Weighting by Gene Tree Uncertainty Improves Accuracy of Quartet-based Species Trees

Chao Zhang[1], Siavash Mirarab[2,*]

1 Bioinformatics and Systems Biology, UC San Diego, CA, USA
2 Department of Electrical and Computer Engineering, UC San Diego, CA, USA

*** smirarab@ucsd.edu**

Abstract

Phylogenomic analyses routinely estimate species trees using methods that account for gene tree discordance. However, the most scalable species tree inference methods, which summarize independently inferred gene trees to obtain a species tree, are sensitive to hard-to-avoid errors introduced in the gene tree estimation step. This dilemma has created much debate on the merits of concatenation versus summary methods and practical obstacles to using summary methods more widely and to the exclusion of concatenation. The most successful attempt at making summary methods resilient to noisy gene trees has been contracting low support branches from the gene trees. Unfortunately, this approach requires arbitrary thresholds and poses new challenges. Here, we introduce threshold-free weighting schemes for the quartet-based species tree inference, the metric used in the popular method ASTRAL. By reducing the impact of quartets with low support or long terminal branches (or both), weighting provides stronger theoretical guarantees and better empirical performance than vanilla ASTRAL. Significantly, weighting by support makes ASTRAL a statistically consistent estimator with estimated (not true) gene trees as input, assuming support values satisfy certain conditions. More consequentially, weighting dramatically improves accuracy in a wide range of simulations and reduces the gap with concatenation in conditions with low gene tree discordance and high noise. On empirical data, weighting improves congruence with concatenation and increases support. Together, our results show that weighting, enabled by a new optimization algorithm we introduce, dramatically improves the utility of summary methods and can reduce the incongruence often observed across analytical pipelines.

keywords: Phylogenomics, ILS, Summary methods, ASTRAL, gene tree estimation error

1 Introduction

Genome-wide data are increasingly available across the tree of life, giving researchers a chance to systematically resolve the evolutionary relationships among species (i.e., species trees) using phylogenomic data. A central promise of phylogenomics is that processes such as incomplete lineage sorting (ILS) that can cause discordance (Degnan and Rosenberg, 2009; Maddison, 1997) among evolutionary histories of different parts of the genome (i.e., gene trees) can be modeled (Edwards, 2009). There has been much progress in developing the theory and methods for species tree inference in the presence of ILS (Mirarab et al., 2021) and other sources of discordance (Elworth et al., 2019; Smith and Hahn, 2021). These phylogenomics approaches have also been widely and increasingly adopted in practice. Yet, substantial challenges remain. Analyses of real data using different methods often reveal incongruent results (Gatesy et al., 2019; Reddy et al., 2017; Shen et al., 2017; Smith et al., 2015; Walker et al., 2018), sparking debate about the cause. Meanwhile, simulation studies have revealed that the best choice of the method is data-dependent (e.g., Bayzid and Warnow, 2013; Mirarab and Warnow, 2015).

A major challenge in phylogenomics is that when we infer gene trees, often from relatively short sequences, the results tend to be highly error-prone (Mirarab et al., 2014a; Patel, 2013; Springer and Gatesy, 2016). Co-estimation of gene trees and species trees (Szöllõsi et al., 2014) is perhaps the most accurate approach to dealing with such noise (Knowles et al., 2012; Leaché and Rannala, 2011). However, despite some progress (Ogilvie et al., 2017), these methods have remained limited in their scalability to even moderately large numbers of species. The approach that is far more scalable and is used often is the "summary" approach: first estimate gene trees from sequence data independently and then summarize them into a species tree by solving optimization problems that provide guarantees of statistical consistency if we allow ourselves to ignore the error in the input tree.

Many summary methods (e.g., Liu and Yu, 2011; Liu et al., 2010, 2009; Mossel and Roch, 2010; Vachaspati and Warnow, 2015) were developed and proved statistically consistent under the multi-species coalescent (MSC) model (Takahata, 1989) of the discordance caused by ILS. Species trees inferred by these tools can be highly accurate even under high levels of ILS. Among the summary tools, ASTRAL (Mirarab et al., 2014b) is among the most widely used and is integrated into other packages (Alanjary et al., 2019; Wang et al., 2020). ASTRAL simply seeks the species tree that maximizes the number of shared quartets (unrooted four-taxon subtrees) between gene trees and the species tree, an optimization problem that guarantees a statistically consistent estimator under the MSC model. The empirical accuracy and scalability of ASTRAL have compared favorably to other methods (e.g., Mirarab, 2019). Moreover, it has now been shown that ASTRAL is also consistent and/or accurate under gene duplication and loss (Legried et al., 2021; Yan et al., 2021), some HGT models (Davidson et al., 2015), and combined models of ILS and duploss (Markin and Eulenstein, 2021), but not gene flow (Solís-Lemus et al., 2016). Zhang et al. 2020 have further adopted the quartet-based approach to multi-copy inputs.

Nevertheless, all summary methods, ASTRAL included, have a shortcoming: inaccuracies in input gene trees can translate to errors in the output species tree (DeGiorgio and Degnan, 2014; Huang and Knowles, 2016; Lanier and Knowles, 2015; Molloy and Warnow, 2018; Patel, 2013). In fact, Roch et al. 2019 proved that summary methods (and concatenation) are positively misleading under pathological examples even in the absence of much true gene tree discordance. These concerns are not just theoretical and can impact biological analyses. For example, on an order-level avian phylogenomic dataset (Jarvis et al., 2014), summary methods, including ASTRAL, produce species trees contradicting the well-established relationships when given input gene trees that have extremely low support (Bayzid et al., 2015), a condition that motivated Mirarab et al. 2014a to bin multiple genes together. As an alternative, Zhang et al. 2018 showed that contracting very low-support branches before running ASTRAL can improve accuracy in simulations and on biological datasets such as the avian dataset. However, this form of reduction in species tree estimation error comes with caveats. Contracted branches may still include signals that will be lost. In particular, when contraction is overly aggressive (e.g., with moderately high thresholds such as 50% or 75%), filtering is often harmful. More pragmatically, the best choice of threshold is dataset dependent, and making a principled choice is challenging if not impossible.

Threshold-free approaches for incorporating gene tree branch support into summary methods have also been proposed. Multi-locus bootstrapping (MLBS) runs the summary method on the bootstrap replicates of gene trees, repeating the process many times to obtain several species trees, which are then combined using a consensus method (Seo, 2008). MLBS can be understood as weighting inferences made from each gene by their uncertainty, and thus, a way to deal with noise. However, previous studies show that MLBS, in fact, reduces the accuracy compared to using Maximum Likelihood (ML) trees (Mirarab et al., 2016). The related method of simply combining all bootstrap replicates into a single run of the summary method has also not been accurate (Mirarab et al., 2014b). A plausible explanation is that bootstrap replicates have much higher rates of discordance and error than ML trees (Sayyari and Mirarab, 2016), and thus, using them directly as input adds noise, even if it reveals uncertainty.

An alternative to using bootstrap trees is to use ML trees as input but explicitly weight gene tree branches (or their quartets) by their statistical support. We can generalize the moderately

successful gene contraction approach, which effectively assigns weights zero or one to quartets, to weight each quartet shared between an estimated gene tree and the proposed species tree according to the statistical support of the quartet resolution. Such an approach will free us from picking arbitrary contraction thresholds and may lead to better accuracy. However, weighting by branch support has not yet been incorporated into existing summary methods such as ASTRAL for several reasons. *i*) It is hard to define weights for quartets in the first place. Methods like ASTRAL are scalable to large numbers of species (n) because they avoid explicitly examining all $\binom{n}{4} = O(n^4)$ quartets even though they optimize a score defined over all quartets. A scalable weighting method for gene tree quartets needs to define weights implicitly based on weights assigned to the $n-3$ gene tree branches. The existing general (e.g., Avni et al., 2015) and MSC-based weighted quartet methods (Richards and Kubatko, 2021; Yourdkhani and Rhodes, 2020) require weights *explicitly* calculated for every quartet, making them less scalable. *ii*) Even with weights implicitly defined, it remains difficult to design efficient algorithms to optimize a weighted score. In fact, unless implicit weights satisfy certain properties, there is no efficient (better than $O(n^4)$) algorithm even for the much simpler problem of computing the total quartet weights of a gene tree. However, with favorable definitions of weights, these difficulties are not insurmountable.

Here, we introduce implicit weighting schemes that avail themselves to efficient optimization with weights conveniently obtained from tree branch lengths and/or branch support values. We introduce the weighted ASTRAL algorithm, an efficient method that is similar to ASTRAL in optimizing a quartet score but is different in several ways: *i*) Its optimization criteria weights each gene tree quartet. *ii*) Its optimization algorithm is completely different from ASTRAL. While the algorithm is more complex and slower in some cases, it scales much better (linearly instead of quadratically) as the number of genes (k) increases. *iii*) Its software package is implemented from scratch and is in C++ instead of Java. Our results show that weighted ASTRAL is superior to ASTRAL in terms of theoretical guarantees that it provides (e.g., under a set of assumptions about gene tree branch support values, weighted ASTRAL is statistically consistent even given estimated trees), accuracy on simulated data, the accuracy of its branch support values, and congruence with concatenation on real datasets.

2 Result

2.1 Weighted ASTRAL Algorithm

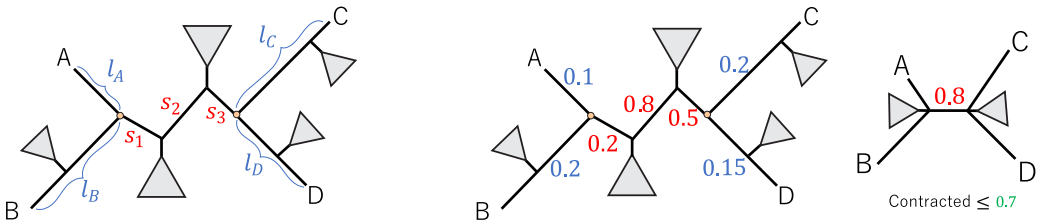
2.1.1 Weight definitions

Unlike ASTRAL-III, where each (resolved) quartet in each gene tree contributes equally to the objective function, weighted ASTRAL assigns each quartet with a weight based on the support or lengths of branches corresponding to it. More specifically, we define three weighting schemes (Fig. 1).

Weighting by support: extends the definition of branch support to a quartet. Let $\mathcal{P}$ be the set of branches on the path between internal nodes of a quartet tree (also called anchors; orange dots in Fig. 1) and let $s(e)$ denote the support of a branch e. We define the support of the quartet as

$$1 - \prod_{e \in \mathcal{P}} \big(1 - s(e)\big)$$

which essentially assumes support values are probabilities of correctness and that branches are independent (both assumptions can be disputed). Given a set of gene trees where each internal branch has a support value, using this definition, we define the weight of each quartet of each gene tree to be its support. The goal is to improve the accuracy by down-weighting quartets with low support. Significantly, while ASTRAL is proved statistically consistent only with true gene trees, Theorem 1 proves that support-weighted ASTRAL is statistically consistent even given estimated



- Original: 1
- Contracting $\leq x$ support branches: $1 - \prod 1_{s_i \leq x}$
- Weighted by Support: $1 - \prod(1 - s_i)$
- Weighted by Length: $e^{-(l_A + l_B + l_C + l_D)}$
- Hybrid Weighting: $(1 - \prod(1 - s_i))e^{-(l_A + l_B + l_C + l_D)}$

- Original: 1
- Contracting ≤ 0.7 support branches: $1 - 1_{0.2 \leq 0.7} 1_{0.8 \leq 0.7} 1_{0.5 \leq 0.7} = \mathbf{1}$
- Weighted by Support: $1 - (1 - 0.2)(1 - 0.8)(1 - 0.5) = \mathbf{0.92}$
- Weighted by Length: $e^{-(0.1 + 0.2 + 0.2 + 0.15)} = \mathbf{0.52}$
- Hybrid Weighting: $= \mathbf{0.92} \times \mathbf{0.52} = \mathbf{0.48}$

Figure 1. Illustration of weighting methods. The generic formula (left) and a specific example (right) of weighting gene tree quartet $AB|CD$ are shown. Trees are annotated with the support of each branch (red) between anchors (orange dots) and the substitution unit length of each leaf-to-anchor path (blue).

gene trees (or any arbitrary set of gene trees) as long as support values satisfy a set of assumptions (see Material and Methods).

Weighting by length down-weights quartets with long terminal branches. Let L be the sum of terminal branch lengths in the gene tree induced to a quartet. We assign e^{-L} as the weight of the quartet and offer two justifications. First, the deeper coalescence events, the longer terminal branch lengths; thus, gene trees that match the species tree are expected, on average, to have shorter branch lengths (see proof of Theorem 2). Thus, down-weighting gene tree quartets with long terminal branches is expected to down-weight genes that do not match the species tree. Doing so can provably provide a bigger gap between the score of the true species tree and alternatives, as shown in Theorem 2 for a strict clock and Theorem 3 for a relaxed clock. Besides the connection to the MSC model, it has also been long appreciated that the so-called long quartets are harder to estimate correctly due to long branch attraction (Erdos et al., 1999; Snir et al., 2008). Many quartet-based methods focus their attention on the so-called short quartets (Nelesen et al., 2012; Warnow et al., 2001). Our weighting scheme naturally achieves the same impact by down-weighting long quartets versus short quartets around difficult species tree branches (see Fig. 1).

Hybrid weighting: We combine both weighting schemes where each quartet is assigned with weight

$$e^{-L}\Big(1 - \prod_{e \in \mathcal{P}} \big(1 - s(e)\big)\Big).$$

This weighting scheme aims to combine strengths of both weighting by support and weighting by length and to improve over both; we will empirically show that such improvements are obtained.

2.1.2 Optimization algorithm

While defining weighting schemes is easy, designing scalable algorithms that optimize a weighted quartet score is much harder. Adopting the existing ASTRAL algorithm to incorporate these per-quartet weights is challenging. A key component of the dynamic programming used in ASTRAL is computing a score for any potential node of the species tree without enumerating all $\binom{n}{4}$ quartets in each gene tree. Instead, it aggregates quartets according to nodes of the gene trees and counts all quartets around a node in one calculation using combinatorics. When each quartet has a different weight, computing the aggregated weights of quartets around a node becomes much more

difficult as simple combinatorial equations become unavailable. A main contribution of this paper is an algorithm to compute weights in an aggregated way (see Optimization algorithm). Despite aggregating quartets by gene tree nodes, this solution would still be slower than optimizing the unweighted scores.

