Bioinformatics, 37(16), 2021, 2317–2324
doi: 10.1093/bioinformatics/btab093
Advance Access Publication Date: 8 February 2021
Original Paper

Phylogenetics

FASTRAL: improving scalability of phylogenomic analysis

Payam Dibaeinia, Shayan Tabe-Bordbar and Tandy Warnow *

Department of Computer Science, University of Illinois, Urbana, IL 61801, USA

*To whom correspondence should be addressed.
Associate Editor: Russell Schwartz

Received on October 28, 2020; revised on February 2, 2021; editorial decision on February 3, 2021; accepted on February 4, 2021

Abstract

Motivation: ASTRAL is the current leading method for species tree estimation from phylogenomic datasets (i.e. hundreds to thousands of genes) that addresses gene tree discord resulting from incomplete lineage sorting (ILS). ASTRAL is statistically consistent under the multi-locus coalescent model (MSC), runs in polynomial time, and is able to run on large datasets. Key to ASTRAL's algorithm is the use of dynamic programming to find an optimal solution to the MQSST (maximum quartet support supertree) within a constraint space that it computes from the input. Yet, ASTRAL can fail to complete within reasonable timeframes on large datasets with many genes and species, because in these cases the constraint space it computes is too large.

Results: Here, we introduce FASTRAL, a phylogenomic estimation method. FASTRAL is based on ASTRAL, but uses a different technique for constructing the constraint space. The technique we use to define the constraint space maintains statistical consistency and is polynomial time; thus we prove that FASTRAL is a polynomial time algorithm that is statistically consistent under the MSC. Our performance study on both biological and simulated datasets demonstrates that FASTRAL matches or improves on ASTRAL with respect to species tree topology accuracy (and under high ILS conditions it is statistically significantly more accurate), while being dramatically faster—especially on datasets with large numbers of genes and high ILS—due to using a significantly smaller constraint space.

Availabilityand implementation: FASTRAL is available in open-source form at https://github.com/PayamDiba/FASTRAL.

Contact: warnow@illinois.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Species tree reconstruction underlies downstream biological studies on mechanism and history of evolution for genes and species. However, biological processes such as incomplete lineage sorting and gene duplication and loss create discordance between gene trees (i.e. evolutionary trees on genomic regions) and the species tree, and make the inference of the species tree challenging (Kubatko and Degnan, 2007). Incomplete lineage sorting (ILS) (Maddison, 1997), which can be modeled by the multi-species coalescent model (MSC) (Kingman, 1982), is believed to be one of the main processes that result in genome-wide discordance.

A simple but commonly used approach to species tree estimation from sequence alignments of multiple genomic regions is to infer the species tree from concatenated alignments using, for example, methods for maximum likelihood [e.g. RAxML (Stamatakis, 2014)]. However, this popular method has been proven to be statistically inconsistent (and even positively misleading) under the MSC, so that they may converge to the wrong tree with probability converging to 1 as the number of genes increases (Roch and Steel, 2015). Furthermore, concatenation analyses can have poor accuracy in the presence of high ILS levels (Kubatko and Degnan, 2007; Mirarab *et al.*, 2014b; Molloy and Warnow, 2018). Alternative approaches that are guaranteed to be statistically consistent have been developed. Perhaps the most accurate methods are those, such as StarBeast (Ogilvie *et al.*, 2017), that co-estimate gene trees and species trees from multi-locus inputs (consisting of multiple sequence alignments for each gene). However, these are generally computationally too intensive to use on large datasets, with difficulty analyzing datasets with 100 or more species, and can even have difficulty with large numbers of genes (Zimmermann *et al.*, 2014). SVDquartets (Chifman and Kubatko, 2014; Vachaspati and Warnow, 2018) is another statistically consistent method for species tree estimation, and operates by computing quartet trees from the input and then combining the quartet trees using a quartet amalgamation method; SVDquartets is popular, but not very scalable to large datasets because of its $\Omega(n^4)$ running time.

Species tree estimation on datasets addressing heterogeneity due to ILS is usually addressed using methods that take a set of trees inferred from various genomic regions (conventionally referred to as gene trees) as input and estimate a species tree by summarizing the input gene trees. Furthermore, several of these methods, referred to as 'summary methods', are statistically consistent and very accurate in practice. Summary methods are potentially scalable to large datasets and can have high accuracy that is competitive or better than the major competing methods, while being faster. To date, several statistically consistent summary methods have been developed including ASTRAL (Mirarab *et al.*, 2014a; Mirarab and Warnow, 2015; Zhang *et al.*, 2018), ASTRID (Vachaspati and Warnow, 2015), MP-EST (Liu *et al.*, 2010) and NJst (Liu and Yu, 2011).

Among these methods, ASTRAL is the most widely used. ASTRAL has several theoretical advantages over the other methods. For example, ASTRAL but not ASTRID, is statistically consistent under the MSC model when species are missing from gene trees under an *i.i.d.* model of missing data (Rhodes *et al.*, 2020).

ASTRAL also has excellent sample complexity (Shekhar *et al.*, 2018), and matches or improves on the other species tree estimation methods that address ILS on large datasets. Furthermore, ASTRAL runs in polynomial time. However, on some datasets, ASTRAL can be computationally intensive, exceeding the allowed time in some computing environments (e.g. 24 hours) (Molloy and Warnow, 2019). Moreover, although biological datasets of interest in phylogenomic analysis may only have tens or a few hundred species (e.g. the Avian Phylogenomics dataset had only 48 species), they can easily have many thousands of genes. As a result, species tree estimation of datasets with even smallish to 'moderate' numbers of species can be very computationally challenging when the number of genes is large.

In this study, we focus on ASTRAL and seek to improve its running time, focusing on addressing the challenge when the number of genes is large (i.e. the common problem in phylogenomics). Although ASTRAL runs in polynomial time, its running time is dominated by the size of a set X of 'allowed bipartitions' that it computes from the input (and the running time is almost quadratic in $|X|$). We show that a change to how ASTRAL computes its set X can be made that substantially reduces its running time without losing statistical consistency. We explore different ways for defining the set X that rely on subsampling from the input gene trees, and devise an approach that enables high accuracy and low running time, and still ensures statistical consistency. Our experimental study validates this approach on both biological and simulated datasets, including on datasets with gene trees having multiple individuals and missing data. Our approach, which we refer to as 'FASTRAL', matches or improves on ASTRAL with respect to topological accuracy and is much faster, especially on datasets with high gene tree heterogeneity and large numbers of genes. In particular, FASTRAL completes in about two minutes on the Avian Phylogenomics project dataset (Jarvis *et al.*, 2014) of 48 species and 14 446 genes, while ASTRAL requires approximately 32 h. Thus, FASTRAL is a very fast alternative to ASTRAL.

