UNIVERSITY OF CALIFORNIA SAN DIEGO

Scalable Algorithms for Inference and Simulation under Complex Phylogenetic Models

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Bioinformatics and System Biology

by

Chao Zhang

Committee in charge:

Professor Siavash Mirarab, Chair
Professor Pavel Pevzner, Co-Chair
Professor Vineet Bafna
Professor Greg Rouse
Professor Glenn Tesler

2022

The Dissertation of Chao Zhang is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2022

DEDICATION

To colleagues and comrades who believe in and advocate for open science and democratization of knowledge, and who strive for a better shared future.

TABLE OF CONTENTS

LIST OF SUPPLEMENTAL FILES

zhang-taper-supplementary-error-pictures.xlsx

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First of all, I would like to acknowledge all my dissertation committee members. Especially, I would like to express my deepest gratitude to Prof. Siavash Mirarab, the chair of my dissertation committee. As many of my fellow Ph.D. students say that the greatest blessing of a Ph.D. student is to have a nice and caring P.I. with great characteristics, fortunately, I am among the most blessed ones. Your great care, flexibility, mentorship, and support helped to shape me and enabled me to become who I am. I really enjoy and appreciate these five years of my Ph.D. experience in your lab.

I would like to thank Prof. Pavel Pevzner, the co-chair of my dissertation committee. You are the one who introduced me to the algorithm side of bioinformatics when I was an undergraduate student and firmed my belief in dedicating myself to this field. Without you, I would not be who I am today. I would also like to thank Prof. Vineet Bafna. Thank you for introducing me to the Bioinformatics and Systems Biology program, and also thank you for introducing me to Prof. Siavash Mirarab. I would like to thank Prof. Glenn Tesler for his high-quality proofreading and polishing this thesis. I really appreciate your advice and your help in Latex writing. I also want to thank Prof. Greg Rouse for his input from a biologist's point of view. I am very glad to have you are a member of my dissertation committee.

I would like to acknowledge and express my appreciation to my lab mates: Erfan Sayyari, Maryam Rabiee, Niema Moshiri, Uyen Mai, Metin Balaban, Eleonora Rachtman, Yueyu Jiang, and Puoya Tabaghi. Your countless conversations greatly enriched my graduate school experience, and I feel blessed working with you all. I specifically thank Erfan Sayyari and Maryam Rabiee for their support on ASTRAL-III. I was lucky being able to take classes from Prof. Daniel Kane, Prof. Li-Fan Lu, and Prof. Jiawang Nie. Their valuable teachings inspire me on various projects. Besides, I want to thank my dear friend and roommate Yuan Wang who takes care of me while I was busy writing this dissertation. I also thank Yong Gan and Zheng Wang who encourage me to make the right decision to join the University of California San Diego.

Finally, I would like to make a formal acknowledgment to the collaborators and co-authors of the papers that I published and used to write this dissertation:

Chapter 2, in full, is a reprint of the material as it appears in " Zhang, C., Rabiee, M., Sayyari, E. & Mirarab, S. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*. **19**, 15-30 (2018) ." The dissertation author was the primary investigator and first author of this paper.

Chapter 3, in full, has been submitted for publication of the material as it may appear in " Zhang, C. & Mirarab, S. Weighting by Gene Tree Uncertainty Improves Accuracy of Quartet-based Species Trees. *Molecular Biology And Evolution*. (2022) ." The dissertation author was the primary investigator and author of this paper.

Chapter 4, in full, is a reprint of the material as it appears in " Zhang, C., Scornavacca, C., Molloy, E. & Mirarab, S. ASTRAL-Pro: quartet-based species-tree inference despite paralogy. *Molecular Biology And Evolution*. **37**, 3292-3307 (2020) ." The dissertation author was the primary investigator and first author of this paper.

Chapter 5, in full, is currently being prepared for submission for publication of the material. " Zhang, C. & Mirarab, S. Scalable Coalescence-aware Ancestry Reconstruction from Aligned Genomes ." The dissertation author was the primary investigator and author of this material.

Chapter 6, in full, is a reprint of the material as it appears in " Zhang, C.†, Zhao, Y.†, Braun, E. & Mirarab, S. TAPER: Pinpointing errors in multiple sequence alignments despite varying rates of evolution. *Methods In Ecology And Evolution*. **12**, 2145-2158 (2021) ." The dissertation author was the co-primary investigator and co-first author of this paper.

Chapter 7, in full, has been submitted for publication of the material as it may appear in " Zhang, C., Bzikadze, A., Safonova, Y. & Mirarab, S. Scalable Models of Antibody Evolution and Benchmarking of Clonal Tree Reconstruction Methods. *Frontiers In Immunology*. (2022) ." The dissertation author was the primary investigator and author of this paper.

VITA

2017 Bachelor of Sciences in Bioengineering: Bioinformatics, University of California San Diego

2017 Bachelor of Sciences in Mathematics – Computer Science, University of California San Diego

2022 Doctor of Philosophy in Bioinformatics and System Biology, University of California San Diego

PUBLICATIONS

1. **Zhang, C.**[†], Zhao, Y.[†], Braun, E. & Mirarab, S. TAPER: Pinpointing errors in multiple sequence alignments despite varying rates of evolution. *Methods In Ecology And Evolution*. **12**, 2145-2158 (2021)

2. **Zhang, C.**, Scornavacca, C., Molloy, E. & Mirarab, S. ASTRAL-Pro: quartet-based species-tree inference despite paralogy. *Molecular Biology And Evolution*. **37**, 3292-3307 (2020)

3. Luebeck, J., Coruh, C., Dehkordi, S., Lange, J., Turner, K., Deshpande, V., Pai, D., **Zhang, C.**, Rajkumar, U., Law, J. & Others AmpliconReconstructor integrates NGS and optical mapping to resolve the complex structures of focal amplifications. *Nature Communications*. **11**, 1-14 (2020)

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

5. Carlin, D., Fong, S., Qin, Y., Jia, T., Huang, J., Bao, B., **Zhang, C.** & Ideker, T. A fast and flexible framework for network-assisted genomic association. *Iscience*. **16** pp. 155-161 (2019)

6. **Zhang, C.**, Rabiee, M., Sayyari, E. & Mirarab, S. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*. **19**, 15-30 (2018)

7. **Zhang, C.**, Sayyari, E. & Mirarab, S. ASTRAL-III: increased scalability and impacts of contracting low support branches. *RECOMB International Workshop On Comparative Genomics*. pp. 53-75 (2017)

8. Petras, D., Nothias, L., Quinn, R., Alexandrov, T., Bandeira, N., Bouslimani, A., Castro-Falcón, G., Chen, L., Dang, T., Floros, D. & **Others**. Mass spectrometry-based visualization of molecules associated with human habitats. *Analytical Chemistry*. **88**, 10775-10784 (2016)

In prep., in review, & under revision:

1. **Zhang, C.** & Mirarab, S. Scalable coalescence-aware ancestries reconstruction from aligned genomes. (in prep.)

2. **Zhang, C.** & Mirarab, S. Weighting by gene tree uncertainty improves accuracy of quartet-based species trees. *Molecular Biology And Evolution*. (2022) (in review)

3. **Zhang, C.**, Bzikadze, A., Safonova, Y. & Mirarab, S. Scalable models of antibody evolution and benchmarking of clonal tree reconstruction methods. *Frontiers in Immunology*. (2022) (in review)

4. **Zhang, C.** & Mirarab, S. ASTRAL-Pro 2: ultrafast species tree reconstruction from multi-copy gene family trees. *Bioinformatics*. (2020) (under revision)

ABSTRACT OF THE DISSERTATION

Scalable Algorithms for Inference and Simulation under Complex Phylogenetic Models

by

Chao Zhang

Doctor of Philosophy in Bioinformatics and System Biology

University of California San Diego, 2022

Professor Siavash Mirarab, Chair
Professor Pavel Pevzner, Co-Chair

Phylogenetics has been widely adopted across biology. Yet, a continuing difficulty in phylogenetics is modeling all biological processes that shape evolution while maintaining computational scalability. My dissertation focuses on several problems, in each case, developing scalable algorithms that advance biological realism. Much of the dissertation focuses on species tree reconstruction confronting discordance among evolutionary histories of genes (gene trees) for biological reasons such as incomplete lineage sorting.

Past work had already developed statistically consistent methods such as ASTRAL for species tree reconstruction given gene trees. However, these methods failed to account for

gene tree error (GTE). Contracting low-support branches was a potential solution, but ASTRAL was not efficient in handling polytomies. Here, I introduce ASTRAL-III, which drastically reduces the computational complexity in handling polytomies and improves robustness to GTE. Not satisfied with the need for a contraction threshold, I also introduce weighted ASTRAL, a method that down-weights error-prune gene tree branches and further improves the accuracy. Furthermore, I propose a method called ASTERISK to infer the species tree directly from multi-sequence alignments (MSAs), forgoing the need to infer error-prone gene trees. Having dealt with gene tree errors, I turn to errors in MSAs, which can impact phylogenetic analyses. I introduce TAPER, a novel two-dimensional outlier detection algorithm that looks for errors in small species-specific stretches of MSAs. TAPER can reduce GTE by finding much of the error while removing very little data.

Another shortcoming of ASTRAL was that it failed to model gene duplication and loss (GDL). I present a new algorithm called ASTRAL-Pro to accommodate datasets with high GDL rates, showing that ASTRAL-Pro is more accurate than alternatives.

Finally, I turn to selective pressure, a process that phylogenetics often fails to model. To benchmark the performance of tools under selection, I develop DIMSIM, an efficient simulator for sequence evolution under selection. I apply DIMSIM to the B-cell affinity maturation process that involves somatic hypermutations to B-Cell sequences followed by selective pressure. My study reveals that phylogenetic reconstruction tools fail to capture key features of clonal tree expansion if applied naively but can be easily rescued by contracting short branches.

Chapter 1

Introduction

Phylogenetic analysis has been widely adopted in different fields of science, including evolutionary biology (Leebens-Mack et al., 2019), cancer biology (Roerink et al., 2018), virology (Pekar et al., 2022), immunology (Kim et al., 2022), and even linguistics (Sagart et al., 2019). Despite its applicability in various fields, phylogenetic inferences remain challenging, especially with the ever-growing amount of data available today.

Species tree reconstruction is arguably one of the most important and computationally challenging task in phylogenetic analysis. One notable challenge is that the evolutionary history of one segment of the genome can differ from the evolutionary history of another segment and thus that of the species (Maddison, 1997; Degnan and Rosenberg, 2009). Such discrepancy has multiple causes, including incomplete lineage sorting (ILS), gene duplication and loss (GDL), horizontal gene transfer (HGT), and hybridization. As omitting the discrepancy may lead into a positively misleading result (Roch and Steel, 2015), discrepancy-aware species tree reconstruction methods are developed to specifically account for the discordance between the histories of genomic segments and that of species.

One popular approach to species tree reconstruction consists of two-steps: first reconstructing the history of each genomic segment, a gene tree, and then reconstructing the history of species, the species tree, by summarizing the information from the gene trees. Another approach directly infers species trees from MSA, circumventing the need for gene trees reconstruction. The former approach is computationally more efficient than the latter approach and is used by many phylogenomic projects (Chen et al., 2020; Zhang et al., 2021; Nissen et al., 2021; Li et al., 2020). ASTRAL (Mirarab et al., 2014) is a summary method that takes as input gene trees and outputs a species tree. ASTRAL maximizes the number of shared quartets – tree topologies induced by four species – between input gene trees and the output species tree. One remarkable result by Markin and Eulenstein (2020) states that ASTRAL is statistically consistent in presence of both ILS and GDL. Besides, Roch and Steel (2015) have also proven that ASTRAL is still consistent under a limited amount of random HGT. Despite being highly accurate under error-free gene trees, ASTRAL is not robust to error-prune input gene trees, both theoretically (Roch et al.,

2019) and practically (Degiorgio and Degnan, 2014; Huang and Lacey Knowles, 2016; Molloy and Warnow, 2018).

One way to improve the accuracy of ASTRAL is by contracting input gene tree branches with very low support, which will create polytomies (nodes with degree more than three) in input trees. As earlier versions of ASTRAL are not efficient in handling polytomies, in Chapter 2, I design a new version of ASTRAL, called ASTRAL-III, which drastically reduced the bottleneck in handling polytomies. I also incorporate various algorithmic techniques into the ASTRAL-III dynamic programming step to improve its running time.

As contracting low-support branches risks losing true biological signal, in Chapter 3, I take one step further and improve the robustness of ASTRAL to error-prune gene trees using a different approach: instead of contracting branches which effectively omits some quartets, I assign each quartet with a weight according to the support values and branch lengths relevant to the quartet. Such weighting scheme further improves the accuracy of ASTRAL under error-prune input gene trees.

Although ASTRAL has been proven consistent under GDL, it is not designed for it and suffers from dramatically increasing sample complexity under high GDL rates, which makes it unusable in practice with high GDL due to low accuracy. In Chapter 4, I first modify the objective function of ASTRAL to a measure of quartet similarity between single-copy and multi-copy trees that specifically accommodates datasets with high GDL rates. I then introduce a method called ASTRAL-Pro (ASTRAL for PaRalogs and Orthologs) to find the species tree that optimizes our quartet similarity measure using dynamic programming. ASTRAL-Pro distinguishes quartets reflecting orthologous relations and quartets reflecting paralogous relations. The former provides true biological signal and the latter does not. We call the former speciation-driven quartets (SDQs). SDQs can be equivalent to each others, and ASTRAL-Pro avoids double-counting equivalent SDQs by assigning SDQs to equivalence classes. One prominent result by ASTRAL-Pro comes from a reanalysis of a plant transcriptome dataset (Wickett et al., 2014) which leads into the One Thousand Plant Transcriptome (OneKP) project (Leebens-Mack et al., 2019). In

the original study, a species tree of 103 plant species is inferred from 424 single-copy genes using ASTRAL. The original study has also inferred 9,683 multi-copy gene trees with up to 2,395 leaves for 80 of the 103 species and three other genomes (a total of 83). However, due to a lack of suitable species tree methods, these gene trees were left unused. ASTRAL-Pro makes it possible to analyze all 9,683 multi-copy gene trees. ASTRAL-Pro successfully recovers at least one well-established biological relationship, which the original study fails to recover using single-copy genes; the species tree by ASTRAL-Pro is also more congruent to the species tree from OneKP consisting of 1,153 species inferred from 410 single-copy genes.

An alternative to the two-step approach is the direct approach. Example of methods using this approach are *BEAST (Heled and Drummond, 2010), SNAPP (Bryant et al., 2012), MrBayes (Ronquist et al., 2012), SVDQuartet (Chifman and Kubatko, 2014), QuCo (Rabiee and Mirarab, 2022). The major shortcoming of those methods is that they are not scalable enough to overhaul the exponentially growing data size, despite some more recent efforts in improving their scalabilities (Ogilvie et al., 2017; Vachaspati and Warnow, 2018; Zhang et al., 2020). For quartet-based site-based methods, one reason for lack of scalability is that they rely on first optimizing each quartet and then summarizing quartets to get the final tree. Even though they can sub-sample quartets, this process is intransigently slow. Alternatively, if each site partitions taxa into multiple groups and all the quartet topologies implied that by partition can be counted at the same time using simple combinatorics instead of iterating through all quartets, then the optimization can be very efficient using a trick similar to what ASTRAL uses. In Chapter 5, I introduce ASTERISK which has the following innovations: i) I introduce quartet site kernel, a new optimization objective computed based on DNA site patterns that is statistically consistent under MSC+GTR, even allowing for changes in rate across sites (with some limitations) and no assumption about species tree branch lengths (including no assumption of ultrametricity). ii) I design a scalable algorithm to optimize the total quartet site kernels for all quartets and all sites. iii) I propose various modifications to quartet site kernel for various applications. I test ASTERISK on a simulated dataset. It shows that ASTERISK dominates concatenation in all

conditions and with abundant genes, ASTERISK levels the performance of weighted ASTRAL.

Phylogenetic analysis relies heavily on accurate multi-sequence alignments (MSAs). Erroneous data can creep into sequence datasets for various reasons. As datasets keep getting larger, it has become difficult to check MSAs visually for errors, and thus, automatic error detection methods are needed more than ever before. In Chapter 6, I introduce a method called TAPER that uses a novel two-dimensional outlier detection algorithm to look for errors in small species-specific stretches of the multiple sequence alignments. Importantly, TAPER adjusts its null expectations per site and species, and in doing so, it attempts to distinguish the real heterogeneity (signal) from errors (noise). TAPER removes very little data yet finds much of the error, and thus, improves the accuracy of downstream phylogenetic analysis.

Currently, the golden standard for benchmarking phylogenetic inferences is through simulation, as true evolutionary histories are difficult to acquire. This is true even for microevolutions such as somatic hypermutations (SHMs) of B cell receptor (BCR) sequences. In Chapter 7, I design and implement DIMSIM, an efficient simulator which simulates the affinity maturation (AM) of B cells. DIMSIM simulates B cell lineages and BCR sequences at the same time, as B cell AM is not under neutral evolution. In fact, B cells during AM are under high selective pressure on their affinities to the antigens. DIMSIM can efficiently simulate sequence evolution under selective pressure based on affinity binding and enables simultaneous simulation of hundreds of B cell lineages at the same time. From benchmarking results using simulations under DIMSIM, I show that maximum likelihood phylogenetic reconstruction methods can fail to capture key features of clonal tree expansion if applied naively.

Bibliography

D. Bryant, R. Bouckaert, J. Felsenstein, N. A. Rosenberg, and A. Roychoudhury. Inferring Species Trees Directly from Biallelic Genetic Markers: Bypassing Gene Trees in a Full Coalescent Analysis. *Molecular Biology and Evolution*, 29(8):1917–1932, 8 2012. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSS086. URL https://academic.oup.com/mbe/article/29/8/1917/1045283.

H. Chen, Y. Zeng, Y. Yang, L. Huang, B. Tang, H. Zhang, F. Hao, W. Liu, Y. Li, Y. Liu, X. Zhang, R. Zhang, Y. Zhang, Y. Li, K. Wang, H. He, Z. Wang, G. Fan, H. Yang, A. Bao, Z. Shang, J. Chen, W. Wang, and Q. Qiu. Allele-aware chromosome-level genome assembly and efficient transgene-free genome editing for the autotetraploid cultivated alfalfa. *Nature Communications 2020 11:1*, 11(1):1–11, 5 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-16338-x. URL https://www.nature.com/articles/s41467-020-16338-x.

J. Chifman and L. Kubatko. Quartet Inference from SNP Data Under the Coalescent Model. *Bioinformatics*, 30(23):3317–3324, 12 2014. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTU530. URL https://academic.oup.com/bioinformatics/article/30/23/3317/206559.

M. Degiorgio and J. H. Degnan. Robustness to Divergence Time Underestimation When Inferring Species Trees from Estimated Gene Trees. *Systematic Biology*, 63(1):66–82, 1 2014. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYT059. URL https://academic.oup.com/sysbio/article/63/1/66/1688532.

J. H. Degnan and N. A. Rosenberg. Gene tree discordance, phylogenetic inference and the multispecies coalescent. *Trends in Ecology & Evolution*, 24(6):332–340, 6 2009. ISSN 0169-5347. doi: 10.1016/J.TREE.2009.01.009.

J. Heled and A. J. Drummond. Bayesian Inference of Species Trees from Multilocus Data. *Molecular Biology and Evolution*, 27(3):570–580, 3 2010. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSP274. URL https://academic.oup.com/mbe/article/27/3/570/999753.

H. Huang and L. Lacey Knowles. Unforeseen Consequences of Excluding Missing Data from Next-Generation Sequences: Simulation Study of RAD Sequences. *Systematic Biology*, 65(3):357–365, 5 2016. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYU046. URL https://academic.oup.com/sysbio/article/65/3/357/2468879.

W. Kim, J. Q. Zhou, S. C. Horvath, A. J. Schmitz, A. J. Sturtz, T. Lei, Z. Liu, E. Kalaidina, M. Thapa, W. B. Alsoussi, A. Haile, M. K. Klebert, T. Suessen, L. Parra-Rodriguez, P. A. Mudd, S. P. Whelan, W. D. Middleton, S. A. Teefey, I. Pusic, J. A. O'Halloran, R. M. Presti, J. S. Turner, and A. H. Ellebedy. Germinal centre-driven maturation of B cell response to mRNA vaccination. *Nature 2022 604:7904*, 604(7904):141–145, 2 2022. ISSN 1476-4687. doi: 10.1038/s41586-022-04527-1. URL https://www.nature.com/articles/s41586-022-04527-1.

J. H. Leebens-Mack, M. S. Barker, E. J. Carpenter, M. K. Deyholos, M. A. Gitzendanner, S. W. Graham, I. Grosse, Z. Li, M. Melkonian, S. Mirarab, M. Porsch, M. Quint, S. A. Rensing, D. E. Soltis, P. S. Soltis, D. W. Stevenson, K. K. Ullrich, N. J. Wickett, L. DeGironimo, P. P. Edger, I. E. Jordon-Thaden, S. Joya, T. Liu, B. Melkonian, N. W. Miles, L. Pokorny, C. Quigley, P. Thomas, J. C. Villarreal, M. M. Augustin, M. D. Barrett, R. S. Baucom, D. J. Beerling, R. M. Benstein, E. Biffin, S. F. Brockington, D. O. Burge, J. N. Burris, K. P. Burris, V. Burtet-Sarramegna, A. L. Caicedo, S. B. Cannon, Z. Çebi, Y. Chang, C. Chater, J. M. Cheeseman, T. Chen, N. D. Clarke, H. Clayton, S. Covshoff, B. J. Crandall-Stotler, H. Cross, C. W. dePamphilis, J. P. Der, R. Determann, R. C. Dickson, V. S. Di Stilio, S. Ellis, E. Fast, N. Feja, K. J. Field, D. A. Filatov, P. M. Finnegan, S. K. Floyd, B. Fogliani, N. García, G. Gâteblé, G. T. Godden, F. Q. Y. Goh, S. Greiner, A. Harkess, J. M. Heaney, K. E. Helliwell, K. Heyduk, J. M. Hibberd, R. G. Hodel, P. M. Hollingsworth, M. T. Johnson, R. Jost, B. Joyce, M. V. Kapralov, E. Kazamia, E. A. Kellogg, M. A. Koch, M. Von Konrat, K. Könyves, T. M. Kutchan, V. Lam, A. Larsson, A. R. Leitch, R. Lentz, F. W. Li, A. J. Lowe, M. Ludwig, P. S. Manos, E. Mavrodiev, M. K. McCormick, M. McKain, T. McLellan, J. R. McNeal, R. E. Miller, M. N. Nelson, Y. Peng, P. Ralph, D. Real, C. W. Riggins, M. Ruhsam, R. F. Sage, A. K. Sakai, M. Scascitella, E. E. Schilling, E. M. Schlösser, H. Sederoff, S. Servick, E. B. Sessa, A. J. Shaw, S. W. Shaw, E. M. Sigel, C. Skema, A. G. Smith, A. Smithson, C. N. Stewart, J. R. Stinchcombe, P. Szövényi, J. A. Tate, H. Tiebel, D. Trapnell, M. Villegente, C. N. Wang, S. G. Weller, M. Wenzel, S. Weststrand, J. H. Westwood, D. F. Whigham, S. Wu, A. S. Wulff, Y. Yang, D. Zhu, C. Zhuang, J. Zuidof, M. W. Chase, J. C. Pires, C. J. Rothfels, J. Yu, C. Chen, L. Chen, S. Cheng, J. Li, R. Li, X. Li, H. Lu, Y. Ou, X. Sun, X. Tan, J. Tang, Z. Tian, F. Wang, J. Wang, X. Wei, X. Xu, Z. Yan, F. Yang, X. Zhong, F. Zhou, Y. Zhu, Y. Zhang, S. Ayyampalayam, T. J. Barkman, N. p. Nguyen, N. Matasci, D. R. Nelson, E. Sayyari, E. K. Wafula, R. L. Walls, T. Warnow, H. An, N. Arrigo, A. E. Baniaga, S. Galuska, S. A. Jorgensen, T. I. Kidder, H. Kong, P. Lu-Irving, H. E. Marx, X. Qi, C. R. Reardon, B. L. Sutherland, G. P. Tiley, S. R. Welles, R. Yu, S. Zhan, L. Gramzow, G. Theißen, and G. K. S. Wong. One thousand plant transcriptomes and the phylogenomics of green plants. *Nature 2019 574:7780*, 574(7780):679–685, 10 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1693-2. URL https://www.nature.com/articles/s41586-019-1693-2.

F. W. Li, T. Nishiyama, M. Waller, E. Frangedakis, J. Keller, Z. Li, N. Fernandez-Pozo, M. S. Barker, T. Bennett, M. A. Blázquez, S. Cheng, A. C. Cuming, J. de Vries, S. de Vries, P. M. Delaux, I. S. Diop, C. J. Harrison, D. Hauser, J. Hernández-García, A. Kirbis, J. C. Meeks, I. Monte, S. K. Mutte, A. Neubauer, D. Quandt, T. Robison, M. Shimamura, S. A. Rensing, J. C. Villarreal, D. Weijers, S. Wicke, G. K. Wong, K. Sakakibara, and P. Szövényi. Anthoceros genomes illuminate the origin of land plants and the unique biology of hornworts. *Nature*

Plants 2020 6:3, 6(3):259–272, 3 2020. ISSN 2055-0278. doi: 10.1038/s41477-020-0618-2. URL https://www.nature.com/articles/s41477-020-0618-2.

W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 9 1997. ISSN 1063-5157. doi: 10.1093/SYSBIO/46.3.523. URL https://academic.oup.com/sysbio/article/46/3/523/1651369.

A. Markin and O. Eulenstein. Quartet-Based Inference Methods are Statistically Consistent Under the Unified Duplication-Loss-Coalescence Model. 4 2020. URL http://arxiv.org/abs/2004.04299.

S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548, 9 2014. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTU462. URL https://academic.oup.com/bioinformatics/article/30/17/i541/200803.

E. K. Molloy and T. Warnow. To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303, 3 2018. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYX077. URL https://academic.oup.com/sysbio/article/67/2/285/4159193.

J. N. Nissen, J. Johansen, R. L. Allesøe, C. K. Sønderby, J. J. A. Armenteros, C. H. Grønbech, L. J. Jensen, H. B. Nielsen, T. N. Petersen, O. Winther, and S. Rasmussen. Improved metagenome binning and assembly using deep variational autoencoders. *Nature Biotechnology 2021 39:5*, 39(5):555–560, 1 2021. ISSN 1546-1696. doi: 10.1038/s41587-020-00777-4. URL https://www.nature.com/articles/s41587-020-00777-4.

H. A. Ogilvie, R. R. Bouckaert, and A. J. Drummond. StarBEAST2 Brings Faster Species Tree Inference and Accurate Estimates of Substitution Rates. *Molecular Biology and Evolution*, 34(8):2101–2114, 8 2017. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSX126. URL https://academic.oup.com/mbe/article/34/8/2101/3738283.

J. E. Pekar, A. Magee, E. Parker, N. Moshiri, K. Izhikevich, J. L. Havens, K. Gangavarapu, L. M. Malpica Serrano, A. Crits-Christoph, N. L. Matteson, M. Zeller, J. I. Levy, J. C. Wang, S. Hughes, J. Lee, H. Park, M.-S. Park, K. Z. Y. Ching, R. T. P. Lin, M. N. Mat Isa, Y. M. Noor, T. I. Vasylyeva, R. F. Garry, E. C. Holmes, A. Rambaut, M. A. Suchard, K. G. Andersen, M. Worobey, and J. O. Wertheim. The molecular epidemiology of multiple zoonotic origins of SARS-CoV-2. *Science (New York, N.Y.)*, page eabp8337, 7 2022. ISSN 1095-9203. doi: 10.1126/SCIENCE.ABP8337/SUPPL{_}FILE/SCIENCE.ABP8337{_}SM{_}DATA{_}S1{_}TO{_}S3.ZIP. URL http://www.ncbi.nlm.nih.gov/pubmed/35881005.

M. Rabiee and S. Mirarab. QuCo: quartet-based co-estimation of species trees and gene trees. *Bioinformatics*, 38(Supplement_1):i413–i421, 6 2022. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTAC265. URL https://academic.oup.com/bioinformatics/article/38/

Supplement_1/i413/6617531.

S. Roch and M. Steel. Likelihood-based tree reconstruction on a concatenation of aligned sequence data sets can be statistically inconsistent. *Theoretical population biology*, 100C: 56–62, 3 2015. ISSN 1096-0325. doi: 10.1016/J.TPB.2014.12.005. URL https://pubmed.ncbi.nlm.nih.gov/25545843/.

S. Roch, M. Nute, and T. Warnow. Long-Branch Attraction in Species Tree Estimation: Inconsistency of Partitioned Likelihood and Topology-Based Summary Methods. *Systematic Biology*, 68(2):281–297, 3 2019. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYY061. URL https://academic.oup.com/sysbio/article/68/2/281/5104882.

S. F. Roerink, N. Sasaki, H. Lee-Six, M. D. Young, L. B. Alexandrov, S. Behjati, T. J. Mitchell, S. Grossmann, H. Lightfoot, D. A. Egan, A. Pronk, N. Smakman, J. Van Gorp, E. Anderson, S. J. Gamble, C. Alder, M. Van De Wetering, P. J. Campbell, M. R. Stratton, and H. Clevers. Intra-tumour diversification in colorectal cancer at the single-cell level. *Nature 2018 556:7702*, 556(7702):457–462, 4 2018. ISSN 1476-4687. doi: 10.1038/s41586-018-0024-3. URL https://www.nature.com/articles/s41586-018-0024-3.

F. Ronquist, M. Teslenko, P. Van Der Mark, D. L. Ayres, A. Darling, S. Höhna, B. Larget, L. Liu, M. A. Suchard, and J. P. Huelsenbeck. MrBayes 3.2: Efficient Bayesian Phylogenetic Inference and Model Choice Across a Large Model Space. *Systematic Biology*, 61(3):539–542, 5 2012. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYS029. URL https://academic.oup.com/sysbio/article/61/3/539/1674894.

L. Sagart, G. Jacques, Y. Lai, R. J. Ryder, V. Thouzeau, S. J. Greenhill, and J. M. List. Dated language phylogenies shed light on the ancestry of Sino-Tibetan. *Proceedings of the National Academy of Sciences of the United States of America*, 116(21):10317–10322, 5 2019. ISSN 10916490. doi: 10.1073/PNAS.1817972116/SUPPL{_}FILE/PNAS.1817972116.SAPP.PDF. URL https://www.pnas.org/doi/abs/10.1073/pnas.1817972116.

P. Vachaspati and T. Warnow. SVDquest: Improving SVDquartets species tree estimation using exact optimization within a constrained search space. *Molecular Phylogenetics and Evolution*, 124:122–136, 7 2018. ISSN 1055-7903. doi: 10.1016/J.YMPEV.2018.03.006.

N. J. Wickett, S. Mirarab, N. Nguyen, T. Warnow, E. Carpenter, N. Matasci, S. Ayyampalayam, M. S. Barker, J. G. Burleigh, M. A. Gitzendanner, B. R. Ruhfel, E. Wafula, J. P. Der, S. W. Graham, S. Mathews, M. Melkonian, D. E. Soltis, P. S. Soltis, N. W. Miles, C. J. Rothfels, L. Pokorny, A. J. Shaw, L. De Gironimo, D. W. Stevenson, B. Surek, J. C. Villarreal, B. Roure, H. Philippe, C. W. De Pamphilis, T. Chen, M. K. Deyholos, R. S. Baucom, T. M. Kutchan, M. M. Augustin, J. Wang, Y. Zhang, Z. Tian, Z. Yan, X. Wu, X. Sun, G. K. S. Wong, and J. Leebens-Mack. Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences of the United States of America*, 111 (45):E4859–E4868, 11 2014. ISSN 10916490. doi: 10.1073/PNAS.1323926111/SUPPL{_}

FILE/PNAS.1323926111.ST02.DOCX. URL https://www.pnas.org/doi/abs/10.1073/pnas.1323926111.

C. Zhang, J. P. Huelsenbeck, and F. Ronquist. Using Parsimony-Guided Tree Proposals to Accelerate Convergence in Bayesian Phylogenetic Inference. *Systematic Biology*, 69(5): 1016–1032, 9 2020. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYAA002. URL https://academic.oup.com/sysbio/article/69/5/1016/5716338.

X. Zhang, S. Chen, L. Shi, D. Gong, S. Zhang, Q. Zhao, D. Zhan, L. Vasseur, Y. Wang, J. Yu, Z. Liao, X. Xu, R. Qi, W. Wang, Y. Ma, P. Wang, N. Ye, D. Ma, Y. Shi, H. Wang, X. Ma, X. Kong, J. Lin, L. Wei, Y. Ma, R. Li, G. Hu, H. He, L. Zhang, R. Ming, G. Wang, H. Tang, and M. You. Haplotype-resolved genome assembly provides insights into evolutionary history of the tea plant Camellia sinensis. *Nature Genetics 2021 53:8*, 53(8):1250–1259, 7 2021. ISSN 1546-1718. doi: 10.1038/s41588-021-00895-y. URL https://www.nature.com/articles/s41588-021-00895-y.

Chapter 2

ASTRAL-III: Polynomial Time Species Tree Reconstruction from Partially Resolved Gene Trees

Background

Evolutionary histories can be discordant across the genome, and such discordances need to be considered in reconstructing the species phylogeny. ASTRAL is one of the leading methods for inferring species trees from gene trees while accounting for gene tree discordance. ASTRAL uses dynamic programming to search for the tree that shares the maximum number of quartet topologies with input gene trees, restricting itself to a predefined set of bipartitions.

Results

We introduce ASTRAL-III, which substantially improves the running time of ASTRAL-II and guarantees polynomial running time as a function of both the number of species (n) and the number of genes (k). ASTRAL-III limits the bipartition constraint set (X) to grow at most linearly with n and k. Moreover, it handles polytomies (nodes with degree more than three) more efficiently than ASTRAL-II, exploits similarities between gene trees better, and uses several techniques to avoid searching parts of the search space that are mathematically guaranteed not to include the optimal tree. The asymptotic running time of ASTRAL-III in the presence of polytomies is $O((nk)^{1.726}D)$ where $D = O(nk)$ is the sum of degrees of all *unique* nodes in input trees. The running time improvements enable us to test whether contracting low support branches in gene trees improves the accuracy by reducing noise. In extensive simulations, we show that removing branches with *very* low support (e.g., below 10%) improves accuracy while overly aggressive filtering is harmful. We observe on a biological avian phylogenomic dataset of 14K genes that contracting low support branches greatly improve results.

Conclusions

ASTRAL-III is a faster version of the ASTRAL method for phylogenetic reconstruction and can scale up to 10,000 species. With ASTRAL-III, low support branches can be removed, resulting in improved accuracy.

2.1 Background

The potential for genome-wide discordance of evolutionary histories (Maddison, 1997; Degnan and Rosenberg, 2009) has motivated the development of several approaches for species phylogeny reconstruction. Reconstructing a collection of gene trees, each inferred from a different part of the genome, and then summarizing them to get a species tree is one such approach and is used by many phylogenomic projects (e.g., Song et al., 2012; Wickett et al., 2014; Jarvis et al., 2014; Laumer et al., 2015; Tarver et al., 2016) (while "gene trees" need not be inferred from functional genes, following the conventions of the field, we will refer to them as such). This two-step approach stands in contrast to concatenation (Rokas et al., 2003), where all the data are combined and analyzed in a single analysis. The two-step approach aims to account for discordance between gene trees and the species tree (but its effectiveness is debated Springer and Gatesy, 2016; Meiklejohn et al., 2016; Edwards et al., 2016; Shen et al., 2017) and is more computationally efficient than statistical co-estimation of gene trees and the species tree (Heled and Drummond, 2010). Incomplete lineage sorting (ILS) is a ubiquitous (Edwards, 2009) cause of discordance. ILS is typically modeled by the multi-species coalescent model (MSCM) (Pamilo and Nei, 1988; Rannala and Yang, 2003), where branches of the species tree represent populations, and lineages are allowed to coalesce inside each branch; lineages that fail to coalesce at the root of each branch are moved to the parent branch.

Many "summary" methods have been developed to infer a species tree from a collection of input trees. Examples include MP-EST (Liu et al., 2010), NJst (Liu and Yu, 2011), ASTRID (Vachaspati and Warnow, 2015), DISTIQUE (Sayyari and Mirarab, 2016a), ASTRAL (Mirarab et al., 2014b; Mirarab and Warnow, 2015) and STAR (Liu et al., 2009), which only use gene tree topologies, and GLASS (Mossel and Roch, 2010) and STEAC (Liu et al., 2009), which also use branch lengths. These methods are all proved statistically consistent under the MSCM, given error-free input gene trees; when input trees are inferred from sequence data, statistical consistency is not guaranteed (Roch and Warnow, 2015). Most methods take rooted

gene trees as input, but some methods (e.g., ASTRAL, NJst/ASTRID and DISTIQUE) use unrooted input trees. ASTRAL-II (Mirarab and Warnow, 2015) is currently one of the commonly used summary methods.

In this paper, we introduce an improved version of ASTRAL called ASTRAL-III. As we will show, compared to ASTRAL-II, the new version has better running time without sacrificing accuracy. The improvements in the running time are both theoretical (reducing the asymptotic running time so that it is guaranteed to grow polynomially with the dataset size) and empirical.

2.2 Methods

2.2.1 Notations and definitions

Let the set of n species be called L and let G be the set of k input gene trees on L. Let $[d]$ represent the set $\{1,2\ldots,d\}$. We use $Q(t)$ to denote the set of quartet trees induced by a tree t. Any subset of L is called a cluster. We define a partition as a set of clusters that are pairwise mutually exclusive; note that we abuse the term "partition" here because the union of all clusters in a partition need not give the complete set. Each node in an unrooted tree defines a partition. A bipartition (tripartition) is a partition with cardinality two (three); a partition with cardinality at least four corresponds to a multifurcation (also referred to as a polytomy). Let X (the constraint bipartition set) be a set of clusters such that for each $A \in X$, we also have $L-A \in X$. We use Y to represent the set of all tripartitions that can be built from X:

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

We use E to denote the set of all unique partitions and their frequency in G. Thus,

$$E = \{(M, \sum_{g \in G} |N(g) \cap \{M\}|) : M \in N(g), g \in G\} \tag{2.1}$$

where $N(g)$ is the set of all partitions representing all internal nodes in the tree g. We also define D as the sum of the cardinalities of unique partitions in gene trees:

$$D = \sum_{(M,c)\in E} |M| \,. \tag{2.2}$$

2.2.2 ASTRAL (old versions)

The problem addressed by ASTRAL is to find the tree that shares the maximum number of induced quartet topologies with the collection of input gene trees:

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

Lafond and Scornavacca recently proved this problem is NP-hard (Lafond and Scornavacca, 2016).

ASTRAL-I and ASTRAL-II algorithms

ASTRAL solves a constrained version of the problem where a set of clusters X restricts bipartitions that the output species tree may include (recall $\forall A \in X : L-A \in X$). Note that setting X to the powerset solves the unconstrained problem. Based on the fact that an unrooted quartet species tree always matches the most likely unrooted quartet gene tree (Allman et al., 2011), ASTRAL is proved statistically consistent (Mirarab et al., 2014b).

ASTRAL uses dynamic programming to solve the problem using the recursive relation:

$$V(A) = \begin{cases} 0 & |A| = 1 \\ \max_{A'\subset A,(A'|A-A'|L-A)\in Y} V(A,A') & |A| > 1 \end{cases}$$

$$V(A,A') = V(A') + V(A-A') + w(A'|A-A'|L-A)$$

where the function $w(T)$ scores each tripartition $T = (A|B|C)$ against each node in each input gene tree. Let partition $M = (M_1|M_2|...|M_d)$ represent an internal node of degree d in a gene

tree. The overall contribution of T to the score of any species tree that includes T is:

$$w(T) = \sum_{g \in G} \sum_{M \in N(g)} \frac{1}{2} QI(T, M) \tag{2.3}$$

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

$$QI(T, M) = \sum_{i \in [d]} \sum_{j \in [d]-\{i\}} \sum_{k \in [d]-\{i,j\}} \frac{a_i + b_j + c_k - 3}{2} a_i b_j c_k \,. \tag{2.4}$$

As previously proved (Mirarab et al., 2014b), $QI(T, M)$ computes twice the number of quartet trees that are going to be shared between any two trees if one includes only T and the other includes only M. ASTRAL-II requires $\Theta(d^3)$ time for computing $QI(.)$, making its overall running time $O(n^3 k|Y|)$ with polytomies of unbounded degrees or $O(nk|Y|)$ in the absence of polytomies.

Noting trivially that $|Y| < |X|^2$, the previously published running time analysis of ASTRAL-II was $O(nk|X|^2)$ for binary gene trees and $O(n^3 k|X|^2)$ for trees with polytomies. A recent result by Kane and Tao (Kane and Tao, 2017) (motivated by the analysis of ASTRAL) proved that $|Y| \leq |X|^{3/log_3(27/4)}$. This result immediately gives us a better upper bound on the running time.

Corollary 2.1. *ASTRAL-II runs in $O(nk|X|^{1.726})$ and $O(n^3 k|X|^{1.726})$, respectively, with and without polytomies in gene trees.*

In ASTRAL-I, X is the set of all bipartitions observed in input gene trees. While sufficient for statistical consistency and often for accuracy, under some conditions, this set X is too restrictive. To address this shortcoming, ASTRAL-II (Mirarab and Warnow, 2015) uses several heuristics (see Appendices 2.A) and further expands the set X. Even though ASTRAL-II tries to limit $|X|$, it does not provide any guarantees as to how it grows with n and k. In the worst case, $|X|$ can grow exponentially, and thus, ASTRAL-II does not guarantee polynomial running time. The relatively high accuracy of ASTRAL-II has been shown in several simulations (Mirarab

and Warnow, 2015; Sayyari and Mirarab, 2016a; Shekhar et al., 2017; Davidson et al., 2015) and it has been adopted by the community as one of the main methods used in phylogenomics. ASTRAL has the ability to compute branch lengths in coalescent units (Degnan and Rosenberg, 2009) and a measure of branch support called local posterior probability (Sayyari and Mirarab, 2016b).

Limitations of ASTRAL-II

Several shortcomings of ASTRAL-II in terms of running time are addressed here (ASTRAL-III); our improvements, in turn, enable new types of analyses.

While ASTRAL-II can analyze datasets with a thousand species and gene trees in reasonable time, it does not easily scale to many tens of thousands of input trees. Datasets with more than ten thousand loci are already available (e.g., Jarvis et al., 2014) and as more genomes are sequenced, more are destined to become available in the near future. Moreover, being able to handle large k (i.e., numbers of input trees) enables using multiple trees per locus (e.g., a Bayesian sample) as input to ASTRAL. The limited scalability of ASTRAL with k has two reasons. First, the set X is not bounded in ASTRAL-II and can grow to become the power set. Thus, in ASTRAL-II, $|X|$ *can* theoretically grow exponentially with n. We fix this in ASTRAL-III by modifying heuristics that form the set X so that they all guarantee that $|X| = O(nk)$. The second cause of the slowdown is that computing each $w(T)$ for a tripartition T requires $\Theta(nk)$. This computation does not exploit similarities between gene trees, a shortcoming that we fix in ASTRAL-III.

Beyond large k, ASTRAL-II, which scales as $O(n^3k|X|^{1.726})$ in the presence of polytomies, can quickly become prohibitively slow for input trees with large polytomies. ASTRAL-III uses a mathematical trick to enable scoring of gene tree polytomies in time similar to binary nodes. The ability to handle large polytomies in input gene trees is important for two reasons. Some of the conditions that are conducive to ILS, namely shallow trees, are also likely to produce identical gene sequence data for several species. The gene tree should leave the relationship

between identical sequences unresolved (FastTree (Price et al., 2010) automatically does it and RAxML, which outputs an arbitrary resolution, warns the user about the input). Moreover, all summary methods, including ASTRAL, are sensitive to gene tree estimation error (Mirarab and Warnow, 2015; Mirarab et al., 2014a; Bayzid et al., 2015; Mirarab et al., 2016; Patel, 2013; Gatesy and Springer, 2014). One way of dealing with gene tree error, previously studied in the context of minimizing deep coalescence (Yu et al., 2011), is to contract low support branches in gene trees and use these unresolved trees as input to the summary method. While earlier studies found no evidence that this approach helps ASTRAL-II when the support is judged by SH-like FastTree support (Mirarab and Warnow, 2015), no study has tested this approach with bootstrap support values. We will for the first time evaluate the effectiveness of contracting branches with low *bootstrap* support and show that conservative filtering of *very* low support branches does, in fact, help the accuracy.

2.2.3 ASTRAL-III

ASTRAL-III has six new features:

1. Heuristics for building the set X are modified to ensure $|X| = O(nk)$. This step alone (without subsequent improvements) guarantees the overall running time is $O((nk)^{2.726})$ for binary gene trees and $O(n^{4.726}k^{2.726})$ for polytomies.

2. Heuristics for building the set X are modified to enlarge X for gene trees with polytomies without breaking $|X| = O(nk)$ guarantees. This can impact the accuracy and empirical running times but not the asymptotic running time.

3. A new way of computing $w(q)$ is introduced to reduce the running time for scoring a gene tree to $O(n)$, instead of $O(n^3)$, in the presence of polytomies. This step, combined with the previous steps, reduces the total running time to $O((nk)^{2.726})$ irrespective of whether gene trees have polytomies.

4. A polytree (a graph with at most one undirected path between any two vertices) is used to represent gene trees, and this enables an algorithm that reduces the overall running time from $O((nk)^{2.726})$ to $O(D(nk)^{1.726})$, which is the final theoretical analysis of ASTRAL-III running time.

5. A new algorithm, similar to A* (Hart et al., 1968), is used to compute an upper-bound on the best possible resolution of a clade; we need not expand a clade recursively when its upper-bound is below the best available score. The worst case asymptotic running time does not change due to this feature.

6. A two-stage heuristic mechanism is designed to further tighten the upper bounds used in pruning unnecessary parts of the search space. The worst case asymptotic running time is not impacted.

A beta version of ASTRAL-III was recently described (Zhang et al., 2017) and that version included features 3–5 but not the others. We next describe each improvement.

New search space: $|X| = O(nk)$

ASTRAL-II uses several heuristic methods to build X (see the original paper (Mirarab and Warnow, 2015) for details). The main method involves computing several extended majority consensus trees from gene trees and then resolving polytomies in these consensus trees using three techniques (mentioned below). These steps are repeated for 10 rounds or more until very few (less than a constant threshold) of the bipartitions observed are new to X. Because the number of rounds is not constant or a function of n and k, we cannot bound how X grows with n and k for ASTRAL-II. In ASTRAL-III, we limit the number of rounds by a constant value (default set to 110). This enables us to provide guarantees of a polynomial growth of $|X|$ with n and k.

To get to $X = O(nk)$, we need further changes. As mentioned, three techniques are used to resolve each polytomy of degree d in extended majority consensus trees. The first technique

uses a precomputed distance matrix to build a UPGMA tree starting from sides of the polytomy and adds the new bipartitions from the UPGMA tree to X. This can only add $O(d) = O(n)$ resolutions. The second technique computes a greedy consensus of gene trees subsampled to randomly selected taxa (one from each side of the polytomy) and adds bipartitions from the greedy consensus to X. This also can only add $O(d)$ new bipartitions. The third resolution samples a taxon from each side of the polytomy, computes d caterpillar trees, each constructed based on decreasing similarity to each sampled taxon, and adds the bipartitions from all these caterpillar trees to X. This quadratic resolution step can add $O(d^2) = O(n^2)$ bipartitions to X. To have $|X| = O(n)$, we need to change this step. Let $d_1 \dots d_r$ be the list of all polytomy degrees in an extended majority consensus tree in the ascending ordered. We find the smallest threshold q such that $\sum_{i=1}^{q} d_i^2 \leq cn$ for some constant c (default $= 25$). In ASTRAL-III, we apply the quadratic resolution technique only for polytomies $d_1 \dots d_q$; this, by definition, ensures no more than $O(d) = O(n)$ bipartitions are added in each round.

New search space: handling gene tree polytomies

We also change the way ASTRAL builds X in the presence of gene tree polytomies. Our goal is to increase $|X|$ compared to ASTRAL-II for multifurcating gene trees. However, $|X|$ is enlarged at most by a constant factor and we retain $|X| = O(nk)$.

If a gene tree includes polytomies, ASTRAL-II adds bipartitions implied by resolutions of that polytomy to the set X using a guide tree g. To build g, a greedy consensus of all gene trees is computed and is further refined to become binary by applying UPGMA to each polytomy of the greedy tree using a precomputed similarity matrix (see the original paper (Mirarab and Warnow, 2015) for details). To resolve a gene tree polytomy of degree d, ASTRAL-II first randomly samples d taxa, each from one side of the polytomy. Let S be the sampled taxa. All bipartitions from the tree g restricted to the set S of leaves are added to X. While in ASTRAL-II this process is done only once, in ASTRAL-III, we repeat the process three times with different random samples S. This increases $|X|$ but at most by a constant factor. The enlarged X can lead

to improved accuracy when input trees include many polytomies.

The second change in ASTRAL-III is that we now use a UPGMA tree inferred based on the similarity matrix as the guide tree. We observed that the UPGMA tree summarizes the input gene trees more accurately than the greedy tree (see Table S2.1). Finally, in ASTRAL-III, we improve the definition of the similarity matrix in the presence of gene tree polytomies. Unlike in ASTRAL-II, we ensure that unresolved quartet trees induced by gene trees do not increase the similarity between pairs of taxa included in those quartets. Note that the similarity matrix, which is based on quartets, should not be confused with the quartet score optimized by ASTRAL.

Efficient handling of Polytomies

Recall that ASTRAL-II uses Equation 2.4 to score a tripartition against a polytomy of size d in $\Theta(d^3)$ time. Our next Lemma shows that this can be improved.

Lemma 2.1. *Let $QI(T,M)$ be twice the number of quartet tree topologies shared between an unrooted tree that only includes a node corresponding to the tripartition $T = (A|B|C)$ and another tree that includes only a node corresponding to a partition $M = (M_1|M_2|...|M_d)$ of degree d; then, $QI(T,M)$ can be computed in time $\Theta(d)$.*

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

$$S_a = \sum_{i \in [d]} a_i \quad \text{and} \quad S_{a,b} = \sum_{i \in [d]} a_i b_i \tag{2.5}$$

where $a_i = |A \cap M_i|$ and $b_i = |B \cap M_i|$; we can also compute S_b, S_c, $S_{a,c}$ and $S_{b,c}$ (similarly defined). Equation 2.4 computes twice the number of quartet tree topologies shared between an unrooted tree with internal node T and another tree with one internal node M (Mirarab and

Warnow, 2015). Equation 2.4 can be rewritten as:

$$\begin{aligned} QI((A|B|C),M) = &\sum_{i\in[d]} \binom{a_i}{2}((S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i) \\ &+ \sum_{i\in[d]} \binom{b_i}{2}((S_a - a_i)(S_c - c_i) - S_{a,c} + a_i c_i) \\ &+ \sum_{i\in[d]} \binom{c_i}{2}((S_a - a_i)(S_b - b_i) - S_{a,b} + a_i b_i) \end{aligned} \tag{2.6}$$

(the derivation is given in the Appendices 2.B). Computing Equation 2.6 instead of Equation 2.4 clearly reduces the running time to $\Theta(d)$ instead of $\Theta(d^3)$. □

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

Corollary 2.2. *Scoring a tripartition (i.e., computing w) can be done in $O(nk)$.*

Gene trees as a Polytree

ASTRAL-II scores each dynamic programming tripartition against each individual node of each gene tree. However, nodes that are repeated in several gene trees need not be recomputed. Recalling the definitions of E and D (Eqs. 2.1 and 2.2),

Lemma 2.2. *The score of a tripartition $T = (A|B|C)$ against all gene trees (i.e., the $w(T)$ score) can be computed in $\Theta(D)$.*

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

$$w(T) = \sum_{(M,c)\in E} c \times QI(T,M)\ . \tag{2.7}$$

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

$$|W \cap A| = \sum_{Z \in \text{children}(W)} |Z \cap A| \tag{2.8}$$

(ditto for $|W \cap B|$ and $|W \cap C|$). All such intersection values can be computed in a post-order traversal of a polytree. In this polytree, all unique clusters in the gene trees are represented as vertices and parent-child relations are represented as edges; note that when a cluster has different resolutions in two different input trees, we arbitrarily choose one set of children in building the polytree. The polytree will include no more than D edges; thus, the time complexity of traversing this polytree (to compute Eq. 2.8) for all nodes is $O(D)$. Once all intersections are computed, in the second step, we simply compute the sum in Eq. 2.7. Each $QI(.)$ computation requires $\Theta(d)$ time by Lemma 2.1. Recalling that $D = \sum_{(M,c) \in E} |M|$, it is clear that computing Equation 2.7 requires $\Theta(D)$ time. Therefore, both steps can be performed in $\Theta(D)$ time. □

Theorem 2.1. *The running time of ASTRAL-III grows as $O(D(nk)^{1.726})$ for both binary and multifurcating gene trees.*

Proof. By results of Kane and Tao (Kane and Tao, 2017), the size of the set Y is $O(|X|^{1.726})$, and for each element in Y, by Lemma 2.2, we require $O(D)$ to compute the weights, regardless of the presence or absence of polytomies. The running time of ASTRAL is dominated by computing the weights (Mirarab and Warnow, 2015). Thus, the overall running time is $O(D|Y|) = O(D|X|^{1.726})$. Moreover, ASTRAL-III forces $|X|$ to grow as $O(nk)$, giving the overall running time of $O(D(nk)^{1.726})$ □

Trimming of the dynamic programming

We now introduce an upper-bound (proved in Appendices 2.B):

$$V(A) \leq U(A) = \frac{w(A|A|L)}{2} - \frac{w(A|A|A)}{3} .$$

Let

$$U(A,A'') = U(A'') + U(A - A'') + w(A''|A - A''|L - A).$$

Since $V(A) \leq U(A)$, for any $(A'|A - A'|L - A') \in Y$ and $(A''|A - A''|L - A'') \in Y$, we no longer need to recursively compute $V(A'')$ and $V(A - A'')$ when $U(A,A'') \leq V(A,A')$. When computing $V(A)$ by maximizing the score over all resolutions of A, imagine that we first encounter A' and then A''. We avoid expanding A'' when $U(A,A'') \leq V(A,A')$. This approach clearly makes the order of processing of the resolutions important. To heuristically improve the efficiency of this approach, we order all $(A'|A - A'|L - A) \in Y$ according to $U(A,A')$. Note that computing $U(A)$ does not require recursive computations down the dynamic programming DAG. Thus, the use of this upper-bound results in the trimming of the search space. However, as far as we can tell, this trimming does not improve the theoretical running time.

