowing the resolution to change for each gene tree. In other words, let $Q(Z)$ be the set of quartets that would be resolved one way or another in any resolution of Z, and note that these are quartets that include two or leaves in Z; then, $U(Z)$ is the number of resolved gene tree quartets that would match *some* resolution of Z and are included in $Q(Z)$. More formally,

$$U(Z) = \sum_{g \in G} \sum_{M \in N(g)} \sum_{T \in Q(Z)} QI(T, M) \ ,$$

where

$$\begin{aligned} Q_1(Z) &= \{\{\{v,w\},\{x\},\{y\}\} | \{x,y\} \subset Z, \{v,w\} \subset L - \{x,y\}\} \ , \\ Q_2(Z) &= \{\{\{v,w\},\{x\},\{y\}\} | \{v,w,x\} \subset Z, y \in L - Z\} \ , \text{ and} \\ Q(Z) &= Q_1(Z) \cup Q_2(Z) \ , Q_1(Z) \cap Q_2(Z) = \emptyset \ . \end{aligned}$$

Clearly, $V(Z) \leq U(Z)$ (equality can be achieved only if all gene trees are compatible with some resolution of Z). Then, letting $d = |M|$ and defining $z_i = |Z \cap M_i|$ and $l_i = |L \cap M_i| = |M_i|$, we have

$$\begin{aligned} & \sum_{\{A,B,C\} \in Q(Z)} QI((A|B|C), M) \\ = & \sum_{\{A,B,C\} \in Q_1(Z)} QI((A|B|C), M) + \sum_{\{A,B,C\} \in Q_2(Z)} QI((A|B|C), M) \\ = & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i\}-[j]} \binom{l_i}{2} z_j z_k \\ + & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i\}-[j]} \binom{z_i}{2} (z_j(l_k - z_k) + (l_j - z_j) z_k) \\ = & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \binom{l_i}{2} \frac{z_j z_k}{2} \\ + & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \binom{z_i}{2} \frac{z_j(l_k - z_k) + (l_j - z_j) z_k}{2} \\ = & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \binom{l_i}{2} \frac{z_j z_k}{2} \\ + & \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \binom{z_i}{2} z_j (l_k - z_k) \ . \end{aligned} \tag{13}$$

Notice that based on Eq. 4,

$$\begin{aligned}
&\frac{QI((Z|Z|L),M)}{2} - \frac{QI((Z|Z|Z),M)}{3} = \\
&\frac{1}{2}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} z_i z_j l_k \frac{z_i+z_j+l_k-3}{2} = \\
-&\frac{1}{3}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} z_i z_j z_k \frac{z_i+z_j+z_k-3}{2} = \\
&\frac{1}{2}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} (\binom{z_i}{2} z_j l_k + z_i\binom{z_j}{2} l_k + z_i z_j \binom{l_k}{2}) \\
-&\frac{1}{3}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} (\binom{z_i}{2} z_j z_k + z_i\binom{z_j}{2} z_k + z_i z_j \binom{z_k}{2}) = \\
&\frac{1}{2}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} (\binom{z_i}{2} z_j l_k + \binom{z_i}{2} z_j l_k + \binom{l_i}{2} z_j z_k) \\
-&\frac{1}{3}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} (\binom{z_i}{2} z_j z_k + \binom{z_i}{2} z_j z_k + \binom{z_i}{2} z_j z_k) = \\
&\frac{1}{2}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} (\binom{l_i}{2} z_j z_k + 2\binom{z_i}{2} z_j l_k) \\
-&\frac{1}{3}\sum_{i\in[\,d\,]}\sum_{j\in[\,d\,]-\{i\}}\sum_{k\in[\,d\,]-\{i,j\}} 3\binom{z_i}{2} z_j z_k = \\
&\sum_{A,B,C\in Q(Z)} QI((A|B|C),M)\;.
\end{aligned} \tag{14}$$