2 Materials and methods

ASTRAL: Given an unrooted tree T on leafset $\mathcal{S}$ with $|S| = n$, we define $C(T)$ to be the set of bipartitions on the leafset of T defined by the edges of T; thus $C(T)$ will contain n trivial bipartitions (that split one leaf off from the other leaves) corresponding to the leaves of T and additional non-trivial bipartitions corresponding to the internal edges of T. If T is binary, then $|C(T)| = 2n - 3$. Given a set $\mathcal{G}$ of gene trees (with leaves taken from $\mathcal{S}$), the **quartet support** of a tree T on leafset $\mathcal{S}$ is $\sum_{t \in \mathcal{G}} |Q(T) \cap Q(t)|$, where $Q(t)$ denotes the set of quartet trees induced by four-leaf trees in t. The input to ASTRAL is a set $\mathcal{G}$ of k gene trees, each leaf-labelled by species drawn from set $\mathcal{S}$ of n species. ASTRAL then uses the input to compute a set X of allowed bipartitions, and uses a polynomial time dynamic programming (DP) algorithm on the pair $(\mathcal{G}, X)$ to find a species tree T on $\mathcal{S}$ that maximizes the total quartet support (with respect to the input gene trees) subject to $C(T) \subseteq X$. This is the Constrained Maximum Quartet Support Species Tree (**Constrained-MQSST**) problem. ASTRAL's DP algorithm operates by implicitly calculating the MQSST criterion score without needing to explicitly examine all $\Theta(n^4)$ quartets. Furthermore, ASTRAL can also take a pair $(\mathcal{G}, X)$ as input and then apply its DP algorithm to that pair, thus allowing the user the flexibility of computing the constraint set using other techniques.

The most recent version of ASTRAL, referred to as ASTRAL-III, runs in $O(D|X|^{1.726})$ where D denotes the number of distinct 'tripartitions' in the input gene trees $\mathcal{G}$ (where a tripartition is defined for each node in each gene tree, and is produced by deleting the node and its incident edges from the gene tree, thus splitting the leafset into three parts). The default way that X is defined in ASTRAL is guaranteed to include all the bipartitions in $\mathcal{G}$, and the first design for ASTRAL (i.e. ASTRAL-I) used only these bipartitions. Hence, in the simplest case, $|X|$ is $O(nk)$. However, as ASTRAL continued to be refined, it expanded the set X to add additional biparititions, but requiring that it not violate the $|X| = O(nk)$ condition. This guarantees that the final running time of ASTRAL-III is $O(D(nk)^{1.726})$. Furthermore, ASTRAL-III is guaranteed statistically consistent, since X always contains the bipartitions from the input gene trees (Theorem 2 from Mirarab *et al.* (2014a)).

FASTRAL: By design, the size of X dominates the running time of ASTRAL, and can make ASTRAL computationally intensive. The key observation that led to the design of FASTRAL is that we can ensure statistical consistency by having X be the bipartitions found in a set of estimated species trees, rather than the gene trees, provided that the set of estimated species trees are computed using statistically consistent methods. Therefore, from a purely theoretical perspective, we can replace the default way that ASTRAL computes X by a set of species trees we can compute using fast and statistically consistent methods. Here we describe how FASTRAL operates, which depend on how it sub-samples from the input set $\mathcal{G}$ of gene trees (Step 1) and the choice of a method M for computing species trees on each sub-sample (Step 2):

1. Step 1: Construct a collection of sub-samples of the gene trees in $\mathcal{G}$.
2. Step 2: For each sub-sample, run M to obtain a tree on S.
3. Step 3: Let X be all bipartitions appearing in any tree obtained in Step 2.
4. Step 4: Run ASTRAL on the pair $(\mathcal{G}, X)$.

As we now show, we can define sub-sampling strategies and choices for M that ensure statistical consistency and polynomial time, and that also provide very good empirical accuracy.

FASTRAL builds set X from bipartitions of species trees or supertrees inferred from the input genes by any auxiliary method of choice M. To increase the diversity among species trees and yet utilizing the full resolution of the input gene trees, FASTRAL divides the input gene trees into $m > 1$ overlapping sets, so that the ith sample contains $t_i \leq k$ gene trees (Fig. 1). There are numerous ways to draw the m sub-samples fed to the auxiliary method. Here, we explore two simple sampling strategies where sub-samples are drawn

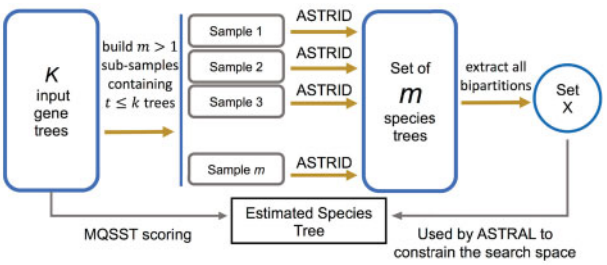

Fig. 1. The FASTRAL pipeline. FASTRAL creates m sub-samples from the input set $\mathcal{G}$ of gene trees, constructs an ASTRID tree on each sub-sample, and uses the bipartitions from the m ASTRID trees for the constraint set X. ASTRAL is then run on input $(\mathcal{G}, X)$

uniformly at random without replacement. In one approach (i.e. same-size sampling), we limit the sub-samples to a specific size (i.e. 25% of input gene trees), while in the other approach (i.e. variable-size sampling), we allow the sub-samples to be of variable sizes. Then a species tree is inferred for each sample using the auxiliary method M. FASTRAL aggregates all unique bipartitions among the m species trees and uses it as the set X of allowed bipartitions. If polytomies (i.e. nodes of degree greater than three) are present in any of the trees constructed by method M, FASTRAL resolves them using a UPGMA tree built from $\mathcal{G}$, which is similar to ASTRAL-III's approach for resolving polytomies. However, in contrast to ASTRAL-II (Mirarab and Warnow, 2015) and ASTRAL-III (Zhang *et al.*, 2018), FASTRAL does not further expand this set X. It is worth mentioning that trees generated by method M are only used to construct the set X, and do not impact ASTRAL's analysis otherwise (e.g. they are not used to define the quartet support criterion or the tripartition weighting performed in ASTRAL).