Two-staged α-trimming

In order to further trim the search space, another upper-bound of $V(A)$ is calculated. For a given $\alpha \geq 1$ and any ordering of the set $\{A' : (A'|A - A'|L - A) \in Y\}$ denoted by $A_1 \ldots A_r$, we define $V_\alpha(A)$ as follows.

$$V_\alpha^i(A) = \left\{ \begin{array}{ll} 0, & i = 0 \\ V_\alpha(A,A_i), & V_\alpha(A,A_i) > \alpha V_\alpha^{i-1}(A) \\ V_\alpha^{i-1}(A), & \text{otherwise} \end{array} \right\} \text{ for } 0 \leq i \leq r$$

$$V_\alpha(A,A_i) = V_\alpha(A_i) + V_\alpha(A - A_i) + w(A_i|A - A_i|L - A) \text{ and } V_\alpha(A) = V_\alpha^r(A)$$

We can compute $V_\alpha(A)$ using an algorithm equivalent to our dynamic programming for computing $V(A)$, except that, as resolutions of a clade A are being tested, a new one is accepted only if it improves upon the previous best resolution by a factor of α (thus, $\alpha = 1$ simply reproduces our existing dynamic programming). When computing $V_\alpha(A)$, for any $i < j$, if $\alpha\big(V_\alpha(A,A_i)\big) \geq U(A,A_j)$, then it is guaranteed that $\alpha\big(V_\alpha(A,A_i)\big) \geq V_\alpha(A,A_j)$, and thus we no longer need to recursively compute $V_\alpha(A_j)$ and $V_\alpha(A-A_j)$. After all $V_\alpha(A)$ values are computed for some choice of α, we turn to computing $V(A)$.

Observe that $V_\alpha(A) \leq V(A) \leq \alpha V_\alpha(A)$. Let $U_\alpha(A,A_j)$ be defined as

$$\min\big(U(A_j), \alpha V_\alpha(A_j)\big) + \min\big(U(A-A_j), \alpha V_\alpha(A-A_j)\big) + w(A_j|A-A_j|L-A)$$

and note that

$$U_\alpha(A,A_j) \geq V(A,A_j) = V(A_j) + V(A-A_j) + w(A_j|A-A_j|L-A)\ .$$

Thus, during the dynamic programming, for $i < j$, if $V(A,A_i) > U_\alpha(A,A_j)$, then it is guaranteed that $V(A,A_i) \geq V(A,A_j)$, and thus we no longer need to recursively compute $V(A_j)$ and $V(A-A_j)$. The hope is that the U_α function will give us tighter upper bounds compared to the U function previously defined. Whether this happens or not depends on the choice of α, the order of visiting clusters, and the particularities of a dataset.

While any choice of $\alpha \geq 1$ would guarantee the correct solution to the dynamic programming, we have empirically selected a heuristic to choose α. We set $\alpha = \frac{U(L)}{g(L)}$, where

$$g(A) = g(A_i) + g(A-A_i) + w(A_i|A-A_i|L-A)$$

where

$$i = \arg\max_j U(A_j) + U(A-A_j) + w(A_j|A-A_j|L-A)$$

and $g(A) = 0$ for $|A| = 1$. Just as before, we order the clusters in the decreasing order of $U(A, A_i)$.

2.3 Results

2.3.1 Experimental setup

We study three research questions:

RQ1: Can *contracting low support branches* improve the accuracy of ASTRAL?

RQ2: How do the *running time and search space* compare between ASTRAL-II and ASTRAL-III?

RQ3: How *accurate* is ASTRAL-III, which guarantees polynomial size search space, compared to ASTRAL-II?

Datasets

Avian biological dataset:

Neoavian relationships show extremely high levels of gene tree discord, perhaps because their ancestors experienced a rapid radiation (Jarvis et al., 2014). A dataset of 48 genomes representing all avian orders has been used to partially resolve this rapid radiation (Jarvis et al., 2014). A set of 14,446 loci (including exons, introns, and UCEs) was used to produce two reference species trees using concatenation and using a coalescent-based method (Jarvis et al., 2014; Mirarab et al., 2014a). We use the set of all unbinned gene trees and compare ASTRAL-III with and without contraction against both reference trees.

Simulated avian-like dataset:

This simulated dataset, previously used to emulate the biological avian dataset (Mirarab et al., 2014a), has three model conditions with respect to the simulated levels of ILS: 1X is the default, whereas 0.5X divides each branch length in half (increasing ILS) and 2X multiplies them by 2 (reducing ILS). Average RF distances between true species tree and true gene trees are 0.35, 0.47, and 0.59, respectively for 2X, 1X, and 0.5X. To further test the impact of gene

tree estimation error, sequence lengths were also varied to create four model conditions: 250bp alignments (0.67 RF distance between true gene trees and estimated gene trees), 500bp (0.54 RF), 1000bp (0.39 RF) and 1500bp (0.30 RF), all based on the 1X ILS. We use 1000 gene trees, and 20 replicates per condition. Gene trees are estimated using RAxML (Stamatakis, 2014) with 200 replicates of bootstrapping.

SimPhy-homogeneous (S100):

We simulated a new 101-taxon dataset using SimPhy (Mallo et al., 2016) with 50 replicates, each with a different species tree. The species trees are simulated under the birth-only process with birth rate 10^{-7}, fixed haploid N_e of 400K, and the number of generations sampled from a log-normal distribution with mean 2.5M. For each replicate, 1000 true gene trees are simulated under the MSCM (exact commands shown in Appendices 2.C and parameters given in Table S2.2). The average normalized RF distance between true species trees and true gene trees was in most replicates in the $[0.3, 0.6]$ range, with an average of 0.46 (Fig. 2.1). We use Indelible (Fletcher and Yang, 2009) to simulate the nucleotide sequences along the gene trees using the GTR evolutionary model (Tavaré, 1986) with 4 different fixed sequence lengths: 1600, 800, 400, and 200bp. We then use FastTree2 (Price et al., 2010) to estimate both ML and 100 bootstrapped gene trees under the GTR+Γ (requiring more than two million runs in total). Gene tree estimation error, measured by the FN rate between the true gene trees and the estimated gene trees, depended on the sequence length as shown in Fig. 2.1 (0.55, 0.42, 0.31, and 0.23 on average for 200bp, 400bp, 800bp, and 1600bp, respectively). We sample 1000, 500, 200, or 50 genes to generate datasets with varying numbers of gene trees.

SimPhy-ASTRAL2 (S200):

This dataset (201 taxa) is from the ASTRAL-II paper (Mirarab and Warnow, 2015). We use its most challenging model conditions with max tree height set to 500K generations and two rates of speciation: 10^{-6} and 10^{-7} (respectively, recent and deep speciation events). Compared to S100, this dataset has a much higher level of ILS. This was the only case in the ASTRAL-II

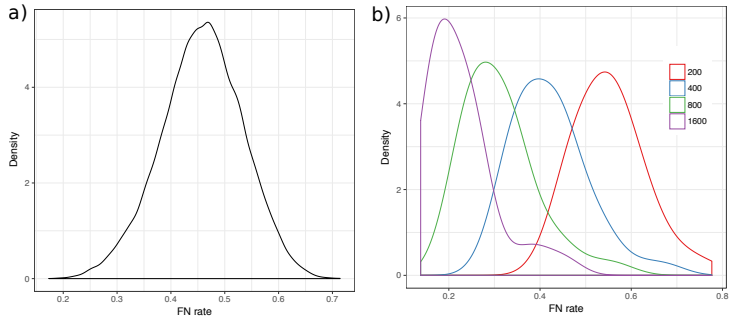

Figure 2.1. *Properties of the S100 dataset. (a)* The density plot of the amount of true gene discordance measured by the FN rate between the true species tree and the true gene trees. *(b)* The density plot of gene tree estimation error measured by FN rate between true gene trees and estimated gene trees for different sets of sequence lengths.

paper where enlarging X substantially impacted accuracy (Mirarab and Warnow, 2015). We use S200 to test if our changes to X have compromised the accuracy. Like S100, gene alignments have varying lengths and mutation rates, leading to a wide range of gene tree error (Mirarab and Warnow, 2015). We analyze the data using 1000, 200, or 50 genes, and each model condition has 50 replicates; following the original paper, three replicates with low signal are removed.

Methods and Evaluation

We compare ASTRAL-III (version 5.5.4) to ASTRAL-II (version 4.11.1) in terms of running time and accuracy. To address RQ1, we draw bootstrap support values on the ML gene trees and then contract branches with bootstrap support up to a threshold (0, 3, 5, 7, 10, 20, 33, 50, and 75%,) using the newick utility package (Junier and Zdobnov, 2010). Together with the original gene trees, we have 10 different versions of ASTRAL-III.

To measure the accuracy of estimated species trees, we use False Negative (FN) rate. Note that in all our species tree comparisons, FN rate is equivalent to normalized Robinson–Foulds (RF) (Robinson and Foulds, 1981) metric because the ASTRAL species trees are fully resolved. All running times are measured on a cluster with servers with Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz; each run was assigned to a single process, sharing cache and memory with other jobs.

2.3.2 RQ1: Impact of contracting low support branches on accuracy

We investigate RQ1 on the two simulated datasets where bootstrapping was feasible (avian and S100) and on the real avian dataset. On S200, due to its size, bootstrapping was not feasible and thus we cannot test RQ1.

S100

On this dataset, contracting *very* low support branches in most cases improves the accuracy (Fig. 2.2 and Table S2.3). However, the excessive removal of branches with high, moderate, or occasionally low support degrades the accuracy. Nevertheless, filtering at 10% is always beneficial on average (Table S2.3). The threshold where contracting starts to become detrimental depends on the condition, especially the number of gene trees and the alignment length, perhaps representing a signal to noise ratio trade-off.

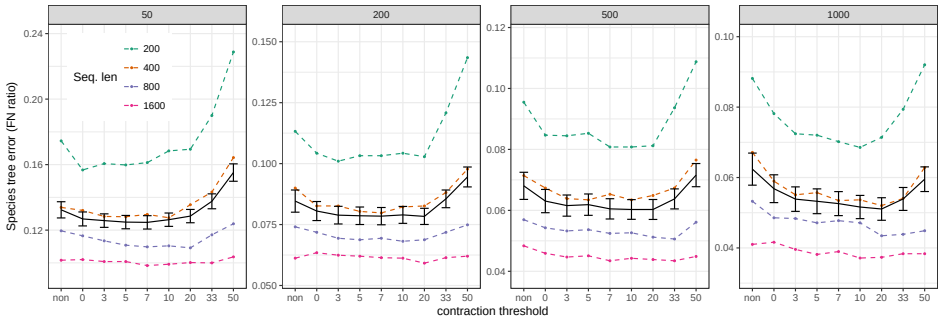

Figure 2.2. *Impact of contraction on the S100 dataset.* The FN error of ASTRAL-III species trees is shown on the S100 dataset given $k = 50$, 200, 500, or 1000 genes (*boxes*) run on the original FastTree gene trees (*non*) or gene trees with branches with $\leq \{0,3,5,7,10,20,33,50\}\%$ support contracted (*x-axis*). Average FN error and standard error bars (200 replicates) are shown with the four alignment lengths combined (*black solid line*). average FN error broken down by alignment length (50 replicates) is also shown (*dashed colored lines*).

As the number of genes increases, the optimal threshold for contracting also tends to increase. Combining all model conditions, the error continues to drop until a 20% contracting threshold with 1000 genes, whereas no substantial improvement is observed after contracting branches with 5% support for 50 genes (Fig. 2.2). Nevertheless, removing branches with 10% or 20% does not increase the error with 50 genes. Perhaps, with few gene trees, removing branches

of low support leaves us with very little information left; thus, regardless of whether we contract or not, we don't get much signal around the most difficult branches. In contrast, when many gene trees are given, perhaps even after removing many branches, still enough gene trees with a resolution around difficult species tree branches are left.

The alignment length and gene tree error also impact the effect of contraction. For short alignments (200bp) and 1000 genes, contracting branches with up to 10% support reduces the species tree error by 21% (from 8.8% with no contraction to 6.9%). As alignment length grows, benefits of gene tree contraction diminish, so that with 1600bp genes, the reduction in error is merely from 4.1% to 3.7%. This pattern is perhaps expected because, with longer alignments, branch support is expected to increase. Thus, with longer gene alignments and consequently better gene trees with higher support, there is less room for improvement by reducing the noise. Consistent with this explanation, grouping replicates based on average gene tree error gives similar results as grouping by alignment length (see Fig. S2.1).

avian-like simulations

On the avian simulated dataset, contracting low support branches helps accuracy marginally, but the extent of impact depends on the model condition (Fig. 2.3). With moderate ILS (2X), we see no improvements as a result of contracting low support branches, perhaps because the average error is below 5% even with no contraction, leaving little room for improvements. Increasing ILS, we start to see improvements using contracted gene trees. Removing branches of up to 5% support reduces the error from 13% to 11% with 0.5X, and from 8% to 7% for the 1X condition.

When ILS is fixed to 1X and sequence length is varied (Fig. 2.3), contracting is helpful mostly with short sequences (e.g., 250bp). With longer sequences, where gene tree estimation error is low, little or no improvement in accuracy is obtained. The best accuracy is typically observed by contracting at 0–5%. The gains in accuracy comparing no contraction to contraction at 0%, 3%, 5% thresholds are statistically significant ($p = 0.017$, 0.028, and 0.013) according to

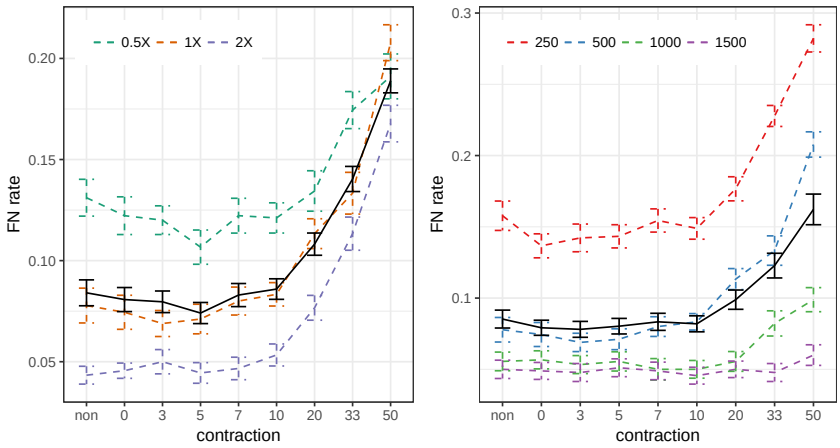

Figure 2.3. *Impact of contraction on the avian simulated dataset.* The FN error of ASTRAL-III species trees is shown on the avian simulated dataset given $k = 1000$ genes with (*left*) fixed sequence lengths $= 500$ and varying levels of ILS, or (*right*) fixed ILS (1X) and varying sequence length, in each case both with full FastTree gene trees (*non*) or trees with branches with $\leq \{0, 3, 5, 7, 10, 20, 33, 50\}\%$ support contracted (*x-axis*). Average and standard error bars are shown for all conditions combined (*black solid line*) and also for each model condition separately (*dashed color lines*). Each model condition has 20 replicates.

one-tailed paired t-tests.

Avian biological dataset

The original analyses on this dataset (Jarvis et al., 2014; Mirarab et al., 2014a) report that MP-EST (Liu et al., 2010) run on 14,446 gene trees produces a tree that conflicts with strong evidence from the literature and other analyses on the same dataset. The statistical binning method was developed to address this shortcoming by combining loci together to reduce gene tree error (Mirarab et al., 2014a; Bayzid et al., 2015). MP-EST run on binned gene trees (i.e., binned MP-EST) produced results (Jarvis et al., 2014; Mirarab et al., 2014a) that were largely congruent with the concatenation using ExaML (Kozlov et al., 2015) and differed in only five branches with low support (Fig. 2.4ab); both trees were used as the reference (Jarvis et al., 2014). Here, we test if simply contracting low support gene tree branches and using ASTRAL-III produces trees congruent with the reference trees.

Similar to MP-EST, when ASTRAL-III is run on 14,446 gene trees with no contraction,

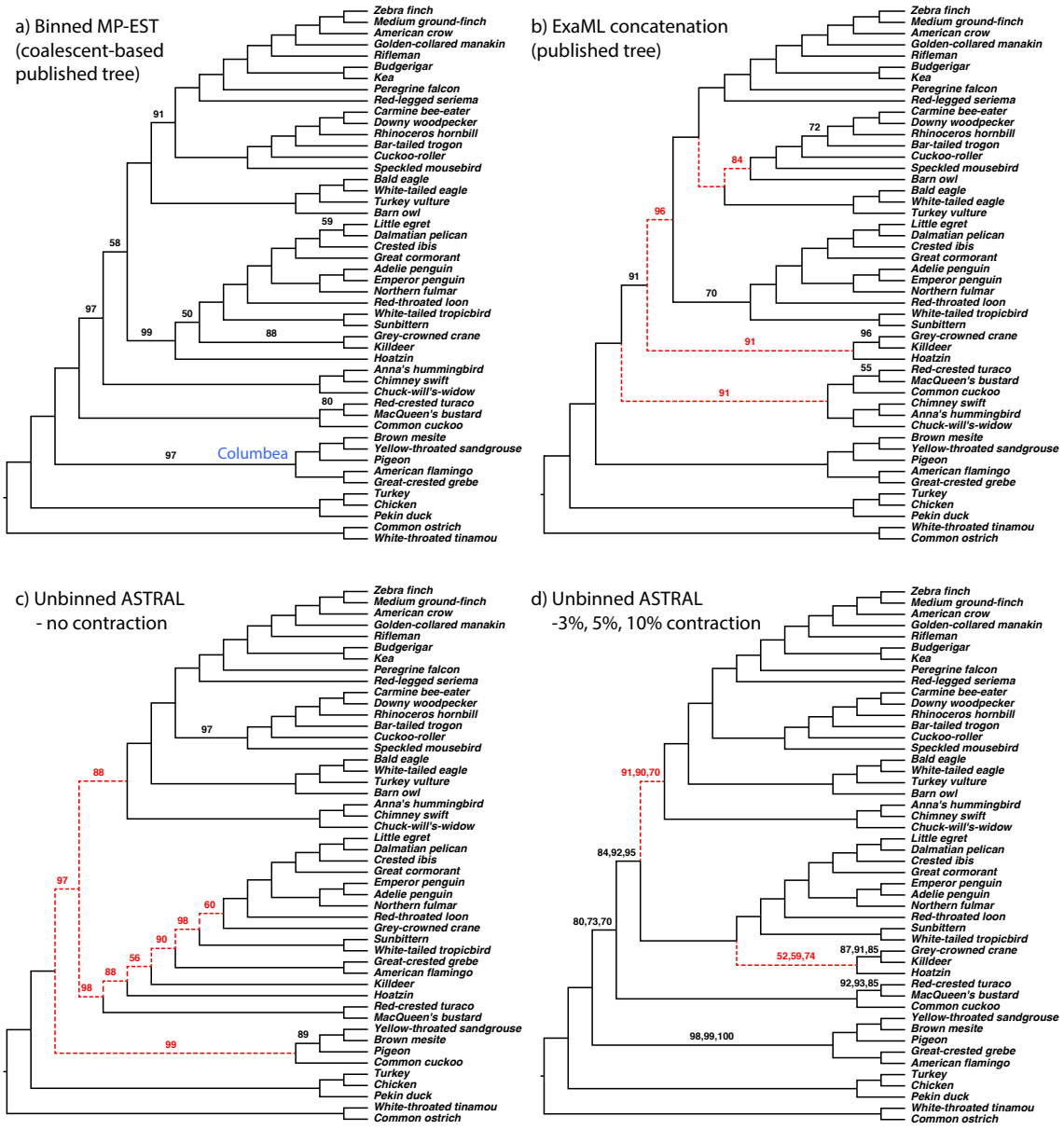

Figure 2.4. *Avian dataset with 14,446 genes.* Shown are reference trees from the original paper (Jarvis et al., 2014) using the coalescent-based binning (*a*) and concatenation (*b*), and two new trees using ASTRAL-III with no contraction (*c*) and with contraction with 3%, 5%, and 10% thresholds (*d*). Support values (bootstrap for *a,b* and local posterior probability for *c,d*) shown for all branches except those with full support; in (d), support is shown for 3%, 5%, and 10%, respectively. Branches conflicting with the reference coalescent-based tree are shown as dotted red lines.

the results differ in nine and 11 branches, respectively, with respect to the reference binned MP-EST and concatenation trees (Fig. 2.4c). Moreover, this tree contradicts some strong results from

the avian analyses (e.g., not recovering the Columbea group, Jarvis et al., 2014). ASTRAL-III with no contraction finishes in 32 hours, but with contraction, depending on the threshold, it takes 3 to 84 hours ($>$ 50 hours for 0% – 20% thresholds and $<$ 26 hours for 33% – 75%). Contracting 0% branches has minimal impact on the discordance (eight discordant branches with binned MP-EST instead of nine). However, contracting low support branches with 3%–33% thresholds dramatically reduces the discordance with the reference tree (2, 2, 4, 2, 3, and 3 discordant branches, respectively, for 3%, 5%, 7%, 10%, 20%, and 33%). Three thresholds (3%, 5%, and 10%) produce an identical tree (Fig. 2.4d). The remaining differences are among the branches that are deemed unresolved by Jarvis *et al.* and change among the reference trees as well (Jarvis et al., 2014). Contracting at 50% and 75% thresholds, however, increases discordance to five and six branches, respectively.

Thus, consistent with simulations, contracting very low support branches seems to produce the best results, when judged by similarity with the reference trees. To summarize, ASTRAL-III obtained on unbinned but collapsed gene trees agreed with all major relations in Jarvis *et al.*, including the novel Columbea group, whereas the unresolved tree missed important clades (Fig. 2.4).

2.3.3 RQ2: Running time improvements

Varying the number of genes (k)

We compare ASTRAL-III to ASTRAL-II on the avian simulated dataset, changing the number of genes from 2^8 to 2^{14} and forcing X to be the same for both versions to enable comparing impacts of improved weight calculation. We allow each replicate run to take up to two days. ASTRAL-III improves the running time over ASTRAL-II and the extent of the improvement depends on k (see Fig. S2.2). With 1000 genes or more, there is at least a 2.1X improvement. With 2^{13} genes, the largest value where both versions could run, ASTRAL-III finishes on average 3.2 times faster than ASTRAL-II (234 versus 758 minutes). ASTRAL-II is not able to finish on the dataset with $k = 2^{14}$, while ASTRAL-III finishes on all conditions.

Moreover, fitting a line to the average running time in the log-log scale graph reveals that on this dataset, the running time of ASTRAL-III on average grows as $O(k^{2.08})$, which is better than that of ASTRAL-II at $O(k^{2.28})$, and both are better than the theoretical worst case, which is $O(k^{2.726})$. These results are consistent with the fact that ASTRAL-III considers similarities between gene tree nodes.

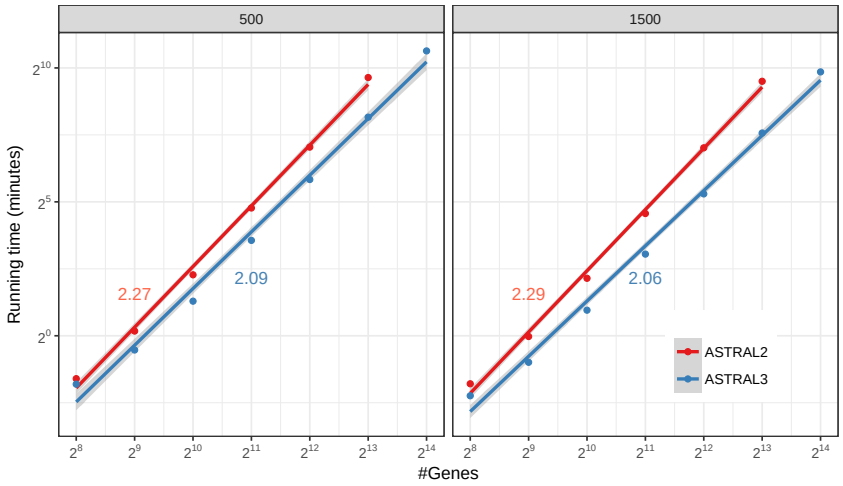

Figure 2.5. *Running time versus k.* Average running times (4 replicates) are shown for ASTRAL-II and ASTRAL-III on the avian dataset with 500bp or 1500bp alignments with varying numbers of gens (k), shown in log scale (see Fig. S2.2 for normal scale). A line is fit to the data points in the log/log space and line slopes are shown. ASTRAL-II did not finish on 2^{14} genes in 48 hours.

Running time for large polytomies

ASTRAL-III has a clear advantage compared to ASTRAL-II with respect to the running time when gene trees include polytomies (Fig. 2.6a and Fig. S2.3). Since ASTRAL-II and ASTRAL-III can have a different set X, we show the running time per each weight calculation (i.e., Eq. 2.3). As we contract low support branches and hence increase the prevalence of polytomies, the weight calculation time quickly grows for ASTRAL-II, whereas, in ASTRAL-III, the weight calculation time remains flat, or even decreases. These results are consistent with a change of asymptotic running time to score a polytomy of size d from $O(d^3)$ in ASTRAL-II to $O(d)$ in ASTRAL-III.

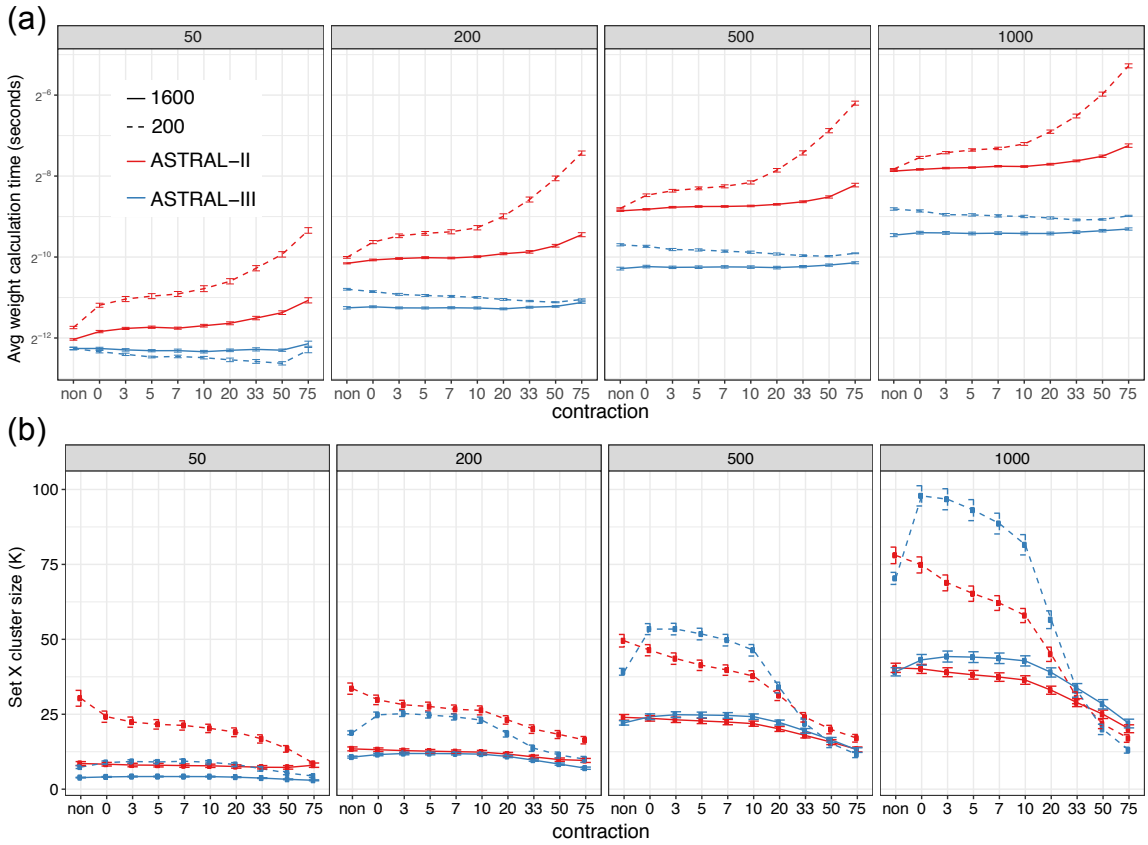

Figure 2.6. *Weight calculation and $|X|$ on S100.* Average and standard error of (a) the time it takes to score a single tripartition using Eq. 2.3 and (b) search space size $|X|$ are shown for both ASTRAL-II and ASTRAL-III on the S100 dataset. Running time is in log scale. We vary numbers of gene trees (*boxes*) and sequence length (200 and 1600). See Fig. S2.3 for similar patterns for with 400 and 800bp alignments.

The search space

Comparing the size of the search space ($|X|$) between ASTRAL-II and ASTRAL-III shows that as intended, the search space is decreased in size for cases with no polytomy but can increase in the presence of polytomies (Fig. 2.6b). With no contraction, on average, $|X|$ is always smaller for ASTRAL-III than ASTRAL-II. With few error-prone gene trees (50 gene trees from 200bp alignments), the search space has reduced dramatically but with many genes or high-quality gene trees, the reductions are minimal. Moreover, the search space for gene trees estimated from short alignments (e.g., 200bp) is several times larger than those based on longer alignments (e.g., 1600bp) for both methods. These are results of the first feature of ASTRAL-III

that forces the search space to grow at $O(nk)$.

Contracting low support branches initially increases the search space. This is because ASTRAL-III unlike ASTRAL-II adds multiple resolutions per polytomy to X. Further contraction results in reductions in $|X|$, presumably because many polytomies exist and they are resolved similarly inside ASTRAL-III.

2.3.4 RQ3: ASTRAL-II versus ASTRAL-III accuracy

Despite limiting $|X|$ to grow at most linearly with n and k, the accuracy of ASTRAL-III remains unchanged compared to ASTRAL-II (Table 2.1 and Figs. S2.4–S2.7). Importantly, even for the very challenging S200 dataset, the accuracy is not reduced substantially even though $|X|$ is reduced by up to 47%. Over all datasets, differences in error are less than 0.002, except for three datasets where the error of ASTRAL-III was less than ASTRAL-II by 0.003, 0.005, and 0.006 and two cases where the error increased by 0.004. Over all datasets, the differences between ASTRAL-II and ASTRAL-III were not statistically significant according to a paired t-test (p-value = 0.496). Since ASTRAL-III has a reduced search space, its quartet scores are typically slightly lower than ASTRAL-II, but these reductions are never more than 0.06%. As expected, the largest drops in the quartet score happen for the challenging S200 dataset with only 50 gene trees. The search space reduces in almost all cases and the reductions can be as much as 72%. Thus, the improved running time of ASTRAL-III does not come at the price of reduced accuracy.

2.4 Discussion

Below we further comment on ASTRAL-III in terms of accuracy and running time. We finish by comparing ASTRAL-III and ASTRAL-III-beta.

Table 2.1. ASTRAL-II versus ASTRAL-III. Average and standard error (inside parenthesis) are shown for changes in accuracy (normalized FN rate), quartet score, and search space size ($|X|$). FN: we show ASTRAL-III − ASTRAL-II; negative numbers indicate ASTRAL-III is more accurate. $|X|$: we show $\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$; negative numbers indicate that ASTRAL-III has a reduced search space. Quartet score: we show $\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$; positive numbers indicate that ASTRAL-III has improved quartet scores. See Figures. S2.4–S2.7 for full distributions.

data set	model condition	FN	$\lvert X\rvert$	quartet score
avian	0.5X-500bp	−0.006 (0.007)	−3% (0)	−0.01% (0.01)
	1X-1000bp	0.001 (0.002)	−1% (0)	0.00% (0.00)
	1X-1500bp	0.004 (0.003)	−1% (0)	0.00% (0.00)
	1X-250bp	0.004 (0.007)	−3% (0)	−0.01% (0.00)
	1X-500bp	−0.001 (0.004)	−2% (0)	0.00% (0.00)
	2X-500bp	−0.003 (0.003)	−2% (0)	0.00% (0.00)
S200	1000gt-10^{-6}	−0.001 (0.000)	0% (0)	0.00% (0.00)
	200gt-10^{-6}	0.000 (0.001)	−5% (1)	0.00% (0.00)
	50gt-10^{-6}	−0.001 (0.001)	−42% (2)	−0.06% (0.01)
	1000gt-10^{-7}	0.001 (0.001)	−1% (0)	0.00% (0.00)
	200gt-10^{-7}	−0.001 (0.001)	−6% (1)	0.00% (0.01)
	50gt-10^{-7}	0.000 (0.002)	−47% (2)	−0.06% (0.01)
S100	1000gt-1600bp	0.000 (0.000)	−3% (0)	0.00% (0.00)
	500gt-1600bp	0.000 (0.000)	−6% (1)	0.00% (0.00)
	200gt-1600bp	0.000 (0.001)	−17% (1)	−0.01% (0.00)
	50gt-1600bp	−0.001 (0.001)	−46% (3)	−0.01% (0.01)
	1000gt-200bp	−0.001 (0.002)	−9% (1)	0.00% (0.00)
	500gt-200bp	−0.001 (0.001)	−19% (1)	−0.01% (0.01)
	200gt-200bp	−0.001 (0.001)	−40% (1)	−0.01% (0.00)
	50gt-200bp	−0.002 (0.002)	−72% (1)	−0.05% (0.01)
	1000gt-400bp	−0.001 (0.002)	−6% (1)	0.00% (0.00)
	500gt-400bp	0.001 (0.001)	−12% (1)	−0.01% (0.00)
	200gt-400bp	0.000 (0.001)	−29% (2)	−0.01% (0.01)
	50gt-400bp	−0.005 (0.001)	−61% (2)	−0.02% (0.01)
	1000gt-800bp	0.000 (0.000)	−4% (0)	0.00% (0.00)
	500gt-800bp	0.001 (0.001)	−9% (1)	0.00% (0.00)
	200gt-800bp	0.001 (0.000)	−22% (2)	−0.01% (0.01)
	50gt-800bp	0.000 (0.001)	−52% (3)	−0.02% (0.01)

2.4.1 Accuracy

Although tree accuracy can improve with contracted gene trees, the gap between performance on true gene trees and estimated gene trees remains wide (Table S2.3). On the S100

dataset, respectively for 50, 200, 500, and 1000 genes, the best average error with 1600bp gene trees among all contraction levels were 9.8%, 5.9%, 4.3%, and 3.7% compared to 7.0%, 3.7%, 2.4%, and 1.5% with true gene trees. Thus, while contracting low support branches helps in addressing gene tree error, improved methods of gene tree estimation remain crucial. Our results also indicate that in the presence of noisy gene trees, increased numbers of genes are needed to achieve high accuracy. For example, on the S100 dataset, with 1000 gene trees of only 200bp and contracting with a 10% threshold, the species tree error was 6.9%, which slightly outperformed the accuracy with only 50 true gene trees. This observation encourages the use of a large number of gene trees; incidentally, a main feature of ASTRAL-III is improved running time with many genes.

The best choice of the threshold of contraction was somewhat sensitive to the dataset. Testing up to 1000 gene trees, we observed that more gene trees clearly increased the optimal threshold, but did not test beyond 1000 genes. One can predict that perhaps the trend may continue but also that the optimal threshold will not indefinitely increase. Similarly, we saw that the amount of gene tree error due to lack of signal impacts the optimal threshold. One may expect that other sources of error, including incorrect orthology, incorrect alignment, and model misspecifications may also impact the optimal threshold. Regardless of the choice of the optimal threshold, it seems that the largest benefits are associated with removing the least supported branches. Overall, a threshold of 10% seemed to provide a good default value.

In most datasets, a substantial accuracy improvement was observed when 0% BS branches were removed. Branches of 0% support are presumably resolved arbitrarily. The use of conserved genes or closely related taxa can increase instances where two or more taxa have identical sequences in some genes. Some tree estimation methods generate binary trees even under such conditions. Removing branches that are arbitrarily resolved make sense and, as our results indicate, improves accuracy.

The main competitor of ASTRAL is NJst (Liu and Yu, 2011) and its fast implementation, ASTRID (Vachaspati and Warnow, 2015), but these tools are not able to handle polytomies

in input gene trees. ASTRAL-III makes it efficient to use unresolved gene trees. Moreover, beyond contracting low support branches, other strategies could be used to reduce impacts of gene tree uncertainty. Previous studies indicate that simply using the set of all bootstrap gene tree replicates as input to ASTRAL increases error (Mirarab et al., 2014b), perhaps due to the increased noise (Mirarab et al., 2016; Sayyari and Mirarab, 2016b). However, using a sample from the Bayesian distribution for each gene tree may improve the accuracy of ASTRAL.

Finally, theoretical implications of removing low support branches are less clear than its empirical impact. In principle, branches that have low support are not necessarily expected to be randomly selected among gene trees. Thus, while our empirical results support the use of (conservative) filtering, the resulting procedure may lose statistical guarantees of consistency. Future work should study conditions where ASTRAL remains statistically consistent with contracted gene trees.

2.4.2 Running time

Large n

To assess limits of ASTRAL-III in terms of scalability, we tested it on 20 replicates of a dataset with 5,000 species and 1000 true gene trees (simulation procedure described in Appendices 2.C and parameters given in Table S2.4). ASTRAL-III took between 2 and 62 hours to run on this dataset (9.4 hours on average). We also attempted to test ASTRAL-III on four replicates of a dataset with 10,000 species and 1000 true gene trees, allowing a week of running time. Of the four replicates, two were able to finish within the allotted time. Thus, depending on the nature of the data, ASTRAL-III may be able to scale to datasets with up to 10,000 species given sufficient running time.

Average running time, $|X|$, and $|Y|$

The ASTRAL-III running time analysis is based on several worst-case assumptions, and real data may grow less rapidly with both n and k. Overall, although the exact value depends

on the dataset and especially the amount of discordance, the running time of ASTRAL seems to grow roughly quadratically with both n and k (i.e., proportionally to n^2k^2); see Figures S2.2 and S2.8.

ASTRAL-III bounds $|X|$ to grow at most linearly with n and k. Empirically, we observe that $|X|$ grows sublinearly with k (close to $O(k^{\frac{3}{4}})$) on the avian simulated dataset (Fig. 2.7a). Note that the avian dataset has one of the highest levels of ILS; the dependence on k is expected to be lower for datasets with lower gene tree discordance. Testing the growth with n is more difficult because as n changes, other factors such as the amount of discordance also change. Nevertheless, across all the datasets that we had available, we tested the change in running time for fixed k as n changes and observed a linear growth (Fig. 2.7b), matching the worst-case scenario.

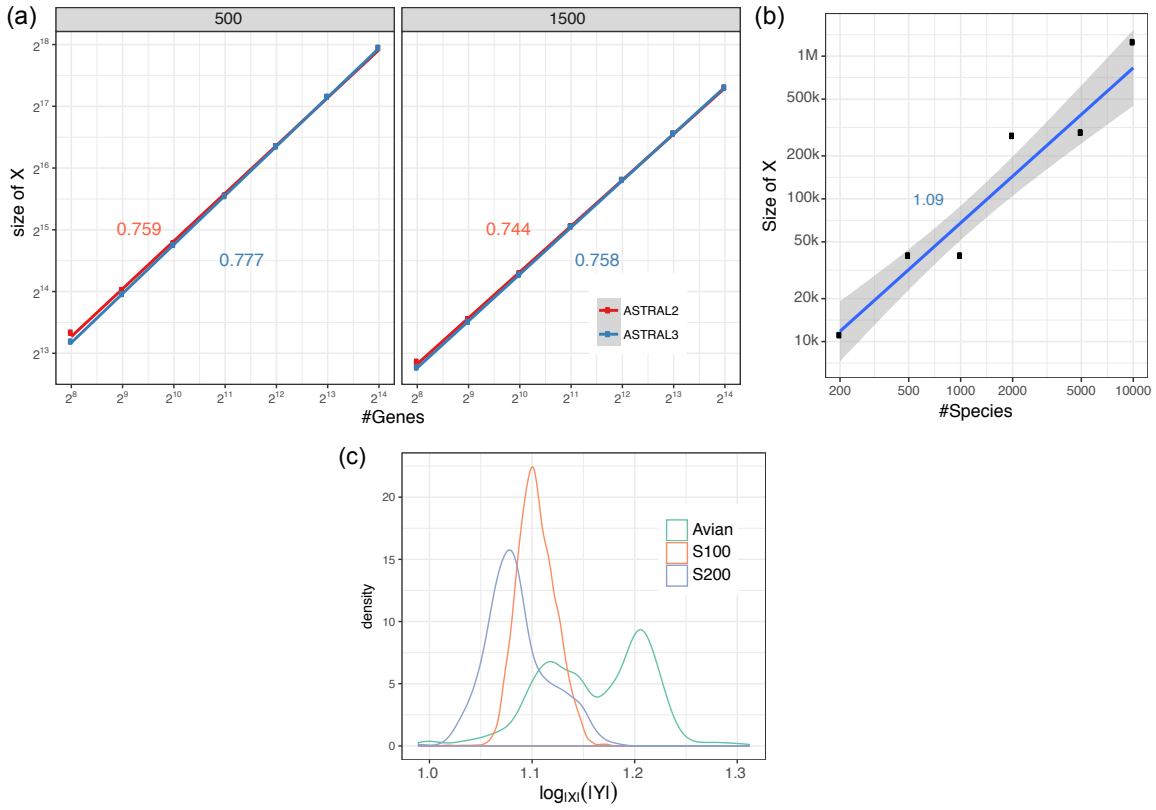

Figure 2.7. *Empirical search space.* (a) $|X|$ is shown for ASTRAL-II and ASTRAL-III for avian-like simulated dataset with varying numbers of genes. (b) $|X|$ is shown for ASTRAL-III for several datasets with varying n. (c) The density plots of $\log_X |Y|$ across all ASTRAL-III runs reported in this paper. Size of the dynamic programming space Y is never above $|X|^{1.312}$ here.

Finally, establishing empirical running time growth requires establishing the rate of the growth of $|Y|$ with respect to $|X|$. The $|Y| \leq |X|^{1.726}$ upper-bound is for specialized formations of the set X (Kane and Tao, 2017). Empirically, as $|X|$ increases, the size of $|Y|$ in ASTRAL-III does not increase as fast as the worst-case scenario implies. Across all of our ASTRAL-III runs in this paper, $|Y|$ ranged in 90% of our runs between $|X|^{1.07}$ and $|X|^{1.20}$, and the overall average was $|X|^{1.11}$ (Fig. 2.7c).

2.4.3 Comparisons to ASTRAL-III-beta

The beta version of ASTRAL-III (Zhang et al., 2017) included features 3–5 but not changes to X (features 1 and 2) or the two-staged α-trimming technique (feature 6). For completeness, we compared ASTRAL-III-beta and ASTRAL-III in terms of accuracy, quartet score, and the running time (Table 2.2). Accuracy and quartet scores are very similar, perhaps with a small improvement since the beta version. The search space is reduced since the beta version (due to features 1 and 2), and the running times are substantially decreased (at least by half in most cases). The reductions in the running time are due to α-trimming, reduced $|X|$, in addition to further improvements in details of our implementation of the polytree data-structure.

Table 2.2. ASTRAL-III-beta vs ASTRAL-III. Columns are defined similar to Table 2.1. Negative numbers indicate ASTRAL-III-beta has a larger value (i.e., has higher error, larger search space, better quartet scores, and is slower).

model condition	contraction	FN	$\lvert X\rvert$	$\lvert Y\rvert$	quartet score	running time
avian-0.5X-500bp	None	−0.003	−3%	−9%	−0.02%	−48%
avian-1X-250bp	None	−0.001	−3%	−9%	0.00%	−56%
avian-1X-500bp	None	−0.001	−2%	−6%	0.00%	−50%
avian-1X-1000bp	None	−0.001	−1%	−4%	0.00%	−58%
avian-1X-1500bp	None	0.001	−1%	−4%	0.00%	−57%
avian-2X-500bp	None	−0.002	−2%	−4%	0.00%	−65%
avian-0.5X-500bp	10%	−0.003	−3%	−29%	−0.01%	−69%
avian-1X-250bp	10%	−0.001	−50%	−40%	0.00%	−81%
avian-1X-500bp	10%	0.003	−18%	−62%	−0.01%	−62%
avian-1X-1000bp	10%	0.000	−5%	−8%	0.00%	−61%
avian-1X-1500bp	10%	0.003	0%	−1%	0.00%	−55%
avian-2X-500bp	10%	−0.002	−14%	−18%	0.00%	−62%

To further demonstrate the impact of the α-trimming feature, we randomly chose 18 species from the avian dataset with 1500bp and 1X ILS. On this limited dataset, we ran ASTRAL-III in its exact mode (i.e., setting X to the power set) with 100 gene trees. Without any trimming of the dynamic programming (i.e., without features 5 and 6), the running time was 40 minutes. Emulating ASTRAL-III-beta, we disabled α-trimming but kept the trimming (feature 5) and the running time reduced to 33 minutes. Adding the α-trimming feature dramatically reduced the running time to 13 minutes. Thus, when X includes many bipartitions that have very little promise in improving the quartet score (as in the exact mode of ASTRAL), the α-trimming approach is very effective in reducing the running time.

2.5 Conclusions

We introduced ASTRAL-III, which compared to ASTRAL-II, improves scalability, especially for datasets with large k and many polytomies. These improvements enabled us to test the accuracy of ASTRAL after contracting low support branches. Overall, we observed improvements in accuracy when very low support branches were contracted, but also evidence that aggressive filtering reduces the accuracy. ASTRAL-III bounds the theoretical running time to $O((nk)^{1.726}.D)$ where $D = O(nk)$ is the sum of degrees of all unique gene tree nodes. In practice, the running time tends to grow no worse than quadratically with both n and k.

2.6 Acknowledgements

Chapter 2, in full, is a reprint of the material as it appears in " Zhang, C., Rabiee, M., Sayyari, E. & Mirarab, S. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*. **19**, 15-30 (2018) ." The dissertation author was the primary investigator and first author of this paper.

Bibliography

E. S. Allman, J. H. Degnan, and J. A. Rhodes. Determining species tree topologies from clade probabilities under the coalescent. *Journal of Theoretical Biology*, 289(1):96–106, 2011. ISSN 00225193. doi: 10.1016/j.jtbi.2011.08.006. URL http://dx.doi.org/10.1016/j.jtbi.2011.08.006.

M. S. Bayzid, S. Mirarab, B. Boussau, and T. Warnow. Weighted Statistical Binning: Enabling Statistically Consistent Genome-Scale Phylogenetic Analyses. *PLoS ONE*, 10(6):e0129183, 1 2015. ISSN 1932-6203. doi: 10.1371/journal.pone.0129183. URL http://dx.doi.org/10.1371/journal.pone.0129183.

R. Davidson, P. Vachaspati, S. Mirarab, and T. Warnow. Phylogenomic species tree estimation in the presence of incomplete lineage sorting and horizontal gene transfer. *BMC Genomics*, 16(Suppl 10):S1, 2015. ISSN 1471-2164. doi: 10.1186/1471-2164-16-S10-S1. URL http://www.biomedcentral.com/1471-2164/16/S10/S1.

J. H. Degnan and N. A. Rosenberg. Gene tree discordance, phylogenetic inference and the multispecies coalescent. *Trends in Ecology and Evolution*, 24(6):332–340, 6 2009. ISSN 01695347. doi: 10.1016/j.tree.2009.01.009. URL http://www.cell.com/ecology-evolution/abstract/S0169-5347(09)00084-6http://www.sciencedirect.com/science/article/pii/S0169534709000846.

S. V. Edwards. Is a new and general theory of molecular systematics emerging? *Evolution*, 63 (1):1–19, 2009. ISSN 1558-5646. doi: 10.1111/j.1558-5646.2008.00549.x.

S. V. Edwards, Z. Xi, A. Janke, B. C. Faircloth, J. E. McCormack, T. C. Glenn, B. Zhong, S. Wu, E. M. Lemmon, A. R. Lemmon, A. D. Leaché, L. Liu, and C. C. Davis. Implementing and testing the multispecies coalescent model: A valuable paradigm for phylogenomics. *Molecular Phylogenetics and Evolution*, 94:447–462, 2016. ISSN 10959513. doi: 10.1016/j.ympev.2015.10.027. URL http://dx.doi.org/10.1016/j.ympev.2015.10.027.

W. Fletcher and Z. Yang. INDELible: A flexible simulator of biological sequence evolution. *Molecular Biology and Evolution*, 26(8):1879–1888, 2009. ISSN 07374038. doi: 10.1093/molbev/msp098.

J. Gatesy and M. S. Springer. Phylogenetic Analysis at Deep Timescales: Unreliable Gene Trees,

Bypassed Hidden Support, and the Coalescence/Concatalescence Conundrum. *Molecular phylogenetics and evolution*, 80:231–266, 2014. ISSN 1095-9513. doi: 10.1016/j.ympev.2014.08.013. URL http://www.ncbi.nlm.nih.gov/pubmed/25152276.

P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. ISSN 0536-1567.

J. Heled and A. J. Drummond. Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution*, 27(3):570–580, 3 2010. ISSN 1537-1719. doi: 10.1093/molbev/msp274. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2822290&tool=pmcentrez&rendertype=abstracthttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2822290&tool=pmcentrez&rendertype=abstract.

E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. W. Ho, B. C. Faircloth, B. Nabholz, J. T. Howard, A. Suh, C. C. Weber, R. R. da Fonseca, J. Li, F. Zhang, H. Li, L. Zhou, N. Narula, L. Liu, G. Ganapathy, B. Boussau, M. S. Bayzid, V. Zavidovych, S. Subramanian, T. Gabaldón, S. Capella-Gutiérrez, J. Huerta-Cepas, B. Rekepalli, K. Munch, M. H. Schierup, B. Lindow, W. C. Warren, D. Ray, R. E. Green, M. W. Bruford, X. Zhan, A. Dixon, S. Li, N. Li, Y. Huang, E. P. Derryberry, M. F. Bertelsen, F. H. Sheldon, R. T. Brumfield, C. V. Mello, P. V. Lovell, M. Wirthlin, M. P. C. Schneider, F. Prosdocimi, J. A. Samaniego, A. M. V. Velazquez, A. Alfaro-Núñez, P. F. Campos, B. Petersen, T. Sicheritz-Ponten, A. Pas, T. Bailey, P. Scofield, M. Bunce, D. M. Lambert, Q. Zhou, P. Perelman, A. C. Driskell, B. Shapiro, Z. Xiong, Y. Zeng, S. Liu, Z. Li, B. Liu, K. Wu, J. Xiao, X. Yinqi, Q. Zheng, Y. Zhang, H. Yang, J. Wang, L. Smeds, F. E. Rheindt, M. J. Braun, J. Fjeldså, L. Orlando, F. K. Barker, K. A. Jønsson, W. Johnson, K.-P. Koepfli, S. O'Brien, D. Haussler, O. A. Ryder, C. Rahbek, E. Willerslev, G. R. Graves, T. C. Glenn, J. E. McCormack, D. W. Burt, H. Ellegren, P. Alström, S. V. Edwards, A. Stamatakis, D. P. Mindell, J. Cracraft, E. L. Braun, T. Warnow, W. Jun, M. T. P. Gilbert, and G. Zhang. Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331, 12 2014. doi: 10.1126/science.1253451. URL http://www.sciencemag.org/content/346/6215/1320.abstract.

T. Junier and E. M. Zdobnov. The Newick utilities: high-throughput phylogenetic tree processing in the UNIX shell. *Bioinformatics*, 26(13):1669–1670, 2010. ISSN 13674803. doi: 10.1093/bioinformatics/btq243.

D. Kane and T. Tao. A bound on partitioning clusters. *arXiv:11702.00912*, 2017. URL http://arxiv.org/abs/1702.00912.

A. M. Kozlov, A. J. Aberer, and A. Stamatakis. ExaML version 3: a tool for phylogenomic analyses on supercomputers. *Bioinformatics*, 31(15):2577–2579, 8 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btv184. URL https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btv184http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btv184.

M. Lafond and C. Scornavacca. On the Weighted Quartet Consensus problem. *arXiv*, 610.00505, 10 2016. URL http://arxiv.org/abs/1610.00505.

C. E. Laumer, A. Hejnol, and G. Giribet. Nuclear genomic signals of the 'microturbellarian' roots of platyhelminth evolutionary innovation. *eLife*, 4, 3 2015. ISSN 2050-084X. doi: 10.7554/eLife.05503. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=4398949&tool=pmcentrez&rendertype=abstract.

L. Liu and L. Yu. Estimating species trees from unrooted gene trees. *Systematic Biology*, 60: 661–667, 2011. ISSN 10635157. doi: 10.1093/sysbio/syr027.