(going from the fourth term to the fifth is accomplished by changing the order of sums). Therefore,

$$\begin{aligned}
U(Z) &= \sum_{g\in G}\sum_{M\in N(g)} (\frac{QI((Z|Z|L),M)}{2} - \frac{QI((Z|Z|Z),M)}{3}) \\
&= \frac{w(Z|Z|L)}{2} - \frac{w(Z|Z|Z)}{3}\;.
\end{aligned} \tag{15}$$

B Simulations and Commands

Simulation Setup

S100: In order to generate the gene trees and species trees using the Simphy we use this command:

Table 2. Species tree and gene tree generation parameters used for simphy [41], and sequence evolution parameters for the GTR model used for Indelible [42] for the S100 dataset.

Parameter name	Parameter value
Speciation rate	0.0000001
Extinsion rate	0
Number of leaves	100
Ingroup divergence to the ingroup ratio	1.0
Generations	LogN(1.470055e + 01, 2.500000e-01)
Haploid effective population size	400000
Global substitution rate	LogN(−1.727461e + 01, 6.931472e-01)
Lineage specific rate gamma shape	LogN(1.500000e + 00, 1)
Gene family specific rate gamma shape	LogN(1.551533e + 00, 6.931472e-01)
Gene tree branch specific rate gamma shape	LogN(1.500000e + 00, 1)
Seed	9644
Sequence length	1600, 800, 400, 200
Sequence base frequencies	Dirichlet(A = 36, C = 26, G = 28, T = 32)
Sequence transition rates	Dirichlet(TC = 16, TA = 3, TG = 5, CA = 5, CG = 6, AG = 15)

```
simphy −rs 50 −rl f:1000 −rg 1 −sb f:0.0000001 −sd f:0
−st ln:14.70055,0.25 −sl f:100 −so f:1 −si f:1 −sp
f:400000 −su ln:−17.27461,0.6931472 −hh f:1 −hs ln:1.5,1
−hl ln:1.551533,0.6931472 −hg ln:1.5,1 −cs 9644 −v 3
−o ASTRALIII −ot 0 −op 1 −od 1
```

Larege-*n* Simulated Dataset: In order to compare running time performances of ASTRAL-II and ASTRAL-III, we created another dataset with very large numbers of species using Simphy and under the MSCM. Since we are only comparing running times, we only use true gene trees to infer the ASTRAL species trees. We have three sub-datasets with 5000, 2000, and 1000 species (plus one outgroup). Each sub-dataset has 4 replicates, and each replicate has a different species tree with 500 gene trees. Species trees are generated based on the birth-death process with birth and date rates from log uniform distributions. We sampled the number of generations and effective population size from log normal and uniform distributions respectively such that we have medium amounts of ILS. The average FN rates between the true gene trees and the species tree ranges between 4% and 23% for 1K, between 21% and 58% for 2K, and between 21% and 33% for 5K.

In order to generate the gene trees and true species trees using the Simphy we use parameters given in Table 3 and the following command.

1K:

```
simphy −rs 20 −rl f:1000 −rg 1 −sb lu:0.0000001,0.000001 −sd
lu:0.0000001,sb −st ln:16,1 −sl f:1000 −so f:1 −si f:1 −sp
u:10000,1000000 −su ln:−17.27461,0.6931472 −hh f:1 −hs ln:1.5,1 −hl
ln:1.551533,0.6931472 −hg ln:1.5,1 −cs 9644 −v 3 −o 5k.species −ot 0
−op 1 −od 1
```

Table 3. Species tree and gene tree generation parameters in Simphy [41] for 1K-taxon, 2K-taxon and 5K-taxon datasets