Here, we used ASTRID (Vachaspati and Warnow, 2015), which is statistically consistent under the MSC and one of the few methods that can run on very large datasets (thousands of genes and species). ASTRID uses a distance-based approach where the first step computes an 'average internode distance matrix' (i.e. matrix of pairwise distances averaged across the gene trees, using the number of internal nodes on the path as the distance), and the second step computes a tree from the distance matrix. As shown by Allman *et al.* (Allman *et al.*, 2018), the average internode distance matrix converges to an additive matrix for the true species tree with probability converging to 1, and so ASTRID [when used with methods such as FastME (Lefort *et al.*, 2015) or Neighbor Joining (Saitou and Nei, 1987)] is statistically consistent under the MSC.

Prior studies comparing ASTRID and ASTRAL shows that both have better accuracy on datasets with large numbers of species than other statistically consistent methods, and the relative performance between them is mixed: in some cases they are tied, sometimes ASTRAL is more accurate and sometimes ASTRID is more accurate (Vachaspati and Warnow, 2015).

2.1 Statistical consistency

Theorem 1.Assume that the random sampling step of FASTRAL generates m sub-samples, such that there is *at least one subsample*S_j whose size also increases to infinity with the number of genes (i.e.$|S_j| = \lceil f_j(|G|) \rceil$ where $f_j : R \rightarrow R^+$ is a real valued increasing function satisfying $lim_n f_j(n) = \infty$). Assume also that M is a summary method (i.e.M takes as input a set of unrooted gene tree topologies and estimates a species tree) that is statistically consistent under the MSC. Then, FASTRAL is statistically consistent under the MSC model when used with M and this sampling strategy.

Proof. Under the conditions of the theorem, as the number k of genes in $\mathcal{G}$ increases to infinity, the number of genes in sub-sample S_j also increases to infinity, and so the tree obtained using M on S_j will converge to the true species tree with probability converging to 1. Furthermore, since tree topologies are discrete objects, for every $\epsilon > 0$ there is a number k_ϵ of genes so that the probability that M returns the true species tree topology for sub-sample S_j given k_ϵ genes is at least $1 - \epsilon$. Therefore, as the number k of genes in $\mathcal{G}$ increases, with probability converging to 1, the set X constructed by FASTRAL will include all the bipartitions that are present in the true species tree. Hence, for a sufficient number of genes, the true species tree will be a feasible solution to the Constrained-MQSST optimization problem solved by ASTRAL. Note that the proof for Theorem 2 from Mirarab *et al.* (2014a) that establishes ASTRAL to be statistically consistent under the MSC using its default technique for computing X (which sets X to the bipartitions from the input gene trees) only depends on X containing, in the limit, all the bipartitions from the species set; hence, the same argument ensures that FASTRAL is statistically consistent under the MSC. □

2.2 Asymptotic running time

On each sub-sample S_i of $\mathcal{G}$ containing $t_i \leq k$ gene trees on n taxa, ASTRID runs in $O(t_i n^2 + n^3)$. After aggregating the bipartitions of ASTRID's species tree into set X, ASTRAL takes $O(D|X|^{1.726})$ to run where D is the number of distinct tripartitions in the input set $\mathcal{G}$ of gene trees and $|X| = O(mn)$ when generating m sub-samples. Therefore, the total asymptotic running time of FASTRAL will be $O(mkn^2 + mn^3 + D(mn)^{1.726})$.

Theorem 2.*When used with ASTRID for computing species trees on the sub-samples, FASTRAL runs in* $O(mkn^2 + mn^3 + D(mn)^{1.726})$, *where* n *is the number of species,* k *is the number of genes,* D *is the number of distinct tripartitions in the input gene trees, and* m *is the number of sub-samples it analyzes. When* $k > n$ *(which is typical for phylogenomic datasets), the running time simplifies to* $O(mkn^2 + D(mn)^{1.726})$.

Comments. Note that when using a sampling strategy in which m is much smaller than k (even when $m = k/c$ for some constant $c > 1$, such as we explore in this study), FASTRAL's asymptotic running time is much faster than the asymptotic running time for ASTRAL, which is $O(D(nk)^{1.726})$, and this good running time would still hold if ASTRID was replaced by another method that ran in $O(n^3)$ time. With respect to statistical consistency, FASTRAL depends on its algorithmic parameters: how it selects M (i.e. the method for computing trees on subsets of the genes) and its sub-sampling strategy, and the conditions under which FASTRAL is guaranteed statistically consistent under the MSC are very modest (i.e.M is a statistically consistent summary method and the sub-sampling strategy includes at least one sub-sample whose size increases to infinity as the number of genes increases to infinity). However, empirical performance (i.e. accuracy on data) can be impacted by the choices for M and sub-sampling strategy, and in ways that are more complex. For example, picking only one sub-sample will mean that FASTRAL is identical to M on the sub-sample, which is clearly not a good strategy. More generally, what is wanted is a large enough number m of sub-samples that the set X that is created does not constrain the search space too much, but when m is very large then the running time will increase. Therefore, while the theorem regarding statistical consistency holds for many random sampling strategies and choices of M, for accuracy and running time considerations, each must be chosen with care.

3 Experimental study

Overview. In our design and evaluation of FASTRAL, we chose ASTRID as the method M to compute trees on the sub-sampled collections of genes. Therefore, the remaining algorithmic parameter to determine is the sub-sampling strategy, and its impact on species tree error. we computed species tree error using the FN error rate, which is the fraction of the number of bipartitions that appear in the true species tree but not in the estimated species tree [this is identical to the Robinson-Foulds (Robinson and Foulds, 1981) error rate since the species trees are binary]. We performed three experiments. In Experiment 1, we compared two sampling strategies to evaluate the impact on species tree accuracy, and selected one for further analysis. In Experiment 2, we compared FASTRAL to ASTRAL on simulated datasets where each gene tree has a single leaf for each species. In Experiment 3, we compared FASTRAL to ASTRAL on simulated datasets where the genes have multiple individuals per species, and some genes may be incomplete (i.e. may be missing species). In Experiment 4, we compared FASTRAL to ASTRAL on the Avian Phylogenomics project dataset with 48 birds and 14 446 genes (Jarvis *et al.*, 2014).