L. Liu, L. Yu, D. K. Pearl, and S. V. Edwards. Estimating species phylogenies using coalescence times among sequences. *Systematic Biology*, 58(5):468–477, 10 2009. ISSN 10635157. doi: 10.1093/sysbio/syp031. URL http://www.ncbi.nlm.nih.gov/pubmed/20525601.

L. Liu, L. Yu, and S. V. Edwards. A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC Evolutionary Biology*, 10(1):302, 2010. URL http://www.biomedcentral.com/1471-2148/10/302.

W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 9 1997. ISSN 10635157. doi: 10.2307/2413694. URL http://sysbio.oxfordjournals.org/cgi/content/abstract/46/3/523http://www.jstor.org/stable/2413694?origin=crossrefhttp://sysbio.oxfordjournals.org/content/46/3/523.short.

D. Mallo, L. De Oliveira Martins, and D. Posada. SimPhy : Phylogenomic Simulation of Gene, Locus, and Species Trees. *Systematic Biology*, 65(2):334–344, 3 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syv082. URL http://biorxiv.org/content/early/2015/06/30/021709.abstracthttp://sysbio.oxfordjournals.org/content/early/2015/12/04/sysbio.syv082.short?rss=1https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syv082.

K. A. Meiklejohn, B. C. Faircloth, T. C. Glenn, R. T. Kimball, and E. L. Braun. Analysis of a Rapid Evolutionary Radiation Using Ultraconserved Elements: Evidence for a Bias in Some Multispecies Coalescent Methods. *Systematic Biology*, 65(4):612–627, 7 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syw014. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syw014.

S. Mirarab and T. Warnow. ASTRAL-II: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 6 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btv234. URL http://bioinformatics.oxfordjournals.org/cgi/content/long/31/12/i44.

S. Mirarab, M. S. Bayzid, B. Boussau, and T. Warnow. Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, 346(6215):1250463–1250463, 12 2014a. ISSN 0036-8075. doi: 10.1126/science.1250463. URL http://www.sciencemag.org/

cgi/doi/10.1126/science.1250463.

S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548, 9 2014b. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu462. URL http://bioinformatics.oxfordjournals.org/cgi/content/long/30/17/i541http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btu462.

S. Mirarab, M. S. Bayzid, and T. Warnow. Evaluating Summary Methods for Multilocus Species Tree Estimation in the Presence of Incomplete Lineage Sorting. *Systematic Biology*, 65(3):366–380, 5 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syu063. URL http://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063%5Cnhttp://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063.abstract%5Cnhttp://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063.full.pdf%5Cnhttp://www.n.

E. Mossel and S. Roch. Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 7(1):166–171, 1 2010. ISSN 1557-9964. doi: 10.1109/TCBB.2008.66. URL http://dl.acm.org/citation.cfm?id=1719272.1719288.

P. Pamilo and M. Nei. Relationships between gene trees and species trees. *Molecular biology and evolution*, 5(5):568–583, 1988. ISSN 0737-4038. URL http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&dopt=Citation&list_uids=3193878.

S. Patel. Error in Phylogenetic Estimation for Bushes in the Tree of Life. *Journal of Phylogenetics & Evolutionary Biology*, 01(02):110, 2013. ISSN 23299002. doi: 10.4172/2329-9002.1000110. URL http://esciencecentral.org/journals/error-in-phylogenetic-estimation-for-bushes-in-the-tree-of-life-2329-9002.1000110.pdf.

M. N. Price, P. S. Dehal, and A. P. Arkin. FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 3 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0009490. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2835736&tool=pmcentrez&rendertype=abstract.

B. Rannala and Z. Yang. Bayes estimation of species divergence times and ancestral population sizes using DNA sequences from multiple loci. *Genetics*, 164(4):1645–1656, 2003.

D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, 1981. URL http://www.sciencedirect.com/science/article/pii/0025556481900432.

S. Roch and T. Warnow. On the Robustness to Gene Tree Estimation Error (or lack thereof) of Coalescent-Based Species Tree Methods. *Systematic Biology*, 64(4):663–676, 2015. ISSN 1076836X. doi: 10.1093/sysbio/syv016.

A. Rokas, B. L. Williams, N. King, and S. B. Carroll. Genome-scale approaches to resolving incongruence in molecular phylogenies. *Nature*, 425(6960):798–804, 10 2003. ISSN 1476-4687. doi: 10.1038/nature02053. URL http://www.ncbi.nlm.nih.gov/pubmed/14574403.

E. Sayyari and S. Mirarab. Anchoring quartet-based phylogenetic distances and applications to species tree reconstruction. *BMC Genomics*, 17(S10):101–113, 11 2016a. ISSN 1471-2164. doi: 10.1186/s12864-016-3098-z. URL http://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-016-3098-z.

E. Sayyari and S. Mirarab. Fast Coalescent-Based Computation of Local Branch Support from Quartet Frequencies. *Molecular Biology and Evolution*, 33(7):1654–1668, 7 2016b. ISSN 0737-4038. doi: 10.1093/molbev/msw079. URL http://mbe.oxfordjournals.org/content/early/2016/04/15/molbev.msw079.abstracthttp://mbe.oxfordjournals.org/lookup/doi/10.1093/molbev/msw079https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msw079.

S. Shekhar, S. Roch, and S. Mirarab. Species tree estimation using ASTRAL: how many genes are enough? *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, PP(99):1–1, 2017. ISSN 1545-5963. doi: 10.1109/TCBB.2017.2757930. URL http://ieeexplore.ieee.org/document/8053780/.

X.-x. Shen, C. T. Hittinger, and A. Rokas. Contentious relationships in phylogenomic studies can be driven by a handful of genes. *Nature Ecology & Evolution*, 1(5):0126, 4 2017. ISSN 2397-334X. doi: 10.1038/s41559-017-0126. URL http://dx.doi.org/10.1038/s41559-017-0126http://www.nature.com/articles/s41559-017-0126.

S. Song, L. Liu, S. V. Edwards, and S. Wu. Resolving conflict in eutherian mammal phylogeny using phylogenomics and the multispecies coalescent model. *Proceedings of the National Academy of Sciences*, 109(37):14942–7, 9 2012. ISSN 1091-6490. doi: 10.1073/pnas.1211733109. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3443116&tool=pmcentrez&rendertype=abstracthttp://www.pnas.org/cgi/content/long/109/37/14942.

M. S. Springer and J. Gatesy. The gene tree delusion. *Molecular Phylogenetics and Evolution*, 94(Part A):1–33, 7 2016. ISSN 10557903. doi: 10.1016/j.ympev.2015.07.018. URL http://www.sciencedirect.com/science/article/pii/S1055790315002225http://linkinghub.elsevier.com/retrieve/pii/S1055790315002225.

A. Stamatakis. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014. ISSN 14602059. doi: 10.1093/bioinformatics/btu033.

J. E. Tarver, M. dos Reis, S. Mirarab, R. J. Moran, S. Parker, J. E. O'Reilly, B. L. King, M. J. O'Connell, R. J. Asher, T. Warnow, K. J. Peterson, P. C. Donoghue, and D. Pisani. The Interrelationships of Placental Mammals and the Limits of Phylogenetic Inference. *Genome*

Biology and Evolution, 8(2):330–344, 2 2016. ISSN 1759-6653. doi: 10.1093/gbe/evv261. URL http://gbe.oxfordjournals.org/cgi/content/long/evv261v1http://gbe.oxfordjournals.org/lookup/doi/10.1093/gbe/evv261.

S. Tavaré. Some Probabilistic and Statistical Problems in the Analysis of DNA Sequences. *Lectures on Mathematics in the Life Sciences*, 17:57–86, 1986.

P. Vachaspati and T. Warnow. ASTRID: Accurate Species TRees from Internode Distances. *BMC Genomics*, 16(Suppl 10):S3, 2015. ISSN 1471-2164.

N. J. Wickett, S. Mirarab, N. Nguyen, T. Warnow, E. J. Carpenter, N. Matasci, S. Ayyampalayam, M. S. Barker, J. G. Burleigh, M. A. Gitzendanner, B. R. Ruhfel, E. Wafula, J. P. Der, S. W. Graham, S. Mathews, M. Melkonian, D. E. Soltis, P. S. Soltis, N. W. Miles, C. J. Rothfels, L. Pokorny, A. J. Shaw, L. DeGironimo, D. W. Stevenson, B. Surek, J. C. Villarreal, B. Roure, H. Philippe, C. W. DePamphilis, T. Chen, M. K. Deyholos, R. S. Baucom, T. M. Kutchan, M. M. Augustin, J. Wang, Y. Zhang, Z. Tian, Z. Yan, X. Wu, X. Sun, G. K.-S. Wong, and J. Leebens-Mack. Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences*, 111(45):4859–4868, 10 2014. ISSN 0027-8424. doi: 10.1073/pnas.1323926111. URL http://www.pnas.org/cgi/content/long/111/45/E4859.

Y. Yu, T. Warnow, and L. Nakhleh. Algorithms for MDC-based multi-locus phylogeny inference: beyond rooted binary gene trees on single alleles. *Journal of Computational Biology*, 18(11):1543–1559, 11 2011. ISSN 1557-8666. doi: 10.1089/cmb.2011.0174. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3216099&tool=pmcentrez&rendertype=abstracthttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3216099&tool=pmcentrez&rendertype=abstract.

C. Zhang, E. Sayyari, and S. Mirarab. ASTRAL-III: Increased Scalability and Impacts of Contracting Low Support Branches. In J. Meidanis and L. Nakhleh, editors, *Lecture Notes in Computer Science*, volume 10562 LNBI, pages 53–75. Springer International Publishing, Cham, 2017. ISBN 978-3-319-67979-2.

Appendices

2.A Supplementary method details

2.A.1 Defining the set X

ASTRAL-II uses several techniques to augment the set X, which we describe below. We also describe how ASTRAL-III modifies each technique.

2.A.2 Similarity matrix

All bipartitions from a UPGMA tree based on a quartet-based measure of distance are added to X. In ASTRAL-III, we improve the distance matrix when gene trees have polytomies. Unlike ASTRAL-II, in ASTRAL-III we make sure that unresolved quartets in input gene trees contribute exactly 0 to our counts of different quartet topologies used in building the similarity matrix. Note that this similarity matrix is separate from and has no impact on the quartet scores.

2.A.3 Greedy trees

ASTRAL-II uses a set of heuristics based on the greedy consensus of gene trees to augment the set X. It first constructs a set of greedy consensus trees using a set of thresholds for minimum frequency of bipartitions. The polytomies in the greedy consensus trees are resolved in three different ways and resulting bipartitions are added to X (see Algorithm S2.1). Of the methods used to resolve the polytomy with degree d, two of them (i.e., using a UPGMA tree started from sides of the polytomy and a greedy consensus of gene trees subsampled to randomly selected taxa) can only add $O(d)$ new bipartitions. The third resolution samples a taxon from

each side of the polytomy; it then computes a caterpillar tree constructed based on decreasing similarity to each sampled taxon and adds the bipartitions from all these caterpillar trees to the search space. This step can add $O(d^2)$ bipartitions to the search space. In ASTRAL-III, to guarantee $|X| = O(nk)$, we need to constrain this step. Let $d_1 \dots d_r$ be the list of all polytomies, ordered from the smallest to the largest. Then, we find the smallest threshold q such that $\sum_{i=1}^{q} d_i^2 \leq cn$ for a constant c, set by default to 25. In ASTRAL-III, we only compute and add bipartitions using caterpillar resolutions for polytomies $d_1 \dots d_q$ (see Algorithm S2.1). By construction, this will ensure that at most $O(n)$ bipartitions are added in this step. Finally, these resolutions can happen in multiple rounds. In ASTRAL-III, we make sure these rounds of resolutions do not grow beyond a constant (default: 100).

2.A.4 Gene tree polytomies

If a gene tree includes polytomies, ASTRAL-II adds bipartitions implied by resolutions of that polytomy to the set X. ASTRAL-II computes a single "reference" tree by computing a greedy consensus of all gene trees and forcing the consensus to be fully resolved with further refinements using the UPGMA algorithm. To resolve a gene tree polytomy, it samples a taxon from each cluster defined by each side of the polytomy, finds the reference tree induced on the sampled taxa, and adds the resulting resolution to the search space. In ASTRAL-III, the definition of the reference tree is modified to use the UPGMA tree inferred on the similarity matrix used by ASTRAL. We observed that the UPGMA tree summarizes the input gene trees more accurately than the greedy trees (Table S2.1). Moreover, unlike ASTRAL-II, in ASTRAL-III, this process is repeated three times with different random samplings.

Algorithm S2.1. Additions to X using greedy consensus. $greedy(\mathscr{G},t,b)$ returns the greedy consensus of $\mathscr{G}$, including only branches with frequency $\geq t$; if b is true, polytomies in the consensus are randomly resolved. $updateX(t)$ adds bipartitions from tree t to the set X; when edges in t are labelled with a frequency label (e.g., frequencies in the greedy consensus), it returns the maximum label of any *new* bipartition added to X. $clusters(p)$ returns the taxon partitions defined by an unrooted node p. $upgma(S,C)$ runs the UPGMA algorithm using the similarity matrix S; when C is given, UPGMA starts by groups defined in C. $randSample(p)$ selects a random taxon from each subtree around a node p, and $resolve(p,r)$ resolves polytomy p according to a tree r on such a sampling. Operator $\restriction$ restricts a tree or a matrix to a subset. $pectinate(O)$ returns a pectinate tree based on O, an ordered list of taxa. *sortBy* sorts a list of taxa based on their decreasing similarity to a given taxon. Constants: $THS = \{0, \frac{1}{100}, \frac{1}{50}, \frac{1}{20}, \frac{1}{10}, \frac{1}{4}, \frac{1}{3}\}$; $MIT = 10$; $RWD = 2$; and $FRQ = LTH = \frac{1}{100}$; $MAXR = 100$.

function ADDBYGREEDY($\mathscr{G},S$)
  for $t \in THS$ **do**
    $gc \leftarrow greedy(\mathscr{G},t,False)$
    for $p \in polytomies(gc)$ **do**
      if $degree(p) \geq POLYLIMIT$ **then**
        $quadratic \leftarrow FALSE$
      else
        $quadratic \leftarrow True$
      $updateX(upgma(S, start = clusters(p)))$
      $c \leftarrow 0$ and $max \leftarrow MIT$
      while $c < max$ **do**
        $c \leftarrow c+1$
        $sample \leftarrow randSample(p)$
        $r \leftarrow greedy(\mathscr{G} \restriction sample, 0, True)$
        $mt \leftarrow updateX(resolve(p,r))$
        if $mt \geq FRQ$ AND $max \leq MAXR$ **then**
          $max \leftarrow max + RWD$
        $updateX(resolve(p, upgma(S \restriction sample)))$
        if $t \leq LTH$ and $c < MIT$ and $quadratic$ **then**
          for $s \in sample$ **do**
            $r \leftarrow pectinate(sortBy(S,s,sample)$
            $updateX(resolve(p,r))$

2.B Derivations

2.B.1 Derivation of Equation 2.6

First note that:

$$\begin{aligned} QI((A|B|C),M) &= \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \frac{a_i+b_j+c_k-3}{2} a_i b_j c_k \\ &= \sum_{i\in[d]} \binom{a_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} b_j c_k \\ &\quad + \sum_{i\in[d]} \binom{b_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} a_j c_k \\ &\quad + \sum_{i\in[d]} \binom{c_i}{2} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} a_j b_k \,. \end{aligned} \tag{2.9}$$

Now, we note that:

$$\begin{aligned} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} b_j c_k &= \sum_{j\in[d]-\{i\}} b_j \sum_{k\in[d]-\{i,j\}} c_k \\ &= \sum_{j\in[d]-\{i\}} b_j (S_c - c_i - c_j) \\ &= -b_i(S_c - c_i - c_i) + \sum_{j\in[d]} b_j (S_c - c_i - c_j) \\ &= 2b_i c_i - S_c b_i + S_b S_c - S_b c_i - S_{b,c} \\ &= (S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i \end{aligned} \tag{2.10}$$

Replacing this (ditto for other terms) in Equation 2.9 directly gives us the Equation 6:

$$\begin{aligned} QI((A|B|C),M) &= \sum_{i\in[d]} \binom{a_i}{2} ((S_b - b_i)(S_c - c_i) - S_{b,c} + b_i c_i) \\ &\quad + \sum_{i\in[d]} \binom{b_i}{2} ((S_a - a_i)(S_c - c_i) - S_{a,c} + a_i c_i) \\ &\quad + \sum_{i\in[d]} \binom{c_i}{2} ((S_a - a_i)(S_b - b_i) - S_{a,b} + a_i b_i) \end{aligned} \tag{2.11}$$

2.B.2 Derivation of the upper bound $U(Z)$

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

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

where

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

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

$$
\begin{aligned}
&\sum_{\{A,B,C\}\in Q(Z)} QI((A|B|C),M) \\
&= \sum_{\{A,B,C\}\in Q_1(Z)} QI((A|B|C),M) + \sum_{\{A,B,C\}\in Q_2(Z)} QI((A|B|C),M) \\
&= \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i\}-[j]} \binom{l_i}{2} z_j z_k \\
&\quad + \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i\}-[j]} \binom{z_i}{2} (z_j(l_k-z_k)+(l_j-z_j)z_k) \\
&= \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \binom{l_i}{2} \frac{z_j z_k}{2} \\
&\quad + \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \binom{z_i}{2} \frac{z_j(l_k-z_k)+(l_j-z_j)z_k}{2} \\
&= \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \binom{l_i}{2} \frac{z_j z_k}{2} \\
&\quad + \sum_{i\in[d]} \sum_{j\in[d]-\{i\}} \sum_{k\in[d]-\{i,j\}} \binom{z_i}{2} z_j(l_k-z_k)\,.
\end{aligned}
\tag{2.12}
$$

Notice that based on Equation 2.4,

$$
\begin{aligned}
&\frac{QI((Z|Z|L),M)}{2} - \frac{QI((Z|Z|Z),M)}{3} \\
&\quad = \frac{1}{2}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} z_i z_j l_k \frac{z_i+z_j+l_k-3}{2} \\
&\qquad - \frac{1}{3}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} z_i z_j z_k \frac{z_i+z_j+z_k-3}{2} \\
&\quad = \frac{1}{2}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} \left(\binom{z_i}{2} z_j l_k + z_i \binom{z_j}{2} l_k + z_i z_j \binom{l_k}{2}\right) \\
&\qquad - \frac{1}{3}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} \left(\binom{z_i}{2} z_j z_k + z_i \binom{z_j}{2} z_k + z_i z_j \binom{z_k}{2}\right) \\
&\quad = \frac{1}{2}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} \left(\binom{z_i}{2} z_j l_k + \binom{z_i}{2} z_j l_k + \binom{l_i}{2} z_j z_k\right) \\
&\qquad - \frac{1}{3}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} \left(\binom{z_i}{2} z_j z_k + \binom{z_i}{2} z_j z_k + \binom{z_i}{2} z_j z_k\right) \\
&\quad = \frac{1}{2}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} \left(\binom{l_i}{2} z_j z_k + 2\binom{z_i}{2} z_j l_k\right) \\
&\qquad - \frac{1}{3}\sum_{i\in[d]}\sum_{j\in[d]-\{i\}}\sum_{k\in[d]-\{i,j\}} 3\binom{z_i}{2} z_j z_k \\
&\quad = \sum_{A,B,C\in Q(Z)} QI((A|B|C),M)\ .
\end{aligned}
\tag{2.13}
$$

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

Therefore,

$$
\begin{aligned}
U(Z) &= \sum_{g\in G}\sum_{M\in N(g)} \left(\frac{QI((Z|Z|L),M)}{2} - \frac{QI((Z|Z|Z),M)}{3}\right) \\
&= \frac{w(Z|Z|L)}{2} - \frac{w(Z|Z|Z)}{3}\ .
\end{aligned}
\tag{2.14}
$$

2.C Simulations and commands

2.C.1 Simulation setup

S100

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

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

Larege-*n* simulated dataset

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

In order to generate the gene trees and true species trees using the Simphy we use parameters given in Table S2.4 and the following command.

1K:

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

2K:

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

5K:

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

10K: For the 10K-taxon dataset of S2 we use this command

```
simphy -rs 20 -rl f:1000 -rg 1 -sb lu:0.0000001,0.000001 -sd lu:0.0000001,sb
-st ln:16.2,1 -sl f:10000 -so f:1 -si f:1 -sp u:10000,1000000 -su
ln:-17.27461,0.6931472 -hh f:1 -hs ln:1.5,1 -hl ln:1.551533,0.6931472 -hg
ln:1.5,1 -cs 9644 -v 3 -o 10k.species -ot 0 -op 1 -od 1
```

2.C.2 Commands

Contracting branches

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

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

Drawing bootstrap support on ML gene trees:

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

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

Then we draw bootstrap supports on the branches:

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

Gene tree estimation

We used FastTree version 2.1.9 Double precision. In order to estimated best ML gene trees we used the following command: `fasttree -nt -gtr -nopr -gamma -n <num> <all-genes.phylip>` where we have all the alignments in the PHYLIP format in the file all-genes.phyip for each replicate, and $< num >$ is the number of alignments in this file.

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

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

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

Running ASTRAL

ASTRAL-II in this paper refers to ASTRAL version 4.11.2 and ASTRAL-III refers to ASTRAL version 5.5.4. Both versions can be found in the link below:

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

Both versions of ASTRAL program were run with following command:

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

2.D Supplementary Figures and Tables

Table S2.1. The accuracy of UPGMA tree and Greedy tree of two model conditions of dataset S100

Contraction threshold	Greedy tree RF	UPGMA tree RF
0%	0.168	0.1461
10%	0.169	0.1451

Table S2.2. Species tree and gene tree generation parameters used for Simphy, and sequence evolution parameters for the GTR model used for Indelible for the S100 dataset.

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

Table S2.3. Species tree error (FN ratio) for all model conditions of the S100 dataset, with true gene trees (*true*), no filtering (*non*), and all filtering thresholds (*columns*).

Genes	Alignment	true	non	0	3	5	7	10	20	33	50	75
50	200bp	7.0	17.4	**15.7**	16.1	16.0	16.1	16.8	16.9	19.0	22.9	31.4
50	400bp		13.4	13.2	**12.8**	**12.8**	13.0	**12.8**	13.6	14.3	16.4	20.7
50	800bp		12.0	11.7	11.3	11.1	11.0	11.0	**10.9**	11.7	12.4	15.4
50	1600bp		10.2	10.2	10.1	10.1	**9.8**	9.9	10.0	10.0	10.4	11.9
200	200bp	3.7	11.3	10.4	**10.1**	10.3	10.3	10.4	10.3	12.1	14.3	20.5
200	400bp		9.0	8.3	8.3	**8.0**	**8.0**	8.2	8.3	8.8	9.8	12.9
200	800bp		7.4	7.2	6.9	6.9	6.9	**6.8**	6.9	7.2	7.5	8.9
200	1600bp		6.1	6.3	6.2	6.2	6.1	6.1	**5.9**	6.1	6.2	7.3
500	200bp	2.4	9.5	8.5	8.4	8.5	**8.1**	**8.1**	**8.1**	9.4	10.9	15.7
500	400bp		7.1	6.7	6.4	**6.3**	6.5	**6.3**	6.5	6.7	7.7	9.9
500	800bp		5.7	5.4	5.3	5.4	5.2	5.3	**5.1**	**5.1**	5.6	6.4
500	1600bp		4.8	4.6	4.5	4.5	**4.3**	4.4	4.4	**4.3**	4.5	5.0
1000	200bp	1.5	8.8	7.8	7.2	7.2	7.0	**6.9**	7.1	7.9	9.2	12.5
1000	400bp		6.7	5.9	5.5	5.6	5.3	5.4	**5.2**	5.4	6.3	7.9
1000	800bp		5.3	4.9	4.8	4.7	4.8	4.7	**4.3**	4.4	4.5	5.4
1000	1600bp		4.1	4.2	4.0	3.8	3.9	**3.7**	**3.7**	3.8	3.8	4.1

Table S2.4. Species tree and gene tree generation parameters in Simphy for 1K-taxon, 2K-taxon and 5K-taxon datasets

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

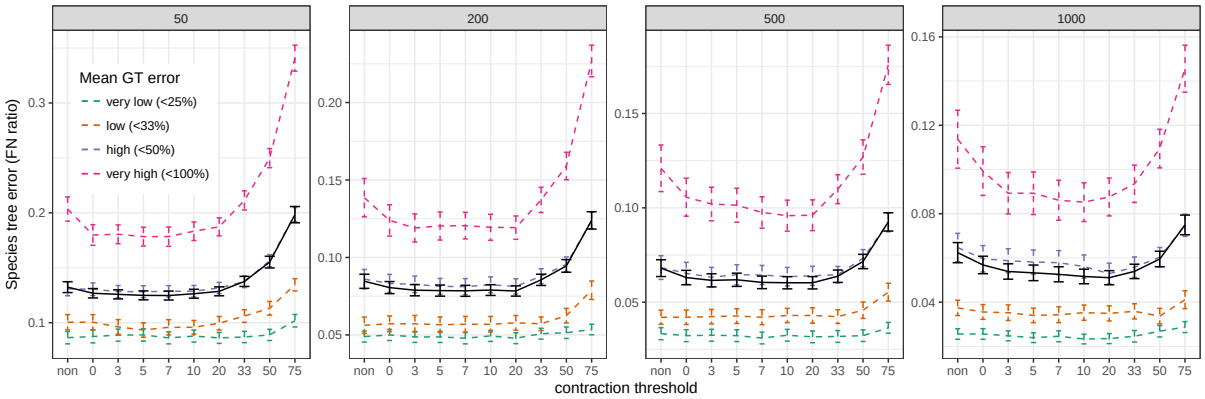

Figure S2.1. Impact of contraction on the S100 dataset. The error in species trees estimated by ASTRAL-III on the S100 dataset given $k = 50$, 200, 500, or 1000 genes (*boxes*) and with full FastTree gene trees (*non*) or trees with branches with $\leq \{0,3,5,7,10,20,33,50\}\%$ support contracted (*x-axis*). Average FN error and standard error bars are shown for all 50 replicates with the four alignment lengths combined (*black solid line*); average FN error broken down by gene tree error is also shown (*dashed colored lines*). We divide the replicates based on their average gene tree error (normalized RF) into four categories: $[0, \frac{1}{4}]$,$(\frac{1}{4}, \frac{1}{3}]$,$(\frac{1}{3}, \frac{1}{2}]$,$(\frac{1}{2}, 1]$.

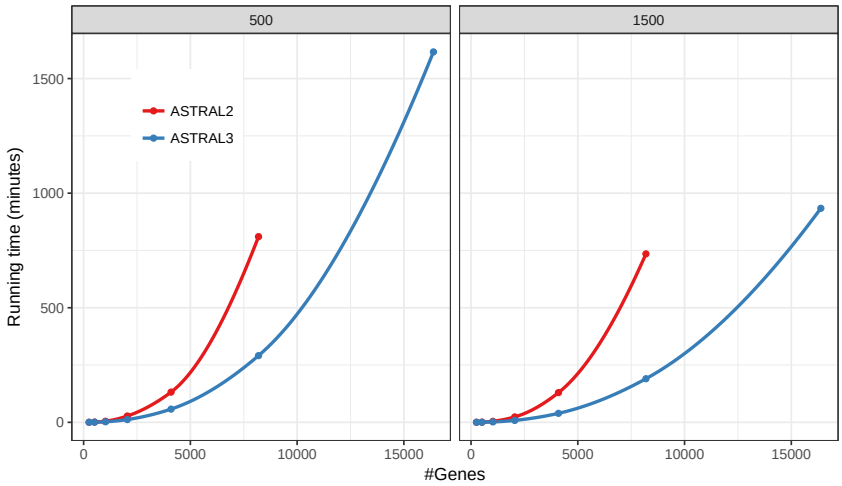

Figure S2.2. Running time versus k. Average running time of ASTRAL-II versus ASTRAL-III on the avian dataset with 500bp or 1500bp alignments with varying numbers of gens (k), shown in normal scale. A LOESS curve is fit to the data points. ASTRAL-II could not finish on 2^{14} genes in the allotted 48-hour time slot. Averages are over 4 runs.

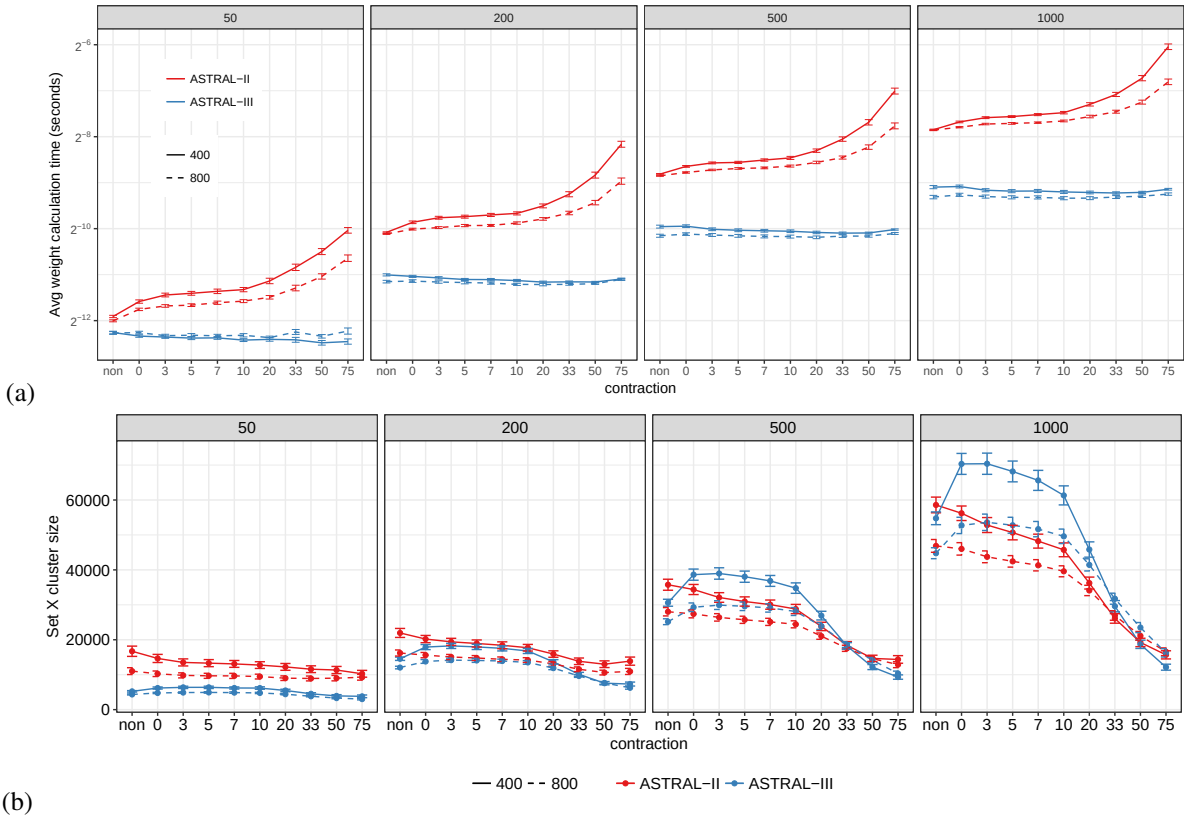

Figure S2.3. Weight calculation and $|X|$ on S100. Average and standard error of (a) the time it takes to score a single tripartition using Eq. 3 and (b) search space size $|X|$ for both ASTRAL-II (red) and ASTRAL-III (blue) on the S100 dataset. Running time is in log scale for varying numbers of gene trees (*boxes*) and sequence length 400 and 800 (*line types*).

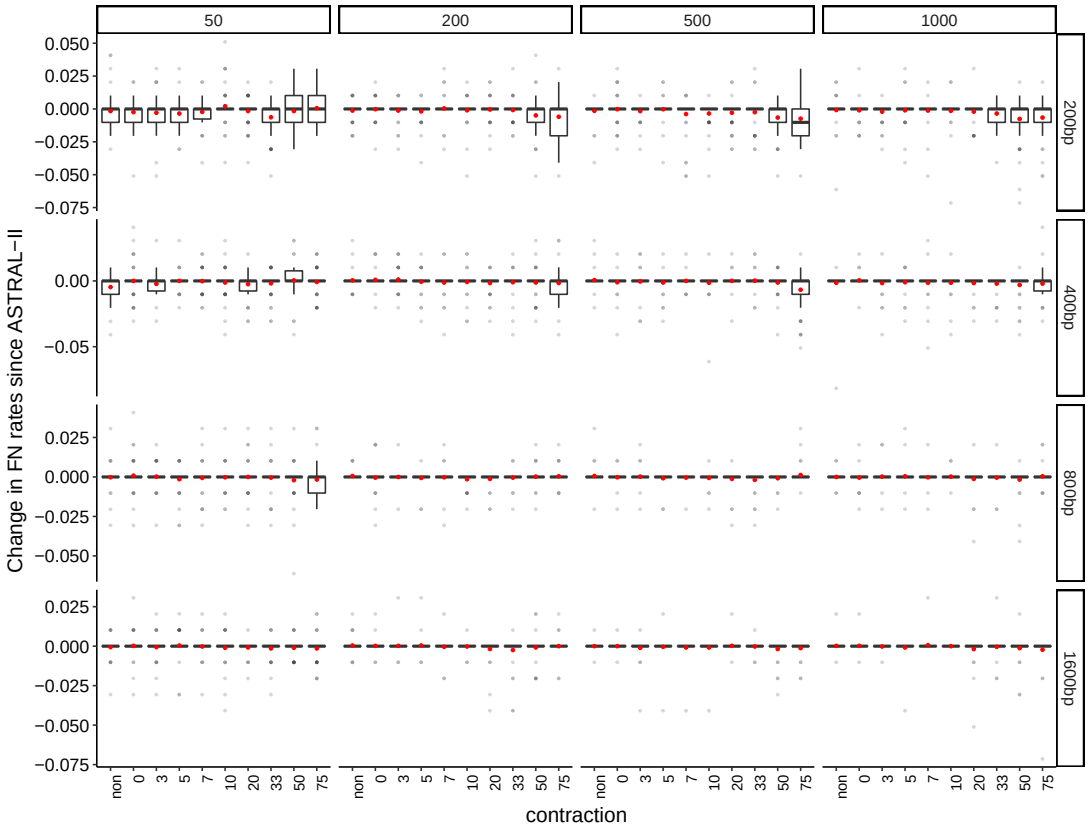

Figure S2.4. Change in species tree FN rates between ASTRAL-II and ASTRAL-III (ASTRAL-III − ASTRAL-II) for S100 dataset varying number of genes, number of base pairs, and contraction levels. Negative values indicate improvements over ASTRAL-II.

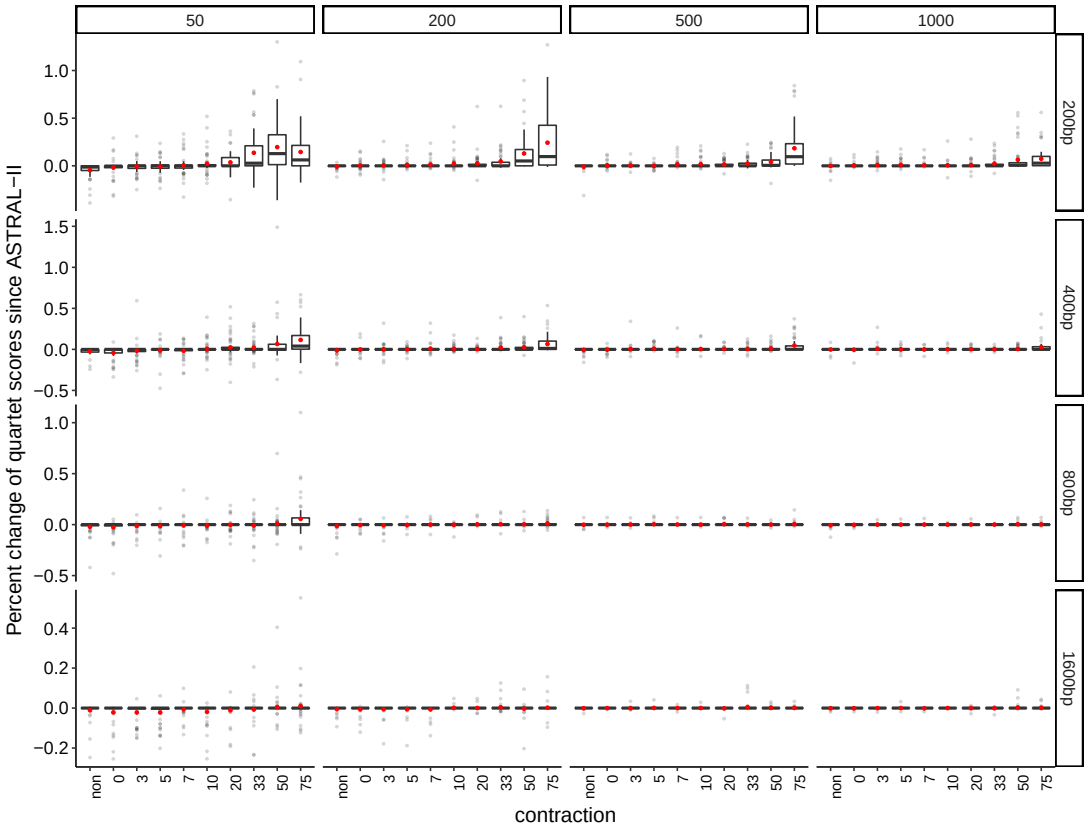

Figure S2.5. Percent change in species tree quartet scores between ASTRAL-II and ASTRAL-III ($\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$) for S100 dataset varying number of genes, number of base pairs, and contraction levels. Positive values indicate improvements over ASTRAL-II.

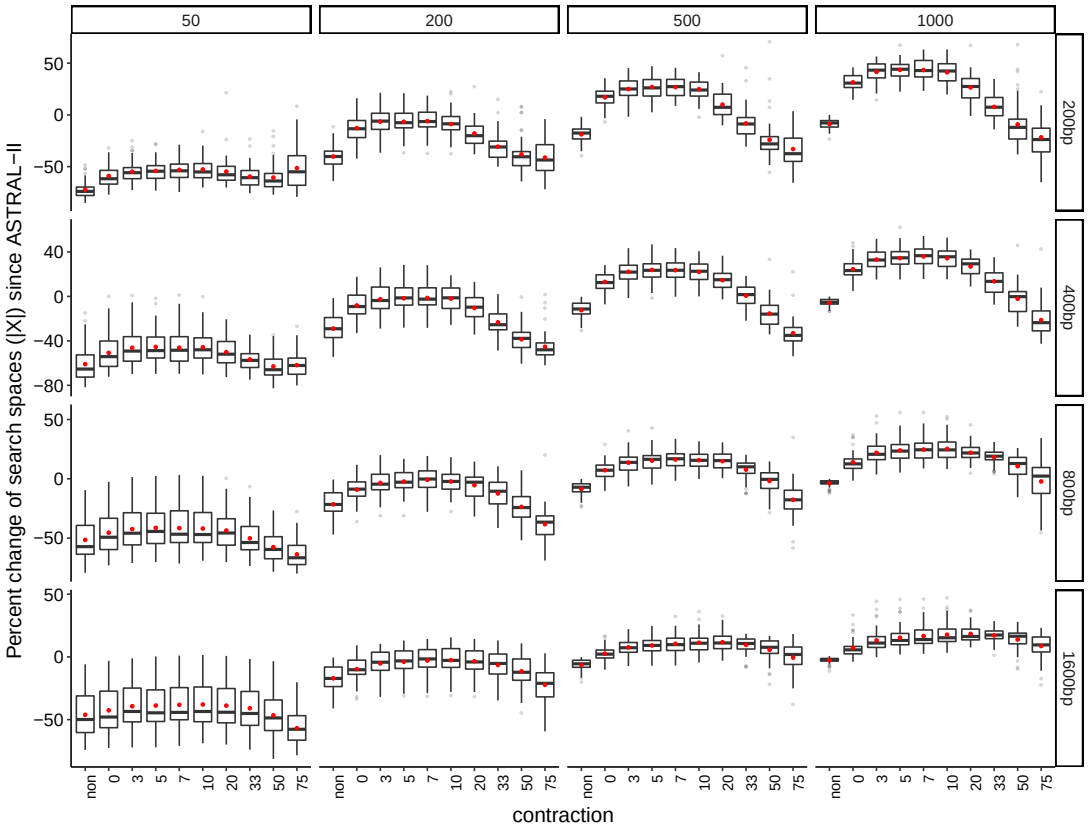

Figure S2.6. Percent change in species tree search space ($|X|$) between ASTRAL-II and ASTRAL-III ($\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$) for S100 dataset varying number of genes, number of base pairs, and contraction levels. Positive values indicate larger search space over ASTRAL-II.

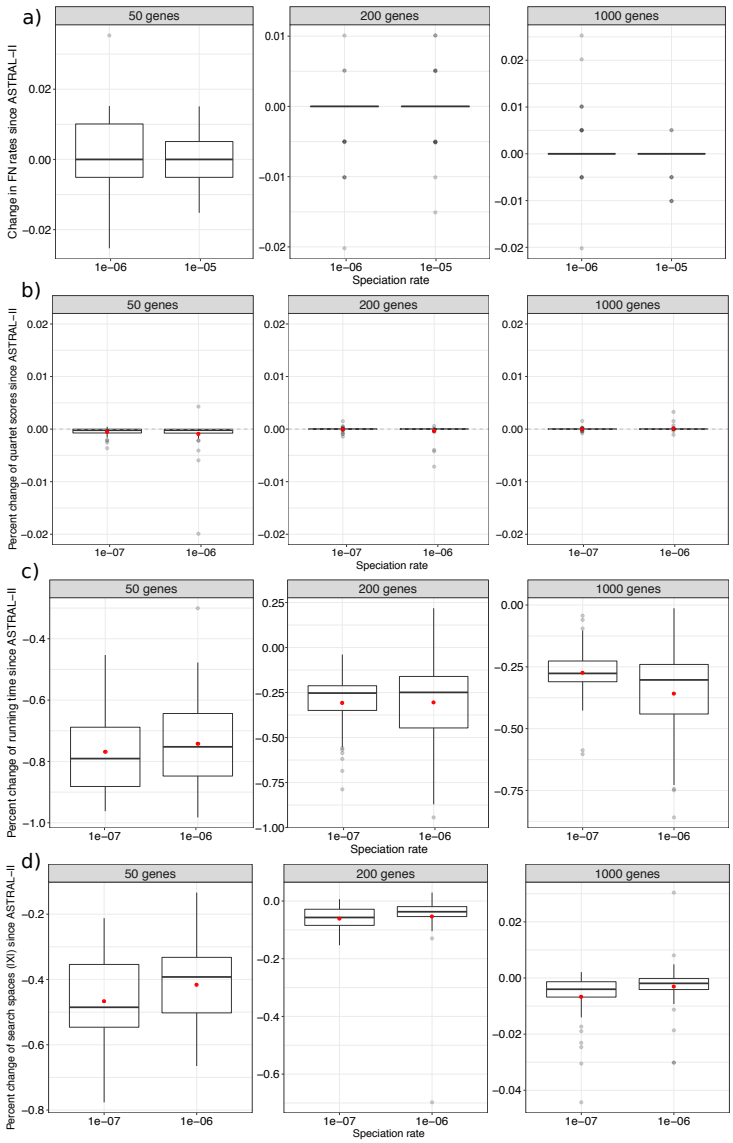

Figure S2.7. (a) Change in species tree FN rates between ASTRAL-II and ASTRAL-III (ASTRAL-III − ASTRAL-II) for S200 dataset. Negative values indicate improvements over ASTRAL-II. (b) Percent change in species tree quartet scores between ASTRAL-II and ASTRAL-III ($\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$) for S200 dataset. Positive values indicate improvements over ASTRAL-II. (c) Percent change in running time between ASTRAL-II and ASTRAL-III ($\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$) for S200 dataset. Positive values indicate longer running times over ASTRAL-II. (d) Percent change in species tree search space ($|X|$) between ASTRAL-II and ASTRAL-III ($\frac{\text{ASTRAL-III}-\text{ASTRAL-II}}{\text{ASTRAL-II}} \times 100$) for S200 dataset. Positive values indicate larger search space over ASTRAL-II.

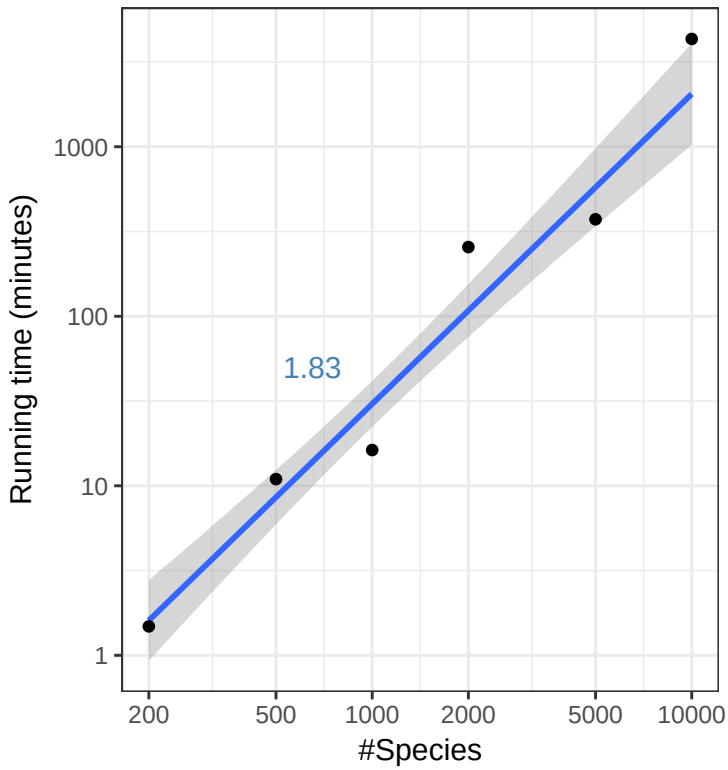

Figure S2.8. Empirical running time of ASTRAL-III with n. Average running time is shown for ASTRAL-III for datasets with varying n. Averages are over 20 replicates. One replicate of 2000 species dataset could not finish in 2 days and is removed from the analysis. Note that these datasets have factors other than n that change as well (e.g., the amount of ILS, etc.). Thus, these running times should be treated as ball-park estimates. Finally, we note that on the 10,000 dataset, we have only 2 replicates and not 20.

Chapter 3

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

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 the original ASTRAL. 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.

3.1 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 (Maddison, 1997; Degnan and Rosenberg, 2009) 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 (Smith and Hahn, 2021; Elworth et al., 2019). 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 (Smith et al., 2015; Reddy et al., 2017; Shen et al., 2017; Walker et al., 2018; Gatesy et al., 2019), 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 (Patel, 2013; Mirarab et al., 2014a; 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 (Leaché and Rannala, 2011; Knowles et al., 2012). 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 et al., 2009; Mossel and Roch, 2010; Liu et al., 2010; Liu and Yu, 2011; 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 (Wang et al., 2020; Alanjary et al., 2019). 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 the gene duplication and loss (GDL) model (Legried et al., 2021; Yan et al., 2021), some horizontal gene transfer models (Davidson et al., 2015), and combined models of ILS and GDL (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; Molloy and Warnow, 2018; Lanier and Knowles, 2015; 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. The idea of weighting can be trace back to Farris (1969). However, weighting by branch support has not yet been incorporated into existing summary methods such as ASTRAL for several reasons. *i*) Quartet weights must be implicitly calculated, as explicitly examining all quartets of n species alone will take $\Theta(n^4)$ time. The existing general (e.g., Avni et al., 2015) and MSC-based weighted quartet methods (Yourdkhani and Rhodes, 2020; Richards and Kubatko, 2021) require weights *explicitly* calculated for every quartet, making them less scalable with n. The reason ASTRAL can scale to a large number of species is that it optimizes a score defined over all quartets without explicitly examining them. Designing a scalable weighting method will require weights that can be implicitly computed based on examining $O(n)$ gene tree branches. *ii*) It is difficult to design efficient algorithms to optimize a weighted score. Unless weights satisfy certain properties, it may not be possible to find an algorithm better than $O(n^4)$ 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 (wASTRAL-bl), branch

support values (wASTRAL-s), or both (wASTRAL-h). 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 entirely 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, accuracy on simulated data, and the accuracy of its branch support values. Weighted ASTRAL is more accurate than CA-ML in our simulations except when there is a large number of inaccurate gene trees or low levels of discordance, where concatenation is slightly more accurate. Most interestingly, weighted ASTRAL is more congruent than the original ASTRAL with concatenation on real datasets.

3.2 Result

3.2.1 Weighted ASTRAL algorithm

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. 3.1a).

Weighting by support

extends the definition of branch support to a quartet. Let $\mathscr{P}$ be the set of branches on the path between internal nodes of a quartet tree (also called anchors; orange dots in Fig. 3.1a) and let $s(e)$ denote the support of a branch e. We define the support of the quartet as

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

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. While we study this goal in our simulation and empirical analyses, we also provide some theoretical results.

Making theoretical statements about estimated gene trees is difficult because we lack an accepted way of modeling gene tree estimation errors. To be able to interrogate theoretical properties of weighted ASTRAL, we propose a simple model of gene tree estimation error called MSC+Error (Material and Methods). In this model, for any true gene tree topology on a quartet Q, the estimated topology is drawn from a distribution that has two features: first, each gene G has a gene-specific level of signal, controlled by a parameter $\alpha_{G,Q}$, and second, all genes can be adversarially biased towards any topology by an amount bounded by a parameter called β_Q. The joint distribution of true and estimated quartet gene trees in the most difficult case can be expressed as a function of $\alpha_{G,Q}$ and β_Q as well as $\theta_Q = 1 - e^{-d}$ where d is the coalescent unit (CU) length of the internal branch of the quartet (Table 3.1 and Fig. 3.1b). Under the MSC+Error model, the distribution of quartet gene tree topologies, written as a vector with the first element corresponding to the species tree, changes (in the worst case) from

$$\frac{1}{3}\begin{bmatrix} 1+2\theta_Q \\ 1-\theta_Q \\ 1-\theta_Q \end{bmatrix} \quad \text{for true gene trees to} \quad \frac{1}{3}\alpha_{G,Q}\begin{bmatrix} 1+2\theta_Q \\ 1-\theta_Q \\ 1-\theta_Q \end{bmatrix} + \frac{1}{3}(1-\alpha_{G,Q})\begin{bmatrix} 1-\beta_Q \\ 1+\beta_Q \\ 1 \end{bmatrix}$$

for estimated gene trees.

The estimated gene tree distribution matches the MSC model when $\alpha_{G,Q} = 1$ and is uniformly random when $\alpha_{G,Q} = \beta_Q = 0$. A choice of $\alpha_{G,Q} < 1$ adds noise to the MSC probabilities,

and any $\beta_Q > 0$ creates an adversarial bias towards the second topology (Fig. 3.1b). Because noise and bias parameters can change across genes and quartets, the MSC+Error model is very general and makes minimal assumptions.

Under the MSC+Error model, the original ASTRAL is statistically consistent with estimated gene trees under limited choices of $\alpha_{G,Q}$ and β_Q. Assuming that the support of a quartet matches the estimated gene tree distribution, we can get our main result. Theorem 3.1 in Material and Methods proves that support-weighted ASTRAL (wASTRAL-s) is statistically consistent under a strictly larger super-set of $\alpha_{G,Q}$ and β_Q parameters than those of unweighted ASTRAL. Thus, there are levels of bias in gene tree estimation (e.g., due to long branch attraction) that, combined with low signal, render unweighted ASTRAL inconsistent (as shown by Roch et al. 2019) but keep wASTRAL-s consistent.

Examining the marginal probabilities and expected weights can illuminate the reason behind the advantage of wASTRAL-s (Fig. 3.1b). First, gene trees with higher levels of noise (i.e., lower $\alpha_{G,Q}$) are down-weighted relative to gene trees with less noise (Fig. 3.1b: note lighted colors as α decreases). Thus, the correct topology benefits from summing weights over gene trees with different $\alpha_{G,Q}$. For example, assume some genes have high noise, and others have low noise following the $\alpha_{G,Q}$ distribution shown in Figure 3.1c. The less noisy genes will be up-weighted such that wASTRAL-s becomes consistent even when unweighted ASTRAL is not (Fig. 3.1d). Second, unless gene trees are extremely noisy (i.e., very low $\alpha_{G,Q}$), wASTRAL-s down-weights the species tree topology less than the other two topologies; in extreme cases, we have scenarios (Fig. 3.1b, bottom, highlighted boxes) where the species tree is dominant with weighted scores but not with unweighted scores. In fact, for fixed α and β, there exists a range of CU quartet internal branch lengths for which ASTRAL is not consistent but wASTRAL-s is (Fig. 3.1e).

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 provided in substitution-per-site units (SU). We assign e^{-L} as the weight of the quartet and offer two justifications. First, deeper coalescence events tend to generate 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 3.2). 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 3.2. 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 (Warnow et al., 2001; Nelesen et al., 2012). Our weighting scheme naturally achieves the same impact by down-weighting long quartets versus short quartets around difficult species tree branches (Fig. 3.1a).

Hybrid weighting

combines both weighting schemes where each quartet is assigned with weight

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

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

While defining weighting schemes is easy, designing scalable algorithms to optimize the weighted quartet score is not. Adopting the existing ASTRAL algorithm to incorporate per-quartet weights is challenging for reasons elaborated in Material and Methods. A major contribution of this paper is designing a set of algorithms (Algorithm S3.1–S3.3) to optimize

the weighted quartet using a set of new techniques paired with a dynamic programming (DP) step similar to ASTRAL. We leave the detailed description of the algorithm to the Optimization algorithm section; esp., see Theorems 3.3, 3.4, and 3.6 for correctness and Theorem 3.5 for the asymptotic running time being $O(kn^{1.5+\varepsilon}H)$ where H is the average gene tree height.