Parameter Name	Parameter value
Speciation rate	LogU[1.000000e-07, 1.000000e-06)
Extinsion rate	LogU[1.000000e-07, SB)
Locus trees	1000
Gene trees	1
Number of leaves	1000, 2000, or 5000
Ingroup divergence to the ingroup ratio	1.0
Generations	LogN(16, 1)
Haploid effective population size	Uniform[10000, 1000000]
Global substitution rate	LogN(−1.727461e + 01, 6.931472e-01)
Lineage specific rate gamma shape	LogN(1.500000e + 00,1)
Gene family specific rate gamma shape	LogN(1.551533e + 00, 6.931472e-01)
Gene tree branch specific rate gamma shape	LogN(1.500000e + 00, 1)
Seed	9644

2K:

```
simphy −rs 20 −rl f:1000 −rg 1 −sb lu:0.0000001,0.000001 −sd
lu:0.0000001,sb −st ln:16,1 −sl f:2000 −so f:1 −si f:1 −sp
u:10000,1000000 −su ln:−17.27461,0.6931472 −hh f:1 −hs ln:1.5,1 −hl
ln:1.551533,0.6931472 −hg ln:1.5,1 −cs 9644 −v 3 −o 5k.species −ot 0
−op 1 −od 1
```

5K:

```
simphy −rs 20 −rl f:1000 −rg 1 −sb lu:0.0000001,0.000001 −sd
lu:0.0000001,sb −st ln:16,1 −sl f:5000 −so f:1 −si f:1 −sp
u:10000,1000000 −su ln:−17.27461,0.6931472 −hh f:1 −hs ln:1.5,1 −hl
ln:1.551533,0.6931472 −hg ln:1.5,1 −cs 9644 −v 3 −o 5k.species −ot 0
−op 1 −od 1
```

Commands

Contracting Branches: In order to contract gene tree branches with bootstrap up to a certain threshold we used this command:

```
nw_ed genetree 'i & (b<=$threshold)' o
```

Drawing Bootstrap Support on ML Gene Trees: In order to draw bootstrap support on best ML gene trees we first reroot both best ML gene tree, and the bootstrap gene trees using this command:

```
nw_support bootstrapgenetrees taxon > bootstrapgenetrees.rerooted
nw_support      bestMLgenetree taxon > bestMLgenetree.rerooted
```

Then we draw bootstrap supports on the branches:

```
nw_support -p
bestMLgenetree.rerooted bootstrapgenetrees.rerooted >
bestMLgenetree.rerooted.final
```

Gene Tree Estimation: We used FastTree version 2.1.9 Double precision. In order to estimated best ML gene trees we used the following command:

```
fasttree -nt -gtr -nopr -gamma -n <num> <all-genes.phylip>
```

where we have all the alignments in the PHYLIP format in the file all-genes.phylip for each replicate, and $< num >$ is the number of alignments in this file.

For bootstrapping analysis, we first generate bootstrapped sequences using RAxML version 8.2.9 with the following command:

```
raxmlHPC  -s alignment.phylip -f j
          -b <seed number> -n BS -m GTRGAMMA -# 100
```

and then we Fasttree to perform the actual ML analyses; for FastTree bootstrap runs, we use the same command and models that we used for best ML gene trees.

Running ASTRAL: ASTRAL-II in this paper refers to ASTRAL version 4.11.1 and ASTRAL-III refers to ASTRAL version 5.2.5. Both versions can be found in the link below:

```
https://github.com/chaoszhang/ASTRAL/releases/tag/paper
```

Both versions of ASTRAL program were run with following command:

```
java -jar <program> -t 0 -i <input> -o <output> &> <log>
```