Datasets. We used biological and simulated datasets from prior studies (Table 1). We selected three model conditions from the ASTRAL-II (Mirarab and Warnow, 2015) simulated datasets; the estimated gene trees for these model conditions were obtained from https://sites.google.com/eng.ucsd.edu/datasets/astral/astral-ii. Using the nomenclature for these models from ASTRID, MC1, MC6 and MC11 have 1000 genes and either 200 (for MC1 and MC6) or 1000

Table 1. Characteristics of the datasets used in this study

Dataset	No. taxa	No. genes	ILS (AD %)	Other
MC1 (Mirarab and Warnow, 2015)	200	1000	69	No. gens.: 500 K, spec. rate: 1e-6
MC6 (Mirarab and Warnow, 2015)	200	1000	9	No. gens: 10 M; spec. rate: 1e–7
MC11 (Mirarab and Warnow, 2015)	1000	1000	35	No. gens: 2 M; spec. rate: 1e-6
D2 (Rabiee *et al.*, 2019)	200	1000	48-53	No. gens: 0.5 M; spec. rate 1e-6
Avian (Jarvis *et al.*, 2014)	48	14446	N/A	Unknown true gene and species trees

Note: The MC1, MC6, MC11 and D2 model conditions are simulated, and their statistics are taken from the cited papers; each has 50 replicates. The avian dataset is biological. For the simulated datasets, the number of generations controls the ILS level (fewer generations results in smaller species tree height and higher ILS) and the speciation rate controls whether the speciation is towards the root (1e7) or towards the leaves (1e6). The ILS level is reported using the Average Discordance (AD) between true gene trees and true species trees, computed using the normalized Robinson-Foulds distance to produce a value between 0 and 1. Thus, MC1 is very high ILS, D2 is high ILS, MC11 is moderate ILS and MC6 is low ILS.

(for MC11) species, and there are 50 replicates per model condition. MC6 has low ILS [i.e. AD = 9%, where AD denotes the average discordance, measured using normalized Robinson-Foulds (Robinson and Foulds, 1981) distances between true gene trees and true species trees], MC11 has moderate ILS (AD = 35%), and MC1 has high ILS (AD = 69%). Also, we used the D2 model condition of simulated datasets from the ASTRAL-multi (Rabiee *et al.*, 2019) study (the estimated gene trees were obtained from https://maryamrabiee.github.io/ASTRAL-multi/). The D2 model condition (50 replicates) has 1000 genes and 200 species with five individuals per species, and high ILS (AD = 48–53%). The avian biological dataset (with 48 species and 14 446 estimated gene trees) was obtained from https://gitlab.com/esayyari/ASTRALIII/-/blob/master/ml.tar.gz. The analysis reported in Jarvis *et al.* (2014) on this avian dataset suggests that it has very high ILS, as all of the estimated gene trees were different from the species tree computed on the datast using ExaML (Kozlov *et al.*, 2015). We used the gene trees for these model conditions, which were estimated using RAxML (Stamatakis, 2014). Since the gene trees are estimated, they all have some gene tree estimation error.

Methods. ASTRAL: Version 5.7.3 was used with default parameters for all the arguments (Zhang *et al.*, 2018).

ASTRID: The linux version of ASTRID-1.4 (https://github.com/pranjalv123/ASTRID-1) was used with the 'auto' mode for distance matrix calculations. Since the internode distance matrix does not have any missing entries, the 'auto' mode uses FastME (Lefort *et al.*, 2015) with Nearest Neighbor Interchanges (NNI) as the distance method for tree estimation.

FASTRAL: Version 1.0.0 was used (https://github.com/PayamDiba/FASTRAL). For intermediate estimation of species trees, we used ASTRID-1.4 (with the same setting as described above) for the MC1, MC6, MC11 model condition datasets and the avian biological dataset, and ASTRID-2 (https://github.com/pranjalv123/ASTRID) for the D2 datasets (since ASTRID-1.4 does not support the multi-individual mode). ASTRID-2 was used with the 'auto' mode for distance matrix calculations. The intermediate species trees found by ASTRID were fed to ASTRAL (flag '-f') for constructing set X, and expansion of set X with heuristics was disabled (flag '-p' set to 0). Moreover, we modified ASTRAL Version 5.7.3 in order to restrict set X to only the union of the bipartitions of the intermediate species trees (i.e. to prevent the inclusion of input gene trees' bipartitions in set X). This modified version of ASTRAL 5.7.3 is distributed with FASTRAL package. In this study we ran FASTRAL with two different general settings: (i) **variable-size sampling**: we generate 51 samples, one of which contains all of the gene trees, 10 samples containing 50% of the gene trees, 20 samples containing 25% of the gene trees and 20 samples containing 10% of the gene trees, and (ii) **same-size sampling**: We generate 51 samples each of which contains 25% of the gene trees. In each case, gene trees are sampled uniformly at random without replacement.

False negative error rate: We computed the number of false negative (FN) branches (i.e. edges in the reference species tree not appearing in the estimated trees) using a script obtained from https://github.com/redavids/phylogenetics-tools/tree/master/comparetrees. We obtain the FN rate by dividing this by n—3, the number of internal branches in a binary tree on n leaves.

4 Results

4.1 Results from experiment 1

We evaluated the impact of sampling strategy (same-size and variable-size sampling, see Methods) on FASTRAL on two model conditions (MC6 and MC11). FASTRAL achieves comparable accuracy under both sampling strategies, but there is a small advantage to using the variable-size sampling (Supplementary Fig. S1). Therefore, we selected the variable-size sampling strategy (FASTRAL_51S_varT) for future analyses. We conjecture that the improvement of variable-size sampling over same-size sampling is due in part to the inclusion of the sample that contains all the gene trees, but future work is needed to fully explore the impact of sampling strategy.