To further enhance scalability, we have also designed a new search algorithm. We build on the idea by Mai and Mirarab 2022 that a new taxon can be added to a tree to maximize the quartet score in subquadratic time in n and adapt it to solve the weighted quartet score placement problem optimally (Theorem 4). Our tree inference algorithm is a combination of multiple rounds of greedy (step-wise) additions to a growing tree and a final round of dynamic programming to combine results of multiple rounds (see Algorithm S2). Luckily, as Theorem 5 and the remark afterward state, this greedy heuristic algorithm retains the statistical consistency properties proved in Theorems 1–3. The running time of this algorithm scales with $O(kHn^2 \log(n))$ where H is the maximum height of gene trees (Theorem 6). Thus, compared to ASTRAL-III with $O((kn)^{2.73})$ (worst case) or $O(k^2n^2)$ (on typical data), the scalability has dramatically improved as the number of genes, k, increases but not as n increases. To improve scaling with n, we use a divide-and-conquer strategy. This algorithm first builds a backbone tree with a smaller number of species (growing with $\sqrt{n}$) by applying the Algorithm S2 and then uses the backbone tree to cluster the rest of the species. It then builds one subtree per cluster and merges those subtrees to get a complete species tree (see Algorithm S3 for details). The statistical consistency is retained by the divide-and-conquer algorithm (Theorem 7), and its running time scales with $O(n^{1.5+\epsilon}Hk)$ (Theorem 8).

2.2 Simulation Results

Comparison of weighting schemes

We start by comparing the accuracy of weighting schemes and different types of branch support on two different datasets. Our default method for computing branch support, used unless otherwise specified, is approximate Bayesian supports from IQ-TREE (aBayes) normalized to range from 0 to 1.

S100. The simphy-simulated 101-taxon S100 dataset is adopted from Zhang et al. 2018 and has gene trees inferred from sequences with varying lengths resulting in various gene tree error levels (see Datasets). On this dataset, bootstrap support computed using RAxML is provided by Zhang et al. 2018. Thus, we also compute weighted ASTRAL trees using Bootstrap supports (wASTRAL-s* and wASTRAL-h*).

In most cases, weighting by support (wASTRAL-s) produces species trees with higher accuracy than weighting by length (Fig. S1), and the improvements are statistically significant; p-value $< 10^{-15}$ according to a repeated-measure ANOVA test with the number of genes (k) and gene length used as extra independent variables (see Statistical tests). The improvement in accuracy varies with k ($p < 10^{-15}$) and perhaps sequence length ($p \approx 0.04$). With all 1000 genes, weighting by length is comparable to weighting by support ($p \approx 0.46$). The accuracy of hybrid weighting (wASTRAL-h) on average is better than the accuracy of wASTRAL-s on all model conditions ($p < 10^{-10}$) and the improvement in accuracy seems to depend on k ($p \approx 0.06$) and sequence length ($p \approx 0.03$). In particular, when $k = 50$, there is no significant evidence that the accuracy of wASTRAL-h is better ($p \approx 0.24$), but as the number of genes increases, wASTRAL-h clearly outperforms the others. With ≥ 500 genes, wASTRAL-h is better than *both* support and length, showing that the combination of the two weighting methods is more powerful than each.

In terms of method of computing support, with support weighting, aBayes support is much better than Bootstrap weighting ($p < 10^{-15}$), but the gap in error becomes significantly ($p < 10^{-9}$ for both) smaller as k and sequence length increase (Fig. S1). With hybrid weighting, aBayes weighting is on average only slightly better than Bootstrap weighting (although the differences are significant with $p \approx 0.0008$, the mean error increases across all conditions by only 0.2%, which is a 3% increase in the error).

S200. This 200-taxon Simphy-simulated dataset has species trees sampled under two birth rates ($10^6, 10^7$), which control deep versus shallow divergences, and tree heights, which control levels of ILS (see Datasets). On this dataset, bootstrapped gene trees are not available; instead, local SH-like support from FastTree-II is available, and we also computed weighted ASTRAL trees using these SH-like supports (wASTRAL-s* and wASTRAL-h*). Patterns of accuracy across wASTRAL versions are similar to S100 (Fig. S2). The accuracy of wASTRAL-h on average is better than the accuracy of wASTRAL-s on all model conditions ($p < 10^{-6}$) and the improvement in accuracy seems to depend on k ($p \approx 10^{-4}$), ILS level ($p < 10^{-7}$), and birth rate ($p < 10^{-10}$). With hybrid weighting, SH-like weighting is on average worse than aBayes weighting (the differences are significant with $p < 10^{-15}$; the error across all conditions is 0.6% more, which is a 9% increase in error). Comparing S100 and S200, while aBayes is better than both bootstrap and SH-like support, its improvements are marginal compared to bootstrap but substantial over SH-like support.

Comparison of topological accuracy to other methods

Impact of gene tree estimation error (S100 dataset). Comparing wASTRAL and ASTRAL-III on the S100 dataset (Fig. 2a and S3), we observe that wASTRAL-h is more robust to gene tree estimation error not only compared to vanilla ASTRAL-III but also ASTRAL-III with low bootstrap support branches ($\leq 5\%$ BS) contracted (ASTRAL-III-5%). The worst accuracy belongs to ASTRAL-III with no branch contraction. As Zhang et al. 2018 reported, contracting low support branches improves the accuracy of ASTRAL-III. However, weighting improves accuracy even more. For example, the average error with 1000 gene trees from 200bp sequences is 9% with ASTRAL-III and reduces to 7% with contracting $\leq 5\%$ BS branches, and wASTRAL-h further reduces the error to 6%.

In all conditions, both wASTRAL versions have better accuracy on average than ASTRAL-III on estimated gene trees ($p < 10^{-15}$), and the difference of accuracy varies according to the sequence length ($p < 10^{-6}$). The mean error of wASTRAL-h (but not wASTRAL-s) is also lower than ASTRAL-III-5% *in every condition*. Over all conditions, both versions of wASTRAL are significantly better than ASTRAL-III-5% ($p \approx 3 \times 10^{-7}$ for wASTRAL-s and $p < 10^{-15}$ for wASTRAL-h). The difference of accuracy between wASTRAL-s and ASTRAL-III-5% seems to vary with k ($p \approx 0.02$) but not sequence length ($p \approx 0.2$). On the other hand, the difference of accuracy between wASTRAL-h and ASTRAL-III-5% varies with sequence length ($p \approx 0.003$) but not k ($p \approx 0.3$). Similar to wASTRAL-h, wASTRAL-h* has mean error lower that that of ASTRAL-III-5% in every condition ($p < 10^{-11}$) and there is no clear evidence that the difference in error varies with k ($p \approx 0.94$) or sequence length ($p \approx 0.16$).

Impact of ILS level and comparison to concatenation (S200 dataset). On the S200 dataset that controls levels of ILS (see Datasets), overall, error rates of wASTRAL-h are lower than that of ASTRAL-III (Fig. 2b and S4) and the improvements are significant ($p < 10^{-15}$). The improvements of wASTRAL-h compared to ASTRAL-III increase with more gene trees ($p < 7 \times 10^{-4}$) but appear to decrease with more ILS ($p \approx 0.08$). Mirarab and Warnow 2015 reported no obvious improvement in accuracy when contracting branches with low SH-like FastTree support values. However, we observed that running ASTRAL-III on gene trees with branches with low aBayes support ($p < 90\%$) contracted (ASTRAL-III-90%) does improve its accuracy. Nevertheless, compared to ASTRAL-III-90%, wASTRAL-h still has lower error ($p \approx 1.6 \times 10^{-6}$), and the improvements of wASTRAL-h are significantly larger for the 10^{-7} birth rates (which correspond to deeper speciations) than the 10^{-6} rate (more recent speciations) ($p \approx 1.5 \times 10^{-5}$).

The comparison between wASTRAL-h and concatenation using ML (CA-ML) significantly depends on several factors (birth rate, $p \approx 4 \times 10^{-8}$; ILS, $p < 10^{-15}$; and, k, $p \approx 2 \times 10^{-12}$). Overall, CA-ML is less robust to ILS levels and is always worse than wASTRAL-h when ILS is high and in most cases when ILS is at the medium level. However, when ILS level is low, wASTRAL-h is better than CA-ML with the 10^6 birth rate ($p \approx 1.7 \times 10^{-5}$), corresponding to recent speciation; by contrast, CA-ML is better than wASTRAL-h with 10^7 birth rate ($p \approx 9 \times 10^{-12}$), corresponding to

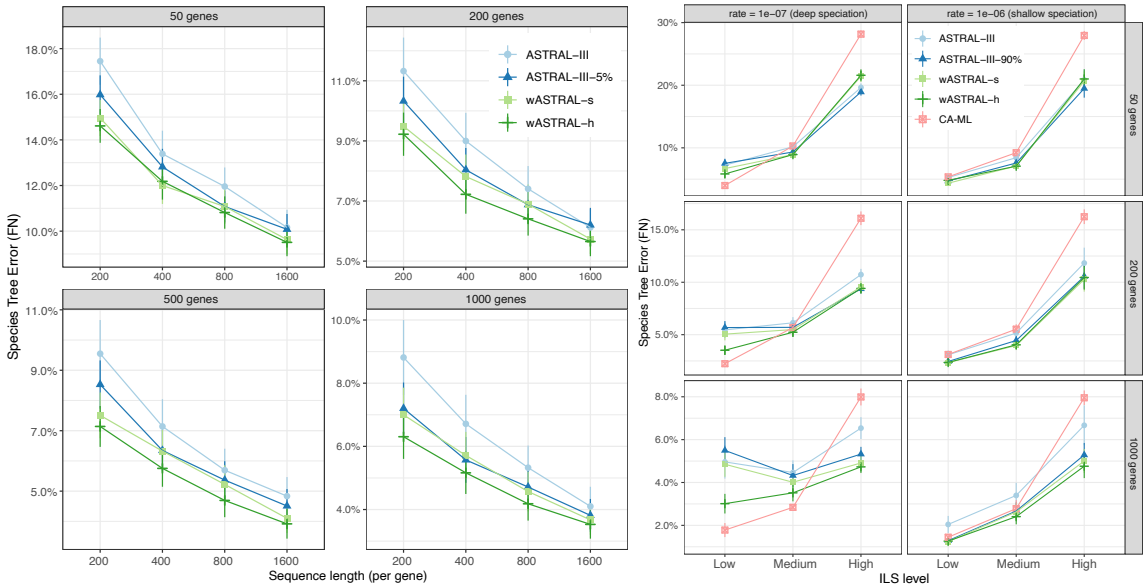


Figure 2. Species tree topological error on simulated datasets. (a) Results on the S100 dataset with $k = \{50, 200, 500, 1000\}$ gene trees (boxes) and gene sequence length $\{200, 400, 800, 1600\}$ (x-axis) for weighted ASTRAL using support weights (wASTRAL-s) and hybrid weights (wASTRAL-h), and ASTRAL-III using fully resolved gene trees and using gene trees where branches with BS $< 5\%$ are contracted (ASTRAL-III-5%). (b) Results on the S200 dataset with $k = \{50, 200, 1000\}$, rates of speciation 1E-6 and 1E-7, and ILS levels. ASTRAL-III is run on FastTree-II gene trees with no further contraction or after contracting branches with aBayes support $< 90\%$ (ASTRAL-III-90%). Concatenation is performed using FastTree-II (see Fig. S3 and S4 for box plots).

deeper speciation events. Thus, in the deep speciation scenarios with low ILS, wASTRAL-h has reduced but not eliminated the gap between ASTRAL and CA-ML. For example, given 1000 gene trees and low ILS with 10^{-6} birth rate, ASTRAL-III has 5% error, which is not helped by branch contraction, whereas wASTRAL-h has 3%, which is much closer to the 2% achieved by CA-ML. To summarize, wASTRAL-h retains and magnifies the advantages of ASTRAL-III over CA-ML for high ILS conditions and eliminates or reduces the advantages of CA-ML under medium and low ILS conditions.

Support accuracy

In addition to obtaining more accurate trees, by accounting for gene tree uncertainty, wASTRAL promises to obtain more realistic measures of support (Branch support). We empirically test if this goal is achieved on simulated datasets using several measures of support accuracy (see Evaluation Criteria).

S100. For both ASTRAL-III and wASTRAL-h, correct branches have much higher support values on average than incorrect branches (Fig. S8). As the support value increases, the proportion of correct branches also increases for both methods (Fig. 3c). Ideal support values are calibrated; i.e., the support matches the probability of correctness of the branch. For both methods, a strong correlation exists between support and the correctness frequency; however, the support is not perfectly calibrated in all conditions. Both methods can under-estimate or over-estimate depending on the sequence length and k.