3.2.2 Simulation results

Comparison of weighting schemes

We start by comparing the accuracy of weighting schemes and branch support types on two simulated datasets (S100 and S200). 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.

This dataset adopted from Zhang et al. 2018 has gene trees inferred from sequences with varying lengths resulting in various levels of gene tree error (see Datasets). In most cases, weighting by support (wASTRAL-s) produces species trees with higher accuracy than weighting by length (wASTRAL-bl), and the improvements are statistically significant (Fig. S3.1); p-value $< 10^{-15}$ according to a repeated-measure ANOVA test (see Statistical tests). The improvement in accuracy varies with k ($p < 10^{-15}$) and perhaps sequence length ($p \approx 0.04$). 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 may depend on k ($p \approx 0.06$) and sequence length ($p \approx 0.03$). With ≥ 500 genes, wASTRAL-h is better than *both* support and length, showing that combining the two weightings makes wASTRAL-h more powerful.

On this dataset, bootstrap support computed using FastTree-2 is provided by Zhang et al. 2018. Thus, we also compute weighted ASTRAL trees using bootstrap supports (wASTRAL-s* and wASTRAL-h*). For weighting by support, aBayes weighting is much better than bootstrap

weighting ($p < 10^{-15}$), but the gap in error significantly ($p < 10^{-9}$ for both) shrinks as k and sequence length increase (Fig. S3.1). For hybrid weighting, aBayes weighting is, on average, only slightly better than bootstrap weighting (the mean error increases across all conditions by only 0.2%).

S200.

This 200-taxon dataset has species trees sampled under two birth rates ($10^{-6}, 10^{-7}$), which control whether speciations are dispersed at random or closer to the tips (Fig. S3.2), 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-2 is available, which we use (wASTRAL-s* and wASTRAL-h*). Patterns of accuracy across wASTRAL versions are similar to S100 (Fig. S3.3) as wASTRAL-h is more accurate than wASTRAL-s on all model conditions ($p < 10^{-6}$), and the improvements depend on k ($p \approx 10^{-4}$), ILS level ($p < 10^{-7}$), and birth rate ($p < 10^{-10}$). Using SH-like support with wASTRAL-h is, on average worse than aBayes support, increasing the error by 9%.

Comparison of topological accuracy to other methods

We next compare wASTRAL-h, the most accurate version of wASTRAL, to other methods.

Impact of gene tree estimation error (S100 dataset).

On the S100 dataset (Fig. 3.2a and S3.4), wASTRAL-h is more robust to gene tree estimation error than ASTRAL-III, regardless of whether low bootstrap support (BS) branches ($\leq 5\%$) are contracted. While contracting low support branches improves the accuracy of ASTRAL-III, weighting improves accuracy even more. For example, the average error with 1000 200bp genes goes down from 9% with ASTRAL-III to 7% after contracting $\leq 5\%$ BS branches and 6% with wASTRAL-h. While wASTRAL-h dominates ASTRAL-III in all conditions with or without contraction ($p < 10^{-15}$), the difference in accuracy varies across sequence

lengths ($p < 10^{-6}$ without contraction and $p \approx 0.003$ with contraction). Similar to wASTRAL-h, wASTRAL-h* has mean error lower than that of ASTRAL-III-5% in every condition ($p < 10^{-11}$).

The clearest patterns are observed when comparing wASTRAL-h and concatenation using ML performed using ML (CA-ML). While increasing the sequence length (and hence reducing the gene tree error) dramatically reduces the error of all ASTRAL variants, it has a much more subdued impact on CA-ML. As a result, the relative accuracy significantly depends on k ($p < 10^{-15}$) and gene sequence length ($p < 10^{-9}$) and the choice of the best method varies across conditions. Generally, wASTRAL-h tends to be more accurate than CA-ML under smaller k and greater sequence lengths. With $k \leq 200$, wASTRAL-h dominates CA-ML for all sequence lengths. With $k > 200$, CA-ML is better for smaller gene alignments, and wASTRAL-h is better for longer alignments, with the only conditions when CA-ML has noticeable improvements over wASTRAL-h corresponding to 200bp genes.

Impact of ILS level (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. 3.2b and S3.5) and the improvements are significant ($p < 10^{-15}$). The improvements of wASTRAL-h compared to ASTRAL-III increase with more gene trees ($p \approx 7 \times 10^{-4}$) but appear to decrease with more ILS ($p \approx 0.08$). While Mirarab and Warnow 2015 reported no improvement in accuracy when contracting branches with low SH-like support, contracting branches with aBayes support $< 90\%$ (ASTRAL-III-90%) does improve accuracy. Nevertheless, wASTRAL-h has yet lower error ($p < 10^{-5}$). Also, improvements of wASTRAL-h are significantly larger for the 10^{-7} birth rates, which tend to have earlier speciations (Fig. S3.2), than the 10^{-6} rate ($p \approx 1.5 \times 10^{-5}$).

The comparison between wASTRAL-h and CA-ML significantly depends on several factors (birth rate: $p < 10^{-7}$; ILS: $p < 10^{-15}$; k: $p < 10^{-11}$). 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, with low ILS and birth rate $= 10^{-6}$ (more recent speciation),

wASTRAL-h is better than CA-ML ($p \approx 1.7 \times 10^{-5}$) while with low ILS and birth rate $= 10^{-7}$ (earlier speciation), CA-ML is better ($p < 10^{-11}$). Thus, in some conditions with low enough 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

We next test whether, by accounting for gene tree uncertainty, wASTRAL improves support values computed using the local Posterior Probability (PP) measure (see Branch support). We examine the calibration of support (i.e., whether the support matches the probability of correctness of a branch), its ability to distinguish correct and incorrect branches examined through Receiver operating characteristic (ROC) curves, and distributions of support (see Evaluation criteria).

S100.

While wASTRAL-h generally gives higher support values than ASTRAL-III (Fig. S3.7), it has fewer cases of highly supported incorrect branches, especially with higher k and shorter sequences (Fig. 3.3a). For both ASTRAL-III and wASTRAL-h, while increased support often leads to increased frequency of correctness (Fig. 3.3b), support under-estimation or over-estimation can also be observed for certain sequence length and k combinations. For example, wASTRAL-h has a tendency to overestimate for large k values and short sequences. In terms of predictive power, for any desired false positive rate (FPR), the recall of wASTRAL-h is as good as or better than ASTRAL-III in all conditions (Fig. 3.3c), though the improvements in ROC can be 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.

S200.

Support values on the S200 dataset exhibit similar patterns to S100 (Fig. 3.3d-f). The most notable difference is that when $k = 1000$, wASTRAL-h has a clear advantage over ASTRAL-III in trading off precision and recall according to ROC curves (Fig. 3.3f and S3.9). This advantage shrinks as k decreases. Here, wASTRAL-h has a slight tendency to under-estimate support values < 1 (Fig. S3.10 and S3.11), and this tendency is most pronounced with 50 genes, high ILS level, and birth rate 10^{-6} (Fig. 3.3e and S3.11).

Comparison of the optimization algorithms

Assigning weights to quartets forced us to develop a new optimization algorithm, which can also be used for unweighted optimization. We next study whether the new optimization algorithm (denoted as DAC) is as effective as that of ASTRAL-III (denoted as A3) when no weights are used.

Testing on the S200 dataset, without missing data, DAC is in most cases slower than the A3 (Figs. 3.4a and 3.4c), a pattern that is pronounced with lower ILS levels. The change in relative running time with ILS levels is due to the dependence of the search space of A3 but not DAC on gene tree discordance levels (Zhang et al., 2018). In terms of accuracy, DAC and A3 are comparable for low and medium ILS levels (Fig. 3.4c). However, in the high ILS case, A3 is clearly better with only 50 genes, slightly better with 200 genes, and perhaps slightly worse with 1000 genes. Cases with reduced accuracy also have reduced quartet scores for the 50 genes scenario and high ILS (Fig. 3.4a), showing that A3 is preferable, especially with few gene trees. Thus, the improved accuracy of wASTRAL over ASTRAL-III is *despite* the fact that its DAC optimization algorithm is not always as effective as A3.

These patterns change when we add low levels of missing data by randomly removing 5% of leaves in each gene tree (Figs. 3.4b and 3.4d). DAC becomes closer to A3 in terms of running time in most cases and is even faster with high ILS and $k = 1000$ (Fig. 3.4b). Regarding accuracy, A3 and DAC are comparable in low and medium ILS levels (Fig. 3.4d). However, in

the high ILS case, the error of A3 is slightly less, comparable, and slightly higher with 50, 200, and 1000 genes, respectively. Substantial changes in accuracy are caused by changes in quartet scores (Fig. 3.4b). Thus, DAC is competitive or better than the A3 in the presence of even low levels of missing data found to varying degrees in biological datasets.

3.2.3 Biological data

We next study seven biological datasets (Datasets). On the canis dataset, which was the only input with at least 5 hours of running time for wASTRAL-h (Table S3.2), we also examine the running time.

OneKp

Overall, 47 out of 1175 (4%) branches change between the published ASTRAL-III tree and our wASTRAL-h tree. Most of these branches had low support in the ASTRAL-III tree (mean: 62%, max: 99%) but not in the wASTRAL-h tree (Fig. S3.12). OneKP Initiative 2019 focused most of their attention on 20 branches, corresponding to nine major evolutionary events that have been historically hard to resolve (e.g., early Eudicot diversification). Among 47 branches that change in wASTRAL-h, four of them are among the 20 focal branches. Beyond topological changes, the support values tend to increase in wASTRAL-h (Fig. 3.5a). In particular, all of the 20 focal branches that had less than full support in the original ASTRAL-III tree have increased support in the wASTRAL-h tree, leaving only four with support below 0.95 (as opposed to 12 branches with ASTRAL-III).

Significantly, all four focal branches that change from ASTRAL-III to wASTRAL-h become consistent with CA-ML, whereas the original ASTRAL-III tree was inconsistent with CA-ML. At the base of eudicots, Vitales (grapes) becomes sister to Santalales in wASTRAL-h tree with moderate support (0.87), which is consistent with CA-ML (Fig. 3.5b). Two branches in the so-called TUC clade also change: ASTRAL-III breaks down the class Ulvophyceae by uniting Bryopsidales with Chlorophyceae while wASTRAL-h recovers Ulvophyceae as sister to

Chlorophyceae, which is the traditional resolution and is in agreement with CA-ML. Finally, the early diversification of ferns differs between CA-ML and ASTRAL-III but is identical between CA-ML and wASTRAL-h. Thus, wASTRAL-h makes coalescent analyses more congruent with CA-ML for the focal branches.

Canis

On the canis dataset of Gopalakrishnan et al. 2018 that spans a relatively shallow time scale (many branches are among populations of the same species), the majority of branches of the ASTRAL-III tree are shorter than 0.1 CU (Fig. 3.5c). Despite that, 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. S3.13). The ASTRAL-MP tree (on 100k gene trees) is identical to the published consensus tree, while the wASTRAL-h tree (on 450k gene tees) differs from it in only one branch (i.e., placement of the Egyptian dogs).

The linear running time scaling of wASTRAL-h with respect to k enables us to analyze randomly sampled subsets of 1000–450000 genes (Fig. 3.5c). The shortest branches need very many genes to achieve universal full support. Using fewer genes (even as many as 100,000) always leaves at least one branch with less than 99% support. Since many of the shortest branches are within species, 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; the 21 longest branches have at least 99% support with as few as 1000 gene trees. Furthermore, wASTRAL-h is more scalable compared to ASTRAL-III with respect to the number of genes k (Fig. 3.5d). As Theorem 3.3 predicts, the running time of wASTRAL-h scales almost linearly with k, while ASTRAL-III scales close to quadratically (Fig. 3.5d and Fig. S3.14). ASTRAL-III fails to finish for $k \geq 2 \times 10^3$ within 24 hours, and ASTRAL-MP with 16 cores takes more than 36 hours for $k = 10^5$. By contrast, wASTRAL-h finishes on $k = 4.5 \times 10^5$ within 18 hours and 2 hours with one and 16 cores, respectively. Even when $k = 10^3$, ASTRAL-III takes $4\times$ more than wASTRAL-h due to the

high levels of gene tree discordance and abundance of missing data, both of which increase the running time of ASTRAL-III but not wASTRAL-h.

Avian

On the avian dataset, the wASTRAL-h tree fully agrees with the ASTRAL-III trees after contracting low support branches and is very similar to original trees published by Jarvis et al. 2014 based on CA-ML (only five branches differ) and statistical binning (only two branches differ). This is in contrast to the ASTRAL-III tree without contraction from Zhang et al. 2018, which is in conflict with strong results from the literature and other methods. 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. S3.15). Interestingly, the only branch that experiences a reduction in support, the placement of Caprimulgimorphae as sister to Telluraves (core land-birds), is a branch that disagrees with both the published CA-ML and statistical binning trees. Finally, four branches with 99-100% support in wASTRAL-h are found by all coalescent-based methods (wASTRAL-h, ASTRAL-III and binned MP-EST) but not CA-ML, possibly pointing to a consistent signal that can be recovered only using coalescent-based analyses.

Cetaceans

The wASTRAL-h tree (Fig. S3.16) is similar to ASTRAL-multi and CA-ML trees reported by McGowen et al. 2020 with only a few differences (three branches to ASTRAL-multi and four to CA-ML). Interestingly, wASTRAL-h agrees with CA-ML and earlier studies (McGowen et al., 2009) 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 break the monophyly of the genus Tursiops as *Tursiops truncatus* moves away from *Tursiops aduncus* and Stenella with high support. The question of the monophyly of Tursiops, supported by morphology, has been answered differently in two recent analyses and remains likely (Moura et al., 2020) but uncertain due to evidence for gene

flow Guo et al. (2022). Close to Tursiops is also the placement of the two *Stenella clymene* individuals, which is a known hybrid species evolved from *Stenella longirostris* and *Stenella coeruleoalba*. Interestingly, the two *Stenella clymene* individuals are placed apart, one as sister to *Stenella longirostris* and the other at the most recent common ancestor of *Stenella longirostris* and *Stenella coeruleoalba*. This placement is in contrast to CA-ML, which puts both individuals as sister to *Stenella longirostris*. Beyond Delphininae, two branches, the placements of *Orcinus orca* and *Neophocaena phocaenoides*, disagree with both ASTRAL-multi and CA-ML, but both branches have very low support in wASTRAL-h and cannot be trusted. These two are among 11 species where McGowen et al. 2009 used data from existing genomes and transcriptomes instead of their own targeted capture, and it is possible that differences in the analytical pipeline may have caused the low support in wASTRAL.

Insect datasets

On all three insect datasets, the differences between wASTRAL-h and ASTRAL-III are minimal and strictly 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 in two low support branches (Fig. S3.17). On the Lepidoptera dataset, only seven out of 200 branches differ between wASTRAL-h and ASTRAL-III, and all of these branches have support below 75% (Fig. S3.18). Across the tree, wASTRAL-h has 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. S3.19).

3.3 Discussion

We introduced a family of new weighting schemes for quartet-based species tree estimation, including weighting quartets by terminal branch length (wASTRAL-bl), internal branch support (wASTRAL-s), or both (wASTRAL-h). We saw that the combined method (wASTRAL-

h) has the best accuracy among the three and dominates unweighted ASTRAL in terms of accuracy. We next further comment on more subtle patterns observed in the data and end by pointing out directions for future research.

3.3.1 Further observations based on the results

The choice between CA-ML and summary methods has been a long-standing debate (Simmons and Gatesy, 2015; Giarla and Esselstyn, 2015; Leaché et al., 2015; Edwards et al., 2016; Meiklejohn et al., 2016). While CA-ML is inconsistent under MSC (Roch and Steel, 2015), the most careful simulation studies have found that the best method depends on the dataset: CA-ML 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 in conditions with lower ILS or heightened gene tree error (Fig. 3.2). Overall, our results point to wASTRAL-h being a reasonable, if not always optimal, choice *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 was reduced in wASTRAL by half or more in most conditions (Fig. 3.3ad), 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. S3.7 and S3.10) saw *increased* support with wASTRAL. Two aspects of how we compute support have changed (Branch support). One is the handling of missing data (see (3.8)); it can be easily shown that, all else being equal, this change will decrease the localPP. 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.

Branch support used as input by weighted ASTRAL 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 (Fig. S3.1). In contrast, SH-like support was noticeably less effective. IQ-TREE's aBayes is a local measure of support (i.e., computed for the nearest neighbor interchanges 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 many branches, effectively down-weighting that gene. In such situations, global support may be more robust.

3.3.2 Limits and future work

The wASTRAL-s optimization, when solved exactly, gives a statistically constant species tree estimator given *estimated* gene trees under our MSC+Error+Support model. While this model is general, our assumptions about support values are 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 MSCand MSC+Error+Support models, respectively, and hope that future works can prove wASTRAL-h is also consistent. Even more intriguing is whether wASTRAL (which can take multi-individual/multi-copy trees as input) is statistically consistent under combined models of GDL and ILS, as ASTRAL-multi is (Markin and Eulenstein, 2021; Hill et al., 2020). This question is particularly important for datasets where assumptions of MSC are violated. For example, on the OneKP dataset, examining the relative support for the three topologies around each branch (Fig. 3.5b) reveals that the quartet frequencies do not always follow the MSC expectations (one high frequency and two equal low frequencies). We believe weighting will continue to be beneficial for models of GDL. However, it is unclear whether weighting by branch length is profitable when gene tree discordance is due to GDL and especially horizontal transfer; thus, we caution the use of branch length when these processes are suspected. Finally, future work can incorporate weighting in 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 (DAC) was slower (in many conditions) and less accurate (in some conditions) than the old algorithm (A3). While DAC tended to be as accurate or more accurate in the presence of missing data (Fig. 3.4b), our simulation results had no missing data, showing that the improved accuracy of wASTRAL-h was due to a better optimization objective, not a better optimization algorithm. Similar to A3, DAC is also 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, and future work can further optimize the code and implement GPU parallelization.

Our simulations, like any other, lacked some of the complexities of real biological data (Springer and Gatesy, 2018; Philippe et al., 2017). We did not include recombination, horizontal transfer, gene flow, hidden paralogy, alignment error, mistaken homology, violations of the model of sequence evolution, or missing data. It can be hoped that weighting helps alleviate the effects of some of these other sources of error as well. However, since many of these can lead to high support for the wrong trees, there is no guarantee that weighting would not leave these misleading signals intact or even amplified. Methods for simulating many of these effects are available and can be used in future studies to compare wASTRAL with both CA-ML and unweighted ASTRAL. A related promising avenue for future research is exploring other ways of weighting quartets. For example, 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 remember that not every weighting method will allow fast optimization using DP.

Finally, several features of 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., $CU + 2 \times SU$). Future work can examine ways to compute branch lengths in 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.

3.4 Material and Methods

3.4.1 Common notations and background

Let $\mathscr{L}_S := \{1, \ldots, n\}$ be a set of n species. Let us suppose that we are given a set of input binary gene trees $\mathscr{G}$ with $k := |\mathscr{G}|$. For each tree $G \in \mathscr{G}$, let its leaf set be $\mathscr{L}_G$ and its edge set be E_G. For each branch $e \in E_G$, we let $l_G(e)$ note its length. For a species set A, let $G \restriction A$ denotes G restricted to A. We refer to a set of four species as a quartet and define $\mathscr{Q}(G) := \{Q : |Q| = 4, Q \subseteq \mathscr{L}_G\}$ as the set of all quartets in G. We define $\delta_G(ab|cd) := 1$ when $\{a,b,c,d\} \in \mathscr{Q}(G)$ and $G \restriction \{a,b,c,d\}$ has topology $ab|cd$; otherwise we define $\delta_G(ab|cd) := 0$. For nodes u and v of a gene tree G, we let $\mathscr{P}_G(u,v)$ denote the set of branches on the path between u and v and let $l_G(u,v) := \sum_{e \in \mathscr{P}_G(u,v)} l_G(e)$. For a quartet $Q = \{a,b,c,d\}$, we denote $\mathscr{P}_G(Q) := \mathscr{P}_G(u,v)$, for u and v being nodes of g corresponding to the internal nodes (called the *anchors*) of $G \restriction Q$; i.e., in case that $G \restriction Q$ has topology $ab|cd$, anchors are the only node on $\mathscr{P}_G(a,b) \cap \mathscr{P}_G(a,c) \cap \mathscr{P}_G(b,c)$ and on $\mathscr{P}_G(b,c) \cap \mathscr{P}_G(b,d) \cap \mathscr{P}_G(c,d)$.

We assume each true gene tree G^* is generated from the true species tree S^* under the MSC model. Branch lengths of G^* are in coalescent units (CUs). For each quartet $Q = \{a,b,c,d\} \subseteq \mathscr{Q}(S^*)$ with topology $ab|cd$ in the species tree, let $\theta_Q = 1 - e^{-d}$ where d is the CU length of the internal branch of the quartet. Under MSC, for each true gene tree G^*, the

following holds (Degnan, 2013): $\mathrm{P}\big(\delta_G(ab|cd)=1\big)=\frac{1}{3}(1+2\theta_Q)$ and $\mathrm{P}\big(\delta_G(ac|bd)=1\big)=\mathrm{P}\big(\delta_G(ad|bc)=1\big)=\frac{1}{3}(1-\theta_Q)$. The input set $\mathscr{G}$ is a set of estimated gene trees, not true gene trees. In practice, these gene trees are estimated from sequence data using methods such as ML with branch lengths $l_G(e)$ given in the substitution-per-site units (SU). Moreover, input gene trees are furnished with support values: $s_G(e)$ maps each edge e of G to a support value in $[0,1]$.

3.4.2 Theoretical results: improved consistency and sample complexity

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

$$W(S,\mathscr{G}) := \sum_{G\in\mathscr{G}} \sum_{Q\in\mathscr{Q}(S)} w_G(S\restriction Q)\,, \tag{3.1}$$

where w_G is a function mapping a quartet of G to a number. In unweighted ASTRAL, for any $\{a,b,c,d\}$,

$$w_G(ab|cd) := \delta_G(ab|cd)\,. \tag{3.2}$$

In this paper, we introduce three new ways of defining w_G. Weighting by support sets:

$$w_G(ab|cd) := \Big(1-\prod_{e\in\mathscr{P}_G(\{a,b,c,d\})}\big(1-s_G(e)\big)\Big)\,\delta_G(ab|cd)\,. \tag{3.3}$$

Weighting by branch length uses

$$w_G(ab|cd) := e^{-\left(l_G(a,b)+l_G(c,d)\right)}\,\delta_G(ab|cd)\,. \tag{3.4}$$

Finally, the hybrid weighting scheme combines weighting by support and weighting by length and uses:

$$w_G(ab|cd) := \Big(1-\prod_{e\in\mathscr{P}_G(u,v)}\big(1-s(e)\big)\Big)e^{-\left(l_G(a,b)+l_G(c,d)\right)}\,\delta_G(ab|cd)\,. \tag{3.5}$$

We study hybrid weighting only empirically but provide theoretical justifications for weighting by support (for estimated gene tree topologies) and weighting by length (for true gene tree topologies).

Weighting by support

Genes have varying levels of signal, and hence gene tree estimation error,and estimated gene trees can also be biased towards a specific topology due to factors such as long branch attraction. When bias goes against the species tree topology, unweighted ASTRAL can be positively misleading (Roch et al., 2019). It is reasonable to assume that gene trees with lower signals have lower support regardless of bias. By down-weighting those genes, wASTRAL-s can rescue consistency. To formalize this intuition, we introduce a model of gene tree error that allows us to make a more formal statement, showing that support weighted ASTRAL is consistent under some conditions where unweighted ASTRAL is not.

MSC+Error+Support model. We assume each input estimated gene tree G is a draw from a distribution that depends on the true gene tree G^*. For each quartet $Q = \{a,b,c,d\} \subseteq \mathcal{Q}(S^*)$ and each gene G, let $\alpha_{G,Q} \in [0,1]$ denote a parameter controlling the quality of the estimated quartet gene tree $G \restriction Q$. We assume $\alpha_{G,Q}$ is independently drawn from the topology of G^* and we let the expected value and variance of $\alpha_{G,Q}$ across genes be denoted by $\bar{\alpha_Q}$ and σ^2_α. For each true gene tree topology, with probability $\alpha_{G,Q}$, we simply set the estimated gene tree to the true topology. With probability $1-\alpha_{G,Q}$, we choose among the three topologies with probabilities $p_1^{G,Q}, p_2^{G,Q}, p_3^{G,Q}$. When these numbers are equal, there is no bias in gene tree estimation, and ASTRAL remains consistent (easy to prove). However, in our model, we allow systematic bias towards any topology. Let $\beta_Q = \max_G \big(\max(3p_1^{G,Q}-1, 3p_2^{G,Q}-1, 3p_3^{G,Q}-1, 1-3p_1^{G,Q}, 1-3p_2^{G,Q}, 1-3p_3^{G,Q})\big)$ be the maximum bias towards or away any topology across genes. Under this model, the joint probability of true and estimated gene trees would follow the inequalities laid out in Table 3.2. For example, in the worst case, where $3p_1^{G,Q} = 1-\beta_Q$, $3p_2^{G,Q} = 1+\beta_Q$, and $3p_3^{G,Q} = 1$, the joint distribution of true and estimated gene trees is given in Table 3.1 and

depicted in Figure 3.1b.

Table 3.1. Joint probabilities (top) and weights (bottom) of estimated and true gene tree topologies under the MSC+Error+Support with the worst-case scenario when $3p_1 = 1-\beta$, $3p_2 = 1+\beta$, and $3p_3 = 1$ for all genes; note that parameters are per quartet and per gene but we omit Q and G superscript for brevity.

$\mathbb{E}[(\cdot)(\cdot)\vert\alpha]$	$\delta_{G^*}(ab\vert cd)$	$\delta_{G^*}(ac\vert bd)$	$\delta_{G^*}(ad\vert bc)$
$\delta_G(ab\vert cd)$	$\frac{1}{3}(1+2\theta)\big(\alpha+\frac{1}{3}(1-\alpha)(1-\beta)\big)$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1-\beta)\big)$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1-\beta)\big)$
$\delta_G(ac\vert bd)$	$\frac{1}{3}(1+2\theta)\big(\frac{1}{3}(1-\alpha)(1+\beta)\big)$	$\frac{1}{3}(1-\theta)\big(\alpha+\frac{1}{3}(1-\alpha)(1+\beta)\big)$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1+\beta)\big)$
$\delta_G(ad\vert bc)$	$\frac{1}{3}(1+2\theta)\big(\frac{1}{3}(1-\alpha)\big)$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)\big)$	$\frac{1}{3}(1-\theta)\big(\alpha+\frac{1}{3}(1-\alpha)\big)$
$w_G(ab\vert cd)$	$\frac{1}{3}(1+2\theta)\big(\alpha+\frac{1}{3}(1-\alpha)(1-\beta)\big)^2$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1-\beta)\big)^2$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1-\beta)\big)^2$
$w_G(ac\vert bd)$	$\frac{1}{3}(1+2\theta)\big(\frac{1}{3}(1-\alpha)(1+\beta)\big)^2$	$\frac{1}{3}(1-\theta)\big(\alpha+\frac{1}{3}(1-\alpha)(1+\beta)\big)^2$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)(1+\beta)\big)^2$
$w_G(ad\vert bc)$	$\frac{1}{3}(1+2\theta)\big(\frac{1}{3}(1-\alpha)\big)^2$	$\frac{1}{3}(1-\theta)\big(\frac{1}{3}(1-\alpha)\big)^2$	$\frac{1}{3}(1-\theta)\big(\alpha+\frac{1}{3}(1-\alpha)\big)^2$

Table 3.2. Joint probabilities (top) and weights (bottom) of estimated and true gene tree topologies under the MSC+Error+Support will follow the inequalities shown here. We omit Q and G superscript for brevity.

$\mathbb{E}\big[(\cdot)(\cdot)\big\vert\alpha_{G,Q}\big]$	$\delta_G(ab\vert cd)$	$\delta_G(ac\vert bd)$
$\delta_{G^*}(ab\vert cd)$	$\geq \frac{1}{3}(1+2\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1+2\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\delta_{G^*}(ac\vert bd)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1-\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\delta_{G^*}(ad\vert bc)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\mathbb{E}\big[(\cdot)(\cdot)\big\vert\alpha_{G,Q}\big]$	$w_G(ab\vert cd)$	$w_G(ac\vert bd)$
$\delta_{G^*}(ab\vert cd)$	$\geq \frac{1}{3}(1+2\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1+2\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$
$\delta_{G^*}(ac\vert bd)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1-\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$
$\delta_{G^*}(ad\vert bc)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$

We assume that for each quartet, the quartet support defined using (3.3) matches the probability of that topology being observed given the true gene tree. Thus, with our model for estimated gene tree distributions, the support of the quartet topology i is $\alpha_{G,Q}+(1-\alpha_{G,Q})p_i^{G,Q}$ if it matches the true tree and $(1-\alpha_{G,Q})p_i^{G,Q}$ if it does not, leading to expected topology weights $w_G(.)$ given in Tables 3.1–3.2.

We now state our main results. Proofs of all results are given in Appendix Proofs.

Proposition 3.1. *For each estimated gene tree G, $\mathbb{E}\big[\delta_G(ab\vert cd)-\delta_G(ac\vert bd)\big] \geq \theta_Q\bar{\alpha}_Q - \frac{2}{3}(1-\bar{\alpha}_Q)\beta_Q$ and $\mathbb{E}\big[w_G(ab\vert cd)-w_G(ac\vert bd)\big] \geq \frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha}_Q{}^2+\sigma_\alpha^2)+\frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha}_Q - \frac{4}{9}(1-\bar{\alpha}_Q)\beta_Q$.*

For consistency of ASTRAL and wASTRAL-s, we need $\mathbb{E}\big[\delta_G(ab\vert cd)-\delta_G(ac\vert bd)\big]\geq 0$ and $\mathbb{E}\big[w_G(ab\vert cd)-w_G(ac\vert bd)\big]\geq 0$, respectively. Figure 3.1e depicts the RHS of equations of

Proposition 3.1, solving for θ_Q and setting σ_α^2 to zero (which is the worst-case for wASTRAL-s). It shows that wASTRAL-h is consistent for a larger set of species tree CU branch lengths, even in absence of any variation in gene tree quality. We next state this observation formally.

Theorem 3.1. *Given estimated gene trees furnished with support generated under MSC+Error+Support model, there exist conditions where* (3.3) *guarantee a statistically consistent estimator of S^* but* (3.2) *does not, and the reverse is not true.*

Weighting by length

Our next result shows that using the length-based weighting function (3.4) leads to a larger gap than unweighted ASTRAL between the expected score of the true species tree and the alternative trees and thus has better sample complexity. Shekhar et al. 2017 has 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. Following that paper, we focus on the regime with $k = \Theta(f^{-2})$ gene trees and show a constant factor improvement in sample complexity. All theoretical results in this section assume that an input gene tree G matches the true gene tree G^* in *topology*.

The improved sample complexity essentially follows from the fact that under the MSC model, gene trees that match the species tree have shorter CU terminal branch lengths on average because discordance is caused by deep coalescence. However, a theoretical difficulty is that input gene trees have SU branch lengths instead of CU length. Thus, we need a model to translate CU lengths in G^* to SU lengths in G, capturing the effects of change in mutation rates and population sizes. We examine two such models.

Naive model. We start with a simple choice akin to a strict clock. Under this naive model, all branches of G are scaled from branches of G^* using a fixed multiplier λ.

Variable rate model. Let branches of the species tree S^* be broken into segments of arbitrary length (Fig. S3.20). For each gene tree G^*, a species tree in SU units $S^\dagger$ is drawn from a fixed distribution $\mathscr{D}$ (which does not change with G^*). $S^\dagger$ matches S^* in topology. The length of each

segment I in $S^\dagger$ is scaled from the length of its corresponding segment in S^* using a multiplier $\Lambda^I_{S^\dagger}$. The set of all multipliers can be jointly drawn from any distribution as long as for each segment I, $\mathbb{E}_{S^\dagger}[\Lambda^I_{S^\dagger}] = \lambda$. Segments in S^* can be used to divide G^* into segments defined at the same points along each branch (Fig. S3.20). The gene tree G is obtained from G^* by multiplying the CU length of each of its segments by the multiplier assigned to that segment in $S^\dagger$. Because segments have different multipliers (even though they have the same expectation), gene tree $G^\dagger$ deviates from ultrametricity. Because multipliers are drawn separately for each gene, deviations from ultrametricity happen in different ways across different genes.

We now state the results. Let $X_G := w_G(ab|cd) - w_G(ac|bd)$ and $Y_G := \delta_G(ab|cd) - \delta_G(ac|bd)$. Then,

Proposition 3.2. *For a true quartet species tree S^* with topology $ab|cd$ and input gene trees $\mathscr{G}$ generated under the naive model with any multiplier λ, let f be the distance between anchors of S^*. As $f \to 0$, given $k = \Theta(f^{-2})$ gene trees, we have $Var[X_G] = \Theta_f(1)$ and*

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

Similarly, under the variable rates model and assuming limited variance of rates across genes, we prove:

Proposition 3.3. *For a true quartet species tree S^* with topology $ab|cd$ and input gene trees $\mathscr{G}$ 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 that for every branch segment I, the variance of its multiplier is bounded above: $Var(\Lambda^I_{S^\dagger}) \leq \varepsilon^2$ where $\varepsilon^2 = \frac{e^{-\lambda L}}{(16+32\lambda)+(6+32\lambda+32\lambda^2)L}\left(\frac{20(\lambda+\lambda^2)}{9(1+2\lambda)^2}\right)^3$. As $f \to 0$, given $k = \Theta(f^{-2})$ gene trees, we have $Var[X_G] = \Theta_f(1)$ and*

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

These propositions lead us to the main result.

Theorem 3.2. *Under the conditions of Proposition 3.2 or Proposition 3.3,*

$$P\Big(\sum_{G\in\mathscr{G}} w_G(ab|cd) \le \sum_{G\in\mathscr{G}} w_G(ac|bd)\Big) \le P\Big(\sum_{G\in\mathscr{G}} \delta_G(ab|cd) \le \sum_{G\in\mathscr{G}} \delta_G(ac|bd)\Big).$$

3.4.3 Optimization algorithm

The objective of our optimization task is to find S maximizing $W(S,\mathscr{G})$ given in (3.1) for one of the w_G functions (3.2)–(3.5). For a species tree S, let $\mathscr{T}_S$ denote the set of tripartitions corresponding to the internal nodes of S. For a tripartition $A|B|C \in \mathscr{T}_S$ corresponding to an internal node v in 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. Then,

$$W(A|B|C,G) = \frac{1}{2}\sum_{\substack{a\in A\cap\mathscr{L}_G\\ b\in B\cap\mathscr{L}_G\\ c\in C\cap\mathscr{L}_G}} \Big(\sum_{d\in A\cap\mathscr{L}_G-\{a\}} w_G(ad|bc) + \sum_{d\in B\cap\mathscr{L}_G-\{b\}} w_G(bd|ac) + \sum_{d\in C\cap\mathscr{L}_G-\{c\}} w_G(cd|ab)\Big),$$

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

ASTRAL-III uses a traversal of gene trees to compute $W(A|B|C,G)$ with weight function (3.2) without enumerating all $\binom{n}{4}$ quartets. At each gene tree node, the total number of shared quartets between that node and v is computed using simple combinatorics. When quartets are weighted differently using weight functions (3.3)–(3.5), computing the aggregated weights of quartets around a node becomes more difficult as simple combinatorial equations become unavailable in the general case. Thus, we cannot simply use the same algorithm as ASTRAL and instead propose a new algorithm. In its simplest form (called the *base* version), the algorithm works as follows.

1. Starting from an empty tree, add each species to the tree one-by-one with a random order to obtain a full tree (see the Placement algorithm section and Algorithm S3.1). The algorithm also computes and stores tripartition scores $W(A|B|C,G)$ for all tripartitions of the output

tree.

2. Repeat the previous step for r rounds; by default $r \in [16, 32]$ (see details under Placement algorithm).

3. Combine results of the r rounds using a final dynamic programming (DP) step, which reuses the tripartition weights computed in step 1; each internal node of the output is constrained to be in at least one of the r greedy trees (see Dynamic programming section and Algorithm S3.2).

What makes this approach possible is step 1: a new algorithm that allows each addition to an existing tree to be performed optimally and efficiently. Importantly, while the base algorithm is a greedy heuristic, as Theorem 3.4 and the remark afterward will show, it retains the statistical consistency properties proved in Theorems 3.1–3.2. The running time of the base algorithm scales with $O(kn^3 \log(n))$ in the worst case (Proposition 3.4) and is better with respect to k but worse with respect to n compared to ASTRAL-III, which is $O\big((kn)^{2.73}\big)$ in the worst case and roughly $O(k^2n^2)$ in practice. Thus, we also propose a divide-and-conquer (DAC) algorithm for $n \geq 200$ that uses the base algorithm on subsets of size $O(\sqrt{n})$ (see the DAC algorithm section and Algorithm S3.3). This strategy improves the running time to $O(n^{2.5+\varepsilon}k)$ under some assumptions, as detailed below under Theorem 3.5. The DAC algorithm also retains the statistical consistency guarantees (Theorem 3.6). We next detail each algorithmic component mentioned above.

Placement algorithm

Mai and Mirarab 2022 use the idea pioneered by Brodal et al. 2013 to design a quasi-linear algorithm to find the optimal placement of a species on a backbone tree that minimizes its quartet distance to a set of reference trees (e.g., gene trees). This algorithm traverses a binary (or multifurcating) species tree in a top-down manner and colors species using three (or more) colors, A, B, and C. When entering any node u of the species tree, all species under u are already colored

A and all other species are colored C. At this point, the smaller child of u is recolored with B. The recoloring is done one species at a time; for each species, the path from the associated leaf in each gene tree to the root is visited, and several counters assigned to each gene tree node are updated. These counters enable calculating the score for placing the query on each species tree branch. After this recoloring is done and before moving from u to any of its children v, the sister of v is colored C, and if v is the smaller child of u, then v is changed back to A. This algorithm performs only $O(n \log n)$ species recoloring steps due to the smaller-child trick, which recolors the larger child of each node less often than the smaller child. Moreover, by representing each gene tree using an $O(\log n)$-height tree called HDT adopted from Brodal et al. 2013, it ensures each recoloring takes $O(k \log n)$ time.

We build on the idea by Mai and Mirarab 2022 and adapt it to optimally solve the weighted quartet score placement problem (Algorithm S3.1), changing it in three substantial ways. *i*) We have created a new set of counters that enable us to compute the total *weighted* quartet score of all tripartitions resulting from all possible placements of the query. These counters essentially count the total weight of all the quartets with the same MRCA using recursive equations shown in Figure 3.6 and Table S3.1. The derivation of these counters is the heart of the algorithm but is too complex to detail here. We leave a full description to Proof of Theorem 3.3. *ii*) At each node u, we also recolor the query species as A, B, and C and recompute the counters; this allows us to compute the quartet score for all tripartitions resulting from all placements of the query. *iii*) Since our counters are more complex than Mai and Mirarab 2022, we use input gene trees instead of HDTs, which would be hard to implement. As a result, the cost of a leaf recoloring in our algorithm is $O(kH)$ where H is the average height of gene trees instead of $O(k \log n)$ had we used HDTs. Note that for sufficiently balanced gene trees, $O(kH)$ and $O(k \log n)$ are similar. We next prove that this algorithm finds the optimal solution.

Theorem 3.3. *Let S be a species tree, i be a species not in $\mathscr{L}_S$, $\mathscr{S}$ be the set of possible species tree topologies by placing i onto S, and S' be the output of Algorithm S3.1. Then,*

$W(S', \mathscr{G}) = \max_{\hat{S} \in \mathscr{S}} W(\hat{S}, \mathscr{G})$.

While each individual 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. Empirically, we found $r \geq 4$ to have minimal impact on the accuracy, but small improvements in the quartet score are observed for up to $r = 32$ rounds in outlier cases (Fig. S3.21). Based on these results, we set r (which the user can adjust) using a dynamic heuristic: i) start with 12 rounds and perform the DP algorithm 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.

Dynamic programming

In each greedy search, we add the tripartitions of each S_i and their weights to a lookup table W^*. The DP step computes an optimal species tree restricted to the tripartitions of W^* (Algorithm S3.2). The DP algorithm proceeds almost identically to ASTRAL, with one difference: While the search space in ASTRAL is the set of bipartitions found in all of the S_i trees, here, the search space is the set of all tripartitions. With this change, we do not need to compute weight scores for any tripartition as those are precomputed and stored in W^* in the placement step.

Proposition 3.4. *The time complexity of Algorithm S3.2 is* $O(kHn^2 \log n)$.

Since $H = O(\log n)$ for balanced trees and $H = O(n)$ for caterpillar trees, the time complexity of Algorithm S3.2 is $O(kn^2 \log^2 n)$ when gene trees are roughly balanced and $O(kn^3 \log n)$ when they are not. Note that because the counters are linearly related to counters of children of a node, in theory, the HDT structure can be adopted in our algorithm leading to a $O(kn^2 \log^2 n)$ worst-case complexity. Since adopting HDT would add much more complexity for (potentially) little gain, we do not pursue it further.

Algorithm S3.2 is not guaranteed to find the optimal solution. However, a positive

theoretical result ensures that this lack of optimality does not impede the statistical consistency of the solution:

Theorem 3.4. *If there exists a species tree topology S^* satisfying that for each quartet subtree $ab|cd$,*

$$\sum_{G\in\mathcal{G}} w(ab|cd) > \max\Big(\sum_{G\in\mathcal{G}} w(ac|bd), \sum_{G\in\mathcal{G}} w(ad|bc)\Big), \tag{3.6}$$

then the output of Algorithm S3.2 will be S^*.

Remark. *For a binary true species tree S^*, as $k\to\infty$, S^* satisfies the condition of Theorem 3.4 with an arbitrarily high probability for wASTRAL-s under the assumptions of Theorem 3.1 and for wASTRAL-bl under the assumptions of Theorem 3.2. To see this point, note that due to the consistency of the estimator, for a quartet Q to achieve a high probability $1-\varepsilon'$ a certain $k_{\varepsilon',Q}$ must be sufficient. Setting $\varepsilon' = 1-(1-\varepsilon)^{1/\binom{n}{4}}$ and using a union bound, it is easy to see that $k=\max_Q k_{\varepsilon',Q}$ is enough to achieve the probability $1-\varepsilon$ of correctness for all quartets. Thus, by Theorem 3.4, Algorithm S3.2 is a statistically consistent estimator of the species tree under the assumptions of Theorems 3.1 and 3.2. We conjecture that wASTRAL-h can also be proved statistically consistent under assumptions similar to Theorem 3.1 for topology and support and Theorem 3.2 for branch length.*

DAC algorithm

The DAC procedure (Algorithm S3.3) first computes a backbone tree on fewer species, adds all the remaining species onto the backbone tree, and then locally refines the topology around the backbone branch.

1. To compute a backbone tree S_i, we randomly select $m=\lceil\sqrt{n}\rceil$ leaves and apply the Algorithm S3.2 with $r=\lceil\sqrt{n}\rceil$ rounds of placements to get a backbone tree with m species.

2. For the remaining $n-m$ species, we independently find their optimal placement on S_i using the Algorithm S3.1. We group species placed on the same branch together to obtain $2m-3$ clusters.

3. For each cluster C_e corresponding to a branch e, we sequentially place species in C_e onto S_i using the Algorithm S3.1 and remove any "orphan" species that are not placed on e or its derived branches; the result is called S_e.

4. All trees in $\{S_e : e \in E_{S_i}\}$ induce the same scaffold tree S_i on their shared taxa. Thus, they can be easily merged into a uniquely defined tree S'_i.

5. If S'_i orphan species exist, at the end, we place them onto S'_i using the Algorithm S3.2.

The potential for orphan taxa makes it harder to establish the time complexity of the DAC algorithm theoretically, but a result can be proved:

Theorem 3.5. *When the inequality condition in Theorem 3.4 is satisfied, then the time complexity of the DAC algorithm is $O(n^{1.5+\varepsilon}kH)$ with arbitrarily high probability.*

Similar to the base algorithm, the DAC algorithm retains statistical consistency.

Theorem 3.6. *Under the conditions of Theorem 3.4, the DAC Algorithm S3.3 will output S^*.*

Remark. *Under assumptions of Theorem 3.1 for wASTRAL-s and Theorem 3.2 for wASTRAL-bl, Algorithm S3.3 gives a statistically consistent estimator of the species trees.*

3.4.4 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 with 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. Removing an internal branch e of the species tree and its four adjacent branches defines a quadripartition of species $A|B|C|D$,

and we assume $(A \cup B)|(C \cup D)$ is the bipartition defined by e. Note that any quartet $(a,b,c,d) \in A \times B \times C \times D$ has the same internal branch as e. Let $\mathscr{G}$ denote the subset of gene trees with at least one element from each of A, B, C, and D. We define x_1, the normalized quartet count for branch e, as

$$x_1 = \frac{\sum_{G \in \mathscr{G}} \sum_{(a,b,c,d) \in A \times B \times C \times D} w_G(ab|cd)}{\sqrt{\frac{1}{|\mathscr{G}|} \sum_{G \in \mathscr{G}} \left(\sum_{(a,b,c,d) \in A \times B \times C \times D} w_G(ab|cd) + w_G(ac|bd) + w_G(ad|bc) \right)^2}} . \tag{3.7}$$

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. This form of normalization models the observation that gene trees with higher weights also have higher variance in their weights. 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 $h(x) = 2^x \mathbf{B}(x+1, x_1+x_2+x_3-x+2\lambda)\left(1 - \mathbf{I}_{\frac{1}{3}}(x+1, x_1+x_2+x_3-x+2\lambda)\right)$, $\mathbf{B}$ is the beta function, $\mathbf{I_x}$ is the regularized incomplete beta function, and λ is birth rate in the Yule prior distribution (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 $N_g = |A \cap \mathscr{L}_g| \times |B \cap \mathscr{L}_g| \times |C \cap \mathscr{L}_g| \times |D \cap \mathscr{L}_g|$ 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 $(A \cup B)|(C \cup D)$. Then, the old definition sets $x_1 = \sum_{G \in \mathscr{G}} \frac{n_g}{N_g}$ while the new definitions uses

$$x_1 = \frac{\sum_{G \in \mathscr{G}} n_g}{\sqrt{\frac{1}{|G|} \sum_{G \in \mathscr{G}} N_g^2}} . \tag{3.8}$$

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.

3.4.5 Datasets

Simulated data

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. 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 Robinson and Foulds (1981) (RF) distance between the true species trees and true gene trees. Gene alignments of length $\{200, 400, 800, 1600\}$ bps are simulated using Indelible (Fletcher and Yang, 2009) under the GTR model after assigning SU gene tree branch lengths that deviate from the clock using rate multipliers. Gene trees are reconstructed under the GTR+Γ model using FastTree-2 (Price et al., 2010). The gene tree estimation error, measured by the False Negative (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 FastTree-2 available for each gene tree.

S200 Simulated dataset by Mirarab and Warnow 2015 includes 200 ingroup species and an 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 errors across replicates (mean: 25%, 31%, and 47%, for low, medium, and high ILS). Gene trees are estimated using FastTree-2, but because bootstrap replicates are not available, we compute aBayes support using IQ-Tree with fixed topologies.

By default, we compute branch length and support using IQ-TREE (v 1.6.12) aBayes

option (`--abayes`) under GTR+Γ model. As each support value s is between $\frac{1}{3}$ and 1, we normalize support value to $\frac{3s-1}{2}$ so that the minimum is 0. To run wASTRAL (which currently takes only binary trees as input), we randomly resolve polytomies in input trees with length and support set to 0, which is equivalent to a polytomy for wASTRAL-s and -h.

ASTRAL-III version 5.7.4 is used throughout. ASTRAL-III-5% (S100 dataset) denotes running ASTRAL-III on gene trees with low bootstrap support branches ($< 5\%$) contracted. The 5% threshold is used because Zhang et al. 2018 found it to have the best accuracy overall. On the S200 dataset, because bootstrap support is not available, we instead rely on IQ-TREE aBayes support, which tends to be much higher than bootstrap support. Thus, we contract branches with support below a 0.90 threshold with aBayes, denoted as ASTRAL-III-90%.

CA-ML is performed using unpartitioned ML. On the S200 dataset, CA-ML was available from the original study (where they used FastTree-2 as the ML method) and is used here. On S100, we ran CA-ML using FastTree-2. Thus, on both datasets, the same tool is used for gene tree estimation and CA-ML, ensuring the comparisons are fair.

Biological datasets

Seven biological datasets were used.

OneKP (OneKP Initiative, 2019) dataset includes 1178 species spanning the plant tree of life obtained using transcriptomics. The original study has inferred 410 AA-based gene trees from putative single-copy genes using RAxML with bootstrap support (which we use), an ASTRAL-III species tree, and CA-ML using RAxML.

Canis (Gopalakrishnan et al., 2018) dataset includes 48 genomes across genus Canis with taxon sampling that allows reconstruction at both species and population levels. Loci with roughly 10kbp lengths were selected across the genome at random, leading to 449,450 gene trees. Since ASTRAL-II could not handle this size, the original study 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 available gene trees, which the original

paper estimated using FastTree-2; we were also able to analyze up to 100,000 gene trees using ASTRAL-MP (Yin et al., 2019) (within 48 hours). Due to the large number of genes, we simply use the provided SH-like FastTree-2 support instead of re-estimating support.

Avian (Jarvis et al., 2014) dataset includes 48 species designed to resolve the order-level avian relationships, which experience extremely high levels of gene tree discordance potentially due to a rapid radiation. Authors studied three data types: concatenation of exons per gene (exons), concatenation of introns per gene (introns), and Ultra Conserved Elements (UCE). Here, we analyze all 14,446 input gene trees (8251 exons, 2516 introns, and 3679 UCEs) with bootstrap-annotated branches available from the original study. The main challenge on this dataset is the low gene tree resolution, which led to the development of the statistical binning method (Mirarab et al., 2014a). Without binning, the analyses of all 14,446 loci resulted in species 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.

Cetaceans (McGowen et al., 2020) dataset includes targeted-captured exonic data for 100 individuals from 77 cetacean species and 12 outgroups. The original study estimated gene trees using RAxML under the GTRCAT model but without support for 3191 protein-coding genes. We computed Bayesian local supports and branch lengths for fixed gene tree topologies using IQ-Tree, and reanalyzed the dataset using wASTRAL-h. We compare the results to two trees produced by the original study: a CA-ML tree and an ASTRAL-multi tree that forces individuals of the same species to be grouped together.

Insect datasets. We also tested three insect datasets, in each case, using available gene trees. i) a 32-taxon collection of 853 RAxML gene trees with bootstrap supports obtained from alignments of ultraconserved elements focused on 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 obtained from transcriptomic alignments 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) and obtained from amino-acid alignments of orthologous protein-coding genes.

3.4.6 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 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 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 the 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 thresholds. We also plot the empirical cumulative density function (ECDF) of correct and incorrect branches. We expect higher support for correct branches than for 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 the 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.

3.5 Acknowledgements

Chapter 3, in full, has been submitted for publication of the material as it may appear in " Zhang, C. & Mirarab, S. Weighting by Gene Tree Uncertainty Improves Accuracy of Quartet-based Species Trees. *Molecular Biology And Evolution*. (2022) ." The dissertation author was the primary investigator and author of this paper.