4.2 Results from experiment 2

Tree error. As seen in Figure 2, for all three model conditions, increasing the number of genes results in decreases in error for all methods, with the biggest decrease occurring between 100 and 500 genes, and then a smaller decrease between 500 and 1000 genes. Results under the model conditions are somewhat different, and so are discussed separately.

For the MC6 model condition, which has 200 species and low ILS (AD = 9%), ASTRAL and FASTRAL are essentially tied for accuracy at all numbers of genes, but ASTRID has higher error. For the MC11 model condition, which has 1000 species and moderate ILS (AD = 35%), ASTRAL and FASTRAL have the best accuracy at 100 genes, and then essentially tie with ASTRID for 500 and 1000 genes. For the MC1 model condition, with 200 species and high ILS (AD = 69%), ASTRID has variable accuracy, but FASTRAL is strictly better than ASTRAL at 500 and 1000 genes and ties with ASTRAL at 100 genes. These trends indicate that the number of genes and ILS level affects both the absolute and relative accuracy of species tree estimation methods, and that FASTRAL has an advantage for accuracy under the high ILS condition.

We evaluate the statistical significance (P-value < 0.05) of the difference in species tree error between ASTRAL and FASTRAL (Supplementary Table S1). Under the MC6 and MC11 conditions, ASTRAL and FASTRAL do not have statistically significant differences in species tree accuracy for any model condition and number of genes (P-value > 0.05). Under the MC1 condition, FASTRAL has a statistically significant advantage over ASTRAL for the 1000-gene case, and is almost statistically significantly better on the 500-gene case. Thus, under high ILS, FASTRAL can be significantly more accurate than ASTRAL, and under lower ILS conditions the differences in accuracy between the two methods are not significant.

Running time. The running times on these model conditions show very large differences between methods, but ILS level, number of genes, and number of species impact the time usage (Fig. 2 and Supplementary Table S2). Under all model conditions and numbers of genes, ASTRID is the fastest method, finishing in just seconds,

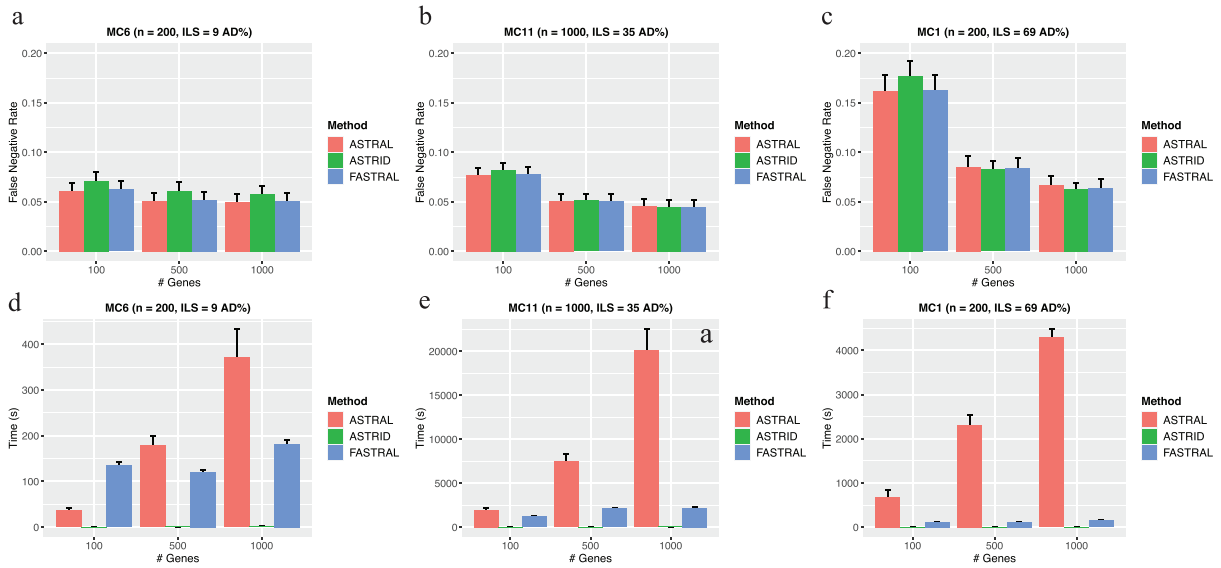

Fig. 2. Experiment 2: Comparison between ASTRAL, ASTRID and FASTRAL, with varying numbers of gene trees. Species tree error is shown in (**A**), (**B**) and (**C**) for low (MC6), moderate (MC11) and high (MC1) ILS model conditions, respectively; running time is shown in (**D**), (**E**) and (**F**) for the same model conditions. Note that the y-axis ranges for the running time differ between the three subfigures. Error bars indicate the standard error from mean

and its running time is not noticeably impacted by the number of genes. A comparison between ASTRAL and FASTRAL shows different trends. On the low ILS model condition (MC6) with 200 species, ASTRAL is faster than FASTRAL given 100 genes, somewhat slower given 500 genes, and again slower given 1000 genes (approx. 371 s compared to approx. 182 s). Thus, FASTRAL is faster than ASTRAL given 500 or 1000 genes, but the ratio is not substantial (2.04), and both methods are very fast under this low ILS condition.

However, under moderate or high ILS, the running times increase and the difference in running time between the methods also increases, so that FASTRAL is faster than ASTRAL for all numbers of genes. Both methods have their highest running times on the MC11 condition with 1000 genes, indicating that the number of species (1000) for this condition has a substantial impact on running time. On the moderate ILS 1000-species MC11 condition with 1000 genes, ASTRAL uses 20125 s on average and FASTRAL uses 2191 s (i.e. ASTRAL is 9.19 times slower than FASTRAL). On the high ILS 200-species MC1 condition with 1000 genes, ASTRAL uses 4293 s and FASTRAL uses 153 s, a ratio of 28.06 in running time. Thus, both methods running times are impacted by the ILS level, number of taxa and number of genes, and FASTRAL's running time advantage over ASTRAL ranges from modest (e.g. a ratio of 2.04 on the MC6 low ILS condition with 200 species) to very large (a ratio of 28.06 on the high ILS MC1 condition).

To understand the different impact of ILS level on the ASTRAL and FASTRAL running times, recall that increases in ILS lead to increases in topological heterogeneity across the gene trees, and this automatically increases the size of $|X|$ as ASTRAL computes it. However, FASTRAL computes its set X from the ASTRID trees computed on sub-sampled sets of gene trees, which ameliorates the impact of increasing numbers of genes on the size of X (and even results in decreases in the size of X as we will discuss next).