In general, wASTRAL-h tends to give higher support values compared to ASTRAL-III. Focusing on full-support branches, ASTRAL-III tends to have high error rates, especially with higher k and

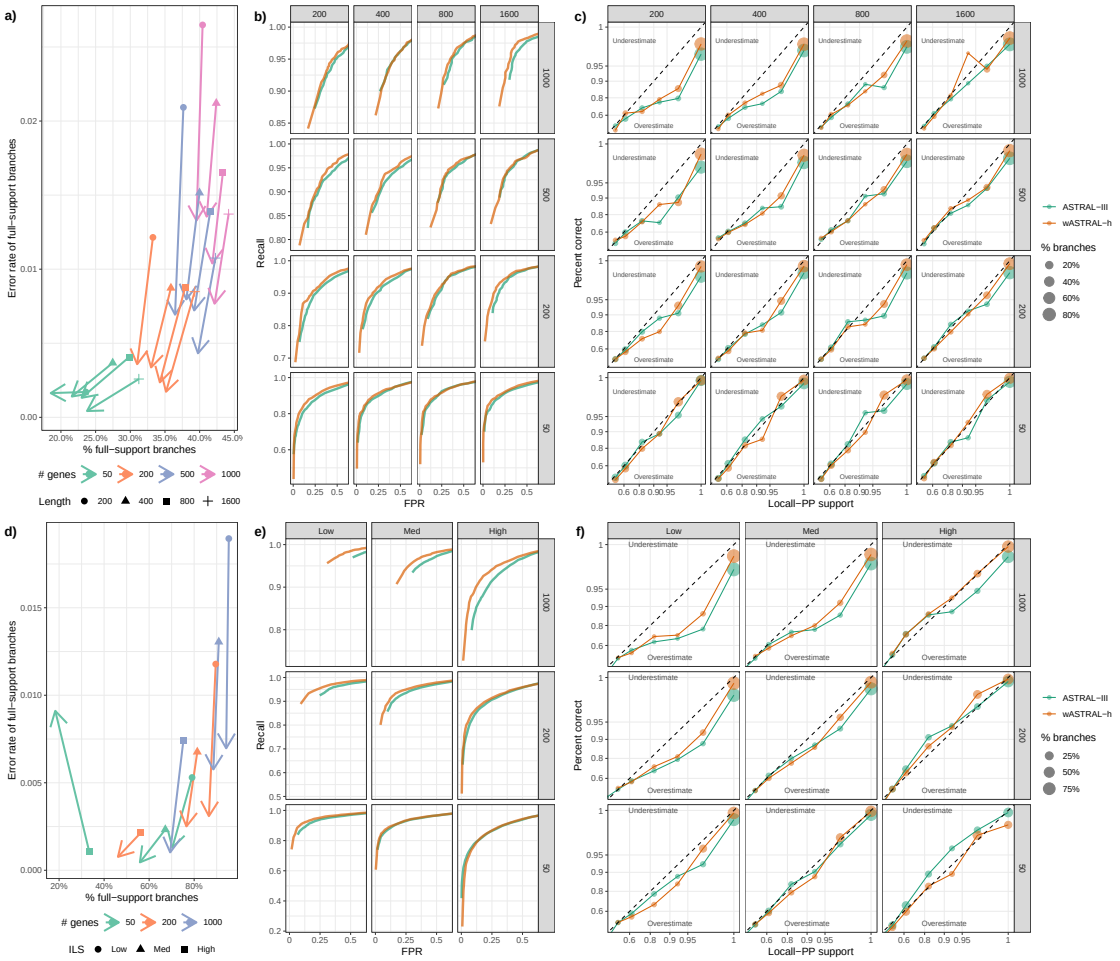


Figure 3. Support accuracy across (a-c) S100 dataset with $k = \{50, 200, 500, 1000\}$ and sequence length $\{200, 400, 800, 1600\}$ and (d-f) S200 dataset with $k = \{50, 200, 1000\}$ and levels of ILS from low to high. (a,d) Change in 100% support branches. Each line shows the portion of full-support branches that are wrong (y-axis) and the percentage of all branches that have full support (x-axis) for wASTRAL-h (the arrowhead) and ASTRAL-III (other shapes). Arrows pointing downwards indicate less frequent error in wASTRAL-h. (b,e) ROC curves where each dot corresponds to a contraction threshold, and TP, FN, FP, TN are measured based on branches above/below threshold that are correct/incorrect (see Evaluation Criteria). (c,f) Support calibration. Branches are binned by their local-PP support, and for each bin, the percentage of branches that are correct are depicted versus the center of the bin. The dotted lines indicate ideal (calibrated) support. Top (bottom) triangle corresponds to the under-estimation (over-estimation) of support. See also Fig. S5-S10.

shorter sequences leading to over-estimation of branch support. However, wASTRAL-h dramatically reduces the error rates on full-support branches, leading to better-calibrated support values (Fig. 3a). Nevertheless, even for wASTRAL-h, a tendency to over-estimate remains for large k values, a tendency that decreases as the sequence length increases, and for $k = 50$ wASTRAL-h sometimes even slightly under-estimates. We next ask which one has better overall performance. By examining the ROC curve as branches with support below varying thresholds are contracted (see Evaluation Criteria), we can detect which method has better recall (i.e., resolution) for a controlled false positive rate (FPR). Results show that for any level of FPR, wASTRAL-h is as good as or better than ASTRAL-III in all conditions (Fig. 3b). However, the improvements in ROC tend to be relatively small. Moreover, in most conditions, the minimum FPR obtained by wASTRAL-h (e.g., at 1.0 support) is lower than the minimum FPR obtained by ASTRAL-III, indicating that fewer of incorrect branches have full support with wASTRAL-h.

S200. Support values on the S200 dataset exhibit similar patterns to S100 (Fig. 3d-f). The most notable difference is that when $k = 1000$, wASTRAL-h has a clear advantage in trading off precision and recall compared to ASTRAL-III according to ROC curves (Fig. 3e and S7). This advantage shrinks as k decreases. One potential reason is that wASTRAL-h has a slight tendency to under-estimate branch confidence for support values < 1 (Fig. S5 and S6), and such tendency is most pronounced with 50 genes, high ILS level, and birth rate 10^{-6} (Fig. 3f and S6).

Comparison of the optimization algorithms

While the main difference between w-ASTRAL and ASTRAL-III is the updated optimization criterion, as we discussed earlier, weighting also forced us to develop a new optimization algorithm. The new optimization algorithm can also be used for unweighted optimization, similar to ASTRAL-III. This leads to a question: which optimization algorithm (ASTRAL-III or the new one) is more effective. The answer depends on whether the datasets include missing data and the amount of gene tree discordance.

In the absence of any missing data, the new algorithm is on average about $10\times$ slower than the ASTRAL-III algorithm on S200 dataset (Figs. 4a and 4c). The slow-down compared to ASTRAL-III is most pronounced with lower ILS levels. In terms of accuracy, the ASTRAL-III and the new algorithms have comparable accuracy in low and medium ILS levels (Fig. 4c). However, in the high ILS case, the error of the ASTRAL-III algorithm is clearly less when only 50 genes are available, slightly less with 200 genes, and perhaps slightly higher with 1000 genes. Cases with reduced accuracy clearly are caused by reductions in the quartet score (so, the ability of the algorithm to optimize the score) for the 50 genes scenario and high ILS (Fig. 4a). Thus, in the absence of missing data, the old algorithm is clearly preferable, especially with few gene trees and very high discordance. Thus, the improved accuracy of weighted ASTRAL over ASTRAL-III in the absence of missing data is *despite* the fact that its optimization algorithm is not as effective as ASTRAL-III in extreme cases.

These patterns substantially change when the dataset includes even very low levels of missing data, as most real biological data do (Figs. 4b and 4d). When we randomly remove 5% of leaves in each gene tree, the comparison between the running time of the new algorithm and the old ASTRAL-III algorithm depends on the model condition (Fig. 4b). While across all conditions, the new algorithm is on average about $3\times$ slower, these slow-downs are mostly for cases with lower ILS levels and lower numbers of genes. By contrast, when ILS level is high and $k = 1000$, the new algorithm runs faster than ASTRAL-III. This is due to the well-documented dependence of ASTRAL-III search space and thus running time on the amount of gene tree discordance, a factor that does not impact the running time of the new algorithm. In terms of accuracy, the old and the new algorithms have comparable accuracy in low and medium ILS levels (Fig. 4d). However, in the high ILS case, the error of the old algorithm is slightly less when only 50 genes are available, comparable with 200 genes, and slightly higher with 1000 genes. Cases with reduced accuracy clearly are caused by reductions in the quartet score for the 50 genes scenario and high ILS; conversely,

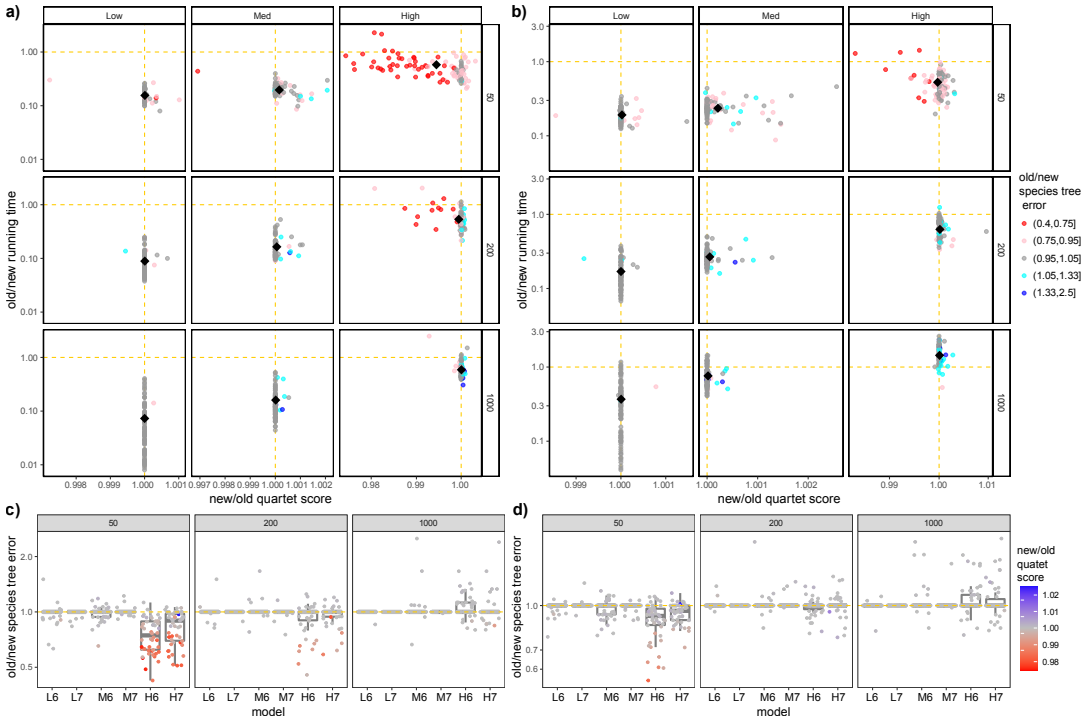


Figure 4. Comparison of the running time, quartet score, and accuracy between the old dynamic programming based and the new insertion-based optimization algorithms (both without weighting) on the S200 dataset. a,b) The ratio between the running time and quartet scores before (a) and after (b) randomly removing 5% of taxa from each gene tree; colors denote the ratio of species tree estimation error between the two methods. Ratios are selected such that the upper-right corner and blue color favor the new algorithm. Results are separated by ILS levels from low to high and by $k = \{50, 200, 1000\}$. c,d) The species tree estimation error using the old algorithm divided by the new algorithm before (c) and after (d) randomly removing of 5% taxa from each gene tree with colors denoting the ratio of quartet scores. L6 to H7 indicate model conditions with low, medium, and high ILS with E1-6 and E1-7 rates.

cases with improved accuracy tend to correspond to better quartet score (Fig. 4b). Thus, the new algorithm is competitive or better than the old algorithm in the presence of even low levels of missing data found to varying degrees in most biological datasets.

2.3 Application on biological data

OneKp

The OneKP dataset of OneKP Initiative 2019 includes 1178 species spanning the plant tree of life. The original study has inferred 410 single-copy AA-based gene trees using RAxML with bootstrap support (which we use), a ASTRAL-III species tree, and CA-ML using RAxML. OneKP Initiative 2019 focused on 20 branches across nine hard-to-resolve areas of the tree (e.g., early Eudicot diversification).

Overall, 47 out of 1175 (4%) branches change between the published ASTRAL-III tree and wASTRAL-h. Among these, four of the changes correspond to one of the 20 focal branches. Beyond topological changes, the support values of the branches recovered by both methods also change, and they tend to increase in wASTRAL-h (Fig. 5a). In particular, all the focal branches that had less than full support in the original ASTRAL-III tree have increased support in the wASTRAL-h tree. Among the 20 focal branches, only four of them have low support (less than 0.95) using wASTRAL-h

in sharp contrast to 12 branches that have low support with ASTRAL-III.

Significantly, all four focal branches that have changed from ASTRAL-III to wASTRAL-h have become consistent with concatenation using ML (CA-ML) where the original ASTRAL-III tree was inconsistent with CA-ML. The order Vitales (grapes) becomes sister to Santalales in wASTRAL-h tree, which is consistent with CA-ML (Fig. 5b). The position of Vitales had low support in the original tree (localPP = 0.44) and has moderate support in the wASTRAL tree (0.87). The sister relationship of Dillenia and Gunnera groups are the only remaining difference between concatenation and wASTRAL-h among the seven basal branches of eudicots, and that grouping has low support in wASTRAL-h. Two of the remaining changes are in the so-called TUC clade, where ASTARL-III breaks down the class Ulvophyceae by uniting Bryopsidales with Chlorophyceae (albeit, with low support); in contrast, wASTRAL-h recovers Ulvophyceae as sister to Chlorophyceae, which is the traditional resolution and is in agreement with the published CA-ML analyses. Finally, the early diversification of ferns was recovered with low support and differently from CA-ML using ASTRAL-III. In contrast, wASTRAL-h resolves early fern diversification identically to CA-ML and with high support. Thus, wASTRAL-h makes coalescent analyses far more congruent with concatenation for the focal branches.

Examining the relative support for the three topologies around each branch (NNIs) reveals that the alternative topologies do not always follow the expectations of MSC (one high frequency and two roughly equal low frequencies). For example, around the branch uniting Polypodiidae and Marattiales, one of the alternative topologies has a support much closer to the main topology than the other topology (Fig. 5b).

Canis

Gopalakrishnan et al. 2018 used a dataset of 48 taxa across genus Canis to resolve the evolutionary history of Canidae at both species and population levels. Since ASTRAL-II could not handle 459,450 gene trees, they partitioned the gene tree into 100 subsets and inferred one ASTRAL-II species tree per subset and published a consensus of those trees. We used wASTRAL-h to analyze all the gene trees, which the original paper estimated using FastTree-II with SH-like local support; we were also able to analyze up to 100,000 gene trees using ASTRAL-MP (Yin et al., 2019) (within 48 hours).