References

1. Song, S., Liu, L., Edwards, S.V., Wu, S.: Resolving conflict in eutherian mammal phylogeny using phylogenomics and the multispecies coalescent model. Proc. Nat. Acad. Sci. **109**(37), 14942–14947 (2012)
2. Wickett, N.J., Mirarab, S., Nguyen, N., et al.: Phylotranscriptomic analysis of the origin and early diversification of land plants. Proc. Nat. Acad. Sci. **111**(45), 4859–4868 (2014)
3. Jarvis, E.D., Mirarab, S., Aberer, A.J., et al.: Whole-genome analyses resolve early branches in the tree of life of modern birds. Science **346**(6215), 1320–1331 (2014)
4. Laumer, C.E., Hejnol, A., Giribet, G.: Nuclear genomic signals of the 'microturbellarian' roots of platyhelminth evolutionary innovation. eLife 4 (2015)
5. Tarver, J.E., dos Reis, M., Mirarab, S., et al.: The interrelationships of placental mammals and the limits of phylogenetic inference. Genome Biol. Evol. **8**(2), 330–344 (2016)
6. Rokas, A., Williams, B.L., King, N., Carroll, S.B.: Genome-scale approaches to resolving incongruence in molecular phylogenies. Nature **425**(6960), 798–804 (2003)

7. Maddison, W.P.: Gene trees in species trees. Syst. Biol. **46**(3), 523–536 (1997)
8. Springer, M.S., Gatesy, J.: The gene tree delusion. Mol. Phylogenet. Evol. **94**(Part A), 1–33 (2016)
9. Meiklejohn, K.A., Faircloth, B.C., Glenn, T.C., Kimball, R.T., Braun, E.L.: Analysis of a rapid evolutionary radiation using ultraconserved elements: evidence for a bias in some multispecies coalescent methods. Syst. Biol. **65**(4), 612–627 (2016)
10. Edwards, S.V., Xi, Z., Janke, A., et al.: Implementing and testing the multispecies coalescent model: a valuable paradigm for phylogenomics. Mol. Phylogenet. Evol. **94**, 447–462 (2016)
11. Shen, X.X., Hittinger, C.T., Rokas, A.: Studies can be driven by a handful of genes. Nature **1**, 1–10 (2017)
12. Heled, J., Drummond, A.J.: Bayesian inference of species trees from multilocus data. Mol. Biol. Evol. **27**(3), 570–580 (2010)
13. Chifman, J., Kubatko, L.S.: Quartet inference from SNP data under the coalescent model. Bioinformatics **30**(23), 3317–3324 (2014)
14. Degnan, J.H., Rosenberg, N.A.: Gene tree discordance, phylogenetic inference and the multispecies coalescent. Trends Ecol. Evol. **24**(6), 332–340 (2009)
15. Edwards, S.V.: Is a new and general theory of molecular systematics emerging? Evolution **63**(1), 1–19 (2009)
16. Pamilo, P., Nei, M.: Relationships between gene trees and species trees. Mol. Biol. Evol. **5**(5), 568–583 (1988)
17. Rannala, B., Yang, Z.: Bayes estimation of species divergence times and ancestral population sizes using DNA sequences from multiple loci. Genetics **164**(4), 1645–1656 (2003)
18. Liu, L., Yu, L., Edwards, S.V.: A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. BMC Evol. Biol. **10**(1), 302 (2010)
19. Liu, L., Yu, L.: Estimating species trees from unrooted gene trees. Syst. Biol. **60**, 661–667 (2011)
20. Sayyari, E., Mirarab, S.: Anchoring quartet-based phylogenetic distances and applications to species tree reconstruction. BMC Genomics **17**(S10), 101–113 (2016)
21. Liu, L., Yu, L., Pearl, D.K., Edwards, S.V.: Estimating species phylogenies using coalescence times among sequences. Syst. Biol. **58**(5), 468–477 (2009)
22. Mossel, E., Roch, S.: Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. IEEE/ACM Trans. Comput. Biol. Bioinform. (TCBB) **7**(1), 166–171 (2010)
23. Roch, S., Warnow, T.: On the robustness to gene tree estimation error (or lack thereof) of coalescent-based species tree methods. Syst. Biol. **64**(4), 663–676 (2015)
24. Mirarab, S., Reaz, R., Bayzid, M.S., et al.: ASTRAL: genome-scale coalescent-based species tree estimation. Bioinformatics **30**(17), i541–i548 (2014)
25. Lafond, M., Scornavacca, C.: On the Weighted Quartet Consensus problem. arxiv: 1610.00505 (2016)
26. Mirarab, S., Warnow, T.: ASTRAL-II: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. Bioinformatics **31**(12), i44–i52 (2015)
27. Allman, E.S., Degnan, J.H., Rhodes, J.A.: Determining species tree topologies from clade probabilities under the coalescent. J. Theor. Biol. **289**(1), 96–106 (2011)
28. Shekhar, S., Roch, S., Mirarab, S.: Species tree estimation using ASTRAL: how many genes are enough? In: Proceedings of International Conference on Research in Computational Molecular Biology (RECOMB) (to appear) (2017)