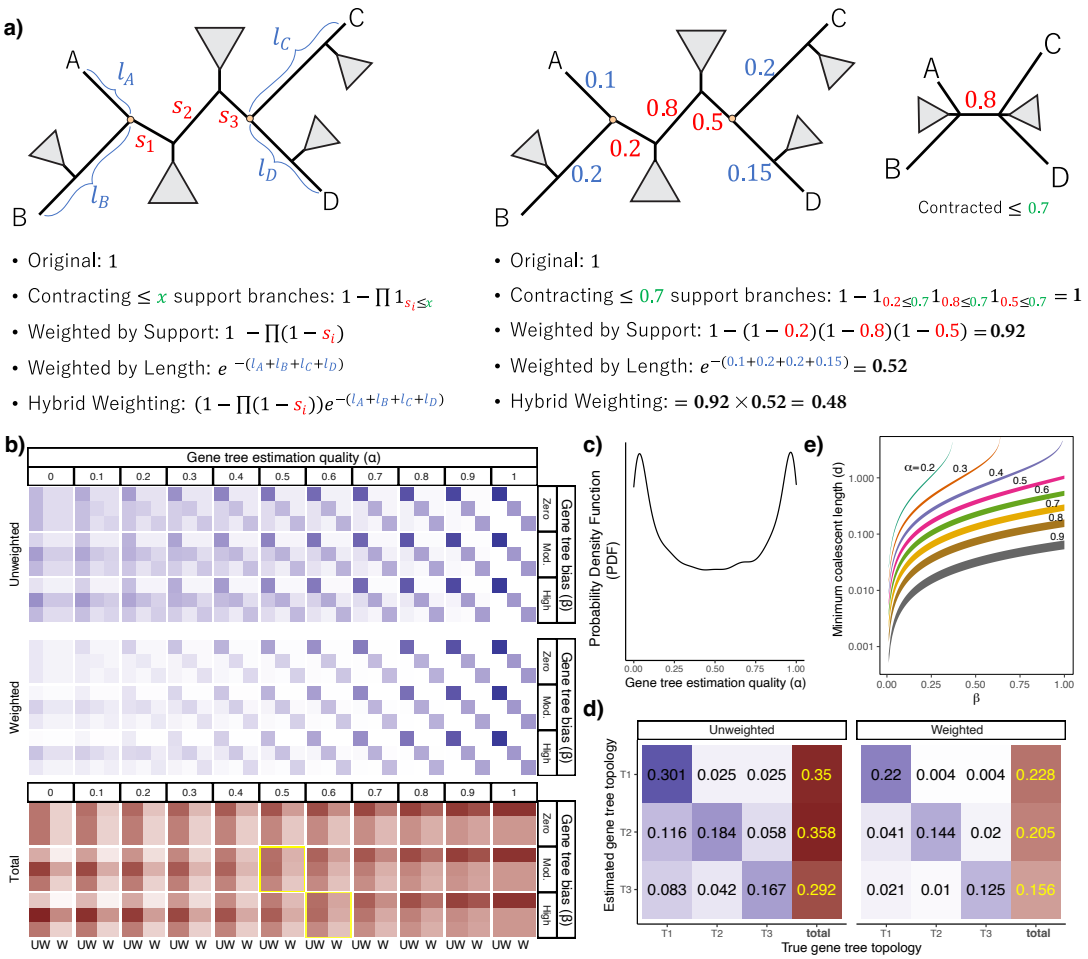

Figure 3.1. (a) Illustration of weighting methods. The generic formula and an example of weighting gene tree quartet $ab|cd$. Trees are annotated with the support (red) of all branches between anchors (orange dots) and the substitution per site unit length of each leaf-to-anchor path (blue). (b-e) Illustration of the impact of weighting using our MSC+Error+Support model for a quartet species tree with internal branch length set to $-\ln 0.75$ CU. (b) Top: each 3×3 square shows the joint probabilities of true (by column) and estimated (by row) gene trees for each of the three possible quartet topologies. The first row/column represents the topology matching species tree, and the second row/column corresponds to the topology towards which gene tree estimation is biased. The gene tree estimation quality α ranges in $[0,1]$, and the bias in gene tree estimation β is set to zero, moderate (0.4), or high (0.6). These probabilities correspond to expected weights in normal ASTRAL. Middle: The expected weights in wASTRAL-s for each scenario. Bottom: The 3×2 grids show the marginal expected score of each topology (rows) for unweighted ASTRAL (UW; first column) and weighted ASTRAL (W; second column). Note the reduced darkness of W columns as α decreases. The two girds highlighted in yellow: the score is highest for the wrong (second row) topology without weights but is higher for the correct topology (first row) with weights. (c) Distribution α drawn from $Beta(0.5,0.5)$ across genes in a toy example. (d) Joint (blue) and marginal (red) probabilities of topologies with and without weighting with moderate bias ($\beta=0.4$) and α drawn from the distribution shown on top. (e) Each band shows the range of coalescent unit (CU) quartet internal branch length where ASTRAL is not consistent but support weighted ASTRAL is, for different α and β valeus.

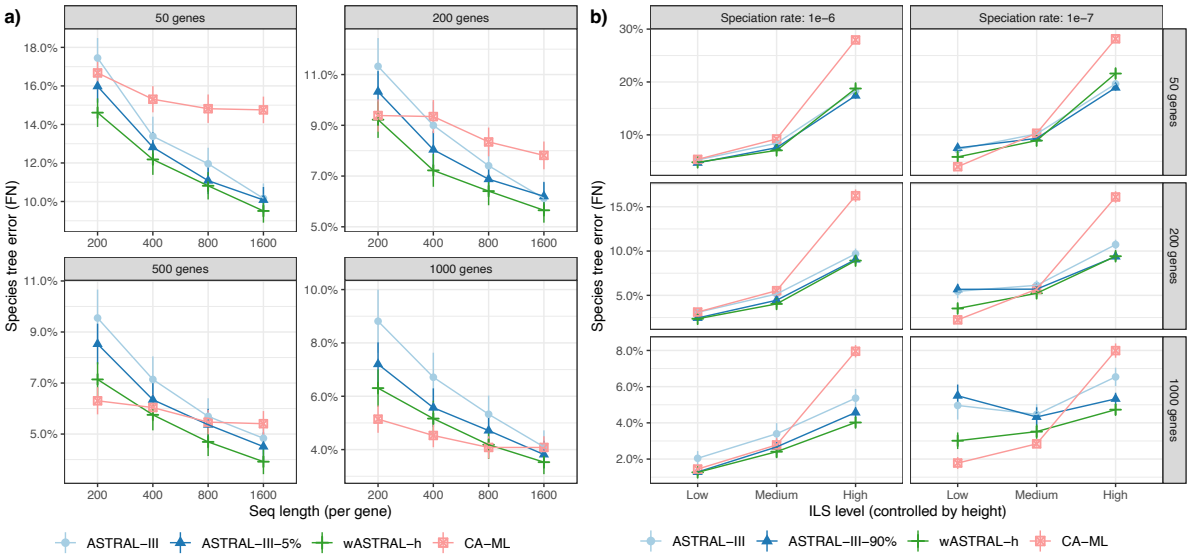

Figure 3.2. Species tree topological error on simulated datasets, comparing weighted ASTRAL hybrid (wASTRAL-h) against ASTRAL-III using fully resolved and contracted gene trees and concatenation using ML (CA-ML). (a) Results on the S100 dataset with $k = \{50, 200, 500, 1000\}$ gene trees (boxes) and gene sequence length $\{200, 400, 800, 1600\}$ (x-axis). Gene trees and CA-ML both inferred using FastTree-2. ASTRAL-III-5% contracts branches with $< 5\%$ BS. (b) Results on the S200 dataset with $k = \{50, 200, 1000\}$, rates of speciation 1E-6 and 1E-7, and three ILS levels. Gene trees and CA-ML both inferred using FastTree-2. ASTRAL-III-90% contracts branches with aBayes support $< 90\%$. See Fig. S3.4 and S3.5 for box plots.

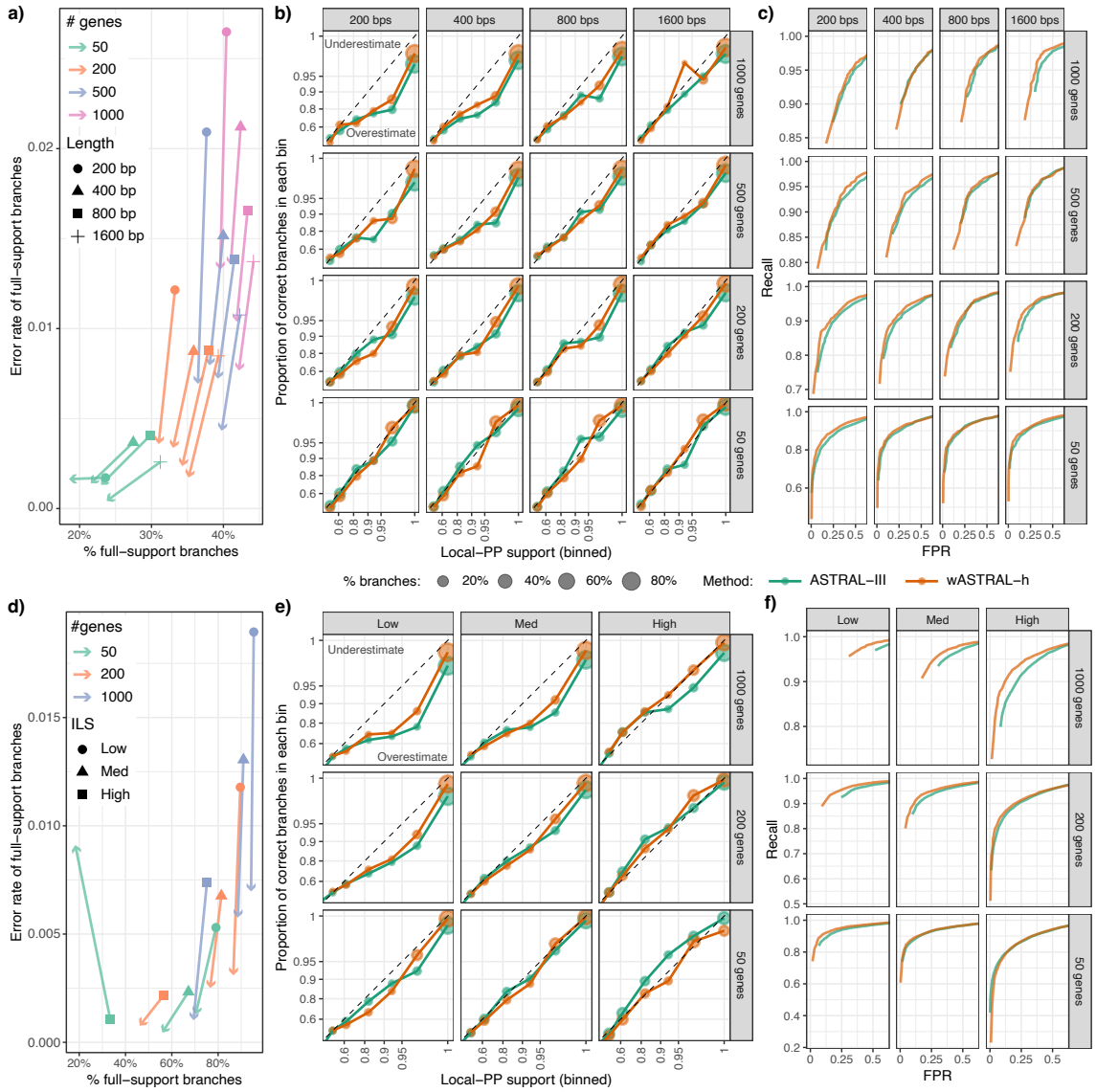

Figure 3.3. Support accuracy across (a-c) S100 dataset with $k = \{50, 200, 500, 1000\}$ and sequence length $\{200, 400, 800, 1600\}$ and (d-f) S200 dataset with $k = \{50, 200, 1000\}$ and levels of ILS from low to high. (a,d) Change in 100% support branches. Each line shows the portion of full-support 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 errors in wASTRAL-h. (b,e) Support calibration. Branches are binned by their 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. (c,f) Receiver operating characteristic (ROC) curves where each dot corresponds to a contraction threshold, (Evaluation criteria). See Figs. S3.6-S3.11.

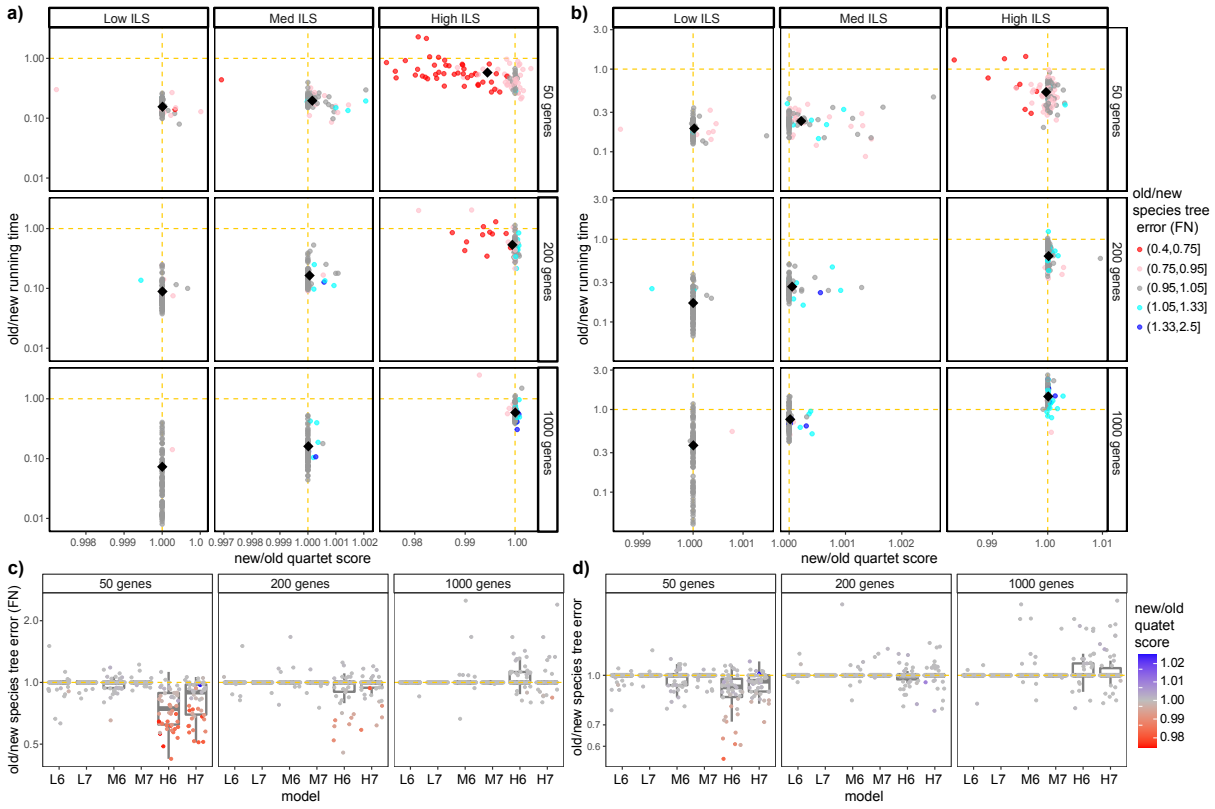

Figure 3.4. Comparison of the running time, quartet score, and accuracy between the old DP-based (A3) and the new optimization algorithms (DAC), both run without weighting on the S200 dataset. a,b) The ratio between the running time and quartet scores before (a) and after (b) randomly removing 5% of taxa from each gene tree; colors denote the ratio of species tree estimation error between the two methods. Note that the upper-right corner and blue color favor DAC. Results are separated by ILS levels from low to high and by $k = \{50, 200, 1000\}$. c,d) The species tree topological error using the A3 algorithm divided by the DAC 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 1E-6 and 1E-7 rates.

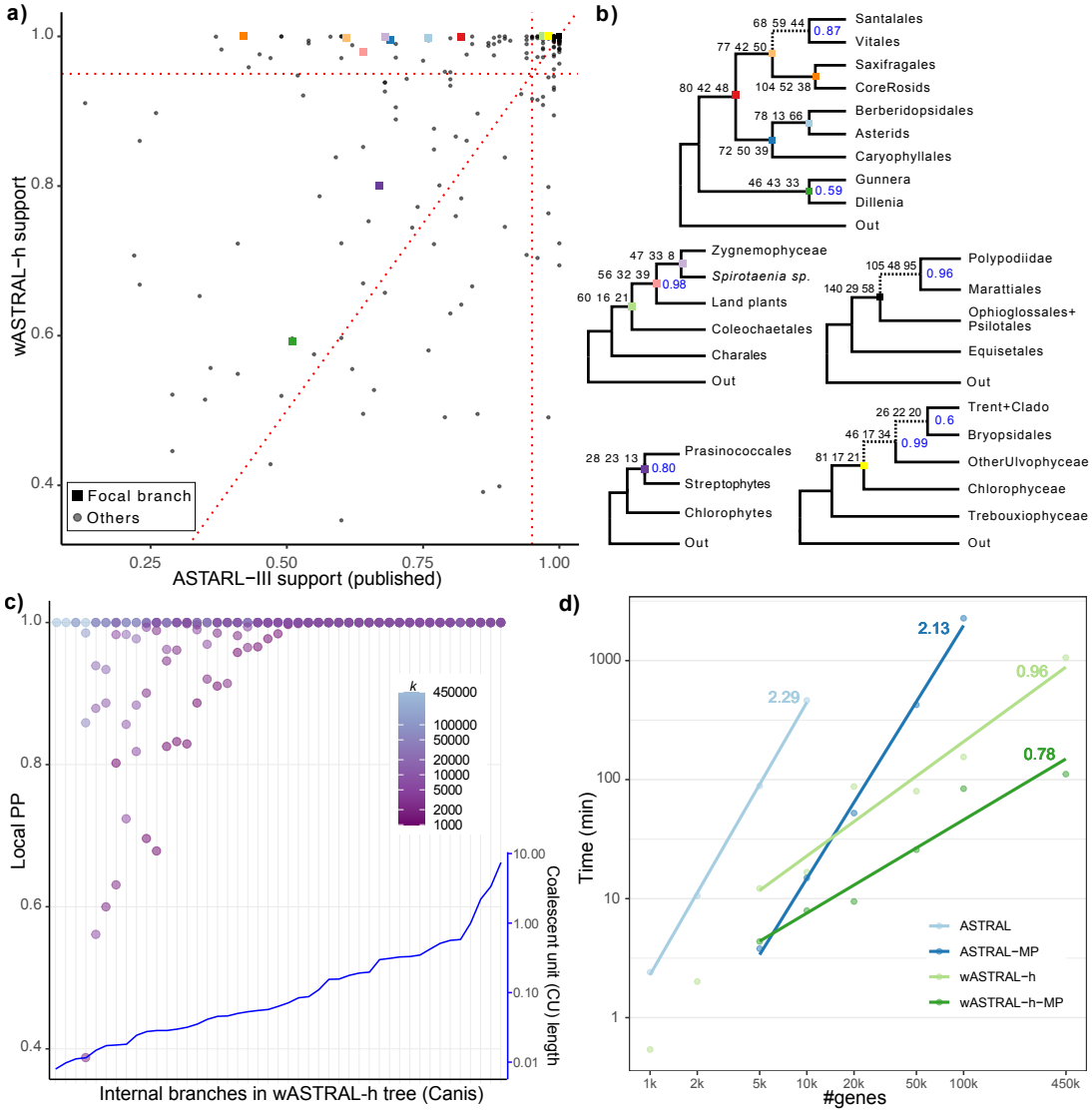

Figure 3.5. Results on OneKp (a,b) and canis (c,d) datasets. (a) Local posterior probabilities (PP) support of all species tree branches shared between wASTRAL-h and the published ASTRAL-III. Focal branches (squares) with support less than 100% in one of the two trees are colored and labeled in panel b. (b) wASTRAL-h resolutions of focal branches that differ from ASTRAL-III in topology or support. Branch labels: total weights of all quartets around each branch for the three possible topologies computed using (3.7) with weights coming from (3.5); the species tree topology is shown first. Node labels: localPP support when not equal to 100%. Dashed: focal branches that differ from ASTRAL-III. (c) Local PP of wASTRAL-h internal branches versus the number of genes k for each branch found in the wASTRAL-h output tree with all gene trees as input (x-axis). The inset with right y-axis scale shows the internal branch lengths in coalescent units on ASTRAL-III tree, sorted from low to high. The leftmost three branches are found only with $k \geq 100000$. 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 dataset for k ranging from 1000 to 450000; Slopes of fitted lines, which estimate asymptotic growth exponent, are labeled. All test cases are performed on a server with AMD EPYC 7742 CPUs.

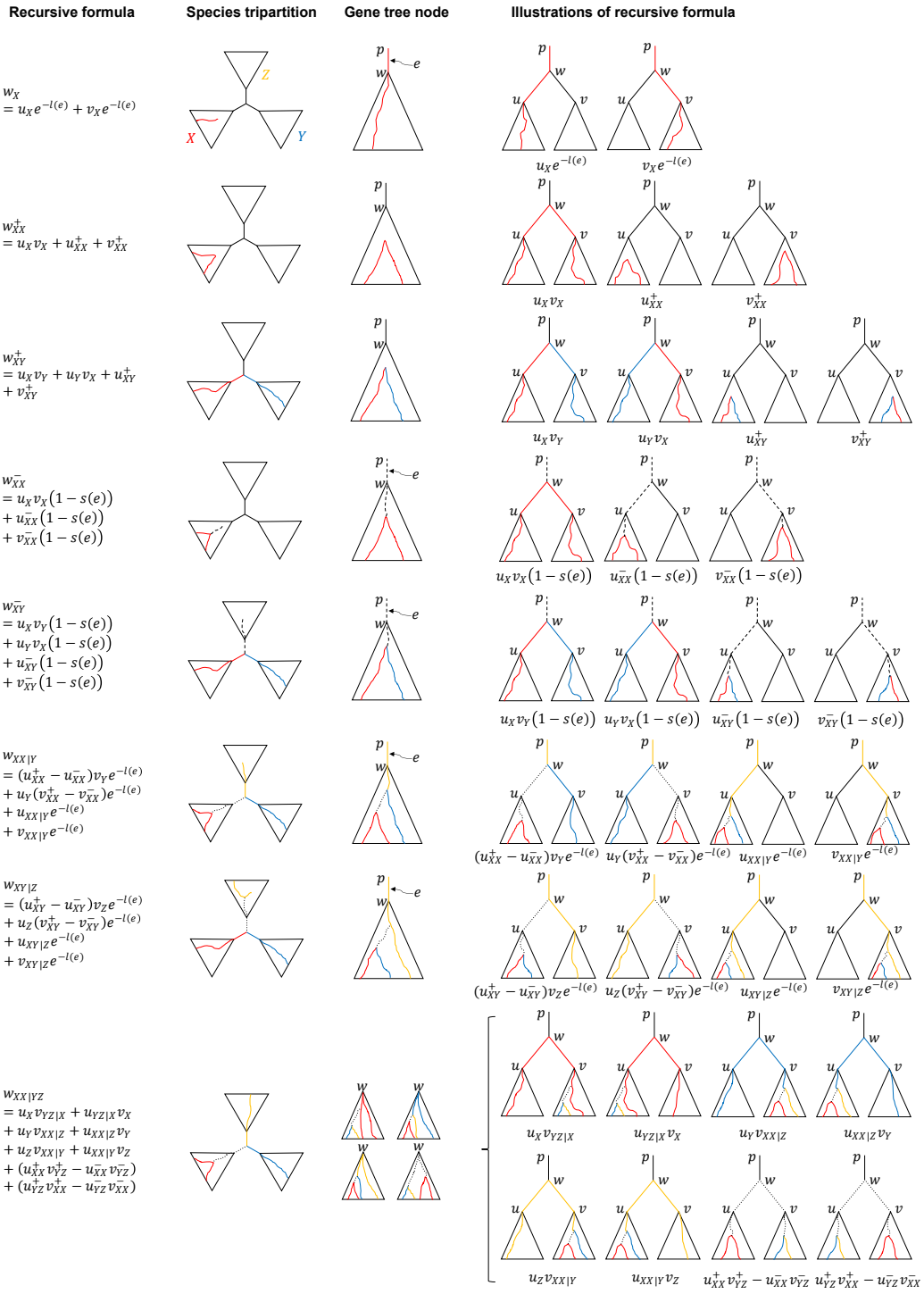

Figure 3.6. Recursive definitions of Counters. For a species tree tripartition $(X|Y|Z)$, and a gene tree node w, we compute the total hybrid weight of all quartets anchored at the species tripartition and with w as the MRCA on the gene tree. Each solid colored path is weighed by the negative exponent of its length; each dashed path is weighted by one minus its support; each dotted path is weighted by its support. See also Table S3.1.

Bibliography

M. Alanjary, K. Steinke, and N. Ziemert. AutoMLST: an automated web server for generating multi-locus species trees highlighting natural product potential. *Nucleic Acids Research*, 47(W1):W276–W282, 7 2019. ISSN 0305-1048. doi: 10.1093/NAR/GKZ282. URL https://academic.oup.com/nar/article/47/W1/W276/5475077.

R. Allio, C. Scornavacca, B. Nabholz, A.-L. Clamens, F. A. Sperling, and F. L. Condamine. Whole Genome Shotgun Phylogenomics Resolves the Pattern and Timing of Swallowtail Butterfly Evolution. *Systematic Biology*, 69(1):38–60, 1 2020. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYZ030. URL https://academic.oup.com/sysbio/article/69/1/38/5486398.

E. Avni, R. Cohen, and S. Snir. Weighted Quartets Phylogenetics. *Systematic Biology*, 64(2): 233–242, 3 2015. ISSN 1076-836X. doi: 10.1093/sysbio/syu087. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syu087.

M. S. Bayzid and T. Warnow. Naive binning improves phylogenomic analyses. *Bioinformatics*, 29(18):2277–84, 2013. ISSN 1367-4811. doi: 10.1093/bioinformatics/btt394. URL http://www.ncbi.nlm.nih.gov/pubmed/23842808.

M. S. Bayzid, S. Mirarab, B. Boussau, and T. Warnow. Weighted Statistical Binning: Enabling Statistically Consistent Genome-Scale Phylogenetic Analyses. *PLoS ONE*, 10(6):e0129183, 1 2015. ISSN 1932-6203. doi: 10.1371/journal.pone.0129183. URL http://dx.doi.org/10.1371/journal.pone.0129183.

S. Bossert, E. A. Murray, A. Pauly, K. Chernyshov, S. G. Brady, and B. N. Danforth. Gene Tree Estimation Error with Ultraconserved Elements: An Empirical Study on Pseudapis Bees. *Systematic Biology*, 70(4):803–821, 6 2021. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYAA097. URL https://academic.oup.com/sysbio/article/70/4/803/6050959.

E. L. Braun, J. Cracraft, and P. Houde. 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, 2019. doi: 10.1007/978-3-030-16477-5{_}6. URL http://link.springer.com/10.1007/978-3-030-16477-5_6.

G. S. Brodal, R. Fagerberg, T. Mailund, C. N. S. Pedersen, and A. Sand. 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, 1 2013. Society for Industrial and Applied Mathematics. ISBN 978-1-61197-251-1. doi: 10.1137/1.9781611973105.130. URL https://epubs.siam.org/doi/10.1137/1.9781611973105.130.

R. Davidson, P. Vachaspati, S. Mirarab, and T. Warnow. Phylogenomic species tree estimation in the presence of incomplete lineage sorting and horizontal gene transfer. *BMC Genomics*, 16(Suppl 10):S1, 2015. ISSN 1471-2164. doi: 10.1186/1471-2164-16-S10-S1. URL http://www.biomedcentral.com/1471-2164/16/S10/S1.

M. DeGiorgio and J. H. Degnan. Robustness to Divergence Time Underestimation When Inferring Species Trees from Estimated Gene Trees. *Systematic Biology*, 63(1):66–82, 1 2014. ISSN 1076-836X. doi: 10.1093/sysbio/syt059. URL http://www.ncbi.nlm.nih.gov/pubmed/23988674https://academic.oup.com/sysbio/article/63/1/66/1688532.

J. H. Degnan. Anomalous Unrooted Gene Trees. *Systematic Biology*, 62(4):574–590, 7 2013. ISSN 1063-5157. doi: 10.1093/sysbio/syt023. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syt023.

J. H. Degnan and N. A. Rosenberg. Gene tree discordance, phylogenetic inference and the multispecies coalescent. *Trends in Ecology and Evolution*, 24(6):332–340, 6 2009. ISSN 01695347. doi: 10.1016/j.tree.2009.01.009. URL http://www.cell.com/ecology-evolution/abstract/S0169-5347(09)00084-6http://www.sciencedirect.com/science/article/pii/S0169534709000846.

S. V. Edwards. Is a new and general theory of molecular systematics emerging? *Evolution*, 63 (1):1–19, 2009. ISSN 1558-5646. doi: 10.1111/j.1558-5646.2008.00549.x.

S. V. Edwards, Z. Xi, A. Janke, B. C. Faircloth, J. E. McCormack, T. C. Glenn, B. Zhong, S. Wu, E. M. Lemmon, A. R. Lemmon, A. D. Leaché, L. Liu, and C. C. Davis. Implementing and testing the multispecies coalescent model: A valuable paradigm for phylogenomics. *Molecular Phylogenetics and Evolution*, 94:447–462, 2016. ISSN 10959513. doi: 10.1016/j.ympev.2015.10.027. URL http://dx.doi.org/10.1016/j.ympev.2015.10.027.

R. A. L. Elworth, H. A. Ogilvie, J. Zhu, and L. Nakhleh. Advances in Computational Methods for Phylogenetic Networks in the Presence of Hybridization. pages 317–360. 2019. doi: 10.1007/978-3-030-10837-3{_}13. URL http://link.springer.com/10.1007/978-3-030-10837-3_13.

P. Erdos, M. Steel, L. Szekely, and T. Warnow. A few logs suffice to build (almost) all trees: Part II. *Theoretical Computer Science*, 221(1-2):77–118, 1999. ISSN 03043975. doi: 10.1016/S0304-3975(99)00028-6.

J. S. Farris. A Successive Approximations Approach to Character Weighting. *Systematic*

Biology, 18(4):374–385, 12 1969. ISSN 1063-5157. doi: 10.2307/2412182. URL https://academic.oup.com/sysbio/article/18/4/374/1699090.

J. Felsenstein and H. Kishino. Is there something wrong with the bootstrap on phylogenies? A reply to Hillis and Bull. *Systematic Biology*, 42(2):193–200, 1993. ISSN 10635157. doi: 10.1093/sysbio/42.2.193. URL http://www.jstor.org/stable/10.2307/2992541.

W. Fletcher and Z. Yang. INDELible: A Flexible Simulator of Biological Sequence Evolution. *Molecular Biology and Evolution*, 26(8):1879–1888, 8 2009. ISSN 0737-4038. doi: 10.1093/molbev/msp098. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msp098.

J. Gatesy, D. B. Sloan, J. M. Warren, R. H. Baker, M. P. Simmons, and M. S. Springer. Partitioned coalescence support reveals biases in species-tree methods and detects gene trees that determine phylogenomic conflicts. *Molecular Phylogenetics and Evolution*, 139:106539, 10 2019. ISSN 10557903. doi: 10.1016/j.ympev.2019.106539. URL https://linkinghub.elsevier.com/retrieve/pii/S1055790318307036.

T. C. Giarla and J. A. Esselstyn. The Challenges of Resolving a Rapid, Recent Radiation: Empirical and Simulated Phylogenomics of Philippine Shrews. *Systematic Biology*, 64(5):727–740, 9 2015. ISSN 1063-5157. doi: 10.1093/sysbio/syv029. URL http://dx.doi.org/10.1093/sysbio/syv029https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syv029.

S. Gopalakrishnan, M.-H. S. Sinding, J. Ramos-Madrigal, J. Niemann, J. A. Samaniego Castruita, F. G. Vieira, C. Carøe, M. d. M. Montero, L. Kuderna, A. Serres, V. M. González-Basallote, Y.-H. Liu, G.-D. Wang, T. Marques-Bonet, S. Mirarab, C. Fernandes, P. Gaubert, K.-P. Koepfli, J. Budd, E. K. Rueness, C. Sillero, M. P. Heide-Jørgensen, B. Petersen, T. Sicheritz-Ponten, L. Bachmann, Ø. Wiig, A. J. Hansen, and M. T. P. Gilbert. Interspecific Gene Flow Shaped the Evolution of the Genus Canis. *Current Biology*, 28(21):3441–3449, 11 2018. ISSN 09609822. doi: 10.1016/j.cub.2018.08.041. URL https://linkinghub.elsevier.com/retrieve/pii/S0960982218311254.

W. Guo, D. Sun, Y. Cao, L. Xiao, X. Huang, W. Ren, S. Xu, and G. Yang. Extensive Interspecific Gene Flow Shaped Complex Evolutionary History and Underestimated Species Diversity in Rapidly Radiated Dolphins. *Journal of Mammalian Evolution*, 29(2):353–367, 6 2022. ISSN 1064-7554. doi: 10.1007/s10914-021-09581-6. URL https://link.springer.com/10.1007/s10914-021-09581-6.

M. Hill, B. Legried, and S. Roch. Species tree estimation under joint modeling of coalescence and duplication: sample complexity of quartet methods. 2020. URL http://arxiv.org/abs/2007.06697.

D. M. Hillis and J. J. Bull. An Empirical Test of Bootstrapping as a Method for Assessing Confidence in Phylogenetic Analysis. *Systematic Biology*, 42(2):182–192, 1993. ISSN 1063-

5157. doi: 10.1093/sysbio/42.2.182. URL http://sysbio.oxfordjournals.org/content/42/2/182.short.

H. Huang and L. L. Knowles. Unforeseen Consequences of Excluding Missing Data from Next-Generation Sequences: Simulation Study of RAD Sequences. *Systematic Biology*, 65(3): 357–365, 5 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syu046. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syu046.

E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. W. Ho, B. C. Faircloth, B. Nabholz, J. T. Howard, A. Suh, C. C. Weber, R. R. da Fonseca, J. Li, F. Zhang, H. Li, L. Zhou, N. Narula, L. Liu, G. Ganapathy, B. Boussau, M. S. Bayzid, V. Zavidovych, S. Subramanian, T. Gabaldon, S. Capella-Gutierrez, J. Huerta-Cepas, B. Rekepalli, K. Munch, M. Schierup, B. Lindow, W. C. Warren, D. Ray, R. E. Green, M. W. Bruford, X. Zhan, A. Dixon, S. Li, N. Li, Y. Huang, E. P. Derryberry, M. F. Bertelsen, F. H. Sheldon, R. T. Brumfield, C. V. Mello, P. V. Lovell, M. Wirthlin, M. P. C. Schneider, F. Prosdocimi, J. A. Samaniego, A. M. V. Velazquez, A. Alfaro-Nunez, P. F. Campos, B. Petersen, T. Sicheritz-Ponten, A. Pas, T. Bailey, P. Scofield, M. Bunce, D. M. Lambert, Q. Zhou, P. Perelman, A. C. Driskell, B. Shapiro, Z. Xiong, Y. Zeng, S. Liu, Z. Li, B. Liu, K. Wu, J. Xiao, X. Yinqi, Q. Zheng, Y. Zhang, H. Yang, J. Wang, L. Smeds, F. E. Rheindt, M. Braun, J. Fjeldsa, L. Orlando, F. K. Barker, K. A. Jonsson, W. Johnson, K.-P. Koepfli, S. O'Brien, D. Haussler, O. A. Ryder, C. Rahbek, E. Willerslev, G. R. Graves, T. C. Glenn, J. McCormack, D. Burt, H. Ellegren, P. Alstrom, S. V. Edwards, A. Stamatakis, D. P. Mindell, J. Cracraft, E. L. Braun, T. Warnow, W. Jun, M. T. P. Gilbert, and G. Zhang. Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331, 12 2014. ISSN 0036-8075. doi: 10.1126/science.1253451. URL http://www.sciencemag.org/content/346/6215/1320.abstracthttp://www.sciencemag.org/cgi/doi/10.1126/science.1253451.

O. Jeffroy, H. Brinkmann, F. Delsuc, and H. Philippe. Phylogenomics: the beginning of incongruence? *Trends in Genetics*, 22(4):225–231, 2006. ISSN 01689525. doi: 10.1016/j.tig.2006.02.003. URL http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&dopt=Citation&list_uids=16490279.

A. Y. Kawahara, D. Plotkin, M. Espeland, K. Meusemann, E. F. A. Toussaint, A. Donath, F. Gimnich, P. B. Frandsen, A. Zwick, M. d. Reis, J. R. Barber, R. S. Peters, S. Liu, X. Zhou, C. Mayer, L. Podsiadlowski, C. Storer, J. E. Yack, B. Misof, and J. W. Breinholt. Phylogenomics reveals the evolutionary timing and pattern of butterflies and moths. *Proceedings of the National Academy of Sciences*, 116(45):22657–22663, 11 2019. ISSN 0027-8424. doi: 10.1073/PNAS.1907847116. URL https://www.pnas.org/content/116/45/22657https://www.pnas.org/content/116/45/22657.abstract.

L. L. Knowles, H. C. Lanier, P. B. Klimov, and Q. He. Full modeling versus summarizing gene-tree uncertainty: Method choice and species-tree accuracy. *Molecular Phylogenetics and Evolution*, 65(2):501–509, 11 2012. ISSN 1095-9513. doi: 10.1016/j.ympev.2012.07.004. URL http://www.sciencedirect.com/science/article/pii/S1055790312002618.

H. C. Lanier and L. L. Knowles. 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, 2 2015. ISSN 10557903. doi: 10.1016/j.ympev.2014.10.022. URL https://linkinghub.elsevier.com/retrieve/pii/S1055790314003820.

A. D. Leaché and B. Rannala. The accuracy of species tree estimation under simulation: A comparison of methods. *Systematic Biology*, 60(2):126–137, 3 2011. ISSN 1076-836X. doi: 10.1093/sysbio/syq073. URL http://www.ncbi.nlm.nih.gov/pubmed/21088009.

A. D. Leaché, B. L. Banbury, J. Felsenstein, A. N. M. De Oca, and A. Stamatakis. Short tree, long tree, right tree, wrong tree: New acquisition bias corrections for inferring SNP phylogenies. *Systematic Biology*, 64(6):1032–1047, 2015. ISSN 1076836X. doi: 10.1093/sysbio/syv053.

B. Legried, E. K. Molloy, T. Warnow, and S. Roch. Polynomial-Time Statistical Estimation of Species Trees Under Gene Duplication and Loss. *Journal of Computational Biology*, 28(5):452–468, 5 2021. ISSN 1557-8666. doi: 10.1089/cmb.2020.0424. URL http://link.springer.com/10.1007/978-3-030-45257-5_8https://www.liebertpub.com/doi/10.1089/cmb.2020.0424.

L. Liu and L. Yu. Estimating Species Trees from Unrooted Gene Trees. *Systematic Biology*, 60(5):661–667, 10 2011. ISSN 1076-836X. doi: 10.1093/sysbio/syr027. URL https://academic.oup.com/sysbio/article/60/5/661/1644054.

L. Liu, L. Yu, D. K. Pearl, and S. V. Edwards. Estimating species phylogenies using coalescence times among sequences. *Systematic Biology*, 58(5):468–477, 10 2009. ISSN 10635157. doi: 10.1093/sysbio/syp031. URL http://www.ncbi.nlm.nih.gov/pubmed/20525601.

L. Liu, L. Yu, and S. V. Edwards. A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC Evolutionary Biology*, 10(1):302, 2010. URL http://www.biomedcentral.com/1471-2148/10/302.

W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 9 1997. ISSN 10635157. doi: 10.2307/2413694. URL http://www.jstor.org/stable/2413694?origin=crossrefhttp://sysbio.oxfordjournals.org/cgi/content/abstract/46/3/523http://sysbio.oxfordjournals.org/content/46/3/523.short.

U. Mai and S. Mirarab. Completing gene trees without species trees in sub-quadratic time. *Bioinformatics*, 38(6):1532–1541, 3 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab875. URL https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btab875/6493250https://academic.oup.com/bioinformatics/article/38/6/1532/6493250.

D. Mallo, L. De Oliveira Martins, and D. Posada. SimPhy : Phylogenomic Simulation of Gene, Locus, and Species Trees. *Systematic Biology*, 65(2):334–344, 3 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syv082. URL http://sysbio.oxfordjournals.org/content/early/2015/12/04/sysbio.syv082.short?rss=1https://academic.oup.com/sysbio/article-lookup/doi/10.1093/

sysbio/syv082http://www.ncbi.nlm.nih.gov/pubmed/265.

A. Markin and O. Eulenstein. Quartet-based inference is statistically consistent under the unified duplication-loss-coalescence model. *Bioinformatics*, page btab414, 5 2021. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab414. URL https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btab414/6287614.

M. R. McGowen, M. Spaulding, and J. Gatesy. Divergence date estimation and a comprehensive molecular tree of extant cetaceans. *Molecular Phylogenetics and Evolution*, 53(3):891–906, 12 2009. ISSN 10557903. doi: 10.1016/j.ympev.2009.08.018. URL https://linkinghub.elsevier.com/retrieve/pii/S1055790309003431.

M. R. McGowen, G. Tsagkogeorga, S. Álvarez-Carretero, M. dos Reis, M. Struebig, R. Deaville, P. D. Jepson, S. Jarman, A. Polanowski, P. A. Morin, and S. J. Rossiter. Phylogenomic Resolution of the Cetacean Tree of Life Using Target Sequence Capture. *Systematic Biology*, 69(3):479–501, 5 2020. ISSN 1063-5157. doi: 10.1093/sysbio/syz068. URL https://academic.oup.com/sysbio/article/69/3/479/5601630.

K. A. Meiklejohn, B. C. Faircloth, T. C. Glenn, R. T. Kimball, and E. L. Braun. Analysis of a Rapid Evolutionary Radiation Using Ultraconserved Elements: Evidence for a Bias in Some Multispecies Coalescent Methods. *Systematic Biology*, 65(4):612–627, 7 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syw014. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syw014.

S. Mirarab. Species Tree Estimation Using ASTRAL: Practical Considerations. *Arxiv preprint*, 1904.03826, 4 2019. URL http://arxiv.org/abs/1904.03826.

S. Mirarab and T. Warnow. ASTRAL-II: Coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 6 2015. ISSN 14602059. doi: 10.1093/bioinformatics/btv234. URL http://bioinformatics.oxfordjournals.org/cgi/content/long/31/12/i44http://bioinformatics.oxfordjournals.org/lookup/doi/10.1093/bioinformatics/btv234.

S. Mirarab, M. S. Bayzid, B. Boussau, and T. Warnow. Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, 346(6215):1250463–1250463, 12 2014a. ISSN 0036-8075. doi: 10.1126/science.1250463. URL http://www.sciencemag.org/cgi/doi/10.1126/science.1250463.

S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548, 9 2014b. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu462. URL http://bioinformatics.oxfordjournals.org/cgi/content/long/30/17/i541http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btu462https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btu462.

S. Mirarab, M. S. Bayzid, and T. Warnow. Evaluating Summary Methods for Multilocus Species Tree Estimation in the Presence of Incomplete Lineage Sorting. *Systematic Biology*, 65(3):366–380, 5 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syu063. URL http://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063%5Cnhttp://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063.abstract%5Cnhttp://sysbio.oxfordjournals.org/content/early/2014/10/13/sysbio.syu063.full.pdf%5Cnhttp://www.n.

S. Mirarab, L. Nakhleh, and T. Warnow. Multispecies Coalescent: Theory and Applications in Phylogenetics. *Annual Review of Ecology, Evolution, and Systematics*, 52(1):247–268, 11 2021. ISSN 1543-592X. doi: 10.1146/annurev-ecolsys-012121-095340. URL https://www.annualreviews.org/doi/10.1146/annurev-ecolsys-012121-095340.

E. K. Molloy and T. Warnow. To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303, 3 2018. ISSN 1063-5157. doi: 10.1093/sysbio/syx077. URL https://academic.oup.com/sysbio/article/67/2/285/4159193.

E. Mossel and S. Roch. Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 7(1):166–171, 1 2010. ISSN 1557-9964. doi: 10.1109/TCBB.2008.66. URL http://dl.acm.org/citation.cfm?id=1719272.1719288.

A. E. Moura, K. Shreves, M. Pilot, K. R. Andrews, D. M. Moore, T. Kishida, L. Möller, A. Natoli, S. Gaspari, M. McGowen, I. Chen, H. Gray, M. Gore, R. M. Culloch, M. S. Kiani, M. S. Willson, A. Bulushi, T. Collins, R. Baldwin, A. Willson, G. Minton, L. Ponnampalam, and A. R. Hoelzel. Phylogenomics of the genus Tursiops and closely related Delphininae reveals extensive reticulation among lineages and provides inference about eco-evolutionary drivers. *Molecular Phylogenetics and Evolution*, 146:106756, 5 2020. ISSN 10557903. doi: 10.1016/j.ympev.2020.106756. URL https://linkinghub.elsevier.com/retrieve/pii/S1055790320300282.

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

H. A. Ogilvie, R. R. Bouckaert, and A. J. Drummond. StarBEAST2 Brings Faster Species Tree Inference and Accurate Estimates of Substitution Rates. *Molecular Biology and Evolution*, 34(8):2101–2114, 8 2017. ISSN 0737-4038. doi: 10.1093/molbev/msx126. URL https://academic.oup.com/mbe/article/34/8/2101/3738283.

O. T. P. T. OneKP Initiative. One thousand plant transcriptomes and the phylogenomics of green plants. *Nature*, 574(7780):679–685, 10 2019. ISSN 0028-0836. doi: 10.1038/s41586-019-1693-2. URL http://www.nature.com/articles/s41586-019-1693-2.

S. Patel. Error in Phylogenetic Estimation for Bushes in the Tree of Life.

Journal of Phylogenetics & Evolutionary Biology, 01(02):110, 2013. ISSN 23299002. doi: 10.4172/2329-9002.1000110. URL http://esciencecentral.org/journals/error-in-phylogenetic-estimation-for-bushes-in-the-tree-of-life-2329-9002.1000110.pdfhttp://www.esciencecentral.org/journals/error-in-phylogenetic-estimation-for-bushes-in-the-tree-of-life-2329-9002.1000110.php?aid=154.

H. Philippe, D. M. d. Vienne, V. Ranwez, B. Roure, D. Baurain, and F. Delsuc. Pitfalls in supermatrix phylogenomics. *European Journal of Taxonomy*, 2017. ISSN 2118-9773. doi: 10.5852/ejt.2017.283.

M. N. Price, P. S. Dehal, and A. P. Arkin. FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 3 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0009490. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2835736&tool=pmcentrez&rendertype=abstract.

S. Reddy, R. T. Kimball, A. Pandey, P. A. Hosner, M. J. Braun, S. J. Hackett, K.-L. Han, J. Harshman, C. J. Huddleston, and S. Kingston. 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, 2017. ISSN 1063-5157.

A. Richards and L. Kubatko. Bayesian-Weighted Triplet and Quartet Methods for Species Tree Inference. *Bulletin of Mathematical Biology*, 83(9):93, 9 2021. ISSN 0092-8240. doi: 10.1007/s11538-021-00918-z. URL https://link.springer.com/10.1007/s11538-021-00918-z.

D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, 1981. URL http://www.sciencedirect.com/science/article/pii/0025556481900432.

S. Roch and M. Steel. Likelihood-based tree reconstruction on a concatenation of aligned sequence data sets can be statistically inconsistent. *Theoretical Population Biology*, 100:56–62, 3 2015. ISSN 00405809. doi: 10.1016/j.tpb.2014.12.005. URL http://www.sciencedirect.com/science/article/pii/S0040580914001075https://linkinghub.elsevier.com/retrieve/pii/S0040580914001075.

S. Roch, M. Nute, and T. Warnow. Long-Branch Attraction in Species Tree Estimation: Inconsistency of Partitioned Likelihood and Topology-Based Summary Methods. *Systematic Biology*, 68(2):281–297, 3 2019. ISSN 1063-5157. doi: 10.1093/sysbio/syy061. URL https://academic.oup.com/sysbio/article/68/2/281/5104882.

E. Sayyari and S. Mirarab. Fast Coalescent-Based Computation of Local Branch Support from Quartet Frequencies. *Molecular Biology and Evolution*, 33(7):1654–1668, 7 2016. ISSN 0737-4038. doi: 10.1093/molbev/msw079. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msw079.

E. Sayyari and S. Mirarab. Testing for Polytomies in Phylogenetic Species Trees Using Quartet Frequencies. *Genes*, 9(3):132, 2 2018. ISSN 2073-4425. doi: 10.3390/genes9030132. URL http://www.mdpi.com/268060http://www.mdpi.com/2073-4425/9/3/132.

E. Sayyari, J. B. Whitfield, and S. Mirarab. DiscoVista: Interpretable visualizations of gene tree discordance. *Molecular Phylogenetics and Evolution*, 122:110–115, 5 2018. ISSN 10557903. doi: 10.1016/j.ympev.2018.01.019. URL https://doi.org/10.1016/j.ympev.2018.01.019https://linkinghub.elsevier.com/retrieve/pii/S1055790317306590.

T.-K. Seo. Calculating Bootstrap Probabilities of Phylogeny Using Multilocus Sequence Data. *Molecular Biology and Evolution*, 25(5):960–971, 2 2008. ISSN 0737-4038. doi: 10.1093/molbev/msn043. URL http://mbe.oxfordjournals.org/cgi/doi/10.1093/molbev/msn043.

S. Shekhar, S. Roch, and S. Mirarab. Species tree estimation using ASTRAL: how many genes are enough? *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(5):1738–1747, 4 2017. ISSN 15579964. doi: 10.1109/TCBB.2017.2757930. URL http://ieeexplore.ieee.org/document/8053780/https://ieeexplore.ieee.org/document/8053780/http://arxiv.org/abs/1704.06831http://dx.doi.org/10.1109/TCBB.2017.2757930.

X.-x. Shen, C. T. Hittinger, and A. Rokas. Contentious relationships in phylogenomic studies can be driven by a handful of genes. *Nature Ecology & Evolution*, 1(5):0126, 4 2017. ISSN 2397-334X. doi: 10.1038/s41559-017-0126. URL http://dx.doi.org/10.1038/s41559-017-0126http://www.nature.com/articles/s41559-017-0126.

M. P. Simmons and J. Gatesy. Coalescence vs. concatenation: sophisticated analyses vs. first principles applied to rooting the angiosperms. *Molecular phylogenetics and evolution*, 91:98–122, 5 2015. ISSN 1095-9513. doi: 10.1016/j.ympev.2015.05.011. URL http://www.sciencedirect.com/science/article/pii/S1055790315001487.

M. L. Smith and M. W. Hahn. New Approaches for Inferring Phylogenies in the Presence of Paralogs. *Trends in Genetics*, 37(2):174–187, 2 2021. ISSN 01689525. doi: 10.1016/j.tig.2020.08.012. URL https://linkinghub.elsevier.com/retrieve/pii/S0168952520302122.

S. A. Smith, M. J. Moore, J. W. Brown, and Y. Yang. Analysis of phylogenomic datasets reveals conflict, concordance, and gene duplications with examples from animals and plants. *BMC Evolutionary Biology*, 15(1):150, 12 2015. ISSN 1471-2148. doi: 10.1186/s12862-015-0423-0. URL https://bmcevolbiol.biomedcentral.com/articles/10.1186/s12862-015-0423-0.

S. Snir, T. Warnow, and S. Rao. Short Quartet Puzzling: A New Quartet-Based Phylogeny Reconstruction Algorithm. *Journal of Computational Biology*, 15(1):91–103, 1 2008. ISSN 1066-5277. doi: 10.1089/cmb.2007.0103. URL http://www.ncbi.nlm.nih.gov/pubmed/18199023http://www.liebertpub.com/doi/10.1089/cmb.2007.0103.

C. Solís-Lemus, M. Yang, and C. Ané. Inconsistency of Species Tree Methods under Gene Flow.

Systematic Biology, 65(5):843–851, 9 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syw030. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syw030.

M. S. Springer and J. Gatesy. The gene tree delusion. *Molecular Phylogenetics and Evolution*, 94(Part A):1–33, 1 2016. ISSN 10557903. doi: 10.1016/j.ympev.2015.07.018. URL http://www.sciencedirect.com/science/article/pii/S1055790315002225http://linkinghub.elsevier.com/retrieve/pii/S1055790315002225http://dx.doi.org/10.1016/j.ympev.2015.07.018https://linkinghub.elsevier.com/retrieve/pii/S1055790315002225.

M. S. Springer and J. Gatesy. On the importance of homology in the age of phylogenomics. *Systematics and Biodiversity*, 16(3):210–228, 4 2018. ISSN 1477-2000. doi: 10.1080/14772000.2017.1401016. URL https://www.tandfonline.com/doi/full/10.1080/14772000.2017.1401016.

E. Susko. Bootstrap support is not first-order correct. *Systematic Biology*, 58(2):211–223, 2009. ISSN 10635157. doi: 10.1093/sysbio/syp016.

A. Suvorov, J. Hochuli, and D. R. Schrider. Accurate Inference of Tree Topologies from Multiple Sequence Alignments Using Deep Learning. *Systematic Biology*, 69(2):221–233, 9 2020. ISSN 1063-5157. doi: 10.1093/sysbio/syz060. URL https://academic.oup.com/sysbio/advance-article/doi/10.1093/sysbio/syz060/5559282.

G. J. Szöllõsi, E. Tannier, V. Daubin, and B. Boussau. The inference of gene trees with species trees. *Systematic Biology*, 64(1):e42–e62, 7 2014. ISSN 1063-5157. doi: 10.1093/sysbio/syu048. URL http://sysbio.oxfordjournals.org/cgi/content/long/64/1/e42.

N. Takahata. Gene genealogy in three related populations: consistency probability between gene and population trees. *Genetics*, 122(4):957–966, 1989. ISSN 0016-6731.

P. Vachaspati and T. Warnow. ASTRID: Accurate Species TRees from Internode Distances. *BMC Genomics*, 16(Suppl 10):S3, 2015. ISSN 1471-2164.

J. F. Walker, J. W. Brown, and S. A. Smith. Analyzing Contentious Relationships and Outlier Genes in Phylogenomics. *Systematic Biology*, 67(5):916–924, 9 2018. ISSN 1063-5157. doi: 10.1093/sysbio/syy043. URL https://academic.oup.com/sysbio/article/67/5/916/5034973.

L. G. Wang, T. T. Y. Lam, S. Xu, Z. Dai, L. Zhou, T. Feng, P. Guo, C. W. Dunn, B. R. Jones, T. Bradley, H. Zhu, Y. Guan, Y. Jiang, and G. Yu. Treeio: An R Package for Phylogenetic Tree Input and Output with Richly Annotated and Associated Data. *Molecular Biology and Evolution*, 37(2):599–603, 2 2020. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSZ240. URL https://academic.oup.com/mbe/article/37/2/599/5601621.

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

2001. ISBN 0898714907.

Z. Yan, M. L. Smith, P. Du, M. W. Hahn, and L. Nakhleh. Species Tree Inference Methods Intended to Deal with Incomplete Lineage Sorting Are Robust to the Presence of Paralogs. *Systematic Biology*, page 498378, 7 2021. ISSN 1063-5157. doi: 10.1093/sysbio/syab056. URL https://www.biorxiv.org/content/10.1101/498378v2https://academic.oup.com/sysbio/advance-article/doi/10.1093/sysbio/syab056/6318793.

J. Yin, C. Zhang, and S. Mirarab. ASTRAL-MP: scaling ASTRAL to very large datasets using randomization and parallelization. *Bioinformatics*, 35(20):3961–3969, 10 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz211. URL https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btz211/5418955https://academic.oup.com/bioinformatics/article/35/20/3961/5418955.