As expected by the shallow time scale (many branches are among populations of the same species), most branches of the ASTRAL-III tree are short (Fig. 5c); e.g., 26 out of 45 branches have length below 0.1 coalescent unit. Despite short branches, due to the large numbers of genes used, both wASTRAL-h and ASTRAL-III produce species trees with at least 99% support on all branches (Fig. S12). The ASTRAL-MP tree (on 100k gene trees) is identical to the published consensus tree, while the wASTRAL-h tree differs from it in only one branch. This sole difference is that wASTRAL-h places Egyptian dogs sister to other Middle Eastern dogs as opposed to Basenji and other Central African dogs.

Analyzing randomly sampled subsets of genes, we observe that the shortest branches need very many genes to achieve the universal full support (Fig. 5c). Using fewer genes (even as many as 100,000), always leaves at least one branch with less than 99% support. With 20,000 gene trees, still, five branches have support below 99%. Moreover, two branches are recovered only if all gene trees are used, and another branch requires 100,000 genes. However, it should be kept in mind that many of the shortest branches are within species, and a tree-like model of evolution is likely insufficient for such branches (Gopalakrishnan et al., 2018). Longer branches, which are mostly across species, do not require large numbers of genes to reach high support. For example, with only 1000 gene trees, 21 branches have at least 99% support.

The increased resolution as a result of using more genes is enabled by the improved theoretical and empirical scalability of wASTRAL-h compared to ASTRAL-III with respect to the number of genes, k (Fig. 5c). The running time of wASTRAL-h scales almost linearly with the number of genes, especially when normalized by the number of rounds of placements (Fig. S11). In contrast, ASTRAL-III scales close to quadratically with k. ASTRAL-III fails to finish on any conditions where $k \geq 2 \times 10^3$ within 24 hours, and it takes ASTRAL-MP with 16 cores more than 36 hours

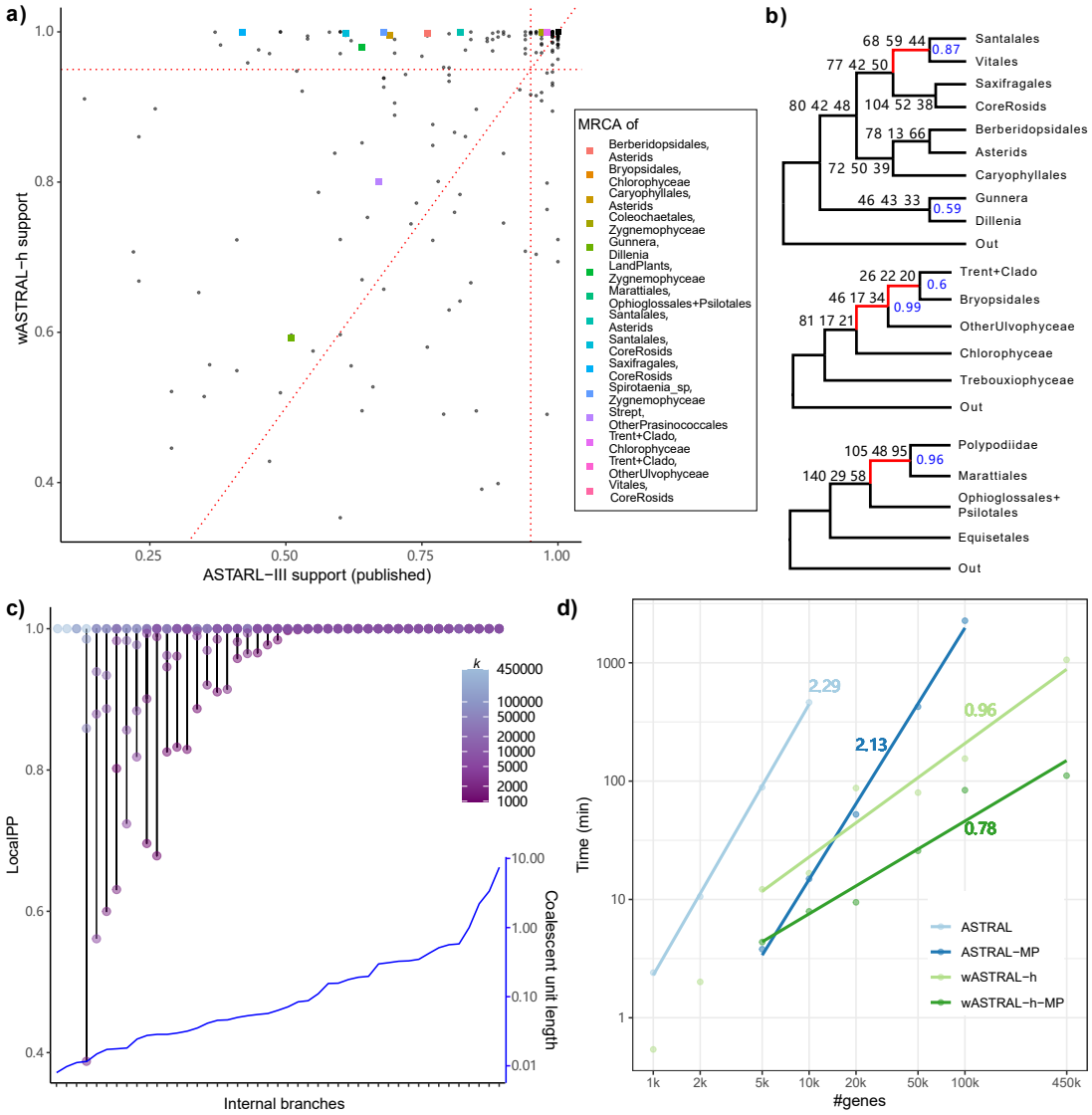


Figure 5. Results on biological datasets on plants (a,b) and dogs (c,d). (a) Weighted ASTRAL support vs ASTRAL-III support for all species tree branches from the OneKp dataset that are shared between the two trees. Focal branches of interest are colored. (b) Weighted ASTRAL resolutions of focal branches with normalized frequencies of all three topologies around each branch shown and local-PP support values colored in blue. Focal branches that differ from ASTRAL-III are colored red. (c) Local-PP values of wASTRAL-h tree internal branches versus the number of genes k; each of the 45 lines corresponds to a branch found in WASTRAL-h output tree with all gene trees as input. The inset with left y-axis scale shows the internal branch lengths in coalescent units on ASTRAL-III tree, sorted from low to high. The three branches to the left are found only with $k = 450000$ or $k = 100000$ consistent with ASTRAL-III topology (top). d) Log-log plot of total running time of ASTRAL-III and wASTRAL-h using both a single core (light colors) and 16 cores (dark colors) vs k on the canis biological dataset for k ranged from 1000 to 450000 with slopes of linear fitting curves labeled. All test cases are performed on a server with AMD EPYC 7742 CPUs.

to finish the $k = 10^5$ condition. By contrast, wASTRAL-h with a single core and 16 cores finishes $k = 4.5 \times 10^5$ case within 18 hours and 2 hours, respectively. Note that on this dataset, even when $k = 10^3$, ASTRAL-III takes 4× time compared to wASTRAL-h, likely because of the very high levels of discordance among gene trees and the absence of some species in some genes, both of which impact running time of ASTRAL-III but not wASTRAL-h.

Avian

Jarvis et al. 2014 used a dataset of 48 species to resolve the order-level avian relationships despite the extremely high levels of gene tree discordance due to a proposed rapid radiation. A major challenge posed by this dataset was the low gene tree resolution , which led to the development of the statistical binning method (Mirarab et al., 2014a). In particular, without binning, the analyses of all 14,446 loci (exons, introns, and UCEs) resulted in trees that were clearly wrong. More recently, species tree inferred from ASTRAL-III without dealing with gene tree error also resulted in incorrect species trees (Zhang et al., 2018); however, contracting low support branches (e.g., $\leq$ 3, 5, and 10%) appeared to solve the problem. Here, we analyze all 14,446 input gene trees with bootstrap-annotated branches using wASTRAL-h.

The wASTRAL-h tree fully agrees with the ASTRAL-III trees after contracting low support branches and is very similar to trees published by Jarvis *et al.* based on concatenation (only five branches differ) and statistical binning (only two branches differ). Moreover, all but one branch in the wASTRAL-h tree has higher or equal support compared to ASTRAL-III with any thresholds of contraction (Fig. S13). Interestingly, the only branch that experiences a reduction in support, the placement of Caprimulgimorphae as sister to Telluraves (core land-birds), is the branch that disagrees with the published concatenation and statistical binning trees. Interestingly, there are four branches with 99-100% support in wASTRAL-h that are also found by all coalescent-based methods (wASTRAL-h, ASTRAL-III and binned MP-EST) but not concatenation.

Cetacean

McGowen et al. 2020 created a UCE dataset of 100 individuals from 77 cetacean species and 12 outgroups to resolve difficult relationships among cetaceans. The original study estimated gene trees using RAxML but without support. We computed Bayesian local supports for gene trees from 3191 protein-coding genes using IQ-Tree and reanalyzed the dataset using wASTRAL-h.

The wASTRAL-h tree is similar to ASTRAL-multi and CA-ML trees reported by the original study (Fig. S14) with only a few differences (three branches to ASTRAL-multi and four to CA-ML). Interestingly, wASTRAL-h agrees with CA-ML and disagrees with ASTRAL-multi tree on the position of the Lissodelphis with high support (though the placement has low support in the ASTRAL-multi). On the other hand, both wASTRAL-h and ASTRAL-III move *Tursiops truncatus* (common bottlenose dolphin) from a clade including *Tursiops aduncus, Stenella attenuata*, and *Stenella frontalis* (as recovered by CA-ML) to a more basal place with high support. Two branches, the placements of *Orcinus orca* and *Neophocaena phocaenoides*, disagree with ASTRAL-multi and CA-ML, but both branches have very low support in wASTRAL-h. These two are among 11 species where McGowen *et al.* used data from previous datasets (genomes, transcriptomes, etc.) instead of sequencing UCEs.

An interesting part of the wASTRAL-h tree is the placement of the two *Stenella clymene* (Clymene dolphin) individuals, which is a known hybrid species evolved from *Stenella longirostris* (spinner dolphin) and *Stenella coeruleoalba*. Interestingly, the two *Stenella clymene* individuals are placed apart, one as sister to *Stenella longirostris* and the other near *Stenella coeruleoalba*. Thus, the hybrid species is placed close to one of the two progenitors, and which one is chosen depends on the individual.

Insect datasets

We also tested three insect datasets: *i*) a 32-taxon collection of 853 RAxML gene trees with bootstrap supports targeting the bee subfamily Nomiinae and particularly genus Pseudapis (Bossert et al., 2021), *ii*) a 203-taxon set of 1930 RAxML gene trees with bootstrap support focused on Lepidoptera (butterflies and moths) (Kawahara et al., 2019), and *iii*) a 61-taxon dataset of the Papilionidae (swallowtail butterflies) with 6407 IQ-TREE gene trees with supports that we computed using aBayes (Allio et al., 2020).

In contrast to previous datasets, on these insect datasets, the differences between wASTRAL-h and ASTRAL-III were minimal and limited to branches with low support. On the Nomiinae dataset, there is no conflict among highly supported branches. wASTRAL-h and ASTRAL-III differ in only one low support branch, and both trees differ from CA-ML (published in the original paper) in two low support branches (Fig. S15). On the Lepidoptera dataset, only seven out of 200 branches differed between wASTRAL-h and ASTRAL-III, and all of these branches had support below 75% (Fig. S16). Across the tree, wASTRAL-h had slightly more branches with support above 95% than ASTRAL-III (173 versus 169). On the Papilionidae datasets, wASTRAL-h tree and ASTRAL-III tree share the same topology, and all branches in both trees have high ($\geq$ 99%) support (Fig. S17).

3 Discussion

3.1 Further observations based on the results

The choice between concatenation and summary methods has been a long-standing debate (Edwards et al., 2016; Giarla and Esselstyn, 2015; Leaché et al., 2015; Meiklejohn et al., 2016; Simmons and Gatesy, 2015). While concatenation is inconsistent under MSC (Roch and Steel, 2015), the most careful simulation studies have found that the best method depends on the dataset: concatenation has been more accurate when gene discordance is low *and* gene signal is limited, and summary methods have been more accurate when discordance is high. Other factors such as deep versus shallow radiations, changes in evolutionary rates across genes, heterotachy, and the number of genes may also matter. Since we cannot reliably predict the superior method in practice, studies often report both types of analyses. We saw that weighting dramatically reduced (but did not fully eliminate) the gap between CA-ML and ASTRAL on the lower ILS dataset (Fig. 2b); wASTRAL-h was nearly as accurate as CA-ML on lower levels of discordance and much more accurate with high levels of discordance (e.g., 1% higher error and 6% lower error, respectively, with 200 genes). As a result, unlike before, we can now rely on the summary method wASTRAL-h *regardless* of the condition. Consistent with simulations, on real datasets, we observed that wASTRAL-h eliminates many of the differences between ASTRAL and CA-ML. Thus, using wASTRAL-h can help reduce the long-standing challenge of getting incongruent results from different analyses.

In our simulations, wASTRAL-h dominates ASTRAL in all model conditions in terms of accuracy, leaving no incentive to prefer ASTRAL in this regard. Contracting low support branches improved ASTRAL trees, but the weighting is more accurate than contracting and does not require hard-to-tune (Bossert et al., 2021) thresholds. Interestingly, the improvements, which were modest in many conditions but substantial in others, appeared more pronounced as the number of genes increased. We speculate the reason is that with more genes, not only the noise in the frequency of observed quartet *topologies* reduces, but also, the quartet weights become less noisy. Thus, having more genes benefits wASTRAL in two ways (less topological noise and better weights), only one of which is enjoyed by ASTRAL.