Overall, we see that FASTRAL has a substantial running time advantage over ASTRAL, but especially when there is moderate to high ILS or when there is a large number of genes or species.

MQSST scores and properties of the constraint set X. To understand these trends, we examine the size of the set X computed by FASTRAL and the density of the true species tree bipartitions in X (i.e. the ratio between $|TP|$ and $|X|$, where TP denotes the true positives in X, or the species tree bipartitions in X). Figure 3 shows results for this analysis on the MC11 datasets, and shows that our approach significantly improves search space efficiency on MC11 datasets (Supplementary Fig. S2 suggests similar improvements on MC1 and MC6). As expected, the constraint set X increases in size for ASTRAL as the number of genes increases. Interestingly, the constraint set X produced by FASTRAL decreases in size as the number of genes increases, which can be explained by the ASTRID trees on the subsets becoming topologically more similar to each other. The decrease in the size of $|X|$ improves the running time, since ASTRAL's running time depends almost quadratically on the size of its search space; hence, the gap in running time between ASTRAL and FASTRAL increases with the number of genes (Fig. 2E). Furthermore, the high density of the true species tree bipartitions shows that we achieve this running time improvement without sacrificing species tree accuracy (Supplementary Fig. S2 suggests similar trends and improvements on MC1 and MC6). We also examined the Maximum Quartet Support Supertree (MQSST) scores produced by ASTRAL and FASTRAL, as the two methods differ only in how they constrain the search space. FASTRAL and ASTRAL are nearly identical on the three model conditions (Supplementary Fig. S3), showing that the change in the constraint space used by FASTRAL is not detrimental.

4.3 Results from experiment 3

Next, we compare FASTRAL to ASTRAL on a challenging simulated dataset containing five individuals per species, and where there can be species missing from gene trees under an *i.i.d.* missing data model (25% of species are missing in 25% of genes). As seen in Figure 4, FASTRAL and ASTRAL have nearly the same accuracy under all the tested conditions (and the differences are not statistically significant, with a P-value of 0.07 and 0.26 for 0% and 25% missing data respectively), and FASTRAL is much faster than ASTRAL. Thus, the relative performance of FASTRAL and ASTRAL for multi-individual datasets is similar here to that observed for the other experiments, even in the presence of *i.i.d.* missing data.

4.4 Results on the avian biological dataset

As illustrated by the results on simulated datasets, FASTRAL's main advantage as compared to ASTRAL is the decreased run-time on datasets with large number of genes. In order to examine the extent of this speed-up on genome-wide biological datasets, we ran both ASTRAL and FASTRAL on the avian biological dataset (Jarvis *et al.*, 2014) and compared the resulting trees and the corresponding run-time and optimization scores. Table 2 shows the running time and MQSST optimization score for both methods (higher scores show higher consistency of the inferred species tree with the quartets

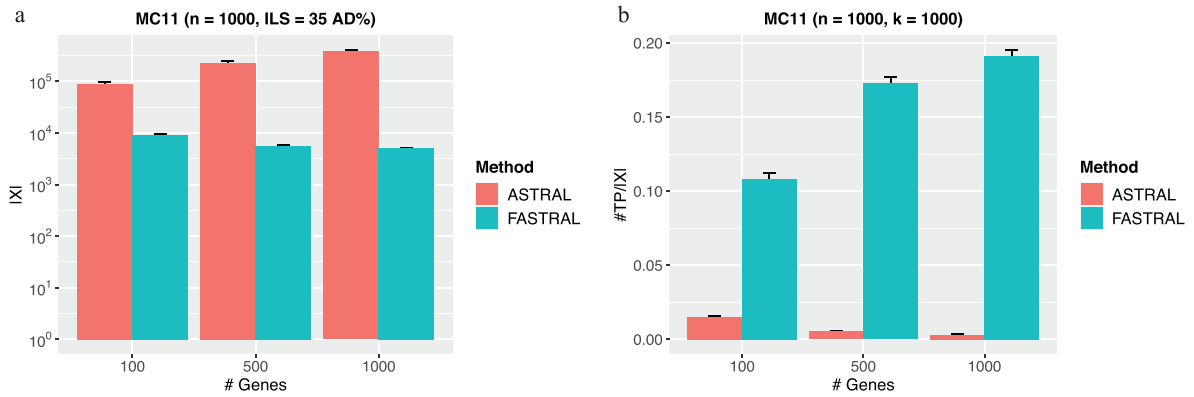

Fig. 3. Understanding the impact of FASTRAL's strategy on the constrained search. A: Comparing size of set X between ASTRAL and FASTRAL on MC11 datasets with 1000 genes (note that the y-axis is logarithmically scaled). B: The density of the true species tree bipartitions in the set X (i.e. space efficiency) for ASTRAL and FASTRAL on MC11 dataset with 1000 genes and 1000 species. Error bars represent standard error

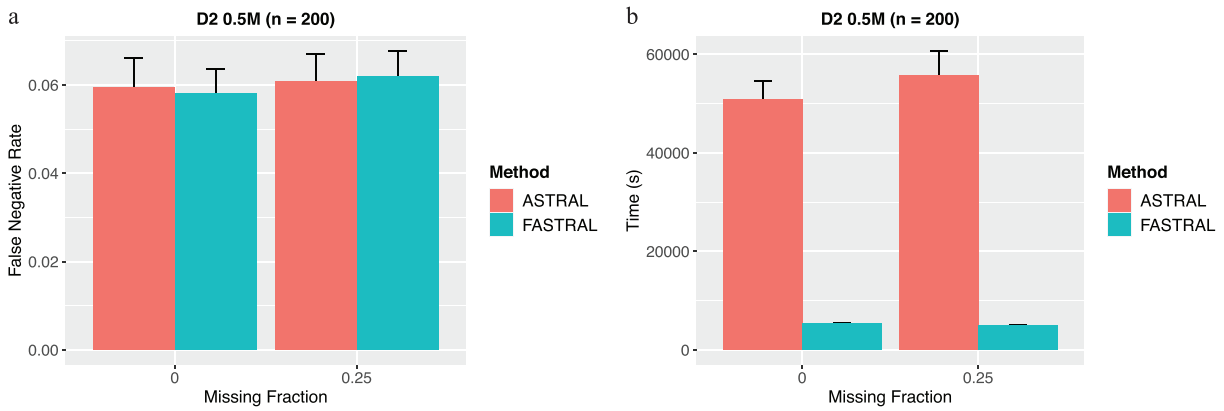

Fig. 4. Experiment 3: Comparing FASTRAL with ASTRAL on a multi-individual dataset in both presence and absence of missing data. A: Comparison between tree error rates (averaged over 50 replicates) ASTRAL and FASTRAL for the ASTRAL-multi D2 dataset (200 species, 1000 genes). B: Comparison between running time of ASTRAL and FASTRAL for the ASTRAL-multi D2 dataset with 200 species and 1000 genes. Error bars represent standard error