S. Yourdkhani and J. A. Rhodes. Inferring Metric Trees from Weighted Quartets via an Intertaxon Distance. *Bulletin of Mathematical Biology*, 82(7):97, 7 2020. ISSN 0092-8240. doi: 10.1007/s11538-020-00773-4. URL http://link.springer.com/10.1007/s11538-020-00773-4.

C. Zhang, M. Rabiee, E. Sayyari, and S. Mirarab. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6):153, 5 2018. ISSN 1471-2105. doi: 10.1186/s12859-018-2129-y. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-018-2129-y.

C. Zhang, C. Scornavacca, E. K. Molloy, and S. Mirarab. ASTRAL-Pro: Quartet-Based Species-Tree Inference despite Paralogy. *Molecular Biology and Evolution*, 37(11):3292–3307, 11 2020. ISSN 0737-4038. doi: 10.1093/molbev/msaa139. URL https://academic.oup.com/mbe/advance-article/doi/10.1093/molbev/msaa139/5850411https://academic.oup.com/mbe/article/37/11/3292/5850411.

Appendices

3.A Commands

3.A.1 Approximate Bayesian Branch Support Annotation

```
iqtree2 -s SEQ_ALIGNMENT -te GENE_TREE -m TVM+I+G4 -abayes
  -pre ANNOTATED_GENE_TREE
```

Note: When inferring support as a post-processing step, the same model used for inferring the tree should be used, a task that requires care when the original trees are inferred using a different tool (e.g., RAxML). TVM+I+G4 is simply an example.

3.A.2 Running wASTRAL

Exact commands when running on gene trees with approximate Bayesian/Bootstrap/SH-like supports.

```
astral-hybrid -x 1 -n 0.333 APPROXIMATE_BAYESIAN_ANNOTATED_GENE_TREE
astral-hybrid -x 100 -n 0 BOOTSTRAP_ANNOTATED_GENE_TREE
astral-hybrid -x 1 -n 0 SH_LIKE_ANNOTATED_GENE_TREE
```

3.B Supplementary Figures and Tables

Table S3.1. 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 the children of w; e be the parental edge of w; p be the parent of w; $\mathscr{P}_{x,w}$ be the path between x and w; $s(\mathscr{P}) = 1-\prod_{\hat{e}\in\mathscr{P}}(1-s(\hat{e}))$; $m(i,j) =$ MRCA of i and j. Counters for leaves are set to zero 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
	$\sum_i e^{-l(\mathscr{P}_{i,p})}$ for all leaf nodes i colored X under w
(w^+_{XX}, w^+_{XY})	$\big(u^+_{XX}+v^+_{XX}+u_Xv_X, u^+_{XY}+v^+_{XY}+u_Xv_Y+u_Yv_X\big)$
	$\sum_{i,j} e^{-l(\mathscr{P}_{i,j})}$ for all leaf nodes i colored X and j colored X/Y under w
(w^-_{XX}, w^-_{XY})	$\big((u^-_{XX}+v^-_{XX}+u_Xv_X)(1-s(e)), (u^-_{XY}+v^-_{XY}+u_Xv_Y+u_Yv_X)(1-s(e))\big)$
	$\sum_{i,j} e^{-l(\mathscr{P}_{i,j})}\big(1-s(\mathscr{P}_{m(i,j),p})\big)$ for all leaf nodes i colored X and j colored X/Y under w
$(w_{XX\|Y}, w_{XY\|Z})$	$\Big(\big(u_{XX\|Y}+v_{XX\|Y}+(u^+_{XX}-u^-_{XX})v_Y+u_Y(v^+_{XX}-v^-_{XX})\big)e^{-l(e)},$ $\big(u_{XY\|Z}+v_{XY\|Z}+(u^+_{XY}-u^-_{XY})v_Z+u_Z(v^+_{XY}-v^-_{XY})\big)e^{-l(e)}\Big)$
	$\sum_{i,j,k} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,p})}s(\mathscr{P}_{m(i,j),m(i,k)})$ for leaf nodes i colored X, j colored X/Y, k colored Z under w, and $m(i,j)$ under $m(i,k)$
$w_{XX\|YZ}$	$v_Xu_{YZ\|X}+u_Xv_{YZ\|X}+u_{XX\|Z}v_Y+v_{XX\|Z}u_Y+u_{XX\|Y}v_Z+v_{XX\|Y}u_Z$ $+(u^+_{YZ}v^+_{XX}-u^-_{YZ}v^-_{XX})+(u^+_{XX}v^+_{YZ}-u^-_{XX}v^-_{YZ})$
	$\sum_{h,i,j,k} w_G(hi\|jk)$ for all leaf nodes h,i colored X, j colored Y, k colored Z, and $w =$ MRCA h,i,j,k
Q	$\sum_{G\in\mathscr{G}}\sum_w(w_{AA\|BC}+w_{BB\|AC}+w_{CC\|AB})$ for internal nodes w in G
	$\sum_{G\in\mathscr{G}}\sum_{h,i,j,k} w_G(hi\|jk)$ 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, $Q = W[A\|B\|C] = \sum_{G\in\mathscr{G}} W(A\|B\|C,G)$ (Proposition 3.5).

Table S3.2. Running time of species tree inference methods on biological datasets. We use 5.17.3 version of ASTRAL-III if not otherwise clarified.

Dataset	n	k	Method	#Cores	Wall-clock time	CPU time
OneKP	1178	410	wASTRAL-h	16	17.1 min	4.57 hr
			ASTRAL-III (5.0.3)	1	17.2 hr	17.2 hr
Canis	48	449450	wASTRAL-h	1	17.7 hr	17.7 hr
Avian	48	14446	wASTRAL-h	16	1.76 min	28.1 min
			ASTRAL-III	16	20.9 min	5.57 hr
Cetacean	98	3191	wASTRAL-h	16	35.2 sec	9.39 min
			ASTRAL-III	16	1.97 min	31.5 min
Nomiinae	32	853	wASTRAL-h	1	5.93 sec	5.93 sec
			ASTRAL-III	1	8.64 sec	8.64 sec
Lepidoptera	203	1930	wASTRAL-h	16	2.02 min	32.3 min
			ASTRAL-III	16	9.14 min	2.44 hr
Papilionidae	61	6405	wASTRAL-h	16	24.8 sec	6.61 min
			ASTRAL-III	16	1.11 min	17.8 min

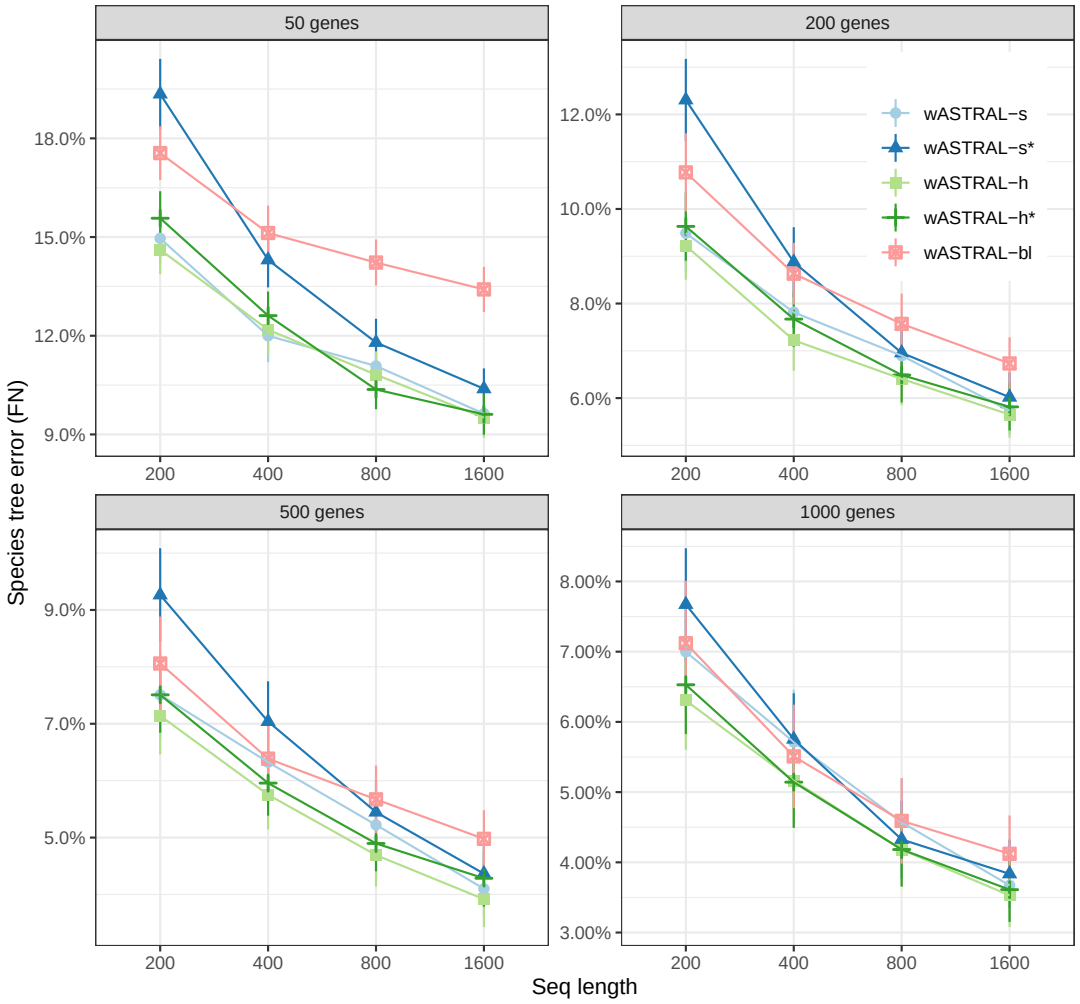

Figure S3.1. Species tree error by weighting scheme on the S100 dataset with $k = \{50, 200, 500, 1000\}$ and gene sequence length $\{200, 400, 800, 1600\}$. Results with aBayes supports are labeled wASTRAL-s and wASTRAL-h; results with bootstrap support are labelled wASTRAL-s* and wASTRAL-h*.

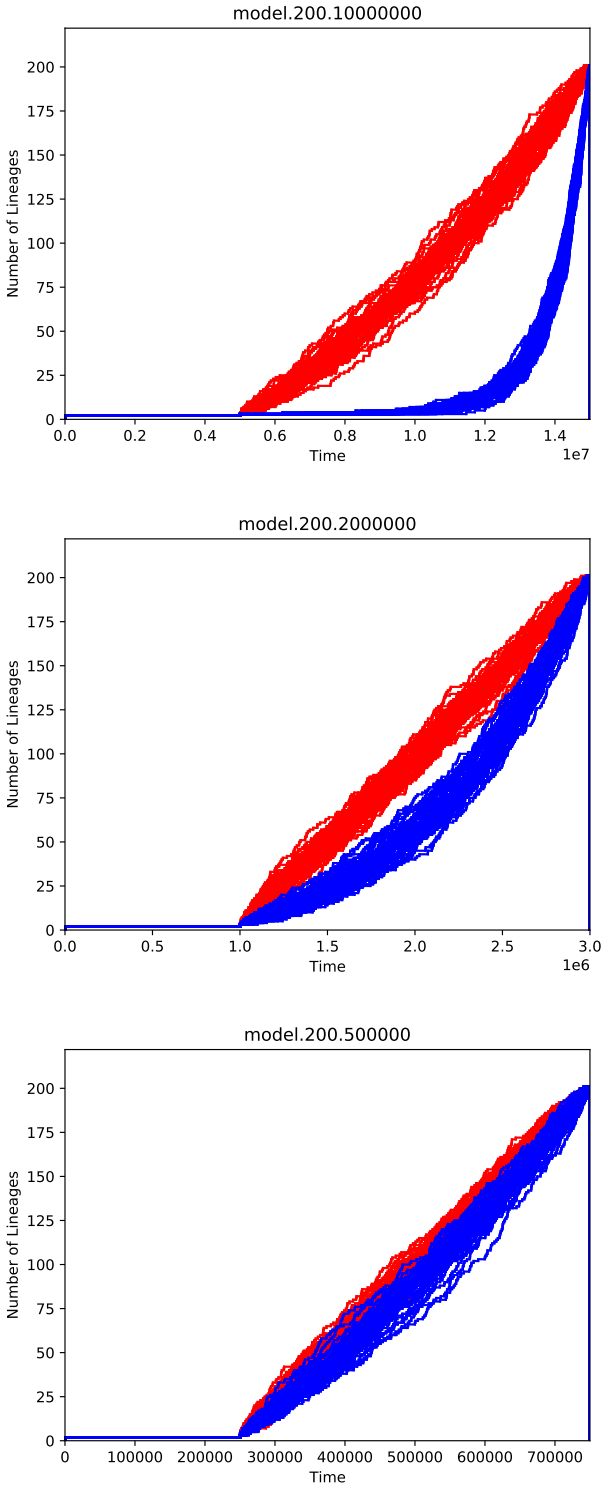

Figure S3.2. Lineage Through Time (LTT) plots for thee simulated model conditions with 10^{-7} (red) and 10^{-6} (blue) rates tend to lead to deeper and shallower speciation.

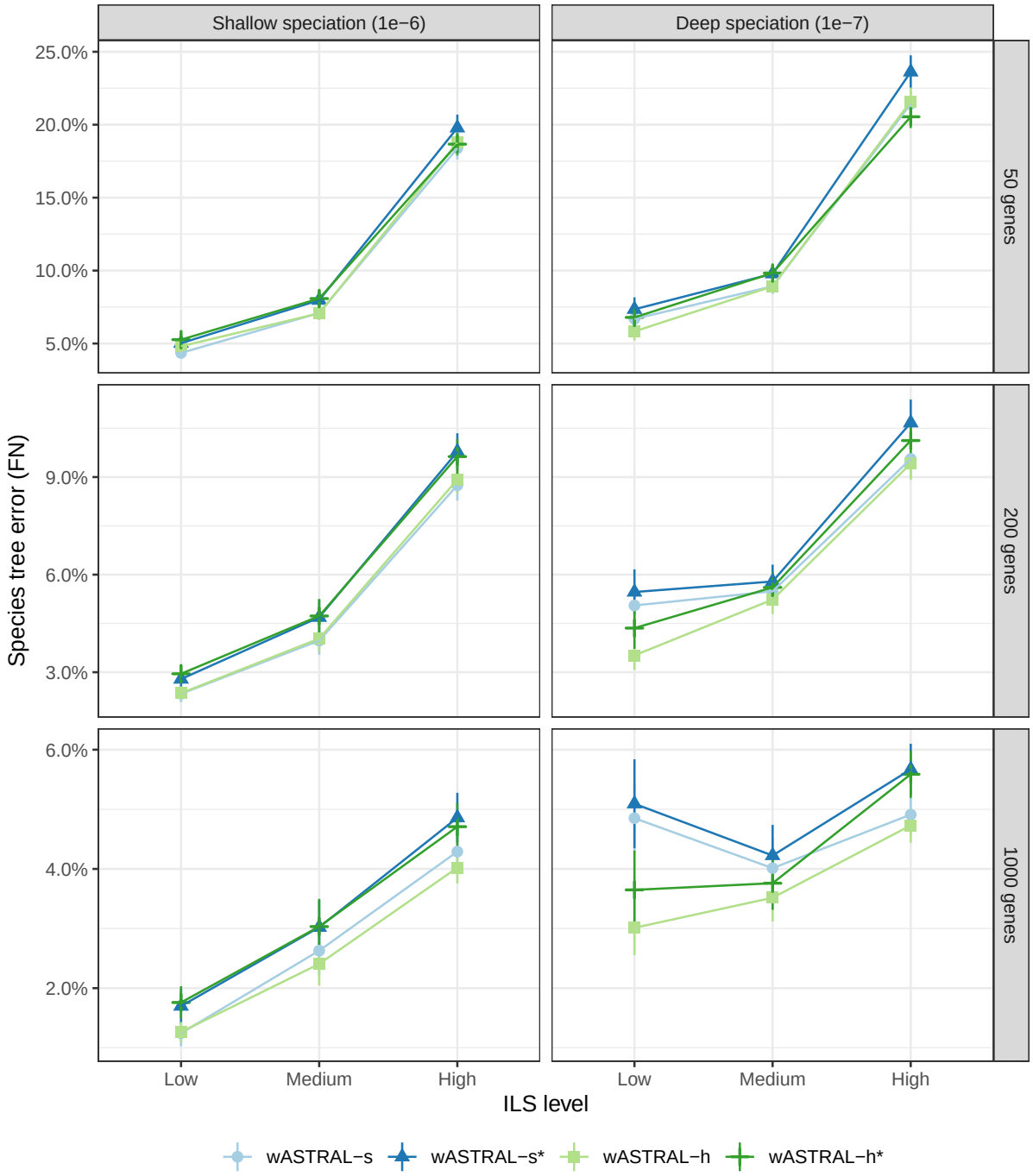

Figure S3.3. Species tree error by weighting scheme on the S200 dataset with $k = \{50, 200, 1000\}$ and population size (ILS levels). Species tree shape with parameters E1-6 and E1-7 are used. Results with aBayes supports are labeled wASTRAL-s and wASTRAL-h; results with SH-like support are labeled wASTRAL-s* and wASTRAL-h*.

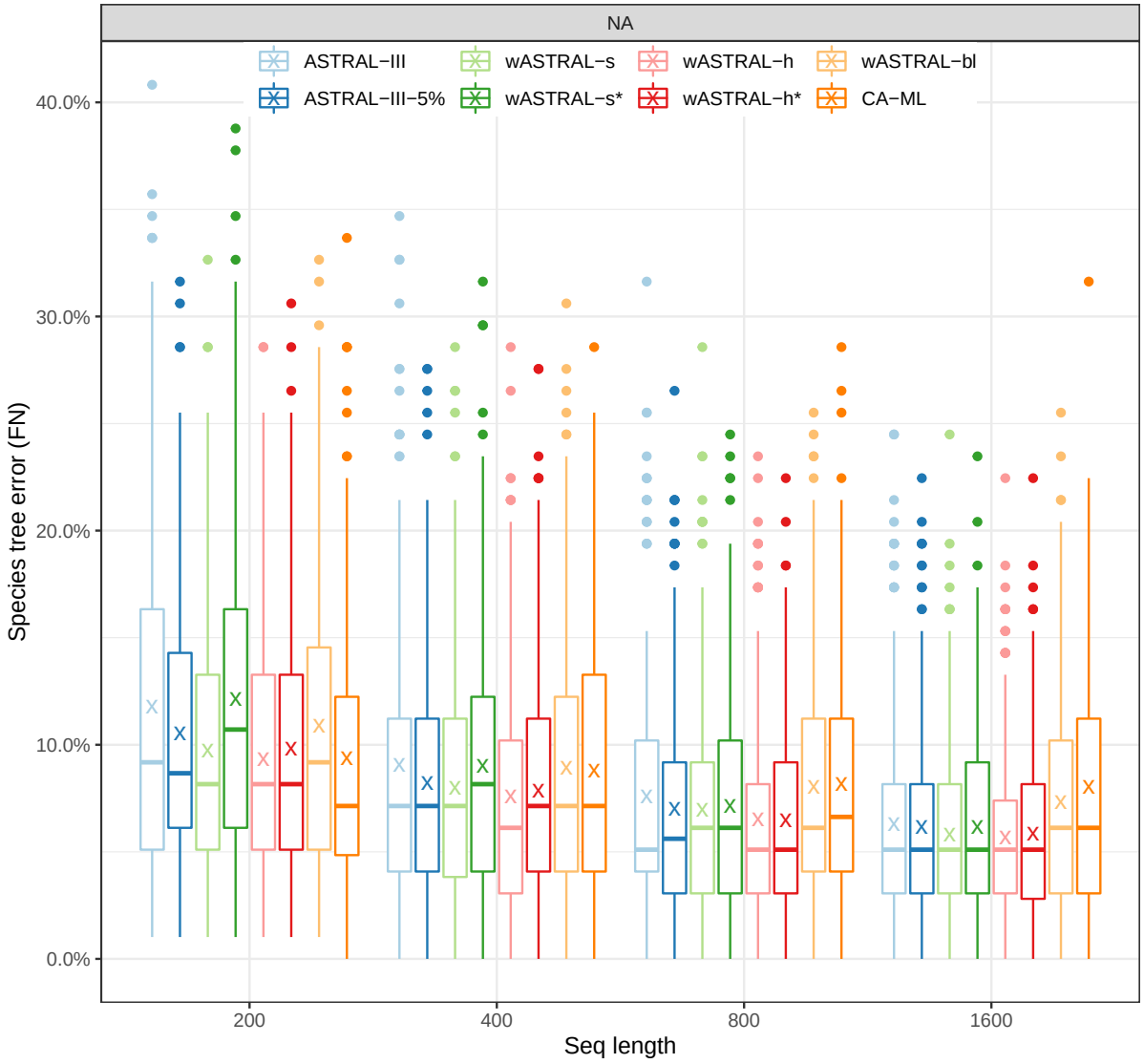

Figure S3.4. Species tree error on the S100 dataset with $k = \{50, 200, 500, 1000\}$ and gene sequence length $\{200, 400, 800, 1600\}$. Results with aBayes supports are labelled wASTRAL-s and wASTRAL-h; results with bootstrap support are labelled wASTRAL-s* and wASTRAL-h*.

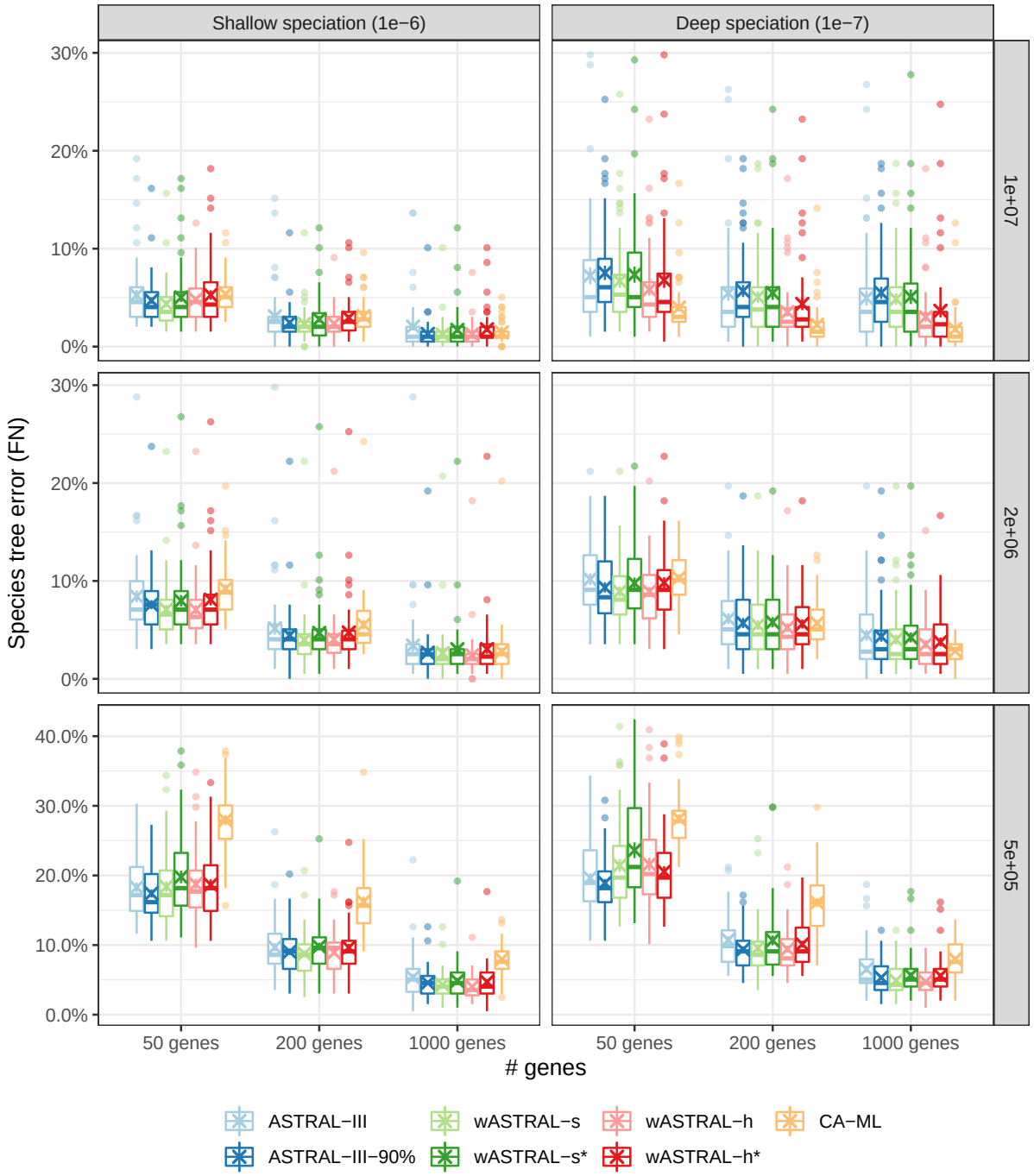

Figure S3.5. Species tree error on the S200 dataset with $k = \{50, 200, 1000\}$ and population size (ILS levels). Species tree shape with parameter E1-6 and E1-7 (box columns) and ILS levels (box rows) low (1e+07), medium (2e+06), and high (5e+05) are used. Results with Bayesian supports are labeled wASTRAL-s and wASTRAL-h; results with SH-like support are labeled wASTRAL-s* and wASTRAL-h*.

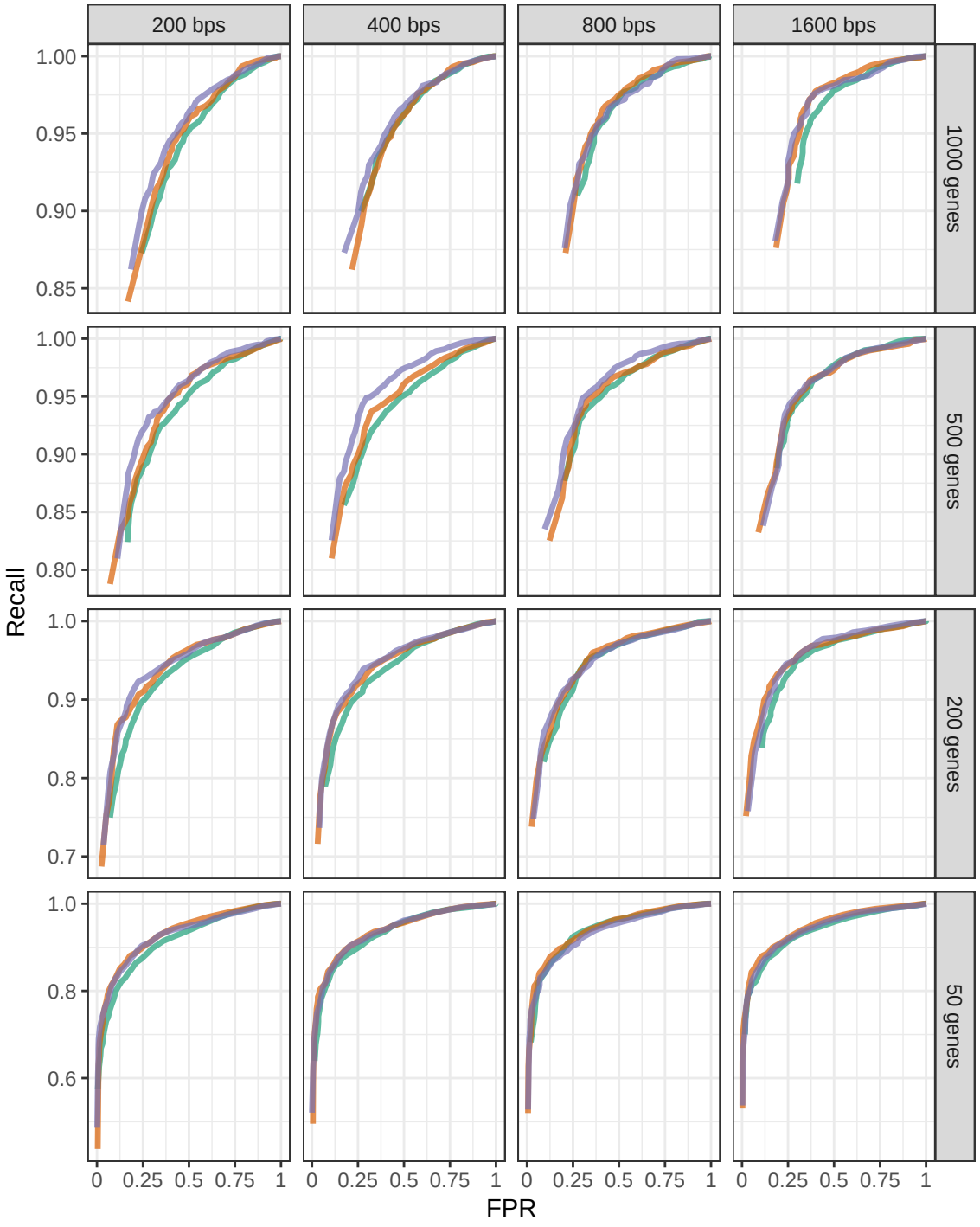

Figure S3.6. ROC of S100 dataset with $k = \{50, 200, 500, 1000\}$ and gene sequence length $\{200, 400, 800, 1600\}$ as we change the threshold of support considered. Results with aBayes supports are labelled wASTRAL-s and wASTRAL-h; results with FastTree-2 bootstrap support are labelled wASTRAL-s* and wASTRAL-h*.

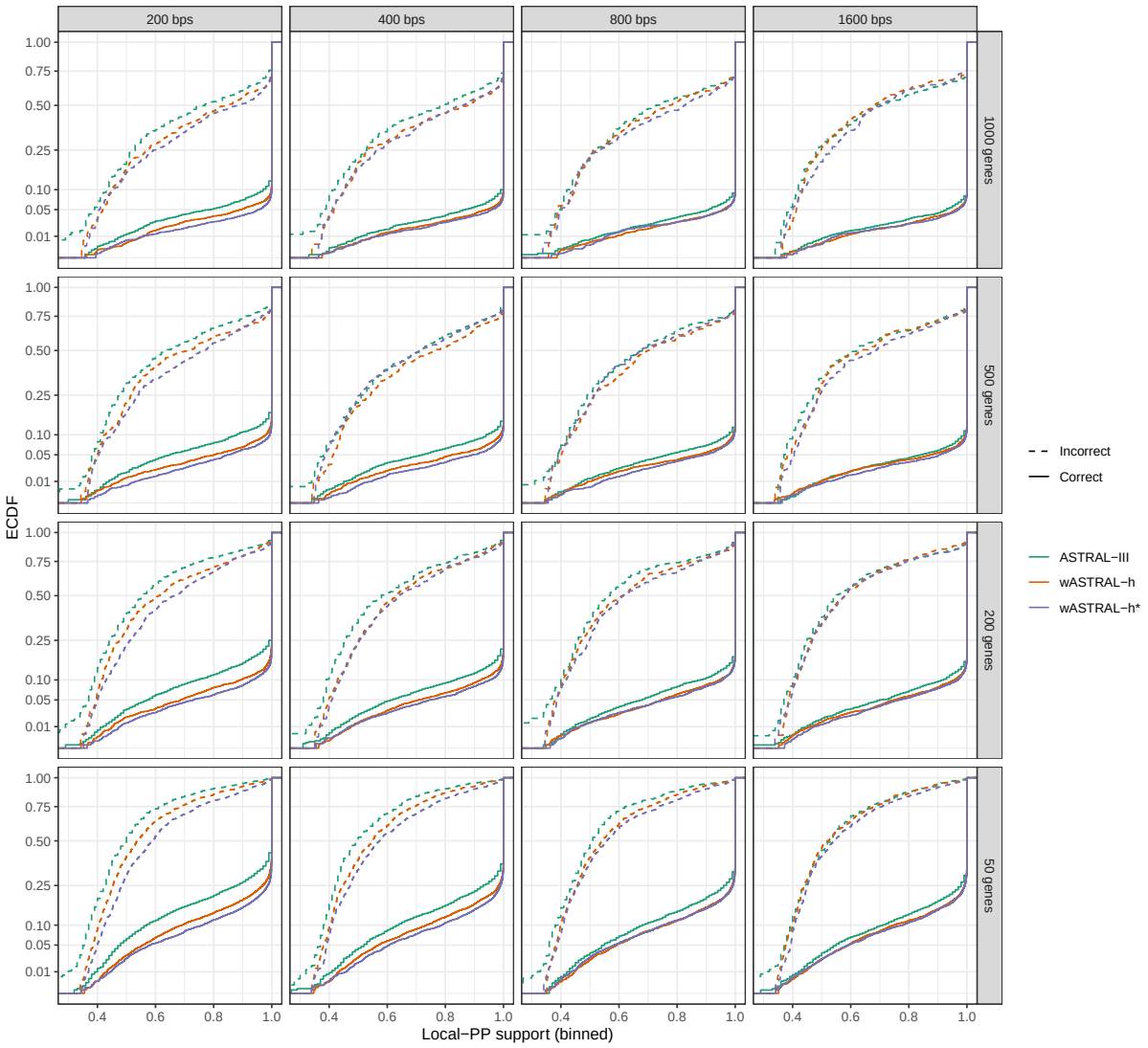

Figure S3.7. ECDF of S100 dataset with $k = \{50, 200, 500, 1000\}$ and gene sequence length $\{200, 400, 800, 1600\}$. Results with aBayes supports are labelled wASTRAL-s and wASTRAL-h; results with FastTree-2 bootstrap support are labelled wASTRAL-s* and wASTRAL-h*.

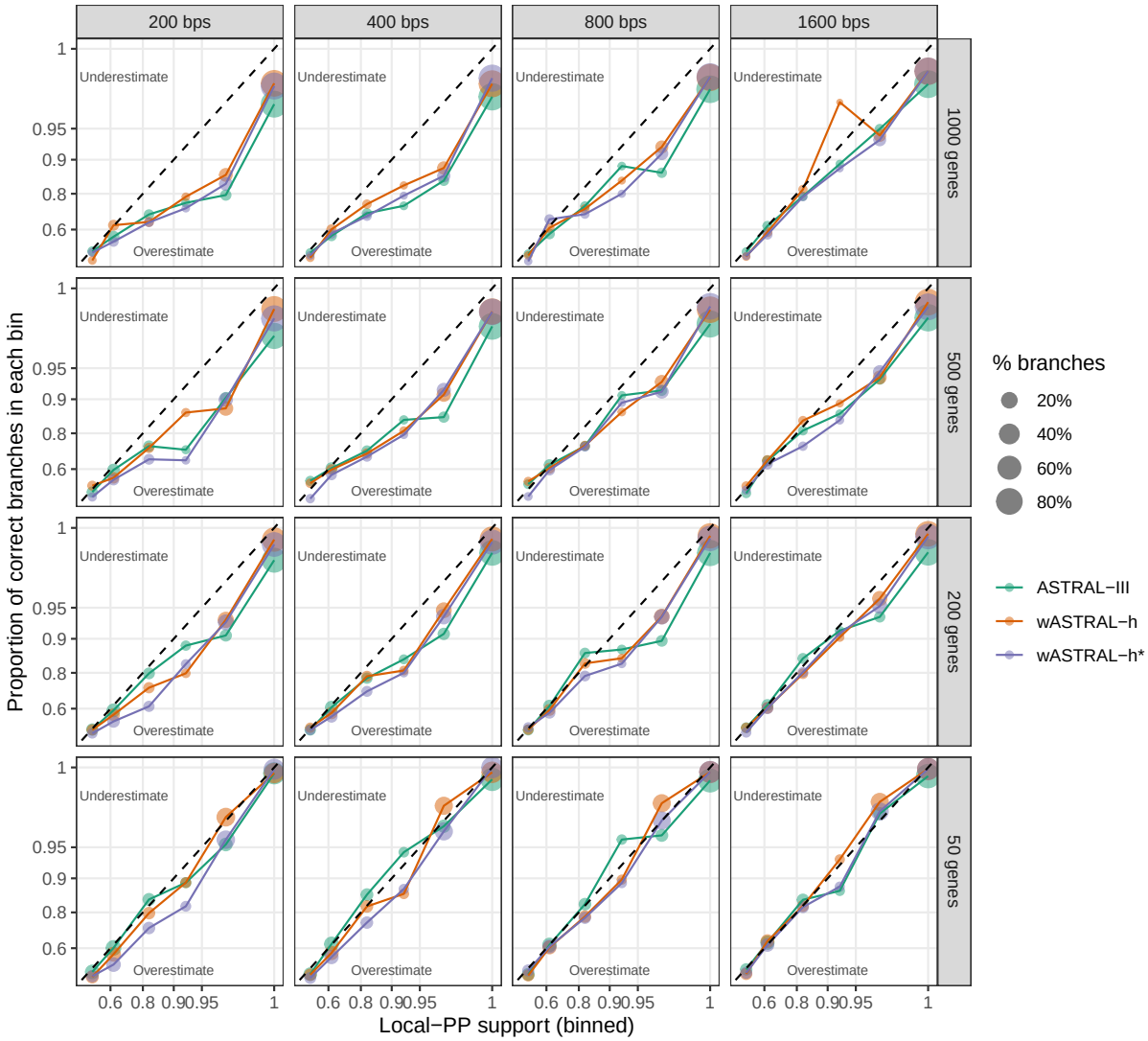

Figure S3.8. Binned accuracy-verses-support plot of S100 dataset with $k = \{50, 200, 500, 1000\}$ and gene sequence length $\{200, 400, 800, 1600\}$. Results with aBayes supports are labelled wASTRAL-s and wASTRAL-h; results with FastTree-2 bootstrap support are labelled wASTRAL-s* and wASTRAL-h*.

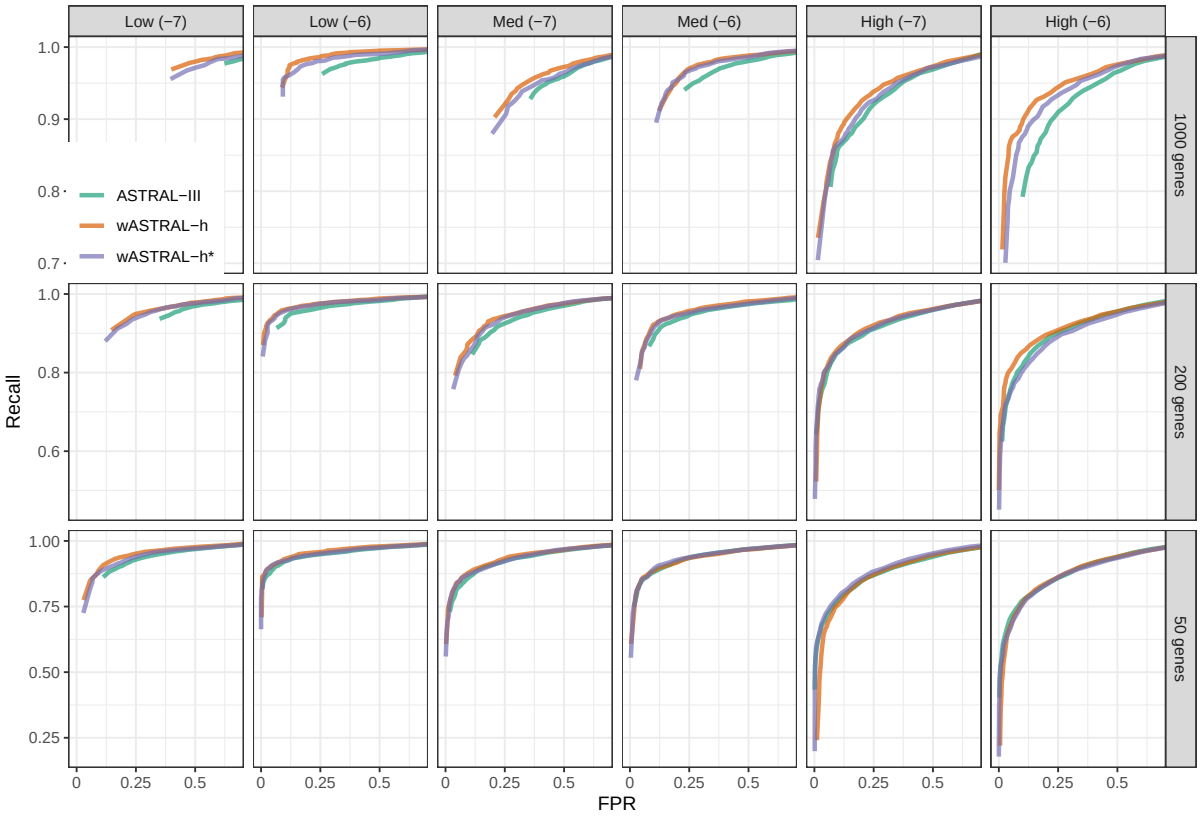

Figure S3.9. ROC of S200 dataset with $k = \{50, 200, 1000\}$ and population size (ILS levels). Species tree shape with parameter E1-6 and E1-7 are used. Results with aBayes supports are labeled wASTRAL-h; results with SH-like support are labelled wASTRAL-h*.

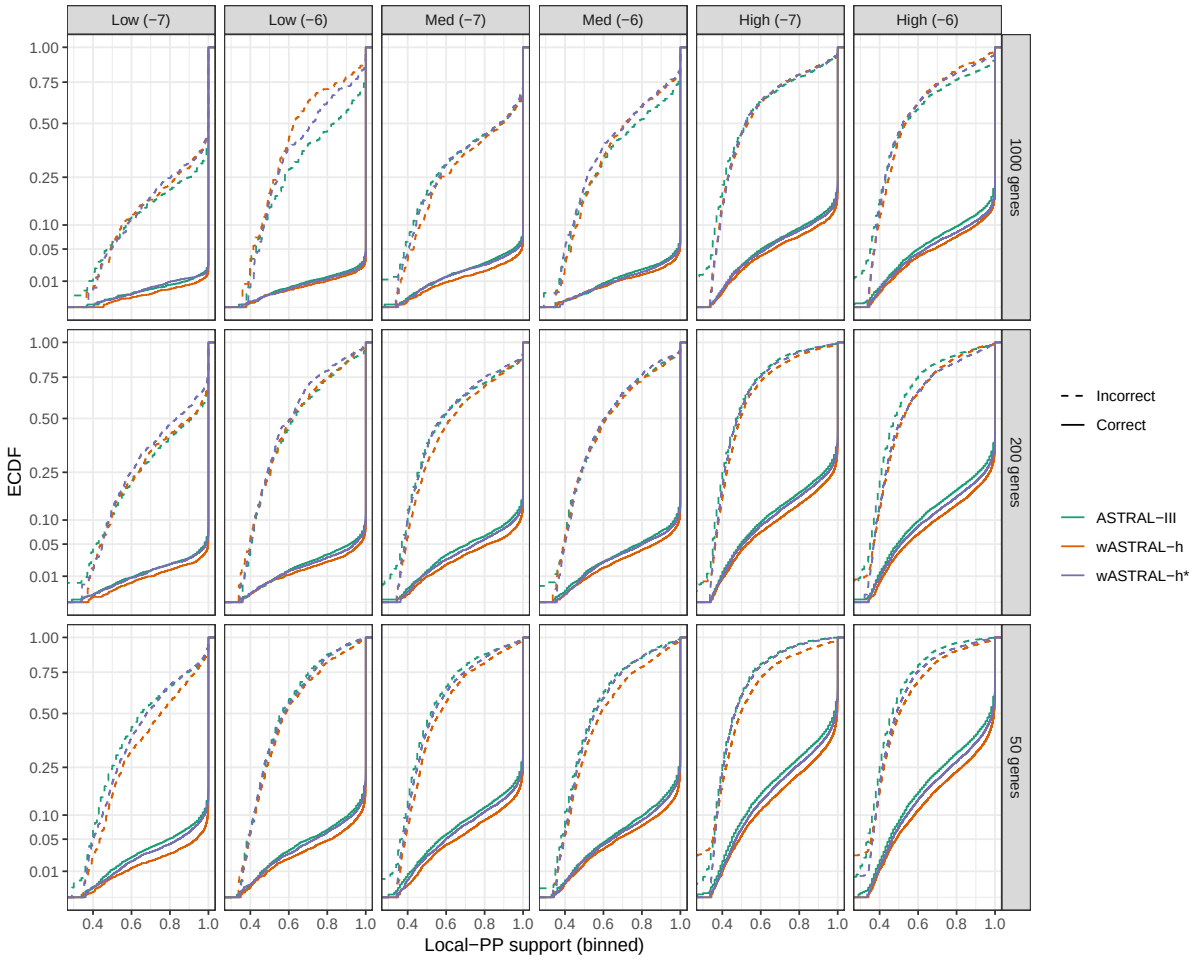

Figure S3.10. ECDF of S200 dataset with $k = \{50, 200, 1000\}$ and population size (ILS levels). Species tree shape with parameter E1-6 and E1-7 (box columns) and ILS levels (box rows) low (1e+07), medium (2e+06), and high (5e+05) are used. Results with aBayes supports are labelled wASTRAL-h; results with SH-like support are labelled wASTRAL-h*.

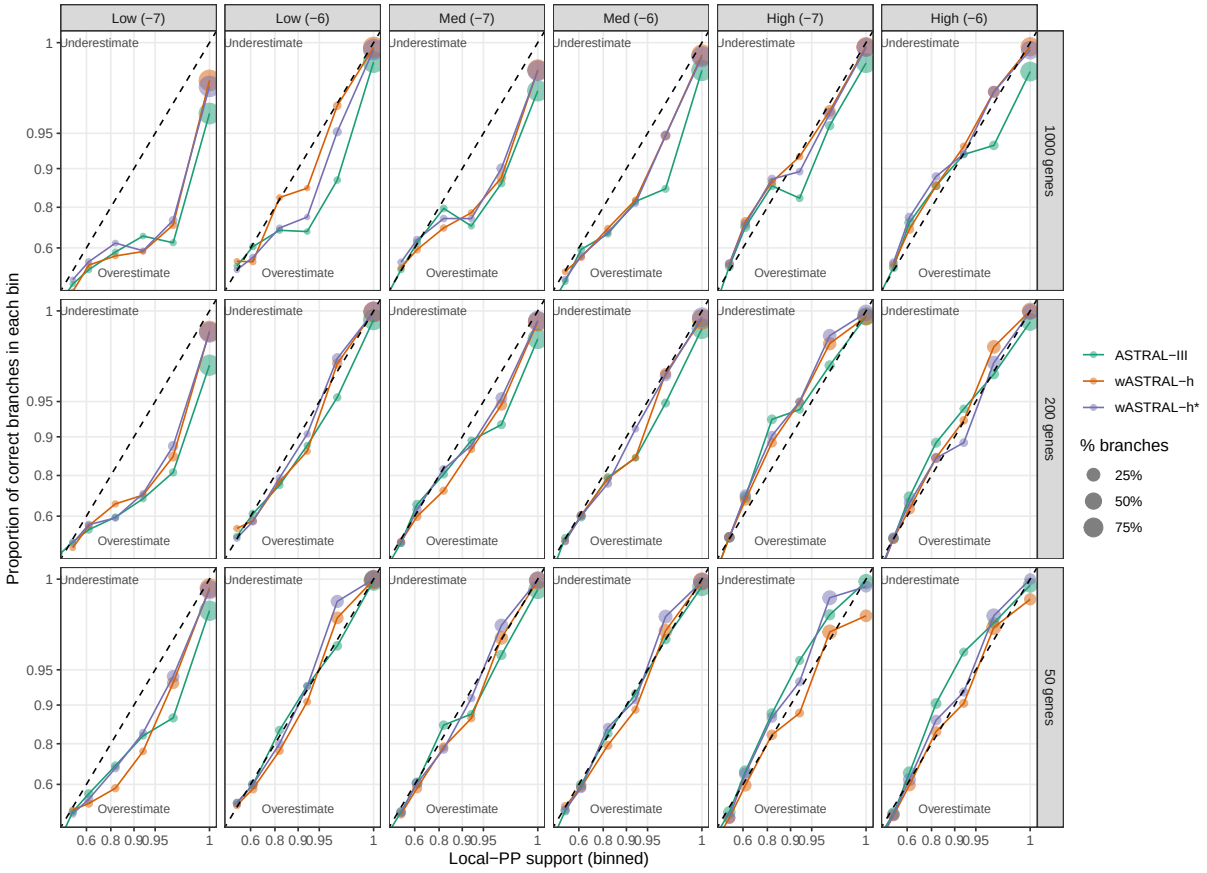

Figure S3.11. Binned accuracy-verses-support plot of S200 dataset with $k = \{50, 200, 1000\}$ and population size (ILS levels). Species tree shape with parameter E1-6 and E1-7 are used. Results with aBayes supports are labeled wASTRAL-h; results with SH-like support are labeled wASTRAL-h*.

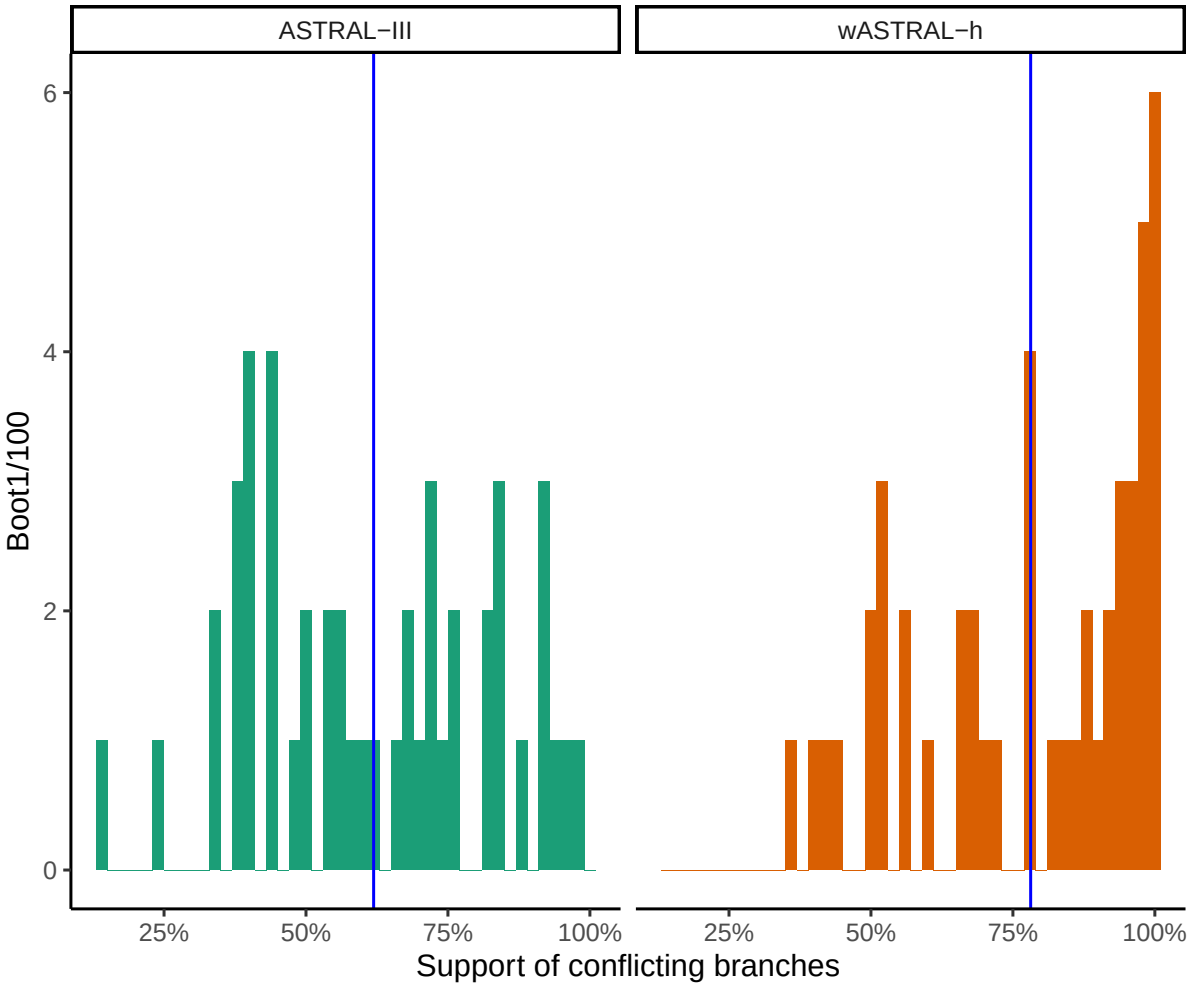

Figure S3.12. The distribution of support values of conflicting branches between wASTRAL-h and ASTRAL-III on the 1kp dataset. The ASTRAL-III conflicting branches range between 14% and 99.00% with a mean of 62%. The wASTRAL-h conflicting branches range between 37% and 99.98% with a mean of 78%.