While topological improvements of wASTRAL-h over ASTRAL were marginal in many cases, the improvements in support were dramatic. The percentage of full support branches that were wrong is reduced in wASTRAL by half or more in most conditions (Fig. 3ad), rendering the full support branches more reliable. This increase in precision did not come at the cost of lowering support. Both real and simulated datasets (e.g., Figs. S5 and S8) saw *increased* supports with wASTRAL. Two aspects of how we compute support have changed (Branch support). One is the handling of

missing data (see (8)); it can be easily shown that, all else being equal, this change will decrease the local-PP. Thus, the increase has to be due to the second change, which is the incorporation of weights. Since localPP support is a function of discordance, the increased support is empirical evidence that down-weighted gene tree quartets tend to be those that are more incongruent with others and the output species tree.

We saw that weighting based on support was superior to branch lengths, and a combination of branch lengths and support was the best. Branch support can be computed in numerous ways with vastly different computational requirements. One practical question is whether one method should be preferred and, if so, which? We tested three ways of computing support on simulated data and noticed that IQ-TREE's aBayes has the best accuracy, closely followed by bootstrapping using RAxML (Fig. S1). In contrast, SH-like support was noticeably less effective. IQ-TREE's aBayes is a local measure of support (i.e., computed for the NNI moves around a branch), and a local notion of support is consistent with how we interpret branch support (i.e., as independent, leading to a product). Moreover, computing local support is much faster than bootstrapping. Thus, while bootstrapping is a good option in terms of accuracy, IQ-TREE's aBayes support can be used to build an accurate *and* efficient pipeline. Nevertheless, note that in the presence of rouge taxa that move widely across a gene tree, local measures of support may provide high support for most branches, whereas global support can result in low support for most branches, effectively down-weighting that gene. In such situations, global support may be more robust.

3.2 Limits and future work

The wASTRAL-s optimization, when solved exactly, gives a statistically constant species tree estimator under the MSC process, given that input *estimated* gene trees and their branch support satisfy certain assumptions. While these assumptions are reasonable, they are also strong, and support estimation methods do not necessarily fulfill them (e.g., see debates in Felsenstein and Kishino, 1993; Hillis and Bull, 1993; Susko, 2009). Thus, the proofs of consistency should be taken more as a theoretical justification of the weighting approach used rather than a prediction of behavior on real data. Support values that over or under-estimate branch supports (compared to our assumptions) may or may not lead to inconsistency of the method, as our assumptions are sufficient but not necessary. Future work can seek more forgiving conditions for support that retain consistency, or conversely, conditions where the method is misleading.

We only proved the statistical consistency of wASTRAL-s and wASTRAL-bl under the MSC model and hope that future works can prove wASTRAL-h is also consistent. Moreover, ASTRAL-multi is now proved consistent under combined models of gene duploss (GDL) and ILS (Markin and Eulenstein, 2021). Weighted ASTRAL can take multi-individual or even multi-copy gene trees as inputs. Future work should explore whether theoretical guarantees of ASTRAL expand to wASTRAL under models that include GDL and should also test the empirical performance of wASTRAL under such models. Future work can also add weighting to the ASTRAL-Pro (Zhang et al., 2020) algorithm that natively supports paralogy.

While wASTRAL-h was more accurate than ASTRAL-III, if we turned off the weights, the new optimization algorithm was slower (in many conditions) and less accurate (in some conditions) than the old algorithm. While the new method tended to be as accurate or more accurate in the presence of missing data (Fig. 4b), our simulation results had no missing data. Thus, the better accuracy of wASTRAL-h was *despite* the new search algorithm being worse and entirely due to the weights. The search algorithm is a heuristic method addressing an NP-hard problem. Just as the speed and accuracy of ASTRAL changed substantially through tweaks to the heuristics from ASTRAL-I to ASTRAL-III, we anticipate that future work can further increase our accuracy, speed, or both. ASTRAL is also finely optimized for CPU, GPU, and vectorization (Yin et al., 2019). Currently, wASTRAL is only trivially parallelized for CPU in two ways, the r rounds of placement and weight computation. Both forms have drawbacks; future work can further optimize the code and implement GPU parallelization.

Perhaps the most promising avenues for future research is exploring other ways of weighting

quartets. For example, since gene trees are estimated from alignments, we can hope that genes with poor alignment lead to low support and low weights. However, alignment errors can also lead to high support. The impact of homology or alignment error (Philippe et al., 2017; Springer and Gatesy, 2018) on wASTRAL should be tested in the future. Furthermore, future work can incorporate homology and alignment quality metrics into the weighting schemes. The weights could also reflect other factors such as evidence of heterotachy impacting gene trees (Braun et al., 2019) and deviations from stationarity Jeffroy et al. (2006). Even more ambitious approaches could be imagined where biases in support estimation could be predicted using machine learning (Suvorov et al., 2020). In designing and testing such weighting schemes, one must keep in mind that not every weighting method will allow fast optimization using dynamic programming.

Finally, several features available in ASTRAL are missing from wASTRAL, but future work can address this limitation. Currently, wASTRAL does not output branch lengths since the natural branch lengths that it could compute would be in a hard to interpret unit (e.g., coalescent plus twice the substitution unit). Future work can examine ways to compute branch lengths in the substitution or coalescent units. Other missing features left to future work are the test of polytomy (Sayyari and Mirarab, 2018), integration with visualization tools such as DiscoVista (Sayyari et al., 2018), and completion of gene trees with respect to each other. Nevertheless, the most valuable features of ASTRAL-III, including handling multi-individual datasets, handling polytomies, and outputting branch support, are all supported.

4 Material and Methods

Let $\mathcal{L}_S := \{1, \ldots, n\}$ be a set of n species. Let us suppose that we are given a set of (estimated or true) binary gene trees $\mathcal{G}$ with $k := |\mathcal{G}|$; for each tree $g = (V_g, E_g) \in \mathcal{G}$, let its leaf set be $\mathcal{L}_g \subseteq \mathcal{L}_S$. For a node v in g, let $\mathcal{L}_v$ be the set of leaves under v. For an estimated gene tree g and for each branch $e \in E_g$, we are given $l_g(e)$ in substitution units (SU) as well as a measure of statistical support $0 \leq s(e) \leq 1$; support can be loosely interpreted as our confidence in the correctness of a bipartition, though we will revisit the assumed meaning of $s(e)$ later. For each two nodes u and v in a gene tree g, we let $\mathcal{P}_g(u, v)$ denote the set of branches on the path between u and v. We also define $l_g(u, v) := \sum_{e \in \mathcal{P}_g(u,v)} l_g(e)$. The notation $g \restriction A$ denotes g restricted to the set A. In each tree g, we refer to a subset Q of four distinct elements of $\mathcal{L}_g$ as a quartet. We define $\mathcal{Q}(g) := \{Q : |Q| = 4, Q \subseteq \mathcal{L}_g\}$ as the set of all quartets in g. We define $\delta_g(ab{\bullet}cd) := 1$ when $\{a, b, c, d\} \in \mathcal{Q}(g)$ and $g \restriction \{a, b, c, d\}$ has bipartition $\{a, b\}{\bullet}\{c, d\}$; otherwise we define $\delta_g(ab{\bullet}cd) := 0$. For any species tree S, let $\mathcal{T}_S$ denote the set of tripartitions corresponding to the internal nodes of S.

Model

We assume each true gene tree G^* is generated according to the multi-species coalescence (MSC) model parameterized by the true species tree S^*. For a quartet species tree, unrooted gene trees are more likely to match the species tree topology than the alternatives (Degnan, 2013). At any point along a branch of the species tree with i gene tree lineages, the distance x in CU units to the next coalescent event, reducing the number of lineages to $i - 1$, is exponentially distributed with the rate $\binom{i}{2}$, resulting in PDF:

$$\frac{i(i-1)}{2} e^{-\frac{i(i-1)}{2} x} ,$$

and the two branches that coalesce are independent of x.

True gene trees have coalescent-unit (CU) lengths, which need to be translated to substitution-unit (SU) before sequence evolution models can be applied. We will consider two models. We start with the simplest choice, akin to a strict clock. Under this naive model, all branches of all gene trees are scaled from CU length to SU length using a fixed ratio λ.

Variable rate model. Let each branch of the species tree be broken into segments of arbitrary length. CU branch lengths of each segment translate to SU units by a multiplier. These multipliers are jointly drawn from a distribution $\mathcal{D}$ defined over all segments (i.e., segment multipliers can be dependent). For each gene tree, its set of multipliers is drawn from $\mathcal{D}$ independently of its topology, its CU branch lengths, and other gene trees. Gene tree branches are broken into segments along the boundaries of the species tree segments. The SU branch lengths of a gene tree are obtained by summing the product of the multiplier and the segment length over all of its segments. Assume the joint distribution of multipliers over all segments is given by $\mathcal{D}$, and let $\mathcal{D}^I$ be the marginal distribution for a segment I. We assume that all $\mathcal{D}^I$ distributions (across all I) share the same expectation λ. Note that the segments are allowed to be arbitrarily small, changing the summation to integration in the limit of infinitesimally small segments.

Because segments have different multipliers (even though they have the same expectation), true gene trees will deviate from ultrametricity. Because multipliers are drawn separately for each gene, deviations from ultrametricity happen in different ways across different genes. Nevertheless, averaged over all genes, the multipliers across the species tree have the same expectations.

4.1 Objective functions

For a given species tree topology S, we define its score against gene tree set $\mathcal{G}$ as:

$$W(S,\mathcal{G}) := \sum_{g\in\mathcal{G}} \sum_{Q\in\mathcal{Q}_S} w_g(S \restriction Q) \tag{1}$$

Our objective is to find the species tree topology S maximizing $W(S,\mathcal{G})$ with a given weighting scheme.

Weighting by support

Let u and v denote the anchors of $g \restriction \{a,b,c,d\}$ (i.e., u is the only node on $\mathcal{P}_g(a,b)\cap\mathcal{P}_g(a,c)\cap\mathcal{P}_g(b,c)$ and v on $\mathcal{P}_g(b,c)\cap\mathcal{P}_g(b,d)\cap\mathcal{P}_g(c,d)$). We define

$$w_g(ab{\bullet}cd) := \Big(1 - \prod_{e\in\mathcal{P}_g(u,v)} \big(1-s(e)\big)\Big)\delta_g(ab{\bullet}cd) \tag{2}$$

Note that if $s(e)$ is considered to be the probability of correctness of a branch, and correctness of branches are considered independent, this weight calculates the probability of correctness of the quartet (Fig. 1). Next, we should how this simple observation can be formalized into a strong proof of consistency under somewhat strong assumptions about the meaning of the support.

Assumptions. We start by assuming true gene trees are generated by *any distribution* (including but not limited to MSC) $P(G^*)$ as long as they satisfy a simple condition (A1) mentioned below. We refer to a random variable corresponding to a true gene tree as G^*. Estimated gene trees can also follow any distribution, as long a second assumption (A2) is correct. The A2 assumption leads to a third assumption on the probabilistic interpretation of support values available for estimated gene trees (A3).

A1. For a true gene tree G^* and a given branch e, let A^*_e be the event that $e \in E_{G^*}$. We assume that for any quartet $\{a,b,c,d\}$ and any given set $\mathcal{A}^* = \{A^*_{e_1},\ldots,A^*_{e_r}\}$ satisfying the condition that $ab{\bullet}cd$, $ac{\bullet}bd$, and $ad{\bullet}bc$ are all compatible with every e_i,

$$\mathbb{E}_{G^*}\left[\delta_{G^*}(ab{\bullet}cd)\middle|\mathcal{A}^*\right] = \frac{1}{3}$$

In words, we assume that according to the distribution generating gene trees, for each quartet, all three topologies have the equal $\frac{1}{3}$ probability even if a set of gene tree edges compatible with all three quartet topologies are known. Note that such an assumption is reasonable for MSC *when* the

true species tree is unknown (and is clearly incorrect if the species tree is known). Thus, we assume that *a priori*, the knowledge of parts of the gene tree does not tell us anything about the other parts.

A2. For an estimated gene tree G corresponding to the true gene tree G^* and a given branch e, let A_e be the event $e \in E_G$. Let $e, e_1, \ldots, e_r$ be any set of pairwise compatible branches and $\mathcal{A} = \{A_{e_1}, \ldots, A_{e_r}\}$. For any two topologies g_1 and g_2, and any edge $e \in E_{g_1} \cap E_{g_2}$ or $e \notin E_{g_1} \cup E_{g_2}$, we assume:

$$\mathrm{P}(A_e|\mathcal{A}, G^* = g_1) = \mathrm{P}(A_e|\mathcal{A}, G^* = g_2)\ .$$

In words, given a branch e is present in the true gene tree, its presence (or absence) in the estimated gene tree is independent of the rest of the true gene tree topology even when conditioned on the presence of any set of other branches in the estimated gene tree. It is harder to say if this assumption is satisfied by methods of gene tree estimation, but it does not seem unreasonable.

Proposition 1. *Let $\{e_1, \ldots, e_r\}$ be any set of pairwise compatible branches, with each element also compatible with a branch e; let $\mathcal{A} = \{A_{e_1}, \ldots, A_{e_r}\}$. Given the assumption A2, for any two topologies g_1 and g_2, regardless of $e \in E_{g_i}$*

$$\frac{P(G^* = g_1|\mathcal{A}, A_e) - P(G^* = g_1|\mathcal{A})}{P(G^* = g_1|\mathcal{A}, A_e^*) - P(G^* = g_1|\mathcal{A})} = \frac{P(G^* = g_2|\mathcal{A}, A_e) - P(G^* = g_2|\mathcal{A})}{P(G^* = g_2|\mathcal{A}, A_e^*) - P(G^* = g_2|\mathcal{A})}$$

For proofs of this and all the following statements, see Appendix Proofs.

A3. We assume that support values estimated on a gene tree G corresponding to the true gene tree G^* are defined as follows (i.e., have specific probabilistic interpretations). Let $\{e_1, \ldots, e_r\}$ be any set of pairwise compatible branches, with each element also compatible with a branch e with support value $s(e)$; let $\mathcal{A} = \{A_{e_1}, \ldots, A_{e_r}\}$; then, we assume

$$s(e) := 1 - \frac{\mathrm{P}(\bar{A}_e^*|\mathcal{A}, A_e)}{\mathrm{P}(\bar{A}_e^*|\mathcal{A})} \quad \Longrightarrow \quad \mathrm{P}(A_e^*|\mathcal{A}, A_e) = s(e) + \big(1 - s(e)\big)\mathrm{P}(A_e^*|\mathcal{A})\ .$$

Particularly, note $s(e) := \frac{\mathrm{P}(A_e^*|A_e) - \mathrm{P}(A_e^*)}{\mathrm{P}(\bar{A}_e^*)}$. We recognize that this is a specific interpretation of support that may not match how support values are actually calculated. Nevertheless, it is not an unreasonable definition. It requires that a branch with support 1 is always correct (i.e., in true gene tree), a branch with support 0 could still be correct at a prior rate (i.e., support 0 means no information), and the probability of correctness linearly increases between these two extremes as the support increases (Fig. S18).