Table 2. Running time (in hours) and optimization score (proportion of quartet trees satisfied) achieved by FASTRAL and ASTRAL on the avian biological dataset from Jarvis *et al.* (2014) with 48 species and 14,446 genes

Method	Time (h)	Optimization score
ASTRAL	~32	0.50038
FASTRAL	~0.04	0.50024

of the input gene trees). FASTRAL runs ~800 times faster that ASTRAL-III, indicating a great improvement in running time. It is worth mentioning that 71% of FASTRAL's run time (~ 104 s) was consumed for building set X (i.e. running ASTRID on sub-samples) and only 29% of its run time (~ 42 s) was used for running ASTRAL. This shows the huge improvement in ASTRAL's running time when operates under an optimal constraint search space (42 s versus 32 h). Furthermore, their inferred species trees (Fig. 5) differ in only three bipartitions. Comparing the resulted trees, we observe that as opposed to ASTRAL, FASTRAL places Red-crested turaco and Houbara bustard close to Common cuckoo. This assignment seems to be in agreement with previously published trees in Jarvis *et al.* (2014) computed using ExaML (Kozlov *et al.*, 2015) or MP-EST combined with statistical binning (Mirarab *et al.*, 2014c). Interestingly, the trees inferred by both ASTRAL and FASTRAL differ in particular branches from the trees inferred from non-coding data [e.g. the intron MP-EST tree in Jarvis *et al.* (2014) and the nucleotide trees in Houde *et al.* (2019) and Reddy *et al.* (2017)]. Such datatype-dependent discordances have been previously reported in the literature (Braun and Kimball, 2021). See Supplementary Figure S4 for the ASTRID tree inferred from these data.

5 Summary and conclusions

Accurate species tree estimation in the presence of incomplete lineage sorting (ILS), as modeled by the multi-species coalescent (MSC), is computationally and statistically challenging. ASTRAL is the leading species tree estimation that is statistically consistent under the MSC model and that can analyze large datasets; however, when given large numbers of genes, ASTRAL can be computationally challenging. Specifically, ASTRAL operates by solving an NP-hard optimization problem (MQSST) within a constrained search space, based on a set X of 'allowed bipartitions' that it computes from the input. When the input set of gene trees is large or there is substantial heterogeneity between gene trees, ASTRAL's set X can become very large, making the running time in some cases excessively large.

Here, we have presented FASTRAL, which uses a generalizable and flexible technique for constructing the set X of allowed bipartitions (compared to how ASTRAL constructs this set) and in so doing improves on ASTRAL. By design, FASTRAL is much faster than ASTRAL because the set of allowed bipartitions is much smaller than ASTRAL's. However, importantly, FASTRAL maintains statistical consistency, is polynomial time, and is as accurate (and in some cases more accurate) than ASTRAL. The improvement of FASTRAL over ASTRAL in terms of accuracy is most noteworthy when there is high ILS and a large number of genes, but FASTRAL

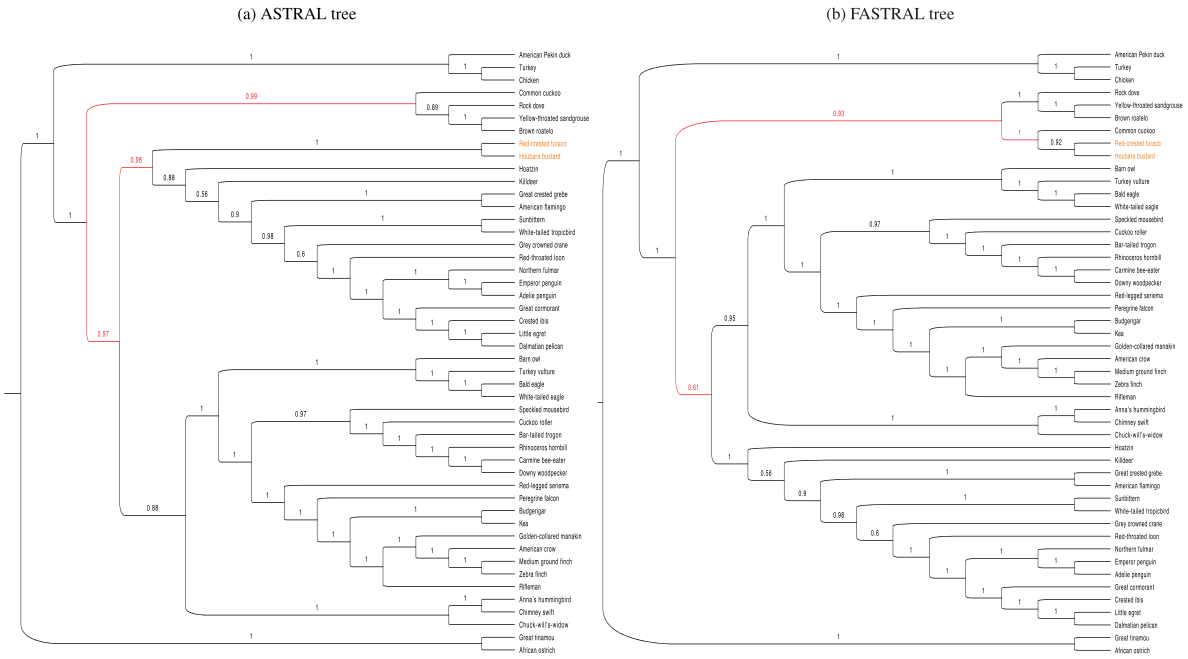

Fig. 5. Experiment 4: Inferred trees on the avian biological dataset with 48 species and 14,446 genes from Jarvis *et al.* (2014), using (left) ASTRAL and (right) FASTRAL. Taxa shown in orange and bipartitions shown in red mark the difference between the two trees. Branch support values (local posterior probabilities) are shown as obtained by ASTRAL. The RF distance between ASTRAL and FASTRAL trees is 3

is almost always much faster than ASTRAL. Thus, this simple approach provides a new and very fast technique to estimate species trees from multi-locus datasets that matches the accuracy of the current leading method, ASTRAL and uses a fraction of the time.