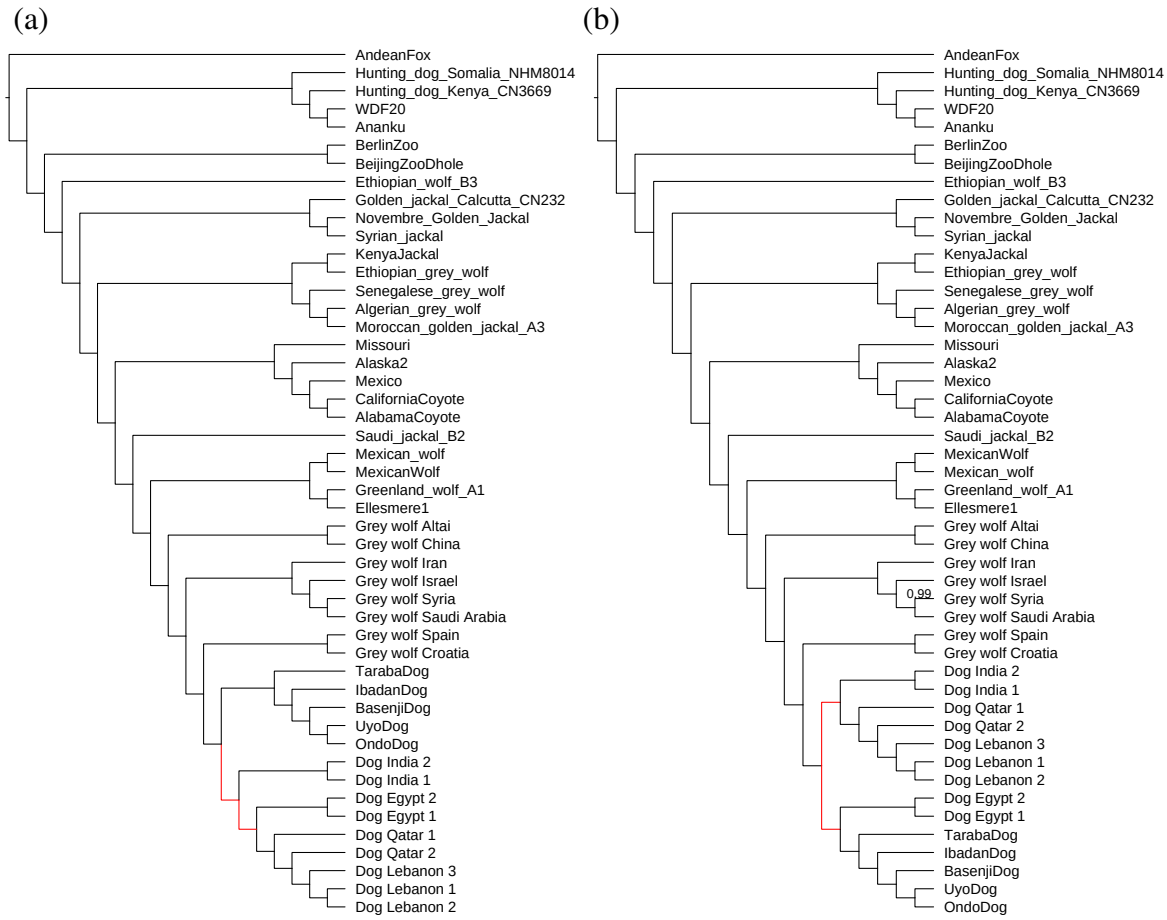

Figure S3.13. Inferred species trees (a) from wASTRAL-hybrid with FastTree-2 branch support values as weights using all 459,450 gene trees and (b) from ASTRAL-III using a subset of 100,000 gene trees on canis dataset. Branches support of 100% are omitted.

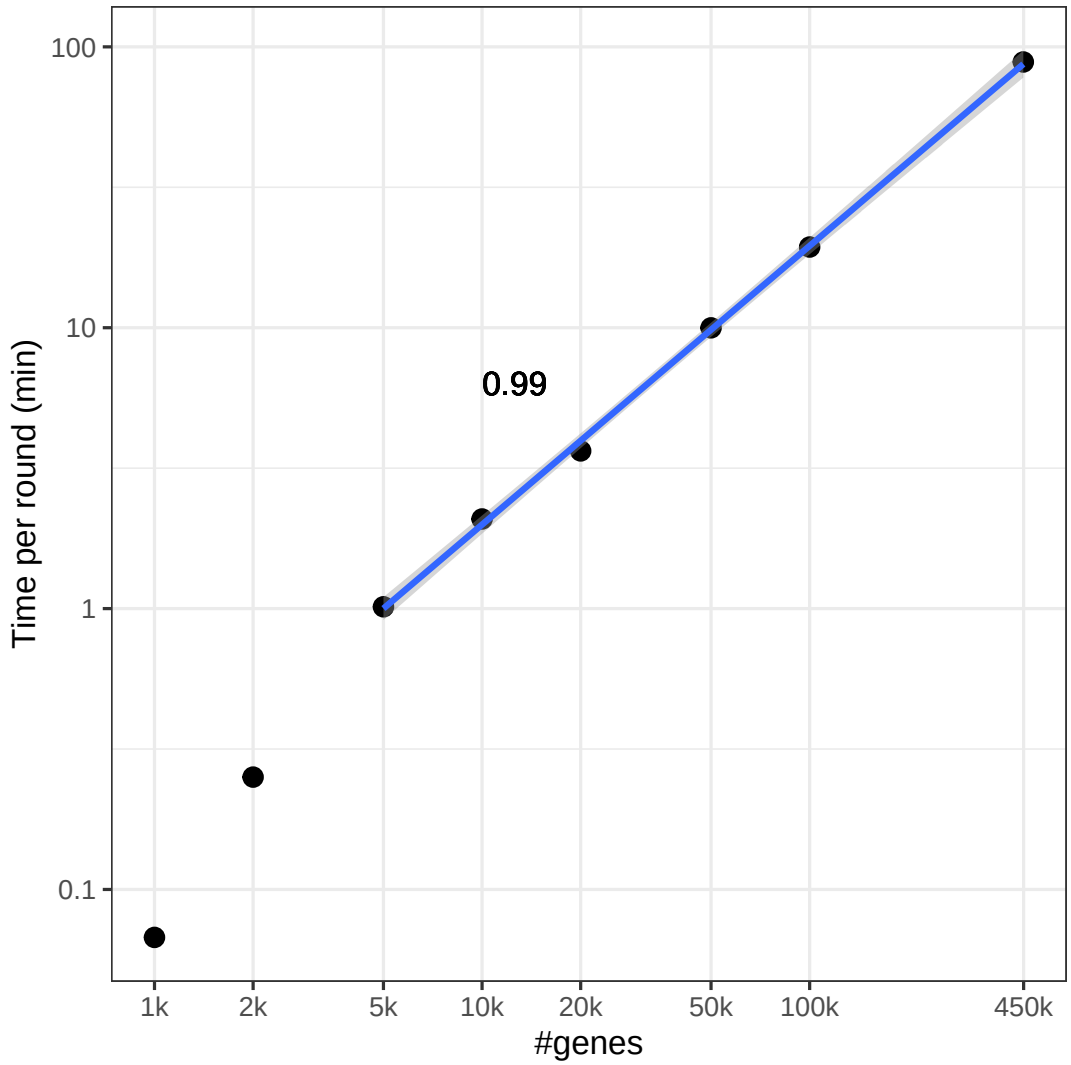

Figure S3.14. Normalized time per round of placement by dividing running time by the total number of rounds of placements for ASTRAL on the Canis dataset for various *k* using the new pipeline.

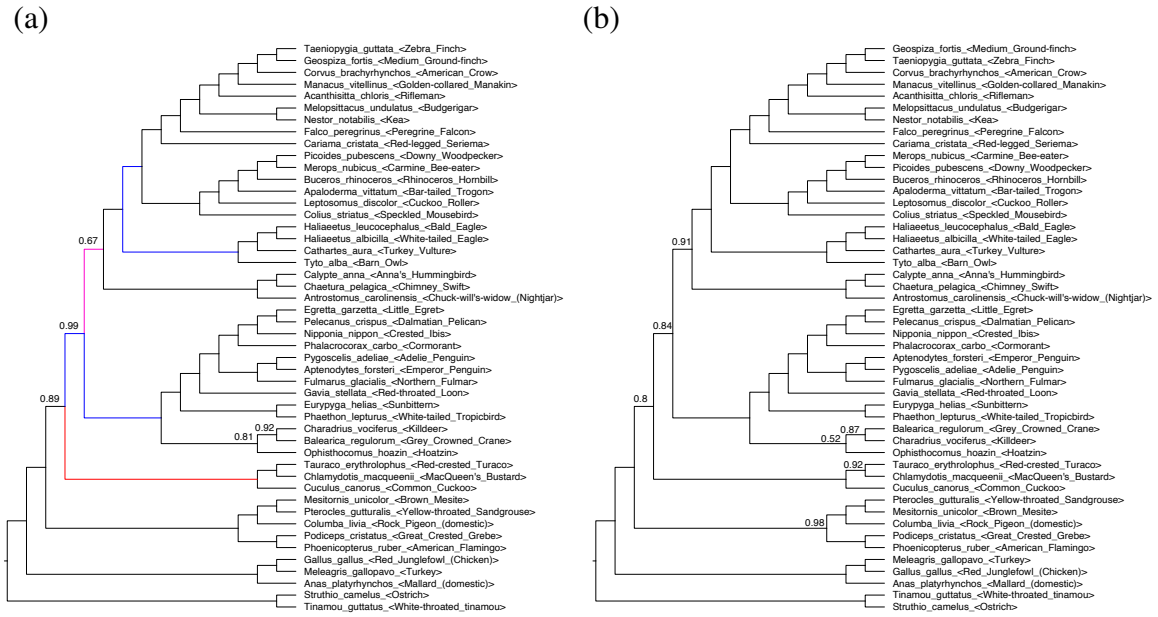

Figure S3.15. Inferred species trees from (a) wASTRAL-hybrid with normalized bootstrap support values as weights and (b) ASTRAL-III on gene trees with low ($< 3\%$ bootstrap) support branches contracted on avian dataset. Branches support of 100% are omitted. Branches that disagree with concatenation (blue), MP-EST binned (red) or both (purple) are identified on the wASTRAL-h tree.

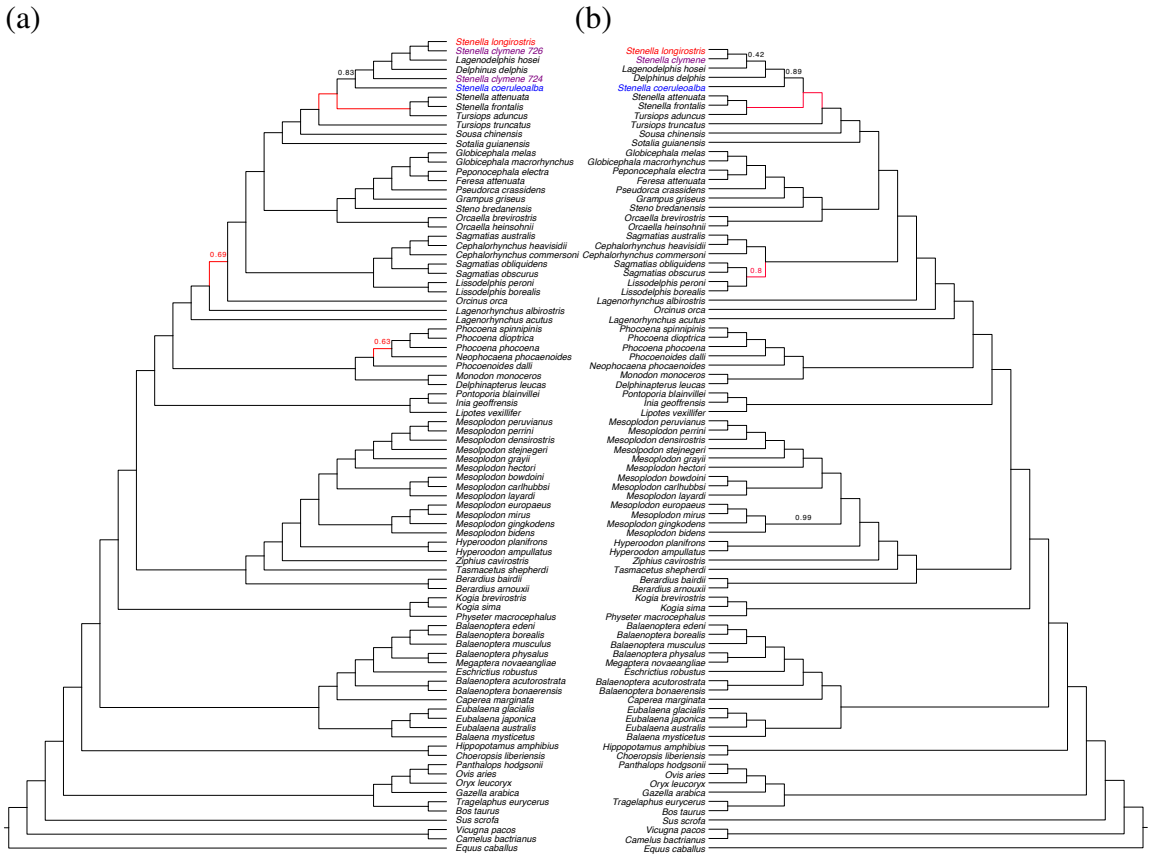

Figure S3.16. Inferred species trees from (a) wASTRAL-hybrid with normalized Bayesian support values as weights (with clades of taxa from the same species contracted) and (b) ASTRAL-multi on cetacean dataset. Branches support of 100% are omitted. Branches conflicting with RAxML concatenation are marked red.

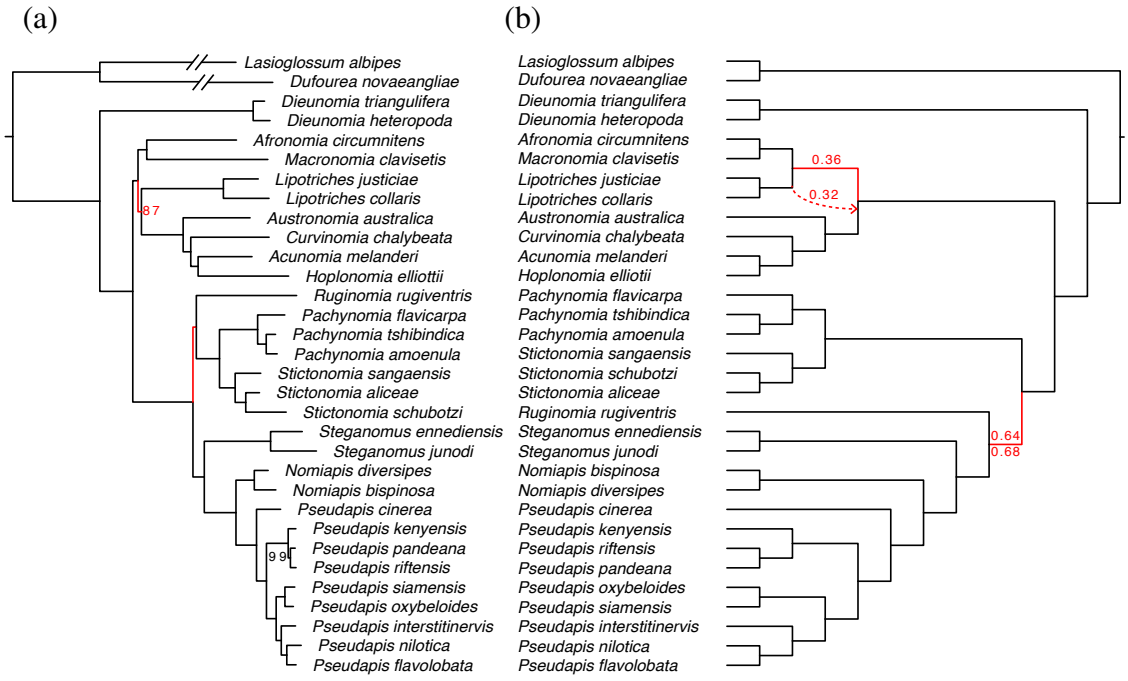

Figure S3.17. (a) RAxML on concatenated genes; (b) wASTRAL-hybrid (top and solid red line) and ASTRAL-III (bottom and dashed red line) on Nomiinae dataset.

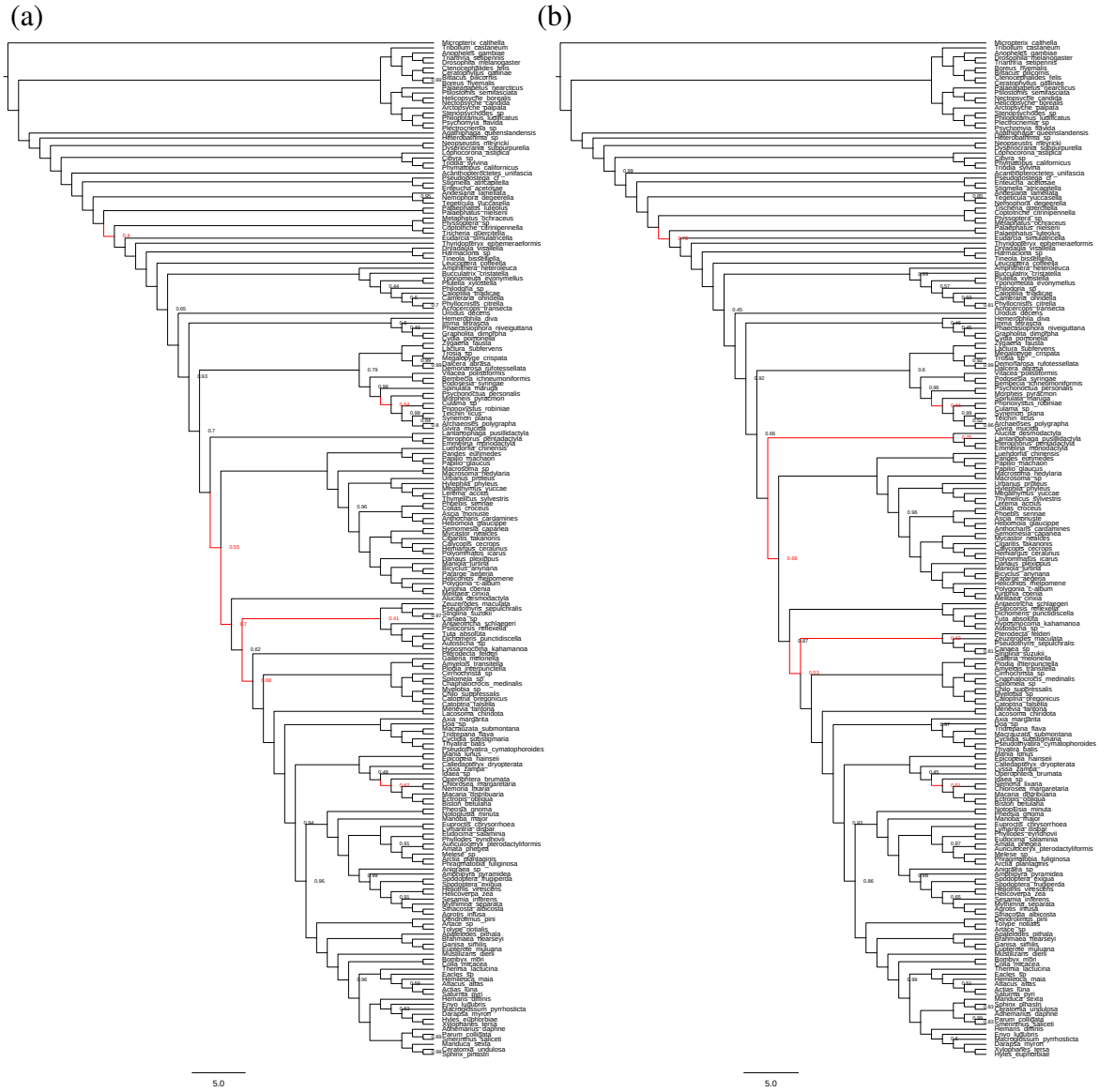

Figure S3.18. Inferred species trees from (a) wASTRAL-hybrid with normalized bootstrap support values as weights and (b) ASTRAL-III on Lepidoptera dataset.

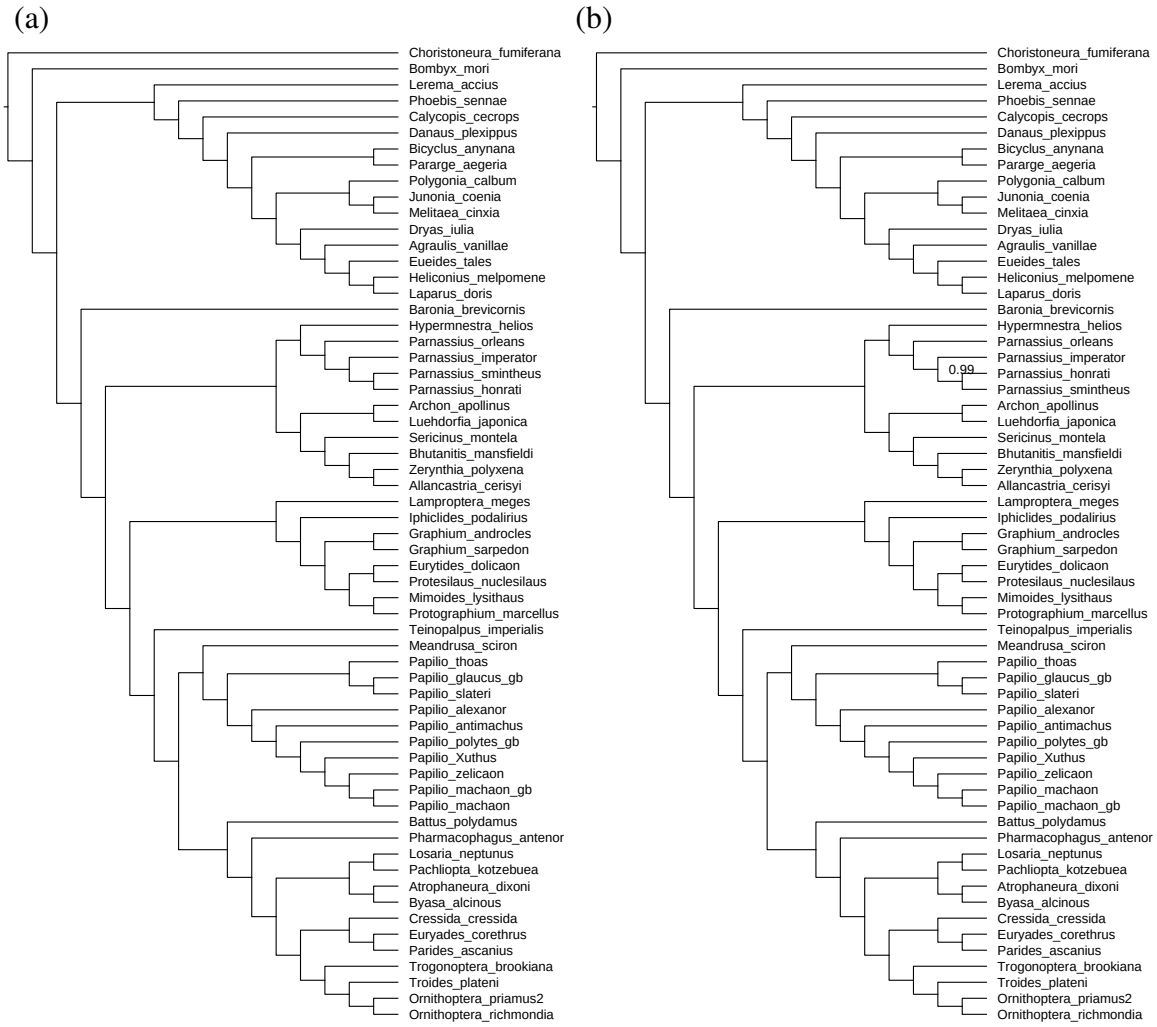

Figure S3.19. Inferred species trees from (a) wASTRAL-hybrid with normalized approximate Bayesian support values as weights and (b) ASTRAL-III on Papilionidae dataset.

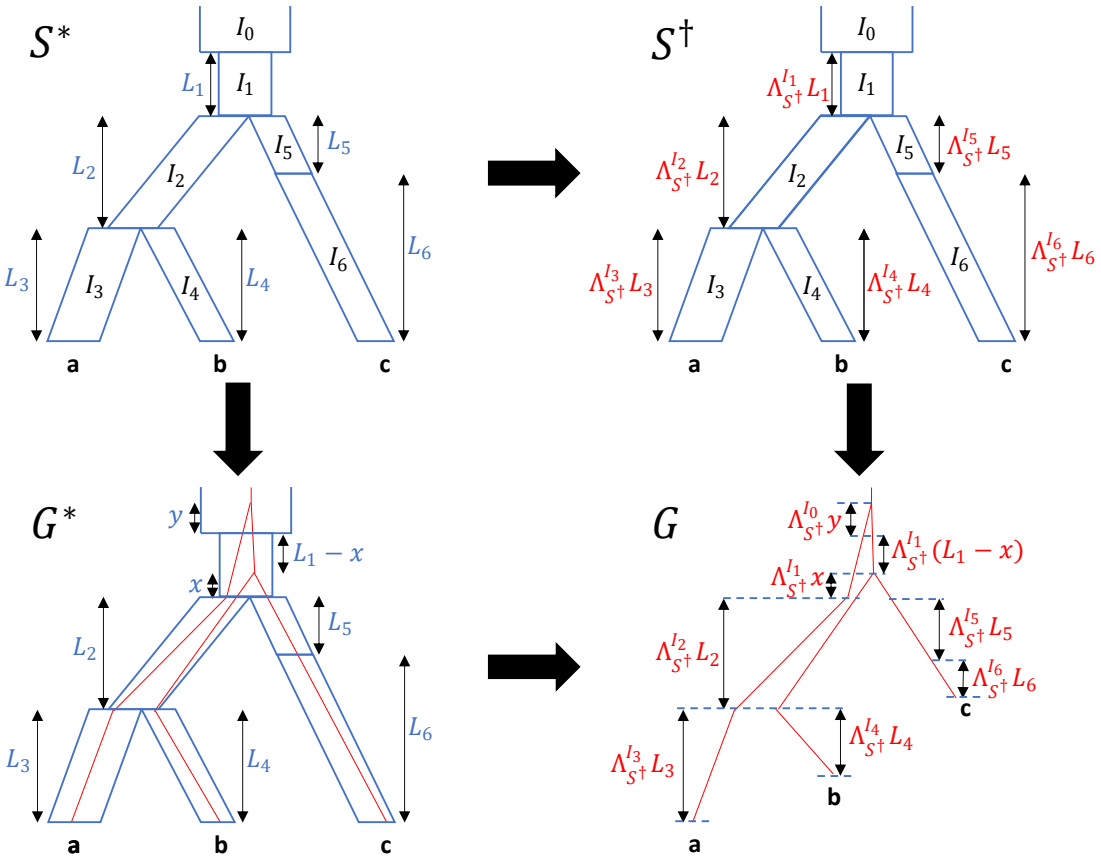

Figure S3.20. An illustration of the process of creating a random gene tree with branch lengths in SU. Branches in the true species tree S^* are broken into intervals $I_0 \ldots I_6$. The species tree with SU branch lengths $S^\dagger$ is created by multiplying each branch length in S^* with a corresponding multiplier; the multipliers are jointly drawn from some distribution and are drawn independently across gene trees. Gene tree G^* is sampled under MSC process from S^* independent of $S^\dagger$. However, it inherits the same division of its lineages into segments as S^* at the same locations. The gene tree with SU branch lengths G is created by translating branch lengths of G^* into SU by multiplying the CU length of each of segment I_i by $\Lambda_{S^\dagger}^{I_i}$, the multiplier associated with the segment I_i in $S^\dagger$ and hence G.

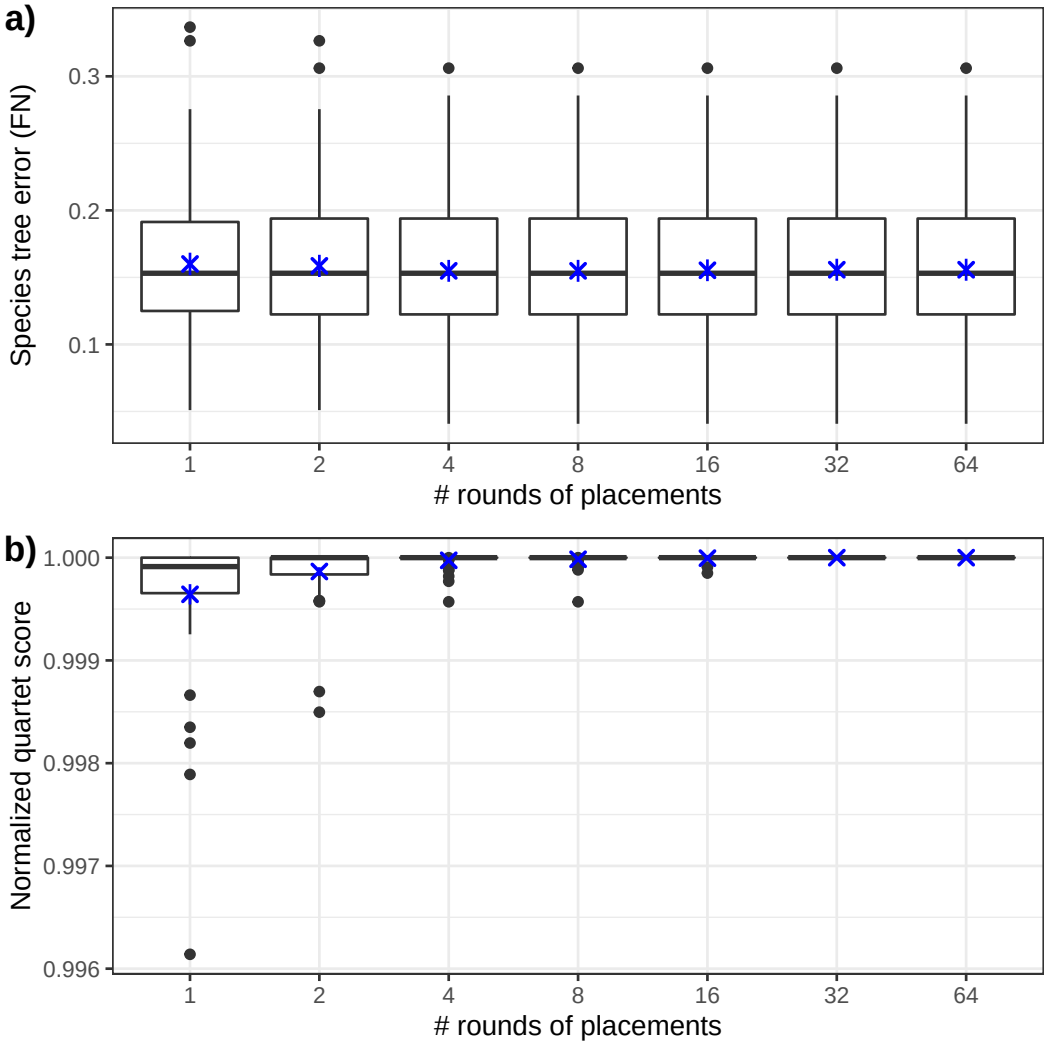

Figure S3.21. The species tree estimation error (FN) of wASTRAL-h on S100 dataset as we change the number of rounds of placements in the base algorithm (r). The most difficult case where gene length= 200 and $k = 50$ is selected. Mean and standard error (50 replicates) are shown in blue.

3.C Supplementary Algorithm

Algorithm S3.1. Recursive placement algorithm. `Place` inserts the species i into an existing species tree S and computes tripartition scores $W(A|B|C,\mathscr{G}) := \sum_{G\in\mathscr{G}} W(A\cap\mathscr{L}_G|B\cap\mathscr{L}_G|C\cap\mathscr{L}_G,G)$ for all tripartitions resulting from adding i onto each branch of S. A global counter Q and a set of per-node counters $w_A, w_B, w_C, w^+_{\cdot\cdot}, w^-_{\cdot\cdot}, w_{\cdot\cdot|\cdot}, w_{\cdot\cdot|\cdot\cdot}$ are all initialized to 0. `OptimalTreeDP` is defined in Algorithm S3.2. Each gene tree is rooted on an arbitrary branch e and the support of e is kept for the branch on one side of the root and zero support is given to the branch on the other side of root. $\mathscr{L}_v$ is the set of leaves under v.

procedure PLACE($i,S,\mathscr{G}$) ▷ Places species i on tree S according to $\mathscr{G}$
  $W \leftarrow$ empty lookup table ▷ global variables
  COLORLEAFSET($\mathscr{L}_S$, C, $\emptyset$, $\mathscr{G}$, W) ▷ Color all leaves of S as C
  COLORLEAFSET($\{i\}$, B, $\emptyset$, $\mathscr{G}$, W) ▷ Color new species i as B
  COLORNODE(the root of $S, i, S, \mathscr{G}, W$) ▷ Traverse S bottom up
  $O \leftarrow$ OPTIMALTREEDP($\mathscr{L}_S\cup\{i\}, \mathscr{L}_S\cup\{i\}, W$)
  return (W, O, edge of S onto which i is added to get O)
procedure COLORLEAFSET($\mathscr{L}^*, X, T, \mathscr{G}, W$) ▷ Condition: Coloring $\mathscr{L}^*$ as X should match T
  for $G\in\mathscr{G}$ **do**
    for $j\in\mathscr{L}^*\cap\mathscr{L}_G$ **do**
      $W[T] \leftarrow$ UPDATECOUNTERS(leaf node corresponding to j in g, X)
procedure COLORNODE($w,i,S,\mathscr{G},W$) ▷ On start: i is B, others are C; On exit: w is A, others kept
  if w is a leaf **then**
    COLORLEAFSET($\mathscr{L}_w$, A, $\mathscr{L}_w|\{i\}|\mathscr{L}_S-\mathscr{L}_w, \mathscr{G}, W$)
  else
    $(u,v) :=$ (the larger child of w, the smaller child of w)
    COLORNODE($v,i,S,\mathscr{G},W$) ▷ Recurse on v, the smaller child
    COLORLEAFSET($\mathscr{L}_v$, C, $\emptyset$, $\mathscr{G}$, W) ▷ Undo coloring of v to enable recursing on u.
    COLORNODE($u,i,S,\mathscr{G},W$) ▷ Recurse on u, the large child
    COLORLEAFSET($\mathscr{L}_v$, B, $\mathscr{L}_u|\{i\}\cup\mathscr{L}_v|\mathscr{L}_S-\mathscr{L}_w, \mathscr{G}, W$) ▷ Tripartition of w when adding i above v
    COLORLEAFSET($\{i\}$, A, $\{i\}\cup\mathscr{L}_u|\mathscr{L}_v|\mathscr{L}_S-\mathscr{L}_w, \mathscr{G}, W$) ▷ Tripartition of w when adding i above u
    COLORLEAFSET($\{i\}$, C, $\mathscr{L}_u|\mathscr{L}_v|\{i\}\cup\mathscr{L}_S-\mathscr{L}_w, \mathscr{G}, W$) ▷ Tripartition of w when adding i above w
    COLORLEAFSET($\{i\}$, B, $\emptyset$, $\mathscr{G}$, W)
    COLORLEAFSET($\mathscr{L}_v$, A, $\mathscr{L}_w|\{i\}|\mathscr{L}_S-\mathscr{L}_w, \mathscr{G}, W$) ▷ Tripartition of the new parent of i and w
procedure RECURSIVEUPDATE(w)
  $(u,v,e) :=$ (the left child of w, the right child of w, the parent branch of w)
  for $(X,Y,Z)\in\{(A,B,C),(B,C,A),(C,A,B)\}$ **do**
    $Q \leftarrow Q - w_{XX|YZ}$
    $w_{XX|YZ} \leftarrow v_X u_{YZ|X} + u_X v_{YZ|X} + u_{XX|Z} v_Y + v_{XX|Z} u_Y + u_{XX|Y} v_Z + v_{XX|Y} u_Z$
      $+(u^+_{YZ}v^+_{XX} - u^-_{YZ}v^-_{XX}) + (u^+_{XX}v^+_{YZ} - u^-_{XX}v^-_{YZ})$
    $Q \leftarrow Q + w_{XX|YZ}$
    if w is not the root **then**
      $(w_X, w_Y, w_Z) \leftarrow \Big((u_X+v_X)e^{-l(e)}, (u_Y+v_Y)e^{-l(e)}, (u_Z+v_Z)e^{-l(e)}\Big)$
      $w^+_{XX} \leftarrow u^+_{XX} + v^+_{XX} + u_X v_X$
      $w^-_{XX} \leftarrow (u^-_{XX} + v^-_{XX} + u_X v_X)\big(1-s(e)\big)$
      $w^+_{YZ} \leftarrow u^+_{YZ} + v^+_{YZ} + u_Y v_Z + u_Z v_Y$
      $w^-_{YZ} \leftarrow (u^-_{YZ} + v^-_{YZ} + u_Y v_Z + u_Z v_Y)\big(1-s(e)\big)$
      $w_{YZ|X} \leftarrow \big(u_{YZ|X} + v_{YZ|X} + (u^+_{YZ} - u^-_{YZ})v_X + u_X(v^+_{YZ} - v^-_{YZ})\big)e^{-l(e)}$
      $w_{XX|Y} \leftarrow \big(u_{XX|Y} + v_{XX|Y} + (u^+_{XX} - u^-_{XX})v_Y + u_Y(v^+_{XX} - v^-_{XX})\big)e^{-l(e)}$
      $w_{XX|Z} \leftarrow \big(u_{XX|Z} + v_{XX|Z} + (u^+_{XX} - u^-_{XX})v_Z + u_Z(v^+_{XX} - v^-_{XX})\big)e^{-l(e)}$
      RECURSIVEUPDATE(the parent of w)
procedure UPDATECOUNTERS(w,X) ▷ w is a leaf, X is a color
  $e :=$ the parent branch of w
  $(w_A, w_B, w_C) \leftarrow (0,0,0)$
  $w_X \leftarrow e^{-l(e)}$
  RECURSIVEUPDATE(the parent of w)
  return Q

Algorithm S3.2. The Algorithm S3.2 of $O(n^2kH\log n)$ running time. At start, the function is called as with $\mathscr{L}_S, \mathscr{G}, r$ as input.

```
1:  procedure NaivePlacement(T, 𝒢, r)
2:      W* ← empty lookup table from tripartitions to their weights
3:      for i ∈ {1, ..., r} do
4:          shuffle T
5:          S_i ← tree with leaves T_1, T_2, and T_3
6:          for j ∈ {4, ..., |T|} do
7:              W_i, S_i, e ← Place(T_j, S_i, 𝒢)
8:          Add all elements of W_i to W*
9:      return OptimalTreeDP(T, T, W*)
10: procedure OptimalTreeDP(P, ℒ, W)
11:     if DPTree(P) available then
12:         return DPTree(P)
13:     if |P| = 1 then
14:         DPScore(P) ← 0
15:         DPTree(P) ← Singleton rooted tree with leafset P
16:     else
17:         X ← −∞
18:         for A ∈ {A : W[A|P − A|ℒ − P] has been computed} do
19:             S_1 ← OptimalTreeDP(A, ℒ, W)
20:             S_2 ← OptimalTreeDP(P − A, ℒ, W)
21:             if DPScore(A) + DPScore(P − A) + W[A|P − A|ℒ − P] > X then
22:                 X ← DPScore(A) + DPScore(P − A) + W[A|P − A|ℒ − P]
23:                 DPTree(P) ← merge subtrees S_1 and S_2 at root
24:         DPScore(P) ← X
25:     return DPTree(P)
```

Algorithm S3.3. The DAC algorithm of $O(n^{1.5+\varepsilon}k)$ running time given some assumptions. OptimalTreeDP and NaivePlacement are defined in Algorithm S3.2, and Place is defined in Algorithm S3.1. At start, the function is called as with $\mathscr{L}_S, \mathscr{G}, r$ as input.

procedure TWOSTEPPLACEMENT($T, \mathscr{G}, r$)
    $W^* \leftarrow$ empty lookup table from tripartitions to their weights
    for $i \in \{1, ..., r\}$ **do**
        $T_i \leftarrow$ a subsample of T by removing each element independently with probability $1 - 1/\sqrt{|T|}$
        $S_i :=$ NAIVEPLACEMENT($T_i, \mathscr{G}, \sqrt{|T|}$)
        for $e \in E_{S_i}$ **do**
            $C_e \leftarrow$ empty list
        for $j \in T - T_i$ **do**
            $W, S_o, e \leftarrow$ PLACE($j, S_i, \mathscr{G}$)
            add T_j to C_e
        $C_\emptyset \leftarrow$ empty list
        $S'_i \leftarrow S_i$
        for $e \in$ branches of S_i **do**
            $S_e \leftarrow S_i$
            for $j \in C_e$ **do**
                $W, S_o, e' \leftarrow$ PLACE($j, S_e, \mathscr{G}$)
                if $e' \in S_i - \{e\}$ **then**
                    add j to $C_\emptyset$
                else
                    $S_e \leftarrow S_o$
            $S'_i \leftarrow$ The merger of compatible trees S_e and S'_i
        for $j \in C_\emptyset$ **do**
            $W_i, S'_i, e \leftarrow$ PLACE($j, S'_i, \mathscr{G}$)
        if $C_\emptyset = \emptyset$ **then**
            $W_i, S'_i, e \leftarrow$ PLACE($\emptyset, S'_i, \mathscr{G}$)
        Add all elements of W_i to W^*
    return OPTIMALTREEDP(T, T, W^*)

3.D Proofs

3.D.1 Weighting by support: Proof of Proposition 3.1 and Theorem 3.1

For ease of reference, we reproduce Table 3.2 from the main paper here:

$\mathbb{E}\big[(\cdot)(\cdot)\big\vert\alpha_{G,Q}\big]$	$\delta_G(ab\vert cd)$	$\delta_G(ac\vert bd)$
$\delta_{G^*}(ab\vert cd)$	$\geq \frac{1}{3}(1+2\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1+2\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\delta_{G^*}(ac\vert bd)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1-\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\delta_{G^*}(ad\vert bc)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)$	$\leq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)$
$\mathbb{E}\big[(\cdot)(\cdot)\big\vert\alpha_{G,Q}\big]$	$w_G(ab\vert cd)$	$w_G(ac\vert bd)$
$\delta_{G^*}(ab\vert cd)$	$\geq \frac{1}{3}(1+2\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1+2\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$
$\delta_{G^*}(ac\vert bd)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1-\theta_Q)\big(\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$
$\delta_{G^*}(ad\vert bc)$	$\geq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2$	$\leq \frac{1}{3}(1-\theta_Q)\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2$

Recall that the expected value and variance of $\alpha_{G,Q}$ across genes is denoted by $\bar{\alpha_Q}$ and σ^2_α.

Proposition 3.1. *For each estimated gene tree G,* $\mathbb{E}\big[\delta_G(ab|cd)-\delta_G(ac|bd)\big] \geq \theta_Q\bar{\alpha_Q} - \frac{2}{3}(1-\bar{\alpha_Q})\beta_Q$ *and* $\mathbb{E}\big[w_G(ab|cd)-w_G(ac|bd)\big] \geq \frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha_Q}^2+\sigma^2_\alpha)+\frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q$.

Proof. To prove the Proposition, we start with the following lemma.

Lemma 3.1. *For each estimated gene tree G with a given* $\alpha_{G,Q}$*,*

$$\mathbb{E}\big[\delta_G(ab|cd)-\delta_G(ac|bd)\big|\alpha_{G,Q}\big] \geq \theta_Q\alpha_{G,Q} - \frac{2}{3}(1-\alpha_{G,Q})\beta_Q$$

and

$$\mathbb{E}\big[w_G(ab|cd)-w_G(ac|bd)\big|\alpha_{G,Q}\big] \geq \frac{1}{9}(3\alpha_{G,Q}-2\beta_Q+2\alpha_{G,Q}\beta_Q+6)\theta_Q\alpha_{G,Q} - \frac{4}{9}(1-\alpha_{G,Q})\beta_Q\,.$$

Proof. From Table 3.2, we can compute

$$\begin{aligned}
&\mathbb{E}\big[\delta_G(ab|cd)-\delta_G(ac|bd)\big|\alpha_{G,Q}\big]\\
=&\mathbb{E}\Big[\big(\delta_G(ab|cd)-\delta_G(ac|bd)\big)\big(\delta_{G^*}(ab|cd)+\delta_{G^*}(ac|bd)+\delta_{G^*}(ad|bc)\big)\Big|\alpha_{G,Q}\Big]\\
\geq&\frac{1}{3}\big((1+2\theta_Q)\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)-\frac{1}{3}\big((1-\theta_Q)\alpha_{G,Q}+\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)\\
=&\theta_Q\alpha_{G,Q}-\frac{2}{3}(1-\alpha_{G,Q})\beta_Q\,;
\end{aligned}$$

similarly,

$$\begin{aligned}
&\mathbb{E}\big[w_G(ab|cd)-w_G(ac|bd)\big|\alpha_{G,Q}\big]\\
=&\mathbb{E}\Big[\big(w_G(ab|cd)-w_G(ac|bd)\big)\big(\delta_{G^*}(ab|cd)+\delta_{G^*}(ac|bd)+\delta_{G^*}(ad|bc)\big)\Big|\alpha_{G,Q}\Big]\\
\geq&\frac{1}{3}(1+2\theta_Q)\alpha_{G,Q}\big(\alpha_{G,Q}+\frac{2}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)+\big(\frac{1}{3}(1-\alpha_{G,Q})(1-\beta_Q)\big)^2\\
&-\frac{1}{3}(1-\theta_Q)\alpha_{G,Q}\big(\alpha_{G,Q}+\frac{2}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)-\big(\frac{1}{3}(1-\alpha_{G,Q})(1+\beta_Q)\big)^2\\
\geq&\frac{1}{9}(3\alpha_{G,Q}-2\beta_Q+2\alpha_{G,Q}\beta_Q+6)\theta_Q\alpha_{G,Q}-\frac{4}{9}(1-\alpha_{G,Q})\beta_Q\,.
\end{aligned}$$

□

From this lemma, we can prove the proposition. First, assume $\alpha_{G,Q}$ is drawn from a discrete distribution. Then,

$$\begin{aligned}
\mathbb{E}\big[\delta_G(ab|cd)-\delta_G(ac|bd)\big]&=\sum_{\alpha_{G,Q}}\mathbb{E}\big[\delta_G(ab|cd)-\delta_G(ac|bd)\big|\alpha_{G,Q}\big]\mathrm{P}(\alpha_{G,Q})\\
&\geq\sum_{\alpha_{G,Q}}\big(\theta_Q\alpha_{G,Q}-\frac{2}{3}(1-\alpha_{G,Q})\beta_Q\big)\mathrm{P}(\alpha_{G,Q})=\theta_Q\bar{\alpha_Q}-\frac{2}{3}(1-\bar{\alpha_Q})\beta_Q
\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}\big[w_G(ab|cd) - w_G(ac|bd)\big] = \sum_{\alpha_{G,Q}} \mathbb{E}\big[w_G(ab|cd) - w_G(ac|bd)\big|\alpha_{G,Q}\big]\mathrm{P}(\alpha_{G,Q}) \\
&\geq \sum_{\alpha_{G,Q}} \big(\frac{1}{9}\big(3\alpha_{G,Q} - 2\beta_Q + 2\alpha_{G,Q}\beta_Q + 6\big)\theta_Q\alpha_{G,Q} - \frac{4}{9}(1-\alpha_{G,Q})\beta_Q\big)\mathrm{P}(\alpha_{G,Q}) \\
&= \frac{1}{9}\theta_Q(3+2\beta_Q)\mathbb{E}[\alpha_{G,Q}^2] + \frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q \\
&= \frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha_Q}^2 + \sigma_\alpha^2) + \frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q\,.
\end{aligned}$$

It is straightforward to change these calculations to use integral instead of sum and $\mathrm{P}(\alpha_{G,Q})$ to the PDF in the case that the distribution of $\alpha_{G,Q}$ is continuous. □

Theorem 3.1. *Given estimated gene trees furnished with support generated under MSC+Error+Support model, there exist conditions where* (3.3) *guarantee a statistically consistent estimator of* S^* *but* (3.2) *does not, and the reverse is not true.*

Proof. Recall that (3.1) states

$$W(S,\mathscr{G}) := \sum_{G\in\mathscr{G}}\sum_{Q\in\mathscr{Q}(S)} w_G(S \restriction Q)\,.$$

It means that in order to produce a statistically consistent estimator using (3.1), the following equation must be satisfied for the true species tree topology S^* and any species tree topology S:

$$\mathbb{E}\big[W(S^*,\mathscr{G}) - W(S,\mathscr{G})\big] = |\mathscr{G}|\sum_{Q\in\mathscr{Q}(S)} \mathbb{E}\big[w_G(S^* \restriction Q) - w_G(S \restriction Q)\big] \geq 0 \tag{3.9}$$

Notice that proving for any quartet $Q = \{a,b,c,d\}$ we have $\mathbb{E}\big[w_G(ab|cd) - w_G(ac|bd)\big] \geq 0$ and $\mathbb{E}\big[w_G(ab|cd) - w_G(ad|bc)\big] \geq 0$ where $S^* \restriction Q = ab|cd$ is sufficient to prove (3.9); on the other hand, proving for any quartet $Q = \{a,b,c,d\}$ where the internal branch

of $S^* \restriction Q$ corresponds to only one branch in S^*, we have $\mathbb{E}\big[w_G(ab|cd) - w_G(ac|bd)\big] \geq 0$ and $\mathbb{E}\big[w_G(ab|cd) - w_G(ad|bc)\big] \geq 0$ where $S^* \restriction Q = ab|cd$ is necessary to prove (3.9).

Thus, from Proposition 3.1, we have guaranteed statistical consistency for weighted ASTRAL for support under

$$D = \bigcap_{Q \in \mathcal{Q}(S)} \{(\theta_Q, \bar{\alpha_Q}, \sigma_\alpha, \beta_Q) \in [0,1]^4 : \frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha_Q}^2 + \sigma_\alpha^2) + \frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q \geq 0\} .$$

Similarly, we have guaranteed statistical consistency for unweighted ASTRAL under

$$D' = \bigcap_{Q \in \mathcal{Q}(S)} \big\{(\theta_Q, \bar{\alpha_Q}, \sigma_\alpha, \beta_Q) \in [0,1]^4 : \theta_Q\bar{\alpha_Q} - \frac{2}{3}(1-\bar{\alpha_Q})\beta_Q \geq 0\big\} .$$

To prove Theorem 3.1, we only need to prove that D' is a proper subset of D.

We can prove $D' \subseteq D$, as for any Q, if $(\theta_Q, \bar{\alpha_Q}, \sigma_\alpha, \beta_Q) \in [0,1]^4$ and $\theta_Q\bar{\alpha_Q} - \frac{2}{3}(1-\bar{\alpha_Q})\beta_Q \geq 0$, then

$$\begin{aligned}
&\frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha_Q}^2 + \sigma_\alpha^2) + \frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q \\
&= \frac{1}{9}\theta_Q(3+2\beta_Q)\sigma_\alpha^2 + \frac{1}{3}\theta_Q(1-\theta_Q)\bar{\alpha_Q}^2 + (\frac{1}{3}\theta_Q\bar{\alpha_Q} + \frac{2}{3})\big(\theta_Q\bar{\alpha_Q} - \frac{2}{3}(1-\bar{\alpha_Q})\beta_Q\big) \geq 0 .
\end{aligned}$$

We can also prove $D' \neq D$, as if for some Q, $\theta_Q = 0.25, \bar{\alpha_Q} = 0.5, \beta_Q = 0.4$,

$$\theta_Q\bar{\alpha_Q} - \frac{2}{3}(1-\bar{\alpha_Q})\beta_Q = -\frac{1}{120} < 0$$

and

$$\frac{1}{9}\theta_Q(3+2\beta_Q)(\bar{\alpha_Q}^2 + \sigma_\alpha^2) + \frac{2}{9}(3-\beta_Q)\theta_Q\bar{\alpha_Q} - \frac{4}{9}(1-\bar{\alpha_Q})\beta_Q = \frac{7}{720} + \frac{19}{180}\sigma_\alpha^2 > 0 .$$

Thus D' is a proper subset of D and we conclude the proof.

□

3.D.2 Weighting by length: Proof of Propositions 3.2 and 3.3 and Theorem 3.2

Before providing the proofs, we remind the reader of one property of the coalescent model. According to the coalescent model, at any point along a branch of the species tree with i gene tree lineages, the time (i.e., distance) x to the next coalescent event, reducing the number of lineages to $i-1$, is exponentially distributed with the rate $\binom{i}{2}$, resulting in probability density function (PDF):

$$\frac{i(i-1)}{2} e^{-\frac{i(i-1)}{2}x}, \tag{3.10}$$

and the two lineages that coalesce are independent of x.

Proposition 3.2. *For a true quartet species tree S^* with topology $ab|cd$ and input gene trees $\mathscr{G}$ generated under the naive model with any multiplier λ, let f be the distance between anchors of S^*. As $f \to 0$, given $k = \Theta(f^{-2})$ gene trees, we have $Var[X_G] = \Theta_f(1)$ and*

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

Proof. We analyze balanced and unbalanced trees separately.

Case 1: Unbalanced trees (i.e., the root of S^* has a terminal branch as a child). W.o.l.g., we assume the root branch is located on branch leading to d.

Let p, q, and r be the MRCA nodes of (a,b), (a,c), and (a,d) on rooted species tree S^*, respectively. Let p' and r' be the points of coalescence of leaves a,b and leaves c,d on the rooted gene tree G, respectively. Let x, y_0, and z be the CU difference in heights of points (p,p'), (q,r), and (r,r'), respectively. Note that f is the length of (p,q). Let $L := l_{S^*}(a,p) + l_{S^*}(b,p) + l_{S^*}(c,r) + l_{S^*}(d,r)$. Notice that $l_G(a,p) + l_G(b,p) + l_G(c,r) + l_G(d,r) = \lambda L$ and $l_G(a,b) + l_G(c,d) = \lambda(2x + 2z + L)$.