Proposition 2. *Given A2 and A3, for any g^* and any observed branch e with support $s(e)$,*

$$P(G^* = g^*|\mathcal{A}, A_e) = s(e)P(G^* = g^*|\mathcal{A}, A_e^*) + (1 - s(e))P(G^* = g^*|\mathcal{A}) \tag{3}$$

Proposition 3. *Given an estimated gene tree G, for every quartet $ab{\bullet}cd$ satisfying $\delta_G(ab{\bullet}cd) = 1$,*

$$w_G(ab{\bullet}cd) = \mathbb{E}_{G^*}\big[\delta_{G^*}(ab{\bullet}cd) - \delta_{G^*}(ac{\bullet}bd)\big|G\big] = \mathbb{E}_{G^*}\big[\delta_{G^*}(ab{\bullet}cd) - \delta_{G^*}(ad{\bullet}bc)\big|G\big] \tag{4}$$

Theorem 1. *Given estimated gene trees with support values satisfying assumptions A1, A2, and A3, the tree maximizing quartet scores weighted by (2) is a statistically consistent estimator of the species tree.*

Weighting by length

Now we consider gene tree branch lengths. In practice, these branch lengths are estimated in the unit of the number of substitutions from sequence data using methods such as ML. We define

$$w_g(ab{\bullet}cd) := e^{-\left(l_g(a,b) + l_g(c,d)\right)}\delta_g(ab{\bullet}cd) \tag{5}$$

The weighting schema (5) leads to a larger gap compared to ASTRAL between the expected score of the true species tree and the alternative trees given true gene trees. Shekhar et al. 2017

established that the number of gene trees required by ASTRAL to recover the species tree scales with f^{-2} as $f \to 0$ where f is the CU length of the shortest species tree branch. We focus on a regime where $k = \Theta(f^{-2})$ gene trees are available and $f \to 0$. Let the (unweighted) ASTRAL quartet score be denoted by:

$$W'(S, \mathcal{G}) := \sum_{g \in \mathcal{G}} \sum_{Q \in \mathcal{Q}_S} \delta_g(S \restriction Q) \ .$$

Theorem 2. *For a true quartet species tree S^* with topology $ab{\bullet}cd$, true gene trees $\mathcal{G}$ generated under the naive model, with f giving the distance between anchors of S^*, as $f \to 0$, given $k = \Theta(f^{-2})$ gene trees,*

$$P\Big(\sum_{G^* \in \mathcal{G}} w_{G^*}(ab{\bullet}cd) \leq \sum_{G^* \in \mathcal{G}} w_{G^*}(ac{\bullet}bd)\Big) \leq P\Big(\sum_{G^* \in \mathcal{G}} \delta_{G^*}(ab{\bullet}cd) \leq \sum_{G^* \in \mathcal{G}} \delta_{G^*}(ac{\bullet}bd)\Big).$$

The proof, given in Appendix Proofs, relies on a proposition. Let $X_{G^*} := w_{G^*}(ab{\bullet}cd) - w_{G^*}(ac{\bullet}bd)$ and $Y_{G^*} := \delta_{G^*}(ab{\bullet}cd) - \delta_{G^*}(ac{\bullet}bd)$, and let $\bar{X_{\mathcal{G}}} = \frac{1}{k}\sum_{G^* \in \mathcal{G}} X_{G^*}$ and $\bar{Y_{\mathcal{G}}} = \frac{1}{k}\sum_{G^* \in \mathcal{G}} Y_{G^*}$. Then,

Proposition 4. *Under conditions of Theorem 2, we have $Var[X_{G^*}] = \Omega(1)$ and*

$$\frac{\mathbb{E}[X_{G^*}]}{\sqrt{Var[X_{G^*}]}} = \frac{1+4\lambda}{1+2\lambda}\sqrt{\frac{3}{2}} f + O(f^2) \ .$$

The proof can extend to the variable rate model that allows rates to change across the tree and across the genes. With the additional constraint that the variation of rates across genes is limited, we prove:

Theorem 3. *For a true quartet species tree S^* with unrooted topology $ab{\bullet}cd$ and true gene trees $\mathcal{G}$ generated under the variable rate model, let f be the distance between anchors of S^* and L be the total length of all other branches. Assume $\forall I$, $Var(\mathcal{D}^I) \leq \varepsilon^2$ where $\varepsilon = \frac{e^{-\lambda L}}{(16+32\lambda)+(6+32\lambda+32\lambda^2)L}\left(\frac{20(\lambda+\lambda^2)}{9(1+2\lambda)^2}\right)^{\frac{3}{2}}$; i.e., the variance of multipliers of all segments are bounded above by ε^2. When $k = \Theta(f^{-2})$, as $f \to 0$, we have*

$$P\Big(\sum_{G^* \in \mathcal{G}} w_{G^*}(ab{\bullet}cd) \leq \sum_{G^* \in \mathcal{G}} w_{G^*}(ac{\bullet}bd)\Big) \leq P\Big(\sum_{G^* \in \mathcal{G}} \delta_{G^*}(ab{\bullet}cd) \leq \sum_{G^* \in \mathcal{G}} \delta_{G^*}(ac{\bullet}bd)\Big).$$

Proposition 5. *Under the conditions of Theorem 3, we have $Var[X_{G^*}] = \Omega(1)$ and*

$$\frac{\mathbb{E}[X_{G^*}]}{\sqrt{Var[X_{G^*}]}} \geq \sqrt{\frac{3}{2}}\Big(1 - \frac{4\lambda^2}{(1+4\lambda)^2}\Big)^{-\frac{1}{2}} f + O(f^2)$$

Hybrid weighting

The hybrid weighting scheme combines weighting by support and weighting by length.

$$w_g(ab{\bullet}cd) := \Big(1 - \prod_{e \in \mathcal{P}_g(u,v)} \big(1 - s(e)\big)\Big) e^{-\left(l_g(a,b) + l_g(c,d)\right)} \delta_{G^*}(ab{\bullet}cd) \ . \tag{6}$$

We showed empirically that hybrid weighting was more accurate than weighting by support and length.

4.2 Optimization algorithm

Similar to ASTRAL, weighted ASTRAL computes $W(S, \mathcal{G})$ by grouping shared quartets according to potential nodes in the output species tree. For a tripartition $A|B|C \in \mathcal{T}_S$ corresponding to an internal node $v \in V_S$ and a gene tree g, let $W(A|B|C, g)$ be the total score of all shared quartets of S and g that anchor at v on the species tree. Similar to ASTRAL,

$$W(A|B|C, g) = \frac{1}{2} \sum_{\substack{a \in A \cap \mathcal{L}_g \\ b \in B \cap \mathcal{L}_g \\ c \in C \cap \mathcal{L}_g}} \Big(\sum_{d \in A \cap \mathcal{L}_g - \{a\}} w_g(ad \bullet bc) + \sum_{d \in B \cap \mathcal{L}_g - \{b\}} w_g(bd \bullet ac) + \sum_{d \in C \cap \mathcal{L}_g - \{c\}} w_g(cd \bullet ab) \Big)$$

and $W(S, \mathcal{G}) = \frac{1}{2} \sum_{A|B|C \in \mathcal{T}_S} \sum_{g \in \mathcal{G}} W(A|B|C, g)$.

Mirarab and Warnow 2015 and Zhang et al. 2018 use a traversal of gene trees (or a polytree) to compute the unweighted score $W'(A|B|C, g)$ in ASTRAL-II and -III. At each node, the total number of shared quartets is computed using simple combinatorics that only need the intersection size between subsets of taxa. Adopting the same approach to weighted quartets required computing the total sum of weights under a node, necessitating many more counters, floating-point operations, and other changes. When all these challenges are overcome, the running time would be much higher than unweighted ASTRAL.

To improve the running time, we change the optimization method. We designed Algorithm S1 for inserting a new species onto an existing tree such that the placement optimizes the quartet score. Our vanilla Algorithm S2 uses Algorithms S1 to heuristically optimize $W(S, \mathcal{G})$ by building S in a greedy fashion, placing one species at a time. We repeat the greedy process r times and use a final round of dynamic programming to combine all the greedy results (Algorithm S2). To further improve the speed for inputs with a large number of taxa ($n \geq 200$), we also introduce the two-step Algorithm S3 that reduces the theoretical and empirical running by repeated application of the Algorithm S2 on subsets of the taxa.

Placement-based Algorithm

Mai and Mirarab 2022 use the idea pioneered by Brodal et al. 2013 to design a quasi-linear algorithm that finds the optimal placement of a taxon on a backbone tree to minimize its quartet distance to a set of reference trees (e.g., gene trees). This algorithm scores tripartitions at all internal nodes of the species tree through successive leaf recoloring and several counters that are kept and updated. The algorithm of Mai and Mirarab 2022 represents gene trees using balanced trees called HDT adopted from Brodal et al. 2013 to ensure each update takes $O(\log n)$ time. Its recoloring process guarantees that *i*) the leaf partition defined by each internal node of the backbone tree matches the leaf colors at some stage, *ii*) in total $O(n \log(n))$ leaf recoloring is performed, and *iii*) the sets of counters provide all information needed to compute the quartet scores for all possible placements.

We changed the Mai and Mirarab 2022 algorithm in two substantial ways. *i*) We have created a set of counters and their recursive formulas (Table 1 and the proof of Proposition 6) for how to update them in a bottom-up fashion following a leaf recoloring. These counters help compute the weighted quartet score of a tripartition without iterating through all quartets. They essentially count the total weight of all the quartets with the same MRCA at once using counters computed for the child nodes. *ii*) Since the new counters are more complex than the unweighted ones, we use input gene trees as they are instead of using HDTs, which would be hard to implement. The cost of coloring in the simplified version is $O(H)$ where H is the maximum height of gene trees. The resulting Algorithm S1 finds the optimal solution.

Theorem 4. *Let S be a species tree, i be a taxon not in $\mathcal{L}_S$, $\mathcal{S}$ be the set of possible species tree topologies by placing i onto S, and S' be the output gene tree of* `Place`$(i, S, \mathcal{G})$. *Then, $W(S', \mathcal{G}) = \max_{\hat{S} \in \mathcal{S}} W(\hat{S}, \mathcal{G})$.*

While each placement is optimal, the greedy search is not guaranteed to find the optimal tree. We run the greedy search r times each of which produces a full tree S_i. We add the tripartitions

Table 1. Counters w_* are defined for each node w in each gene tree, and Q is defined globally. Here, X, Y, Z are distinct colors of A, B, and C. Let u, v be children of w; e be parental edge of w; p be parent of w; $\mathcal{P}_{x,w}$ be path between x and w. $s(\mathcal{P}) = 1 - \prod_{\hat{e}\in\mathcal{P}}(1 - s(\hat{e}))$; $m(i,j)$ = MRCA of i and j. Counters w_* are set to zero for leaves unless explicitly noted. For each counter, we show a recursive equation on top and the equivalent non-recursive definition on the bottom.

w_X	$(u_X + v_X)e^{-l(e)}$ for internal node w; $e^{-l(e)}$ for leaf node w colored X
	$\sum_i e^{-l(\mathcal{P}_{i,p})}$ for all leaf nodes i colored X under w
(w^+_{XX}, w^+_{XY})	$\big(u^+_{XX} + v^+_{XX} + u_X v_X, u^+_{XY} + v^+_{XY} + u_X v_Y + u_Y v_X\big)$
	$\sum_{i,j} e^{-l(\mathcal{P}_{i,j})}$ for all leaf nodes i colored X and j colored X/Y under w
(w^-_{XX}, w^-_{XY})	$\big((u^-_{XX} + v^-_{XX} + u_X v_X)\big(1 - s(e)\big), (u^-_{XY} + v^-_{XY} + u_X v_Y + u_Y v_X)\big(1 - s(e)\big)\big)$
	$\sum_{i,j} e^{-l(\mathcal{P}_{i,j})}\big(1 - s(\mathcal{P}_{m(i,j),p})\big)$ for all leaf nodes i colored X and j colored X/Y under w
$(w_{XX\|Z}, w_{XY\|Z})$	$\Big(\big(u_{XX\|Y} + v_{XX\|Y} + (u^+_{XX} - u^-_{XX})v_Y + u_Y(v^+_{XX} - v^-_{XX})\big)e^{-l(e)},$ $\big(u_{XY\|Z} + v_{XY\|Z} + (u^+_{XY} - u^-_{XY})v_Z + u_Z(v^+_{XY} - v^-_{XY})\big)e^{-l(e)}\Big)$
	$\sum_{i,j,k} e^{-l(\mathcal{P}_{i,j}) - l(\mathcal{P}_{k,p})} s(\mathcal{P}_{m(i,j),m(i,k)})$ for leaf nodes i colored X, j colored X/Y, k colored Z under w, and $m(i,j)$ under $m(i,k)$
$w_{XX\|YZ}$	$v_X u_{YZ\|X} + u_X v_{YZ\|X} + u_{XX\|Z} v_Y + v_{XX\|Z} u_Y + u_{XX\|Y} v_Z + v_{XX\|Y} u_Z$
	$\sum_{h,i,j,k} w_g(hi\bullet jk)$ for all leaf nodes h, i colored X, j colored Y, k colored Z, and w = MRCA h, i, j, k
Q	$\sum_{g\in\mathcal{G}} \sum_w (w_{AA\|BC} + w_{BB\|AC} + w_{CC\|AB})$ for internal nodes w in g
	$\sum_{g\in\mathcal{G}} \sum_{h,i,j,k} w_g(hi\bullet jk)$ for leaf nodes h, i, j, k in g where h, i have the same color and i, j, k have different colors; when species coloring matches all gene trees, $Q = W[A\|B\|C] = \sum_{g\in\mathcal{G}} W(A\|B\|C, g)$ (Proposition 6).

of each S_i and their weights to a lookup table W^*. The dynamic programming step computes an optimal species tree restricted to the tripartitions of W^* (Algorithm S2) and proceeds very similarly to ASTRAL, with one difference: The search space is all tripartitions (not bipartitions) found in all of the S_i trees. Note that we do not need to compute tripartition weights as those are precomputed and stored in W^* in the placement step. Algorithm S2 is not guaranteed to find the optimal solution either. However, a positive theoretical result ensures this lack of optimality does not impede the statistical consistency of the solution.