Future work should explore variants of this approach where other fast methods (besides ASTRID) are used to construct trees on the sub-sampled genes, and other sub-sampling strategies should also be explored. Future work should also evaluate performance (running time and accuracy) on additional simulated and biological datasets to evaluate how FASTRAL performs under a variety of circumstances, and include comparisons to other methods for species tree estimation, including methods such as RevPoMo (Schrempf *et al.*, 2016) not established to be statistically consistent under the multi-species coalescent model at this time. Finally, constrained optimization is a basic technique in many phylogenomic analyses [e.g. SVDquest (Vachaspati and Warnow, 2018) and FastMulRFS (Molloy and Warnow, 2020)], and so this approach could be used in other contexts as well.

Acknowledgements

This work was the result of a final course project by PD and STB for the graduate course, CS 581: Algorithmic Genomic Biology, from Spring 2020 at the University of Illinois.

Data Availability

All datasets evaluated in this study are from prior publications and are available from https://sites.google.com/eng.ucsd.edu/datasets/astral/astral-ii. https://maryamrabiee. //github.io/ASTRAL-multi/, and https:// //gitlab.com/esayyari/ASTRALIII/-/blob/master/ml.tar.gz.

Funding

This work was supported by the Grainger Foundation through a Grainger Engineering Breakthroughs Initiative chair to T.W.

Conflict of Interest: none declared.

References

Allman,E.S. *et al.* (2018) Species tree inference from gene splits by unrooted STAR methods. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **15**, 337–342.

Braun,E.L. and Kimball,R.T. (2021) Data types and the phylogeny of Neoaves. *Birds*, **2**, 1–22.

Chifman,J. and Kubatko,L. (2014) Quartet inference from SNP data under the coalescent model. *Bioinformatics*, **30**, 3317–3324.

Houde,P. *et al.* (2019) Phylogenetic signal of indels and the Neoavian radiation. *Diversity*, **11**, 108.

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

Kingman,J.F.C. (1982) The coalescent. *Stochastic Processes Appl.*, **13**, 235–248.

Kozlov,A.M. *et al.* (2015) ExaML version 3: a tool for phylogenomic analyses on supercomputers. *Bioinformatics*, **31**, 2577–2579.

Kubatko,L. and Degnan,J. (2007) Inconsistency of phylogenetic estimates from concatenated data under coalescence. *Syst. Biol.*, **56**, 17–24.

Lefort,V. *et al.* (2015) FastME 2.0: a comprehensive, accurate, and fast distance-based phylogeny inference program. *Mol. Biol. Evol.*, **32**, 2798–2800.

Liu,L. and Yu,L. (2011) Estimating species trees from unrooted gene trees. *Syst. Biol.*, **60**, 661–667.

Liu,L. *et al.* (2010) A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC Evol. Biol.*, **10**, 302.

Maddison,W.P. (1997) Gene trees in species trees. *Syst. Biol.*, **46**, 523–536.

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

Mirarab,S. *et al.* (2014a) ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, **30**, i541–i548.

Mirarab,S. *et al.* (2014b) Evaluating summary methods for multi-locus species tree estimation in the presence of incomplete lineage sorting. *Syst. Biol.*, **65**, 366–380.

Mirarab,S. *et al.* (2014c) Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, **346**, 1250463–1250463.

Molloy,E.K. and Warnow,T. (2018) To include or not to include: the impact of gene filtering on species tree estimation methods. *Syst. Biol.*, **67**, 285–303.

Molloy,E.K. and Warnow,T. (2019) Statistically consistent divide-and-conquer pipelines for phylogeny estimation using NJMerge. *Algorithms Mol. Biol.*, **14**, 14.

Molloy,E.K. and Warnow,T. (2020) FastMulRFS: fast and accurate species tree estimation under generic gene duplication and loss models. *Bioinformatics*, **36**, i57–i65.

Ogilvie,H.A. *et al.* (2017) StarBEAST2 brings faster species tree inference and accurate estimates of substitution rates. *Mol. Biol. Evol.*, **34**, 2101–2114.

Rabiee,M. *et al.* (2019) Multi-allele species reconstruction using ASTRAL. *Mol. Phylogenet. Evol.*, **130**, 286–296.

Reddy,S. *et al.* (2017) Why do phylogenomic data sets yield conflicting trees? Data type influences the avian tree of life more than taxon sampling. *Syst. Biol.*, **66**, 857–879.

Rhodes,J.A. *et al.* (2020) NJst and ASTRID are not statistically consistent under a random model of missing data. *arXiv Preprint arXiv: 2001.07844*.

Robinson,D. and Foulds,L. (1981) Comparison of phylogenetic trees. *Math. Biosci.*, **53**, 131–147.

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

Saitou,N. and Nei,M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, **4**, 406–425.

Schrempf,D. *et al.* (2016) Reversible polymorphism-aware phylogenetic models and their application to tree inference. *J. Theor. Biol.*, **407**, 362–370.

Shekhar,S. *et al.* (2018) Species tree estimation using astral: how many genes are enough?*IEEE/ACM Trans. Comput. Biol. Bioinf.*, **15**, 1738–1747.

Stamatakis,A. (2014) RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, **30**, 1312–1313.

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

Vachaspati,P. and Warnow,T. (2018) SVDquest: improving SVDquartets species tree estimation using exact optimization within a constrained search space. *Mol. Phylogenet. Evol.*, **124**, 122–136.

Zhang,C. *et al.* (2018) ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, **19**, 153.

Zimmermann,T. *et al.* (2014) BBCA: improving the scalability of BEAST using random binning. *BMC Genomics*, **15**, S11.