29. Davidson, R., Vachaspati, P., Mirarab, S., Warnow, T.: Phylogenomic species tree estimation in the presence of incomplete lineage sorting and horizontal gene transfer. BMC Genomics **16**(Suppl 10), S1 (2015)
30. Sayyari, E., Mirarab, S.: Fast coalescent-based computation of local branch support from quartet frequencies. Mol. Biol. Evol. **33**(7), 1654–1668 (2016)
31. Price, M.N., Dehal, P.S., Arkin, A.P.: FastTree-2 - approximately maximum-likelihood trees for large alignments. PLoS ONE **5**(3), e9490 (2010)
32. Mirarab, S., Bayzid, M.S., Boussau, B., Warnow, T.: Statistical binning enables an accurate coalescent-based estimation of the avian tree. Science **346**(6215), 1250463–1250463 (2014)
33. Bayzid, M.S., Mirarab, S., Boussau, B., Warnow, T.: Weighted statistical binning: enabling statistically consistent genome-scale phylogenetic analyses. PLoS ONE **10**(6), e0129183 (2015)
34. Mirarab, S., Bayzid, M.S., Warnow, T.: Evaluating summary methods for multilocus species tree estimation in the presence of incomplete lineage sorting. Syst. Biol. **65**(3), 366–380 (2016)
35. Patel, S., Kimball, R., Braun, E.: Error in phylogenetic estimation for bushes in the tree of life. Phylogenet. Evol. Biol. **1**(2), 2 (2013)
36. Gatesy, J., Springer, M.S.: Phylogenetic analysis at deep timescales: unreliable gene trees, bypassed hidden support, and the coalescence/concatalescence conundrum. Mol. Phylogenet. Evol. **80**, 231–266 (2014)
37. Yu, Y., Warnow, T., Nakhleh, L.: Algorithms for MDC-based multi-locus phylogeny inference: beyond rooted binary gene trees on single alleles. J. Comput. Biol. **18**(11), 1543–1559 (2011)
38. Vachaspati, P., Warnow, T.: ASTRID: accurate species trees from internode distances. BMC genomics **16**(Suppl 10), S3 (2015)
39. Kane, D., Tao, T.: A bound on partitioning clusters (2017). arXiv:11702.00912
40. Stamatakis, A.: RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics **30**(9), 1312–1313 (2014)
41. Mallo, D., De Oliveira Martins, L., Posada, D.: SimPhy: Phylogenomic simulation of gene, locus and species trees. Syst. Biol. **65**(2), syv082 (2016)
42. Fletcher, W., Yang, Z.: INDELible: a flexible simulator of biological sequence evolution. Mol. Biol. Evol. **26**(8), 1879–1888 (2009)
43. Tavaré, S.: Some probabilistic and statistical problems in the analysis of DNA sequences. Lect. Math. Life Sci. **17**, 57–86 (1986)
44. Junier, T., Zdobnov, E.M.: The Newick utilities: high-throughput phylogenetic tree processing in the UNIX shell. Bioinformatics **26**(13), 1669–1670 (2010)
45. Robinson, D., Foulds, L.: Comparison of phylogenetic trees. Math. Biosci. **53**(1–2), 131–147 (1981)