Theorem 5. *If there exists a species tree topology S^* satisfying that for each quartet subtree $ab|cd$,*

$$\sum_{g\in\mathcal{G}} w(ab|cd) > \max\Big(\sum_{g\in\mathcal{G}} w(ac|bd), \sum_{g\in\mathcal{G}} w(ad|bc)\Big)\,, \tag{7}$$

then the output of Algorithm S2 will be S^*.

Remark. *For a binary true species tree S^*, as $k \to \infty$, S^* satisfies the condition of Theorem 5 with high probability for wASTRAL-s on estimated gene trees (Theorem 1) and wASTRAL-bl on true gene trees (Theorem 2) under proper assumptions. Thus, under those assumptions, wASTRAL-s and wASTRAL-bl using Algorithm S2 remain statistically consistent estimators of the species tree. We conjecture that wASTRAL-h is also statistically consistent given estimated gene trees under certain conditions.*

Theorem 6. *The time complexity of Algorithm S2 is $O(n^2 kH \log n)$.*

Since $H = O(\log n)$ for a balanced tree and $H = O(n)$ for a caterpillar tree, the time complexity of Algorithm S2 is $O(n^2 k \log^2 n)$ when gene trees are roughly balanced and $O(n^3 k \log n)$ when they are caterpillar-like. Thus, because we do not adopt HDTs, the time complexity of our algorithm is slightly worse than Mai and Mirarab 2022 only when trees are unbalanced. As our counters are linearly related to counters of each child of a node, the HDT structure can be adopted in our algorithm, enabling a $O(n^2 k \log^2 n)$ algorithm. Since such an implementation would not necessarily reduce running time in practice, we do not pursue it further.

Two-step algorithm

The two-step algorithm replaces each round of placement in the Algorithm S2 with a two-step procedure: first, a backbone tree is computed using the Algorithm S2 on a smaller set of species; then, all other species are added onto the backbone tree with local refinements. To compute a backbone tree S_i, we randomly select $m = O(\sqrt{n})$ leaves and apply the Algorithm S2 with $r = O(\sqrt{n})$ rounds of placements to get a backbone tree with m species. Then, to complete the backbone tree S_i, for each of the other $n - m$ species, we first find the branch of S_i with optimal placement score; we cluster the species by optimal placement branches. For each cluster C_e corresponding to branch e, we sequentially place species in C_e onto S_i and remove any species that are not placed on e (or its derived branches) to obtain S_e. We merge all trees in $\{S_e : e \in E_{S_i}\}$ into a uniquely defined compatible tree S_i'. If S_i' is not complete (i.e., some taxa were removed because they fall outside their base placement edge), at the end, we place these orphan species onto S_i' using the Algorithm S2 to get a complete tree (Algorithm S3).

Theorem 7. *Under the conditions of Theorem 5, Algorithm S3 will output S^*.*

Remark. *Under proper assumptions, wASTRAL-s and wASTRAL-bl, using Algorithm S3 are statistically consistent estimators of the species trees. We conjecture that wASTRAL-h using this algorithm is also a statistically consistent estimator under certain assumptions.*

Theorem 8. *When the inequality condition in Theorem 5 is satisfied, then the time complexity of the two-step algorithm is $O(n^{1.5+\epsilon}kH)$ with arbitrarily high probability.*

4.3 Branch support

We adopt the quartet-based metric introduced by Sayyari and Mirarab 2016 used for measuring branch support. This metric essentially quantifies the probability of the true quartet score around a species tree branch being more than $\frac{1}{3}$ given the observed quartet topologies assuming that gene trees are fully independent, but the quartets around the branch are fully dependent. The original metric gives all gene trees that have at least one quartet around a branch of interest an equal weight of one. In wASTRAL-h, we instead weight each gene tree by the total support of all three topologies and normalize the counts in a fashion inspired by Lyapunov's Central Limit Theorem. For each species tree quadripartition $A|B|C|D$ representing an internal branch, let $\mathcal{G}$ denote the subset of gene trees that have at least one element from each of A, B, C, and D. We define the x_1, the normalized quartet count for branch $(A \cup B)|(C \cup D)$, as

$$x_1 = \frac{\sum_{g \in \mathcal{G}} \sum_{(a,b,c,d) \in A \times B \times C \times D} w_g(ab|cd)}{\sqrt{\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \left(\sum_{(a,b,c,d) \in A \times B \times C \times D} w_g(ab|cd) + w_g(ac|bd) + w_g(ad|bc) \right)^2}} .$$

The quartet counts for $(A \cup C)|(B \cup D)$ and $(A \cup D)|(B \cup C)$ are similarly defined and are denoted by x_2 and x_3. Using the localPP method of Sayyari and Mirarab 2016, we set the localPP support to: $\frac{h(x_1)}{h(x_1)+h(x_2)+h(x_3)}$, where $h(x) = 2^x \mathbf{B}(x+1, x_1+x_2+x_3-x+2\lambda)\big(1 - \mathbf{I}_{\frac{1}{3}}(x+1, x_1+x_2+x_3-x+2\lambda)\big)$, $\mathbf{B}$ is the beta function, $\mathbf{I_x}$ is the regularized incomplete beta function, and λ is the Yule prior (default: $\frac{1}{2}$).

When all weights are set to 1, as in ASTRAL-III, the new definition is identical to the original one in the absence of missing data but can be different with missing data. Let $N_g = |A \cap \mathcal{L}_g| \times |B \cap \mathcal{L}_g| \times |C \cap \mathcal{L}_g| \times |D \cap \mathcal{L}_g|$ be the number of quartets around the branch of interest present in a gene tree g; let n_g be the number of those quartets that are compatible with $(A \cup B)|(C \cup D)$. The original method sets:

$$\text{Old definition: } x_1 = \sum_{g \in \mathcal{G}} \frac{n_g}{N_g} \qquad \text{New definition: } x_1 = \frac{\sum_{g \in \mathcal{G}} n_g}{\sqrt{\frac{1}{|g|} \sum_{g \in \mathcal{G}} N_g^2}} \tag{8}$$

The two definitions are identical only when all N_g values are the same, which is the case when there is no missing data but can also happen in other scenarios. When patterns of missing data are different, the old calculations made sure all genes had equal weights (each gene has $x_1 + x_2 + x_3 = 1$). In the new definition, since each gene is weighted differently in wASTRAL, to begin with, we also allow genes to have a different total vote depending on their patterns of missing data. In the new formula, each gene votes (contributes to $x_1 + x_2 + x_3$) proportionally to the number of quartets they have around a branch.

Datasets

S100 Simulated dataset by Zhang et al. 2018, includes 100 ingroups and one outgroup, and is simulated using SimPhy (Mallo et al., 2016) with 50 replicates. The species trees are simulated under the birth-only process with birth rate 10^{-7}, a fixed haploid population size of 4×10^5, and the number of generations sampled from a log-normal distribution with mean 2.5×10^6. For each replicate, 1000 true gene trees are simulated under the MSC model. The ILS level substantially varies across replicates, with a mean of 0.46 when measured by the average normalized RF distance between the true species trees and true gene trees. The gene alignments are simulated on gene trees with lengths $\{200, 400, 800, 1600\}$ bps using Indelible (Fletcher and Yang, 2009) under the GTR model with gene tree branch specific mutation rates drawn from a log-normal distribution. Gene trees are reconstructed under the GTR+Γ model using FastTree2 (Price et al., 2010). The gene tree estimation error, measured by the FN rate between the true gene trees and the estimated gene trees is $\{0.55, 0.42, 0.31, 0.23\}$ for lengths $\{200, 400, 800, 1600\}$. The original publication has made bootstrap support obtained from 100 replicates run using FastTree2 available for each gene tree. Estimated gene trees are subsampled to subsets of size $k = \{50, 200, 500, 1000\}$.

S200 Simulated dataset by Mirarab and Warnow 2015 includes 200 ingroup species and one outgroup. Its species trees are generated under two different birth rates $10^{-6}, 10^{-7}$ each with 50 replicates and three different ILS levels, low ($\approx 10\%$), medium ($\approx 35\%$), and high ($\approx 70\%$), controlled by max tree heights $10^7, 2 \times 10^6, 5 \times 10^5$ generations, respectively. The sequence length of each gene is uniformly drawn between 300 and 1500 bps, resulting in a wide range of gene tree estimation error across replicates (mean: 25%, 31%, and 47%, for low, medium, and high ILS). Here, bootstrap replicates are not available, but we compute aBayes support. Estimated gene trees are downsampled with $k = \{50, 200, 1000\}$.

By default, we compute branch length and support using IQ-TREE (v 1.6.12) aBayes option (`--abayes`) under GTR+Γ model to obtain support values. As each support value s is between $\frac{1}{3}$ and 1, we use normalized support value by $\frac{3s-1}{2}$ so that the minimum is 0. IQ-TREE also computes branch lengths for each input gene tree using ML. Unresolved branches are equivalent for wASTRAL-s and -h to any arbitrary resolution with length and support 0. The number of rounds r is dynamically determined by three simple heuristics: *i*) start with 12 rounds and perform DP to get an optimal score; *ii*) run another 4 rounds and perform DP using bipartitions from all previous rounds; *iii*) repeat step (*ii*) until no improvement to the optimal score is obtained or step (*ii*) has been repeated five times.

ASTRAL-III version 5.7.4 is run on available gene trees. We also test contracted gene trees as inputs. We use ASTRAL-III-5% (S100 dataset) to denote running ASTRAL-III on gene trees with low bootstrap support branches ($< 5\%$) contracted. The 5% threshold is used because it had the best accuracy overall in the original study. On the S200 dataset, we do not have bootstrap support values and instead have to rely on IQ-TREE aBayes support. Because aBayes support is much higher than bootstrap support, we decide to contract branches with support below a 0.90 threshold with aBayes. Thus, we compared with ASTRAL-III-90% to denote running ASTRAL-III on gene trees with aBayes support (< 0.90) contracted.

CA-ML is concatenation using unpartitioned ML method. On the S200 dataset, CA-ML was available from the original study (where they used FastTree2 as the ML method) and is used here. Thus, FastTree2 is used both for gene tree estimation and CA-ML on the S200 dataset, ensuring the comparisons are fair.

Evaluation Criteria

To compare topological accuracy, we use the false-negative rate (FN) in recovering bipartitions of the true species tree. Since the true species tree and the reconstructed species tree are both binary, false-negative rate, false-positive rate, and normalized Robinson–Foulds distance (RF) are all the same. For measuring the accuracy of support, we use three methods with different goals.

Calibration plots ask if support values perfectly indicate correctness (i.e., are *calibrated*). We break support values into these bins: $[\frac{1}{3}, 0.5)$, $[0.5, 0.75)$, $[0.75, 0.9)$, $[0.9, 0.95)$, $[0.95, 1)$, and $\{1\}$ (note that all tools 1 means anything rounded to 1 by the tool). For each bin, we compute the average accuracy of branches with support in that range and plot it versus the midpoint of boundaries of that bin. On such plots, points above (below) diagonal indicate under-estimation (over-estimation) of branch support. Even when support is not calibrated, it can be useful if higher support *correlates* with correctness; e.g., if all support values are uniformly increased or decreased (say, divided by two), it can still be perfectly correlated with support. To measure this aspect, we use ROC curves. For a large number of thresholds between 0 and 1, we contract all branches with support below that threshold. We call contracted correct branches FN, contracted incorrect branches TN, kept correct branches TP, kept incorrect branches FP; these allow us to define true positive rate and recall, and thus ROC. Note that the ROC curve remains the same with any monotonic transformation of support values assuming an infinite number of the esholds. We also plot empirical cumulative density function (ECDF) of correct and incorrect branches. We expect higher support for correct branches than incorrect branches; thus, the accuracy can be judged by the gap between ECDF curves of correct and incorrect branches.

Statistical tests. We perform repeated measures ANOVA tests between two species tree reconstruction methods to test significance of topological accuracy differences and whether the gap in accuracy depends on simulation model parameters. We limit the data to only the two methods being compared and for each experimental condition, we use replicates as the repeated measures (i.e., the error term). We perform double-sided ANOVA tests on reconstruction methods vs. experimental conditions and report p-values for the difference between methods and the impact of other variables on that difference.

Availability: The wASTRAL software is available publicly at https://github.com/chaoszhang/ASTER. Data used here are available at https://github.com/chaoszhang/Weighted-ASTRAL_data.

References

Alanjary, M., Steinke, K., and Ziemert, N. (2019). AutoMLST: an automated web server for generating multi-locus species trees highlighting natural product potential. *Nucleic Acids Research*, 47(W1):W276–W282.

Allio, R., Scornavacca, C., Nabholz, B., et al. (2020). Whole Genome Shotgun Phylogenomics Resolves the Pattern and Timing of Swallowtail Butterfly Evolution. *Systematic Biology*, 69(1):38–60.

Avni, E., Cohen, R., and Snir, S. (2015). Weighted Quartets Phylogenetics. *Systematic Biology*, 64(2):233–242.

Bayzid, M. S., Mirarab, S., Boussau, B., and Warnow, T. (2015). Weighted Statistical Binning: Enabling Statistically Consistent Genome-Scale Phylogenetic Analyses. *PLoS ONE*, 10(6):e0129183.

Bayzid, M. S. and Warnow, T. (2013). Naive binning improves phylogenomic analyses. *Bioinformatics*, 29(18):2277–84.

Bossert, S., Murray, E. A., Pauly, A., et al. (2021). Gene Tree Estimation Error with Ultraconserved Elements: An Empirical Study on Pseudapis Bees. *Systematic Biology*, 70(4):803–821.

Braun, E. L., Cracraft, J., and Houde, P. (2019). Resolving the Avian Tree of Life from Top to Bottom: The Promise and Potential Boundaries of the Phylogenomic Era. In *Avian Genomics in Ecology and Evolution*, pages 151–210. Springer International Publishing, Cham.

Brodal, G. S., Fagerberg, R., Mailund, T., et al. (2013). Efficient Algorithms for Computing the Triplet and Quartet Distance Between Trees of Arbitrary Degree. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1814–1832, Philadelphia, PA. Society for Industrial and Applied Mathematics.

Davidson, R., Vachaspati, P., Mirarab, S., and Warnow, T. (2015). Phylogenomic species tree estimation in the presence of incomplete lineage sorting and horizontal gene transfer. *BMC Genomics*, 16(Suppl 10):S1.

DeGiorgio, M. and Degnan, J. H. (2014). Robustness to Divergence Time Underestimation When Inferring Species Trees from Estimated Gene Trees. *Systematic Biology*, 63(1):66–82.

Degnan, J. H. (2013). Anomalous Unrooted Gene Trees. *Systematic Biology*, 62(4):574–590.

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

Edwards, S. V. (2009). Is a new and general theory of molecular systematics emerging? *Evolution*, 63(1):1–19.

Edwards, S. V., Xi, Z., Janke, A., et al. (2016). Implementing and testing the multispecies coalescent model: A valuable paradigm for phylogenomics. *Molecular Phylogenetics and Evolution*, 94:447–462.

Elworth, R. A. L., Ogilvie, H. A., Zhu, J., and Nakhleh, L. (2019). Advances in Computational Methods for Phylogenetic Networks in the Presence of Hybridization. pages 317–360.

Erdos, P., Steel, M., Szekely, L., and Warnow, T. (1999). A few logs suffice to build (almost) all trees: Part II. *Theoretical Computer Science*, 221(1-2):77–118.

Felsenstein, J. and Kishino, H. (1993). Is there something wrong with the bootstrap on phylogenies? A reply to Hillis and Bull. *Systematic Biology*, 42(2):193–200.

Fletcher, W. and Yang, Z. (2009). INDELible: A Flexible Simulator of Biological Sequence Evolution. *Molecular Biology and Evolution*, 26(8):1879–1888.

Gatesy, J., Sloan, D. B., Warren, J. M., et al. (2019). Partitioned coalescence support reveals biases in species-tree methods and detects gene trees that determine phylogenomic conflicts. *Molecular Phylogenetics and Evolution*, 139:106539.

Giarla, T. C. and Esselstyn, J. A. (2015). The Challenges of Resolving a Rapid, Recent Radiation: Empirical and Simulated Phylogenomics of Philippine Shrews. *Systematic Biology*, 64(5):727–740.

Gopalakrishnan, S., Sinding, M.-H. S., Ramos-Madrigal, J., et al. (2018). Interspecific Gene Flow Shaped the Evolution of the Genus Canis. *Current Biology*, 28(21):3441–3449.

Hillis, D. M. and Bull, J. J. (1993). An Empirical Test of Bootstrapping as a Method for Assessing Confidence in Phylogenetic Analysis. *Systematic Biology*, 42(2):182–192.

Huang, H. and Knowles, L. L. (2016). Unforeseen Consequences of Excluding Missing Data from Next-Generation Sequences: Simulation Study of RAD Sequences. *Systematic Biology*, 65(3):357–365.

Jarvis, E. D., Mirarab, S., Aberer, A. J., et al. (2014). Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331.

Jeffroy, O., Brinkmann, H., Delsuc, F., and Philippe, H. (2006). Phylogenomics: the beginning of incongruence? *Trends in Genetics*, 22(4):225–231.

Kawahara, A. Y., Plotkin, D., Espeland, M., et al. (2019). Phylogenomics reveals the evolutionary timing and pattern of butterflies and moths. *Proceedings of the National Academy of Sciences*, 116(45):22657–22663.

Knowles, L. L., Lanier, H. C., Klimov, P. B., and He, Q. (2012). Full modeling versus summarizing gene-tree uncertainty: Method choice and species-tree accuracy. *Molecular Phylogenetics and Evolution*, 65(2):501–509.

Lanier, H. C. and Knowles, L. L. (2015). Applying species-tree analyses to deep phylogenetic histories: Challenges and potential suggested from a survey of empirical phylogenetic studies. *Molecular Phylogenetics and Evolution*, 83:191–199.

Leaché, A. D., Banbury, B. L., Felsenstein, J., et al. (2015). Short tree, long tree, right tree, wrong tree: New acquisition bias corrections for inferring SNP phylogenies. *Systematic Biology*, 64(6):1032–1047.

Leaché, A. D. and Rannala, B. (2011). The accuracy of species tree estimation under simulation: A comparison of methods. *Systematic Biology*, 60(2):126–137.

Legried, B., Molloy, E. K., Warnow, T., and Roch, S. (2021). Polynomial-Time Statistical Estimation of Species Trees Under Gene Duplication and Loss. *Journal of Computational Biology*, 28(5):452–468.

Liu, L. and Yu, L. (2011). Estimating Species Trees from Unrooted Gene Trees. *Systematic Biology*, 60(5):661–667.

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

Liu, L., Yu, L., Pearl, D. K., and Edwards, S. V. (2009). Estimating species phylogenies using coalescence times among sequences. *Systematic Biology*, 58(5):468–477.

Maddison, W. P. (1997). Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536.

Mai, U. and Mirarab, S. (2022). Completing gene trees without species trees in sub-quadratic time. *Bioinformatics*, page btab875.

Mallo, D., De Oliveira Martins, L., and Posada, D. (2016). SimPhy : Phylogenomic Simulation of Gene, Locus, and Species Trees. *Systematic Biology*, 65(2):334–344.

Markin, A. and Eulenstein, O. (2021). Quartet-based inference is statistically consistent under the unified duplication-loss-coalescence model. *Bioinformatics*, page btab414.

McGowen, M. R., Tsagkogeorga, G., Álvarez-Carretero, S., et al. (2020). Phylogenomic Resolution of the Cetacean Tree of Life Using Target Sequence Capture. *Systematic Biology*, 69(3):479–501.

Meiklejohn, K. A., Faircloth, B. C., Glenn, T. C., et al. (2016). Analysis of a Rapid Evolutionary Radiation Using Ultraconserved Elements: Evidence for a Bias in Some Multispecies Coalescent Methods. *Systematic Biology*, 65(4):612–627.

Mirarab, S. (2019). Species Tree Estimation Using ASTRAL: Practical Considerations. *Arxiv preprint*, 1904.03826.

Mirarab, S., Bayzid, M. S., Boussau, B., and Warnow, T. (2014a). Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, 346(6215):1250463–1250463.

Mirarab, S., Bayzid, M. S., and Warnow, T. (2016). Evaluating Summary Methods for Multilocus Species Tree Estimation in the Presence of Incomplete Lineage Sorting. *Systematic Biology*, 65(3):366–380.

Mirarab, S., Nakhleh, L., and Warnow, T. (2021). Multispecies Coalescent: Theory and Applications in Phylogenetics. *Annual Review of Ecology, Evolution, and Systematics*, 52(1):247–268.

Mirarab, S., Reaz, R., Bayzid, M. S., et al. (2014b). ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548.

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

Molloy, E. K. and Warnow, T. (2018). To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303.

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

Nelesen, S. M., Liu, K., Wang, L.-S., et al. (2012). DACTAL: divide-and-conquer trees (almost) without alignments. *Bioinformatics*, 28(12):i274–i282.

Ogilvie, H. A., Bouckaert, R. R., and Drummond, A. J. (2017). StarBEAST2 Brings Faster Species Tree Inference and Accurate Estimates of Substitution Rates. *Molecular Biology and Evolution*, 34(8):2101–2114.

OneKP Initiative, O. T. P. T. (2019). One thousand plant transcriptomes and the phylogenomics of green plants. *Nature*, 574(7780):679–685.

Patel, S. (2013). Error in Phylogenetic Estimation for Bushes in the Tree of Life. *Journal of Phylogenetics & Evolutionary Biology*, 01(02):110.

Philippe, H., Vienne, D. M. d., Ranwez, V., et al. (2017). Pitfalls in supermatrix phylogenomics. *European Journal of Taxonomy*.

Price, M. N., Dehal, P. S., and Arkin, A. P. (2010). FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490.

Reddy, S., Kimball, R. T., Pandey, A., et al. (2017). Why do phylogenomic data sets yield conflicting trees? Data type influences the avian tree of life more than taxon sampling. *Systematic biology*, 66(5):857–879.

Richards, A. and Kubatko, L. (2021). Bayesian-Weighted Triplet and Quartet Methods for Species Tree Inference. *Bulletin of Mathematical Biology*, 83(9):93.

Roch, S., Nute, M., and Warnow, T. (2019). Long-Branch Attraction in Species Tree Estimation: Inconsistency of Partitioned Likelihood and Topology-Based Summary Methods. *Systematic Biology*, 68(2):281–297.

Roch, S. and Steel, M. (2015). Likelihood-based tree reconstruction on a concatenation of aligned sequence data sets can be statistically inconsistent. *Theoretical Population Biology*, 100:56–62.

Sayyari, E. and Mirarab, S. (2016). Fast Coalescent-Based Computation of Local Branch Support from Quartet Frequencies. *Molecular Biology and Evolution*, 33(7):1654–1668.

Sayyari, E. and Mirarab, S. (2018). Testing for Polytomies in Phylogenetic Species Trees Using Quartet Frequencies. *Genes*, 9(3):132.

Sayyari, E., Whitfield, J. B., and Mirarab, S. (2018). DiscoVista: Interpretable visualizations of gene tree discordance. *Molecular Phylogenetics and Evolution*, 122:110–115.

Seo, T.-K. (2008). Calculating Bootstrap Probabilities of Phylogeny Using Multilocus Sequence Data. *Molecular Biology and Evolution*, 25(5):960–971.

Shekhar, S., Roch, S., and Mirarab, S. (2017). Species tree estimation using ASTRAL: how many genes are enough? *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(5):1738–1747.

Shen, X.-x., Hittinger, C. T., and Rokas, A. (2017). Contentious relationships in phylogenomic studies can be driven by a handful of genes. *Nature Ecology & Evolution*, 1(5):0126.

Simmons, M. P. and Gatesy, J. (2015). Coalescence vs. concatenation: sophisticated analyses vs. first principles applied to rooting the angiosperms. *Molecular phylogenetics and evolution*, 91:98–122.

Smith, M. L. and Hahn, M. W. (2021). New Approaches for Inferring Phylogenies in the Presence of Paralogs. *Trends in Genetics*, 37(2):174–187.

Smith, S. A., Moore, M. J., Brown, J. W., and Yang, Y. (2015). Analysis of phylogenomic datasets reveals conflict, concordance, and gene duplications with examples from animals and plants. *BMC Evolutionary Biology*, 15(1):150.

Snir, S., Warnow, T., and Rao, S. (2008). Short Quartet Puzzling: A New Quartet-Based Phylogeny Reconstruction Algorithm. *Journal of Computational Biology*, 15(1):91–103.

Solís-Lemus, C., Yang, M., and Ané, C. (2016). Inconsistency of Species Tree Methods under Gene Flow. *Systematic Biology*, 65(5):843–851.

Springer, M. S. and Gatesy, J. (2016). The gene tree delusion. *Molecular Phylogenetics and Evolution*, 94(Part A):1–33.

Springer, M. S. and Gatesy, J. (2018). On the importance of homology in the age of phylogenomics. *Systematics and Biodiversity*, 16(3):210–228.

Susko, E. (2009). Bootstrap support is not first-order correct. *Systematic Biology*, 58(2):211–223.

Suvorov, A., Hochuli, J., and Schrider, D. R. (2020). Accurate Inference of Tree Topologies from Multiple Sequence Alignments Using Deep Learning. *Systematic Biology*, 69(2):221–233.

Szöllősi, G. J., Tannier, E., Daubin, V., and Boussau, B. (2014). The inference of gene trees with species trees. *Systematic Biology*, 64(1):e42–e62.

Takahata, N. (1989). Gene genealogy in three related populations: consistency probability between gene and population trees. *Genetics*, 122(4):957–966.

Vachaspati, P. and Warnow, T. (2015). ASTRID: Accurate Species TRees from Internode Distances. *BMC Genomics*, 16(Suppl 10):S3.

Walker, J. F., Brown, J. W., and Smith, S. A. (2018). Analyzing Contentious Relationships and Outlier Genes in Phylogenomics. *Systematic Biology*, 67(5):916–924.

Wang, L. G., Lam, T. T. Y., Xu, S., et al. (2020). Treeio: An R Package for Phylogenetic Tree Input and Output with Richly Annotated and Associated Data. *Molecular Biology and Evolution*, 37(2):599–603.

Warnow, T., Moret, B. M. E., and John, K. S. (2001). Absolute convergence: True trees from short sequences. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*.

Yan, Z., Smith, M. L., Du, P., et al. (2021). Species Tree Inference Methods Intended to Deal with Incomplete Lineage Sorting Are Robust to the Presence of Paralogs. *Systematic Biology*, page 498378.

Yin, J., Zhang, C., and Mirarab, S. (2019). ASTRAL-MP: scaling ASTRAL to very large datasets using randomization and parallelization. *Bioinformatics*, 35(20):3961–3969.

Yourdkhani, S. and Rhodes, J. A. (2020). Inferring Metric Trees from Weighted Quartets via an Intertaxon Distance. *Bulletin of Mathematical Biology*, 82(7):97.

Zhang, C., Rabiee, M., Sayyari, E., and Mirarab, S. (2018). ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6):153.

Zhang, C., Scornavacca, C., Molloy, E. K., and Mirarab, S. (2020). ASTRAL-Pro: Quartet-Based Species-Tree Inference despite Paralogy. *Molecular Biology and Evolution*, 37(11):3292–3307.