Let $f_X(x)$ be the probability density that x is the CU difference in heights of (p,p') and

p' is the lowest point of coalescence. Notice that by (3.10):

$$f_X(x) = \begin{cases} e^{-x} & 0 \le x \le f \\ \frac{1}{\binom{2}{3}}\left(e^{-f}\binom{2}{3}e^{-\binom{2}{3}(x-f)}\right) = e^{-3x+2f} & f \le x \le f+y_0 \\ \frac{1}{\binom{2}{4}}\left(e^{-f}e^{-\binom{2}{3}y_0}\binom{2}{4}e^{-\binom{2}{4}(x-f-y_0)}\right) = e^{-6x+5f+3y_0} & f+y_0 \le x \end{cases}$$

Let $f_{Z|X}(z;x)$ be the probability density that z is the CU difference in heights of (r,r'), conditioned on that x is the CU difference in heights of (p,p') and p' is the lowest point of coalescence. Notice that:

$$f_{Z|X}(z;x) = \begin{cases} e^{-z} & 0 \le x \le f+y_0 \text{ and } 0 \le z \\ e^{-\left(z-(x-f-y_0)\right)} = e^{-z+x-f-y_0} & 0 \le x-f-y_0 \le z \end{cases}$$

We specify three coalescence scenarios by indicator functions $\delta_1, \delta_2, \delta_3$: *i)* δ_1 indicates $0 \le x < f$; *ii)* δ_2 indicates $f \le x < f+y_0$; *iii)* δ_3 indicates $f+y_0 \le x$.

Note that

$$\begin{aligned} \mathbb{E}\left[w_G(ab|cd)\right] &= \mathbb{E}\left[(\delta_1+\delta_2+\delta_3)w_G(ab|cd)\right] \\ \mathbb{E}\left[w_G^2(ab|cd)\right] &= \mathbb{E}\left[(\delta_1+\delta_2+\delta_3)w_G^2(ab|cd)\right]. \end{aligned}$$

Similarly, since only scenarios 2 and 3 have deep coalescence events that may lead to gene tree disagreement with the species tree, and by the symmetry of all three topologies under scenarios 2 and 3,

$$\begin{aligned} \mathbb{E}\left[w_G(ac|bd)\right] &= \mathbb{E}\left[(\delta_2+\delta_3)w_G(ab|cd)\right] \\ \mathbb{E}\left[w_G^2(ac|bd)\right] &= \mathbb{E}\left[(\delta_2+\delta_3)w_G^2(ab|cd)\right]. \end{aligned}$$

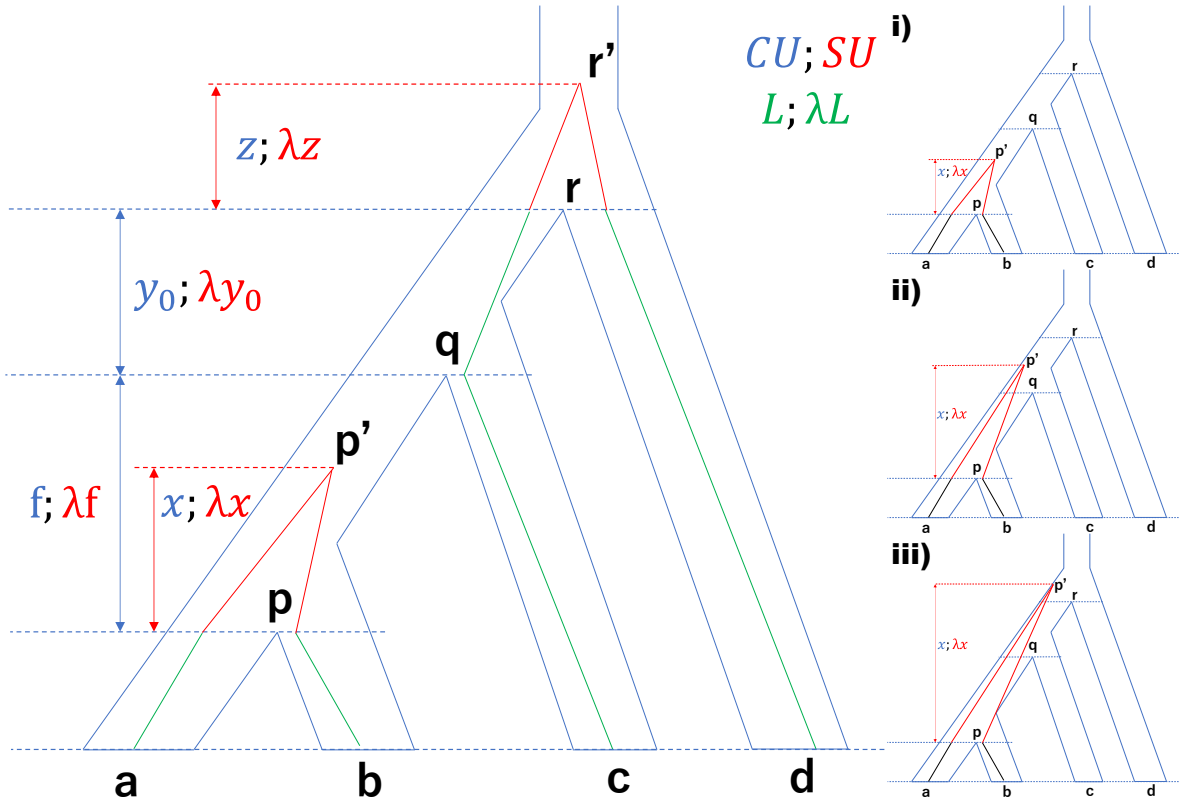

Figure S3.22. Illustration of the unbalanced case. Lengths in CU/SU units are denoted in blue/red. Branches in green have a total length $L/\lambda L$ in CU/SU units. The right-hand side shows the position of p' in relation to q and r in various cases.

Thus,

$$\mathbb{E}[X_G] = \mathbb{E}\big[w_G(ab|cd)\big] - \mathbb{E}\big[w_G(ac|bd)\big] = \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] \,, \tag{3.11}$$

and since $w_G(ab|cd)w_G(ac|bd) = 0$,

$$\begin{aligned} \mathrm{Var}[X_G] =& \mathbb{E}[X_G^2] - \mathbb{E}^2[X_G] = \mathbb{E}\big[w_G^2(ab|cd) + w_G^2(ac|bd)\big] - \mathbb{E}^2[X_G] \\ =& \mathbb{E}\big[(\delta_1 + 2\delta_2 + 2\delta_3) w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G] \,. \end{aligned} \tag{3.12}$$

We next compute both elements of (3.11) as well as some elements of (3.12) (others will not be necessary).

- δ_1: When G has topology $ab|cd$, p' must be the lowest point of coalescence. Thus,

$$\begin{aligned} &\mathbb{E}\big[\delta_1 w_G(ab|cd)\big] \\ &= \int_0^f \int_0^{+\infty} e^{-\lambda(2x+2z+L)} f_X(x) f_{Z|X}(z;x) dz dx \\ &= \int_0^f \int_0^{+\infty} e^{-\lambda(2x+2z+L)} e^{-x} e^{-z} dz dx \\ &= \frac{e^{-\lambda L}(1 - e^{-(1+2\lambda)f})}{(1+2\lambda)^2} \,; \end{aligned}$$

$$\mathbb{E}\big[\delta_1 w_G^2(ab|cd)\big] \leq \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = O(f) \,.$$

- δ_2: When G has topology $ab|cd$, p' must be the lowest point of coalescence. Thus,

$$\begin{aligned} &\mathbb{E}\big[\delta_2 w_G^2(ab|cd)\big] \\ &= \int_f^{f+y_0} \int_0^{+\infty} e^{-\lambda(4x+4z+2L)} f_X(x) f_{Z|Y}(z;y) dz dx \\ &= \int_f^{f+y_0} \int_0^{+\infty} e^{-\lambda(4x+4z+2L)} e^{-3x+2f} e^{-z} dz dx \\ &= \frac{1 - e^{-(3+4\lambda)y_0}}{(1+4\lambda)(3+4\lambda)} e^{-(1+4\lambda)f - 2\lambda L} \,. \end{aligned}$$

- δ_3: When G has the topology $ab|cd$, either p' or q' must be the lowest point of coalescence, and by symmetry, the two cases must have the same PDFs. Thus,

$$\begin{aligned}
&\mathbb{E}\big[\delta_3 w_G^2(ab|cd)\big] \\
&= \int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} e^{-\lambda(4x+4z+2L)} 2 f_X(x) f_{Z|X}(z;x) dz dx \\
&= \int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} e^{-\lambda(4x+4z+2L)} 2 e^{-6x+5f+3y_0} e^{-z+x-f-y_0} dz dx \\
&= \int_{f+y_0}^{+\infty} e^{-4\lambda(x+x-f-y_0)-2\lambda L} 2 e^{-6x+5f+3y_0} \frac{1}{1+4\lambda} dx \\
&= \frac{1}{(3+4\lambda)(1+4\lambda)} e^{-(1+4\lambda)f-(3+4\lambda)y_0-2\lambda L} .
\end{aligned}$$

Replacing in (3.11), we get

$$\mathbb{E}[X_G] = \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = \frac{e^{-\lambda L}(1-e^{-(1+2\lambda)f})}{(1+2\lambda)^2} = \frac{e^{-\lambda L}}{1+2\lambda} f + O(f^2) ;$$

and replacing in (3.12), we get

$$\begin{aligned}
\mathrm{Var}[X_G] =& \mathbb{E}\big[(\delta_1+2\delta_2+2\delta_3) w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G] = \mathbb{E}\big[2(\delta_2+\delta_3) w_G^2(ab|cd)\big] + O(f) \\
=& \frac{2e^{-(1+4\lambda)f-2\lambda L}}{(3+4\lambda)(1+4\lambda)} + O(f) = \frac{2e^{-2\lambda L}}{(3+4\lambda)(1+4\lambda)} + O(f) ,
\end{aligned}$$

from which our assumption of $\mathrm{Var}[X_G] = \Omega(1)$ follows.

Case 2: Balanced tree.

Let p, q, and r be the MRCA nodes of (a,b), (c,d), and (a,d) on rooted species tree S^*, respectively. Let p' and q' be the points of coalescence of leaves a, b and leaves c, d on the rooted gene tree G, respectively. Let x, x_0, y, and y_0 be the CU difference in heights of points (p,p'), (p,r), (q,q'), and (q,r), respectively. Note that $f = x+y$ is CU length of path (p,q). Let $L := l_{S^*}(a,p) + l_{S^*}(b,p) + l_{S^*}(c,q) + l_{S^*}(d,q)$. Notice that $l_G(a,p) + l_G(b,p) + l_G(c,q) + l_G(d,q) = \lambda L$ and $l_G(a,b) + l_G(c,d) = \lambda(2x+2y+L)$.

We specify three coalescence scenarios by indicator functions $\delta_1, \delta_2, \delta_3$: *i)* δ_1 indicates

$0 \leq x < x_0$; *ii)* δ_2 indicates $x_0 \leq x, 0 \leq y < y_0$; *iii)* δ_3 indicates $x_0 \leq x, y_0 \leq y$.

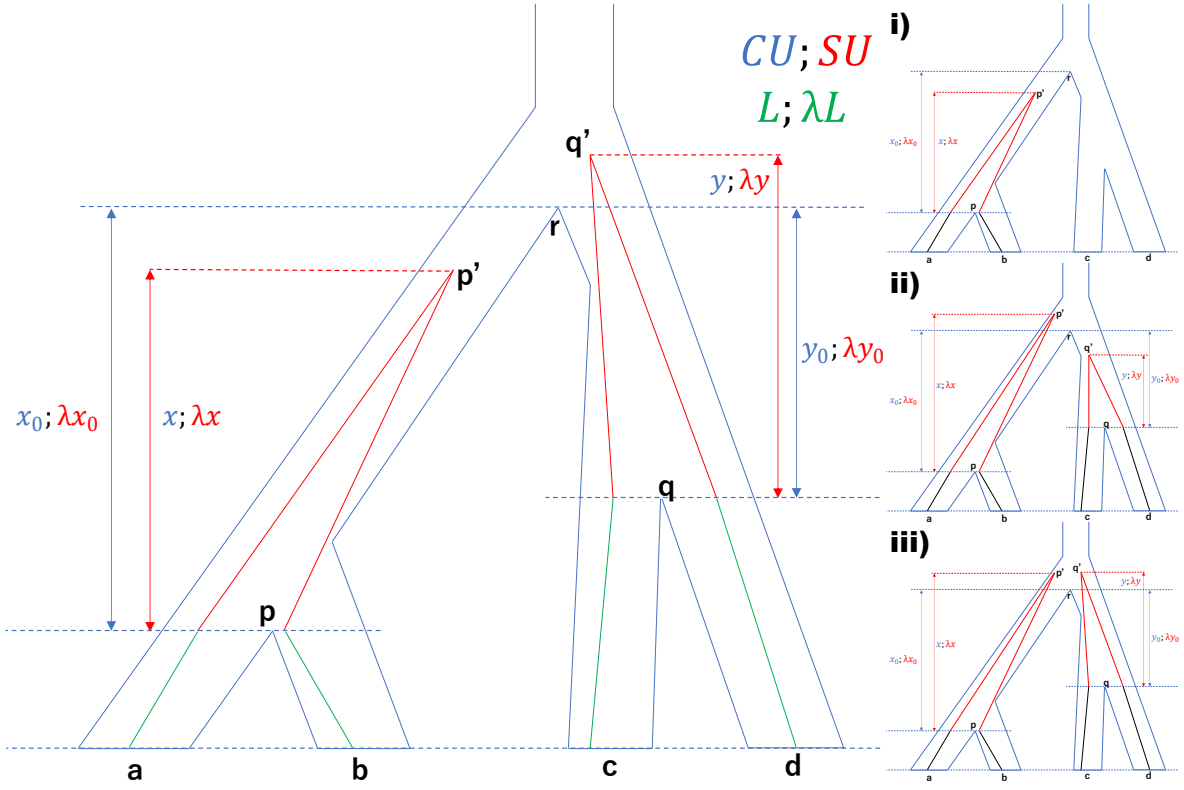

Figure S3.23. Illustration of the balanced case. Lengths in CU/SU units are denoted in blue/red. Branches in green have a total length $L/\lambda L$ in CU/SU units. The right-hand side shows the position of p' and q' in relation to r in various cases.

Note that

$$\mathbb{E}\left[w_G(ab|cd)\right] = \mathbb{E}\left[(\delta_1+\delta_2+\delta_3)w_G(ab|cd)\right]$$
$$\mathbb{E}\left[w_G^2(ab|cd)\right] = \mathbb{E}\left[(\delta_1+\delta_2+\delta_3)w_G^2(ab|cd)\right] .$$

Similarly, since only scenarios 3 have deep coalescence events that may lead to gene tree disagreement with the species tree, and by the symmetry of all three topologies under scenarios 3,

$$\mathbb{E}\left[w_G(ac|bd)\right] = \mathbb{E}\left[\delta_3 w_G(ab|cd)\right]$$
$$\mathbb{E}\left[w_G^2(ac|bd)\right] = \mathbb{E}\left[\delta_3 w_G^2(ab|cd)\right] .$$

Thus,

$$\mathbb{E}[X_G] = \mathbb{E}\big[w_G(ab|cd)\big] - \mathbb{E}\big[w_G(ac|bd)\big] = \mathbb{E}\big[(\delta_1+\delta_2)w_G(ab|cd)\big]\ ; \tag{3.13}$$

and since $w_G(ab|cd)w_G(ac|bd) = 0$,

$$\begin{aligned}\mathrm{Var}[X_G] =& \mathbb{E}[X_G^2] - \mathbb{E}^2[X_G] = \mathbb{E}\big[w_G^2(ab|cd) + w_G^2(ac|bd)\big] - \mathbb{E}^2[X_G] \\ =& \mathbb{E}\big[(\delta_1+\delta_2+2\delta_3)w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G]\ .\end{aligned} \tag{3.14}$$

- δ_1: Here,

$$\begin{aligned}\mathbb{E}\big[\delta_1 w_G(ab|cd)\big] =& \int_0^{x_0}\int_0^{+\infty} e^{-\lambda(2x+2y+L)}e^{-x}e^{-y}dydx \\ =& \frac{e^{-\lambda L}(1-e^{-(1+2\lambda)x_0})}{(1+2\lambda)^2} = \frac{e^{-\lambda L}x_0}{1+2\lambda} + O(x_0^2) = \frac{e^{-\lambda L}x_0}{1+2\lambda} + O(f^2)\ ;\end{aligned}$$

  and

$$\mathbb{E}\big[\delta_1 w_G^2(ab|cd)\big] \le \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = O(f)\ .$$

- δ_2: Here,

$$\begin{aligned}\mathbb{E}\big[\delta_2 w_G(ab|cd)\big] =& \int_{x_0}^{+\infty}\int_0^{y_0} e^{-\lambda(2x+2y+L)}e^{-x}e^{-y}dydx \\ =& \frac{e^{-\lambda L}(1-e^{-(1+2\lambda)y_0})e^{-(1+2\lambda)x_0}}{(1+2\lambda)^2} = \frac{e^{-\lambda L}y_0}{1+2\lambda} + O(f^2)\ ;\end{aligned}$$

  and

$$\mathbb{E}\big[\delta_2 w_G^2(ab|cd)\big] \le \mathbb{E}\big[\delta_2 w_G(ab|cd)\big] = O(f)\ .$$

- δ_3: Similar to the unbalanced case, when G has the topology $ab|cd$, either p' or q' must be the lowest point of coalescence, and by symmetry, the two cases must have the same PDFs.

Thus,

$$\begin{aligned}\mathbb{E}\big[\delta_3 w_G^2(ab|cd)\big] &= \int_{x_0}^{+\infty}\int_{x-x_0+y_0}^{+\infty} e^{-\lambda(4x+4y+2L)} 2e^{-x_0}e^{-y_0}e^{-6x+6x_0}e^{-y+x-x_0+y_0}\,dy\,dx \\ &= \int_{x_0}^{+\infty} e^{-4\lambda(x+x-x_0+y_0)-2\lambda L} 2e^{-x_0}e^{-y_0}e^{-6x+6x_0}\frac{1}{1+4\lambda}\,dx \\ &= \frac{1}{(3+4\lambda)(1+4\lambda)}e^{-(1+4\lambda)(x_0+y_0)-2\lambda L} = \frac{1}{(3+4\lambda)(1+4\lambda)}e^{-(1+4\lambda)f-2\lambda L}\,.\end{aligned}$$

Replacing in (3.13), we get

$$\mathbb{E}[X_G] = \mathbb{E}\big[(\delta_1+\delta_2)w_G(ab|cd)\big] = \frac{e^{-\lambda L}(x_0+y_0)}{1+2\lambda} + O(f^2) = \frac{e^{-\lambda L}f}{1+2\lambda} + O(f^2)\,;$$

and replacing in (3.14), we get

$$\begin{aligned}\mathrm{Var}[X_G] =& \mathbb{E}\big[(\delta_1+\delta_2+2\delta_3)w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G] \\ =& \mathbb{E}\big[2\delta_3 w_G^2(ab|cd)\big] + O(f) \\ =& \frac{2e^{-(1+4\lambda)f-2\lambda L}}{(3+4\lambda)(1+4\lambda)} + O(f) = \frac{2e^{-2\lambda L}}{(3+4\lambda)(1+4\lambda)} + O(f)\,,\end{aligned}$$

from which our assumption of $\mathrm{Var}[X_G] = \Theta_f(1)$ follows.

Thus, in both balanced and unbalanced cases,

$$\frac{\mathbb{E}[X_G]}{\sqrt{\mathrm{Var}[X_G]}} = \frac{\frac{e^{-\lambda L}}{1+2\lambda}f + O(f^2)}{\sqrt{\frac{2e^{-2\lambda L}}{(1+4\lambda)(3+4\lambda)} + O(f)}} = \sqrt{1+\frac{4\lambda+4\lambda^2}{3(1+2\lambda)^2}}\sqrt{\frac{3}{2}}f + O(f^2)$$

□

Proposition 3.3. *For a true quartet species tree S^* with topology $ab|cd$ and input gene trees $\mathscr{G}$ 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 that for every branch segment I, the variance of*

its multiplier is bounded above: $Var(\Lambda^I_{S^\dagger}) \leq \varepsilon^2$ *where* $\varepsilon^2 = \frac{e^{-\lambda L}}{(16+32\lambda)+(6+32\lambda+32\lambda^2)L}\left(\frac{20(\lambda+\lambda^2)}{9(1+2\lambda)^2}\right)^3$. *As* $f \to 0$, *given* $k = \Theta(f^{-2})$ *gene trees, we have* $Var[X_G] = \Theta_f(1)$ *and*

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

Proof. We follow the same logic in proof of Proposition 3.2.

Case 1: Unbalanced trees. Let $P(x)$ be functions to random variables denoting SU difference in heights of points (p,p') where p' is x CU distance above p; let $R(z)$ be functions to random variables denoting SU difference in heights of points (r,r') where r' is z CU distance above r. Note that $P(f+y_0)+R(z) = P(f+y_0+z)$ where $P(f+y_0)$ denote the SU length of (p,r). Let random variable $\Lambda := \big(l_{S^\dagger}(a,p) + l_{S^\dagger}(b,p) + l_{S^\dagger}(c,r) + l_{S^\dagger}(d,r)\big)$ be the total SU terminal branch lengths and the constant value L be the CU distance corresponding to Λ.

- δ_1: When G has topology $ab|cd$, p' must be the lowest point of coalescence. Thus,

$$\begin{aligned}
&\mathbb{E}\big[\delta_1 w_G(ab|cd)\big] \\
=&\mathbb{E}\Big[\int_0^f \int_0^{+\infty} e^{-2P(x)-2R(z)-\Lambda} f_X(x) f_{Z|X}(z;x)\,dz\,dx\Big] \\
=&\mathbb{E}\Big[\int_0^f \int_0^{+\infty} e^{-2P(x)-2R(z)-\Lambda} e^{-x} e^{-z}\,dz\,dx\Big] \\
=&\mathbb{E}\Big[\int_0^f \int_0^{+\infty} e^{-2P(x)-2R(z)-\Lambda-x-z}\,dz\,dx\Big]\,;
\end{aligned}$$

and

$$\mathbb{E}\big[\delta_1 w_G^2(ab|cd)\big] \leq \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = O(f)\,.$$

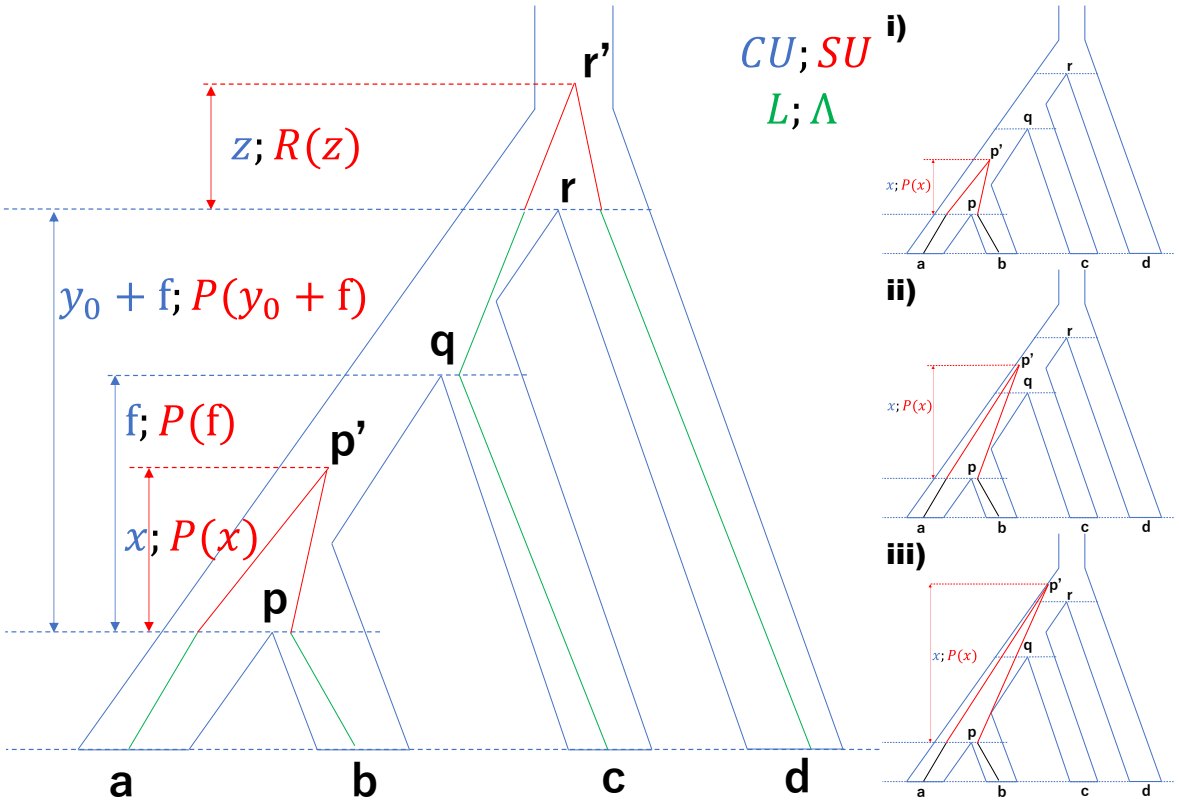

Figure S3.24. Illustration of the unbalanced case. Lengths in CU/SU units are denoted in blue/red. Branches in green have a total length L/Λ in CU/SU units. The right-hand side shows the position of p' in relation to q and r in various cases.

- δ_2: When G has topology $ab|cd$, p' must be the lowest point of coalescence. Thus,

$$\begin{aligned}
&\mathbb{E}\big[\delta_2 w_G^2(ab|cd)\big] \\
=&\mathbb{E}\Big[\int_f^{f+y_0}\int_0^{+\infty} e^{-4P(x)-4R(z)-2\Lambda} f_X(x) f_{Z|Y}(z;y)\,dz\,dx\Big] \\
=&\mathbb{E}\Big[\int_f^{f+y_0}\int_0^{+\infty} e^{-4P(x)-4R(z)-2\Lambda} e^{-3x+2f} e^{-z}\,dz\,dx\Big] \\
=&\int_f^{f+y_0}\int_0^{+\infty} \mathbb{E}\big[e^{-4P(x)-4R(z)-2\Lambda}\big] e^{-3x-z+2f}\,dz\,dx\,.
\end{aligned}$$

- δ_3: When G has the topology $ab|cd$, either p' or q' must be the lowest point of coalescence, and by symmetry, the two cases must have the same PDFs. Thus,

$$\begin{aligned}
&\mathbb{E}\big[\delta_3 w_G^2(ab|cd)\big] \\
=&\mathbb{E}\Big[\int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} e^{-4P(x)-4R(z)-2\Lambda} 2 f_X(x) f_{Z|X}(z;x)\,dz\,dx\Big] \\
=&\mathbb{E}\Big[\int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} e^{-4P(x)-4R(z)-2\Lambda} 2 e^{-6x+5f+3y_0} e^{-z+x-f-y_0}\,dz\,dx\Big] \\
=&\int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} \mathbb{E}\big[e^{-4P(x)-4R(z)-2\Lambda}\big] 2 e^{-5x-z+4f+2y_0}\,dz\,dx\,.
\end{aligned}$$

Replacing in (3.11), by Jensen's inequality, we get

$$\begin{aligned}
\mathbb{E}[X_G] =&\mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = \mathbb{E}\Big[\int_0^f\int_0^{+\infty} e^{-2P(x)-2R(z)-\Lambda-x-z}\,dz\,dx\Big] \\
\geq&\int_0^f\int_0^{+\infty} e^{\mathbb{E}\left[-2P(x)-2R(z)-\Lambda-x-z\right]}\,dz\,dx \\
=&\int_0^f\int_0^{+\infty} e^{-2\lambda x-2\lambda z-\lambda L-x-z}\,dz\,dx \\
=&\frac{e^{-\lambda L}(1-e^{-(1+2\lambda)f})}{(1+2\lambda)^2} = \frac{e^{-\lambda L}}{1+2\lambda} f + O(f^2)\,.
\end{aligned}$$

And replacing in (3.12), we get

$$\begin{aligned}
\mathrm{Var}[X_G] =&\mathbb{E}\big[(\delta_1+2\delta_2+2\delta_3) w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G] = \mathbb{E}\big[2(\delta_2+\delta_3) w_G^2(ab|cd)\big] + O(f) \\
=&\int_f^{f+y_0}\int_0^{+\infty} \mathbb{E}\big[e^{-4P(x)-4R(z)-2\Lambda}\big] 2 e^{-3x-z+2f}\,dz\,dx \\
&+\int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty} \mathbb{E}\big[e^{-4P(x)-4R(z)-2\Lambda}\big] 4 e^{-5x-z+2f+2y_0}\,dz\,dx + O(f)\,,
\end{aligned}$$

from which our assumption of $\mathrm{Var}[X_{G^*}] = \Theta_f(1)$ follows.

Let $F_P(u;x)$, $F_R(v;z)$, and $F_\Lambda(w)$ be the CDF of $P(x)$, $R(z)$, and Λ respectively; let $F_{PR\Lambda}(u,v,w;x,z)$ and $F_{PR\Lambda}(u,v,w;x,z)$ be the joint CDF and the joint PDF. Let $F_P^{-1}(t;x)$, $F_R^{-1}(t;z)$, and $F_\Lambda^{-1}(t)$ be the inverse function of CDF of $P(x)$, $R(z)$, and Λ respectively.

Then,

$$
\begin{aligned}
&\mathbb{E}\Big[e^{-2\left(2P(x)+2R(z)+\Lambda\right)}\Big] \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} e^{-2(2u+2v+w)} F_{PR\Lambda}(u,v,w;x,z)\,dw\,dv\,du \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} e^{-2(2u+2v+w)} \frac{\partial^3 F_{PR\Lambda}}{\partial u\partial v\partial w}\,dw\,dv\,du \\
&= \int_0^{+\infty}\int_0^{+\infty} \Big(e^{-2(2u+2v+w)} \frac{\partial^2 F_{PR\Lambda}}{\partial u\partial v}\Big|_{w=0}^{+\infty} \\
&\quad - \int_0^{+\infty} (-2) e^{-2(2u+2v+w)} \frac{\partial^2 F_{PR\Lambda}}{\partial u\partial v}\,dw \Big)\,dv\,du \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 2e^{-2(2u+2v+w)} \frac{\partial^2 F_{PR\Lambda}}{\partial u\partial v}\,dv\,du\,dw \\
&= \int_0^{+\infty}\int_0^{+\infty} \Big(2e^{-2(2u+2v+w)} \frac{\partial F_{PR\Lambda}}{\partial u}\Big|_{v=0}^{+\infty} - \int_0^{+\infty} (-8) e^{-2(2u+2v+w)} \frac{\partial F_{PR\Lambda}}{\partial u}\,dv \Big)\,du\,dw \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 8e^{-2(2u+2v+w)} \frac{\partial F_{PR\Lambda}}{\partial u}\,du\,dw\,dv \\
&= \int_0^{+\infty}\int_0^{+\infty} \Big(8e^{-2(2u+2v+w)} F_{PR\Lambda}(u,v,w;x,z)\Big|_{u=0}^{+\infty} \\
&\quad - \int_0^{+\infty} (-32) e^{-2(2u+2v+w)} F_{PR\Lambda}(u,v,w;x,z)\,du \Big)\,dw\,dv \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 32e^{-2(2u+2v+w)} F_{PR\Lambda}(u,v,w;x,z)\,dw\,dv\,du \\
&\le \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 32e^{-2(2u+2v+w)} \min\{F_P(u;x), F_R(v;z), F_\Lambda(w)\}\,dw\,dv\,du \\
&= \int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 32e^{-2(2u+2v+w)} \Big(\int_0^1 1_{t\le F_P(u;x)} 1_{t\le F_R(v;z)} 1_{t\le F_\Lambda(w)}\,dt \Big)\,dw\,dv\,du \\
&= \int_0^1\int_0^{+\infty}\int_0^{+\infty}\int_0^{+\infty} 32e^{-2(2u+2v+w)} 1_{u\ge F_P^{-1}(t;x)} 1_{v\ge F_R^{-1}(t;z)} 1_{w\ge F_\Lambda^{-1}(t)}\,dw\,dv\,du\,dt \\
&= \int_0^1 \int_{F_P^{-1}(t;x)}^{+\infty}\int_{F_R^{-1}(t;z)}^{+\infty}\int_{F_\Lambda^{-1}(t)}^{+\infty} 32e^{-2(2u+2v+w)}\,dw\,dv\,du\,dt \\
&= \int_0^1 e^{-2\left(2F_P^{-1}(t;x)+2F_R^{-1}(t;z)+F_\Lambda^{-1}(t)\right)}\,dt\ .
\end{aligned}
$$

Thus, for any $0 < t_0 < 1$,

$$\begin{aligned}
&\mathbb{E}\Big[e^{-2\left(2P(x)+2R(z)+\Lambda\right)}\Big] \\
&\leq \int_0^1 e^{-2\left(2F_P^{-1}(t;x)+2F_R^{-1}(t;z)+F_\Lambda^{-1}(t)\right)}dt \\
&\leq \int_0^{t_0} \overbrace{e^{-2\left(2F_P^{-1}(0;x)+2F_R^{-1}(0;z)+F_\Lambda^{-1}(0)\right)}}^{1} dt + \int_{t_0}^1 e^{-2\left(2F_P^{-1}(t_0;x)+2F_R^{-1}(t_0;z)+F_\Lambda^{-1}(t_0)\right)}dt \\
&\leq t_0 + e^{-2\left(2F_P^{-1}(t_0;x)+2F_R^{-1}(t_0;z)+F_\Lambda^{-1}(t_0)\right)} .
\end{aligned}$$

By Chebyshev's inequality (using $t_0^{-\frac{1}{2}}$ as the constant), $F_P^{-1}(t_0;x) \geq (\lambda - \frac{\varepsilon}{\sqrt{t_0}})x$, $F_R^{-1}(t_0;z) \geq (\lambda - \frac{\varepsilon}{\sqrt{t_0}})y$, and $F_\Lambda^{-1}(t_0) \geq (\lambda - \frac{\varepsilon}{\sqrt{t_0}})L$. Thus,

$$\mathbb{E}\Big[e^{-2\left(2P(x)+2R(z)+\Lambda\right)}\Big] \leq t_0 + e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x+4z+2L)} .$$

Thus,

$$\begin{aligned}
\mathrm{Var}[X_{G^*}] \leq & \int_f^{f+y_0}\int_0^{+\infty}\Big(t_0 + e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x+4z+2L)}\Big)2e^{-3x-z+2f}dzdx \\
&+ \int_{f+y_0}^{+\infty}\int_{x-f-y_0}^{+\infty}\Big(t_0 + e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x+4z+2L)}\Big)4e^{-5x-z+2f+2y_0}dzdx + O(f) \\
= & \int_f^{f+y_0}\Big(2t_0e^{-3x+2f} + \frac{2}{1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}}}e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x+2L)-3x+2f}\Big)dx \\
&+ \int_{f+y_0}^{+\infty}\Big(4t_0e^{-6x+3f+3y_0} + \frac{4}{1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}}}e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(8x-4f-4y_0+2L)-6x+3f+3y_0}\Big)dx + O(f) \\
= & \frac{2}{3}t_0(e^{-f} - e^{-f-3y_0}) + \frac{2}{(1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})(3+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})}\big(e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4f+2L)-f} - e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4f+4y_0+2L)-f-3y_0}\big) \\
&+ \frac{4}{6}t_0e^{-3f-3y_0} + \frac{4}{(1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})(6+8\lambda-\frac{8\varepsilon}{\sqrt{t_0}})}e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4f+4y_0+2L)-3f-3y_0} + O(f) \\
= & \frac{2}{3}t_0 + \frac{2e^{-2L(\lambda-\frac{\varepsilon}{\sqrt{t_0}})}}{(1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})(3+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})} + O(f) .
\end{aligned}$$

Case 2: Balanced tree. Let $P(x)$ be functions to random variables denoting SU difference in heights of points (p,p') where p' is x CU distance above p; let $Q(y)$ be functions to random variables denoting SU difference in heights of points (q,q') where q' is y CU distance above q. Note that $P(x_0+z)-P(x_0) = Q(y_0+z)-Q(y_0)$ where $P(x_0)$ and $Q(y_0)$ denote the SU length of (p,r) and (q,r), respectively. Let random variable $\Lambda := \big(l_{S^\dagger}(a,p) + l_{S^\dagger}(b,p) + l_{S^\dagger}(c,q) +$

$l_{S^\dagger}(d,q)\big)$ be the total SU terminal branch lengths and the constant value L be the CU distance corresponding to Λ.

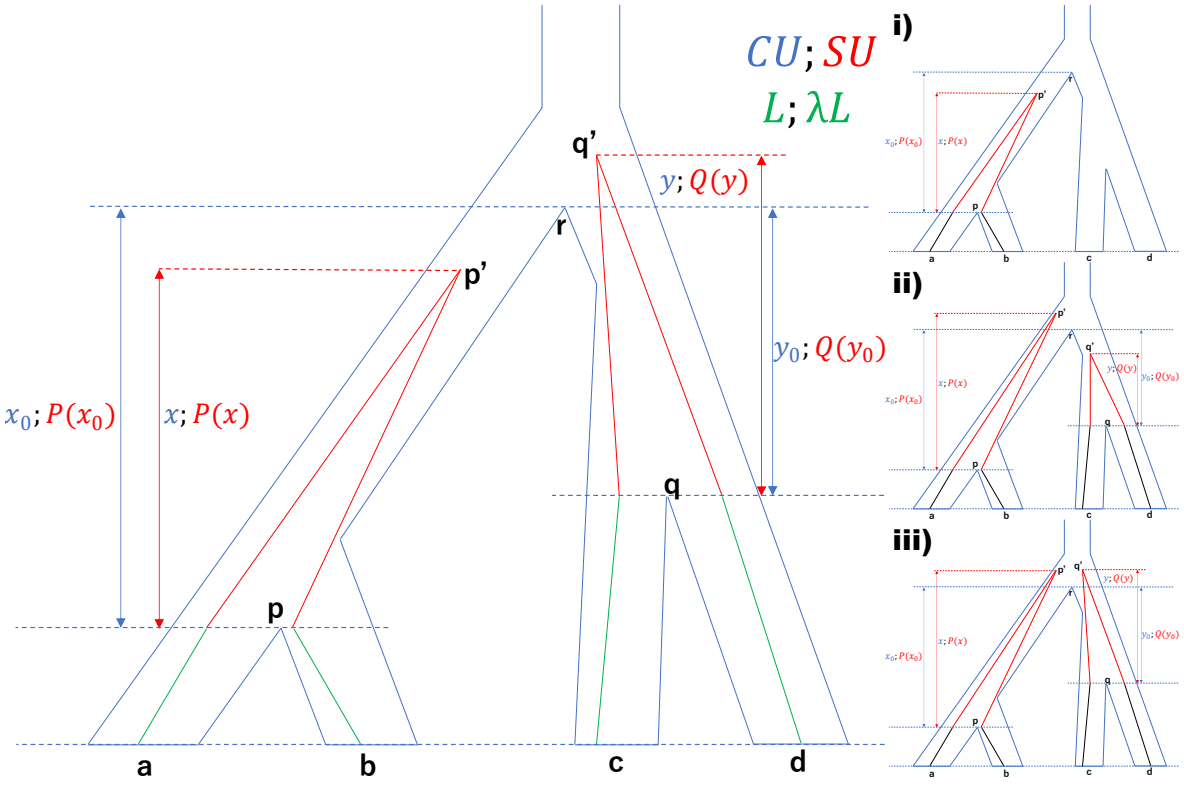

Figure S3.25. Illustration of the balanced case. Lengths in CU/SU units are denoted in blue/red. Branches in green have a total length L/Λ in CU/SU units. The right-hand side shows the position of p' and q' in relation to r in various cases.

- δ_1: Here,

$$\mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = \mathbb{E}\Big[\int_0^{x_0}\int_0^{+\infty} e^{-2P(x)-2Q(y)-\Lambda} e^{-x} e^{-y} dy dx\Big] ;$$

and

$$\mathbb{E}\big[\delta_1 w_G^2(ab|cd)\big] \leq \mathbb{E}\big[\delta_1 w_G(ab|cd)\big] = O(f) .$$

- δ_2: Here,

$$\mathbb{E}\big[\delta_2 w_G(ab|cd)\big] = \mathbb{E}\Big[\int_{x_0}^{+\infty}\int_{0}^{y_0} e^{-2P(x)-2Q(y)-\Lambda}e^{-x}e^{-y}dydx\Big]\ ;$$

and

$$\mathbb{E}\big[\delta_2 w_G^2(ab|cd)\big] \leq \mathbb{E}\big[\delta_2 w_G(ab|cd)\big] = O(f)\ .$$

- δ_3: Similar to the unbalanced case, when G has the topology $ab|cd$, either p' or q' must be the lowest point of coalescence, and by symmetry, the two cases must have the same PDFs. Thus,

$$\begin{aligned}\mathbb{E}\big[\delta_3 w_G^2(ab|cd)\big] &= \mathbb{E}\Big[\int_{x_0}^{+\infty}\int_{x-x_0+y_0}^{+\infty} e^{-4P(x)-4Q(y)-2\Lambda}2e^{-x_0}e^{-y_0}e^{-6x+6x_0}e^{-y+x-x_0+y_0}dydx\Big]\\ &= \int_{x_0}^{+\infty}\int_{x-x_0+y_0}^{+\infty}\mathbb{E}\Big[e^{-4P(x)-4Q(y)-2\Lambda}\Big]2e^{-5x-y+4x_0}dydx\ .\end{aligned}$$

Replacing in (3.13), we get

$$\begin{aligned}\mathbb{E}[X_G] &= \mathbb{E}\big[(\delta_1+\delta_2)w_G(ab|cd)\big]\\ &= \mathbb{E}\Big[\int_0^{x_0}\int_0^{+\infty} e^{-2P(x)-2Q(y)-\Lambda}e^{-x}e^{-y}dydx + \int_{x_0}^{+\infty}\int_0^{y_0} e^{-2P(x)-2Q(y)-\Lambda}e^{-x}e^{-y}dydx\Big]\\ &\geq \int_0^{x_0}\int_0^{+\infty} e^{-2\lambda x-2\lambda y-\lambda L}e^{-x}e^{-y}dydx + \int_{x_0}^{+\infty}\int_0^{y_0} e^{-2\lambda x-2\lambda y-\lambda L}e^{-x}e^{-y}dydx\\ &= \frac{(x_0+y_0)e^{-\lambda L}}{1+2\lambda} + O(f^2) = \frac{fe^{-\lambda L}}{1+2\lambda} + O(f^2)\ ;\end{aligned}$$

and replacing in (3.14), for any $0 < t_0 < 1$,

$$\begin{aligned}
\mathrm{Var}[X_G] =& \mathbb{E}\big[(\delta_1+\delta_2+2\delta_3)w_G^2(ab|cd)\big] - \mathbb{E}^2[X_G] \\
=& \mathbb{E}\big[2\delta_3 w_G^2(ab|cd)\big] + O(f) \\
=& \int_{x_0}^{+\infty}\int_{x-x_0+y_0}^{+\infty} \mathbb{E}\Big[e^{-4P(x)-4Q(y)-2\Lambda}\Big] 4e^{-5x-y+4x_0}dydx \\
\leq& \int_{x_0}^{+\infty}\int_{x-x_0+y_0}^{+\infty} \Big(t_0 + e^{(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x+4y+2L)}\Big) 4e^{-5x-y+4x_0}dydx + O(f) \\
=& \int_{x_0}^{+\infty} \Big(4e^{-6x-y_0+5x_0}t_0 + \frac{4}{1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}}} e^{-6x-y_0+5x_0+(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(8x-4x_0+4y_0+2L)}\Big)dx + O(f) \\
=& \frac{4}{6}e^{-x_0-y_0}t_0 + \frac{4}{(1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})(6+8\lambda-\frac{8\varepsilon}{\sqrt{t_0}})} e^{-x_0-y_0+(-\lambda+\frac{\varepsilon}{\sqrt{t_0}})(4x_0+4y_0+2L)} + O(f) \\
=& \frac{2}{3}t_0 + \frac{2e^{-2L(\lambda-\frac{\varepsilon}{\sqrt{t_0}})}}{(1+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})(3+4\lambda-\frac{4\varepsilon}{\sqrt{t_0}})} + O(f)\,,
\end{aligned}$$

from which our assumption of $\mathrm{Var}[X_G] = \Theta_f(1)$ follows.

Thus, for both balanced and unbalanced trees, the variance is bounded the by same expression, and thus in both cases,

$$\begin{aligned}
\mathrm{Var}[X_{G^*}] \leq& \frac{2}{3}t_0 + 2\frac{\frac{e^{-2\lambda L}}{(1+4\lambda)(3+4\lambda)}}{(1-\frac{4\varepsilon}{(1+4\lambda)\sqrt{t_0}})(1-\frac{4\varepsilon}{(3+4\lambda)\sqrt{t_0}})e^{-\frac{2\varepsilon L}{\sqrt{t_0}}}} + O(f) \\
\leq& \frac{2}{3}t_0 + 2\frac{\frac{e^{-2\lambda L}}{(1+4\lambda)(3+4\lambda)}}{(1-\frac{4\varepsilon}{(1+4\lambda)\sqrt{t_0}})(1-\frac{4\varepsilon}{(3+4\lambda)\sqrt{t_0}})(1-\frac{2\varepsilon L}{\sqrt{t_0}})} + O(f) \\
\leq& \frac{2}{3}t_0 + 2\frac{\frac{e^{-2\lambda L}}{(1+4\lambda)(3+4\lambda)}}{(1-\frac{4\varepsilon}{(1+4\lambda)\sqrt{t_0}}-\frac{4\varepsilon}{(3+4\lambda)\sqrt{t_0}}-\frac{2\varepsilon L}{\sqrt{t_0}})} + O(f) \\
=& \frac{2}{3}t_0 + \frac{2e^{-2\lambda L}}{(3+16\lambda+16\lambda^2)-\frac{\varepsilon}{\sqrt{t_0}}\big((16+32\lambda)+(6+32\lambda+32\lambda^2)L\big)} + O(f)\,.
\end{aligned}$$

Now, let $C := (16+32\lambda)+(6+32\lambda+32\lambda^2)L, t_0 = \big(\frac{C^{\frac{1}{3}}\varepsilon^{\frac{1}{3}}}{(3+16\lambda+16\lambda^2)e^{\frac{2}{3}\lambda L}}\big)^2$, we get

$$\begin{aligned}
\mathrm{Var}[X_{G^*}] \leq& \frac{2e^{-2\lambda L}}{3(3+16\lambda+16\lambda^2)^2}\Big((\varepsilon e^{\lambda L}C)^{\frac{2}{3}} + \frac{9+48\lambda+48\lambda^2}{1-(\varepsilon e^{\lambda L}C)^{\frac{2}{3}}}\Big) + O(f) \\
=& \frac{2e^{-2\lambda L}}{3(3+16\lambda+16\lambda^2)}\Big(\frac{(\varepsilon e^{\lambda L}C)^{\frac{2}{3}}}{3+16\lambda+16\lambda^2} + 3 + \frac{3(\varepsilon e^{\lambda L}C)^{\frac{2}{3}}}{1-(\varepsilon e^{\lambda L}C)^{\frac{2}{3}}}\Big) + O(f)\,.
\end{aligned}$$

Now, recalling that $\varepsilon = \frac{e^{-\lambda L}}{C}\Big(\frac{20(\lambda+\lambda^2)}{9(1+2\lambda)^2}\Big)^{\frac{3}{2}}$,

$$\begin{aligned}
\mathrm{Var}[X_{G^*}] \leq & \frac{2}{3(3+16\lambda+16\lambda^2)(1+2\lambda)^2} \\
& \Big(\frac{\frac{20}{9}(\lambda+\lambda^2)}{3+16\lambda+16\lambda^2}+3+\frac{3(\frac{20}{9})(\lambda+\lambda^2)}{1-\frac{\frac{20}{9}(\lambda+\lambda^2)}{1+4\lambda+4\lambda^2}}\Big)+O(f) \\
\leq & \frac{2}{3(3+16\lambda+16\lambda^2)(1+2\lambda)^2}\Big(\frac{20}{27}\lambda+3+\frac{\frac{20}{3}(\lambda+\lambda^2)}{1-\frac{5}{9}}\Big)+O(f) \\
= & \frac{2}{3(3+16\lambda+16\lambda^2)(1+2\lambda)^2}\Big(\frac{20}{27}\lambda+3+15(\lambda+\lambda^2)\Big)+O(f) \\
< & \frac{2}{3(1+2\lambda)^2}\Big(\frac{3+16\lambda+15\lambda^2}{3+16\lambda+16\lambda^2}\Big)+O(f)\,.
\end{aligned}$$

□

Theorem 3.2. *Under the conditions of Proposition 3.2 or Proposition 3.3,*

$$P\Big(\sum_{G\in\mathscr{G}} w_G(ab|cd) \leq \sum_{G\in\mathscr{G}} w_G(ac|bd)\Big) \leq P\Big(\sum_{G\in\mathscr{G}} \delta_G(ab|cd) \leq \sum_{G\in\mathscr{G}} \delta_G(ac|bd)\Big).$$

Proof. We start with proving this theorem under the conditions of Proposition 3.2. Recall $X_G := w_G(ab|cd) - w_G(ac|bd)$ and $Y_G := \delta_G(ab|cd) - \delta_G(ac|bd)$, and let $\bar{X_{\mathscr{G}}} = \frac{1}{k}\sum_{G\in\mathscr{G}} X_G$ and $\bar{Y_{\mathscr{G}}} = \frac{1}{k}\sum_{G\in\mathscr{G}} Y_G$. Recall also that under Proposition 3.2, proved below, under conditions of Theorem 3.2, we have $\mathrm{Var}[X_G] = \Omega(1)$ and

$$\frac{\mathbb{E}[X_G]}{\sqrt{\mathrm{Var}[X_G]}} = -\sqrt{\frac{3+16\lambda+16\lambda^2}{3+16\lambda+15\lambda^2}}\sqrt{\frac{3}{2}}f+O(f^2)\,. \tag{3.15}$$

Similarly, we can compute the ratio of mean and variance for Y (corresponding to unweighted ASTRAL):

$$\mathbb{E}[Y_G] := \mathbb{E}\big[\delta_G(ab|cd) - \delta_G(ac|bd)\big] = 1-e^{-f} = f+O(f^2)$$

$$\mathrm{Var}[Y_G] := \mathrm{Var}\big[\delta_G(ab|cd) - \delta_G(ac|bd)\big] = \frac{5}{3}e^{-f} - e^{-2f} = \frac{2}{3} + O(f)$$

and thus,

$$\frac{\mathbb{E}[Y_G]}{\sqrt{\mathrm{Var}[Y_G]}} = \sqrt{\frac{3}{2}}f + O(f^2)\,. \tag{3.16}$$

Given Proposition 3.2, we can use Berry–Esseen theorem to derive

$$\begin{aligned}\mathrm{P}\big(\bar{X_{\mathscr{G}}} \le 0\big) =& \mathrm{P}\big(\frac{\sqrt{k}}{\sqrt{\mathrm{Var}[X_G]}}(\bar{X_{\mathscr{G}}} - \mathbb{E}[X_G]) \le -\frac{\sqrt{k}}{\sqrt{\mathrm{Var}[X_G]}}\mathbb{E}[X_G]\big) = \\ & \Phi(-\sqrt{k}\frac{\mathbb{E}[X_G]}{\sqrt{\mathrm{Var}[X_G]}}) + O(\frac{1}{\sqrt{k}})\,,\end{aligned}$$

where Φ denotes CDF of the standard Normal distribution. Since $k = \Theta(f^{-2})$,

$$\mathrm{P}\big(\bar{X_{\mathscr{G}}} \le 0\big) = \Phi(-\sqrt{k}\frac{\mathbb{E}[X_G]}{\sqrt{\mathrm{Var}[X_G]}}) + O(f) \tag{3.17}$$

and

$$\mathrm{P}\big(\bar{Y_{\mathscr{G}}} \le 0\big) = \Phi(-\sqrt{k}\frac{\mathbb{E}[Y_G]}{\sqrt{\mathrm{Var}[Y_G]}}) + O(f)\,, \tag{3.18}$$

Combining equations (3.17) and (3.18) with (3.15) and (3.16), we get

$$\mathrm{P}\big(\sum_{G\in\mathscr{G}} w_G(ab|cd) \le \sum_{G\in\mathscr{G}} w_G(ac|bd)\big) = \Phi(-\sqrt{\frac{3+16\lambda+16\lambda^2}{3+16\lambda+15\lambda^2}}\sqrt{\frac{3}{2}}f\sqrt{k}) + O(f)$$

and

$$\mathrm{P}\big(\sum_{G\in\mathscr{G}} \delta_G(ab|cd) \le \sum_{G\in\mathscr{G}} \delta_G(ac|bd)\big) = \Phi(-\sqrt{\frac{3}{2}}f\sqrt{k}) + O(f)\,.$$

As $f \to 0$, the interval $(-\sqrt{1+\frac{4\lambda+4\lambda^2}{3(1+2\lambda)^2}}\sqrt{\frac{3}{2}}f\sqrt{k}, -\sqrt{\frac{3}{2}}f\sqrt{k})$ does not shrink because

$\Theta(f\sqrt{k}) = \Theta(1)$. Thus, we have

$$\Phi(-\sqrt{\frac{3}{2}}f\sqrt{k}) - \Phi(-\sqrt{1+\frac{4\lambda+4\lambda^2}{3(1+2\lambda)^2}}\sqrt{\frac{3}{2}}f\sqrt{k}) = \Theta(1)$$

ensuring that

$$\mathrm{P}\big(\sum_{G\in\mathscr{G}} w_G(ab|cd) \leq \sum_{G\in\mathscr{G}} w_G(ac|bd)\big) \leq \mathrm{P}\big(\sum_{G\in\mathscr{G}} \delta_G(ab|cd) \leq \sum_{G\in\mathscr{G}} \delta_G(ac|bd)\big)\,.$$

The proof under Proposition 3.3 is similar. Recall that under Proposition 3.3, $\mathrm{Var}[X_{G^*}] = \Theta_f(1)$ and

$$\frac{\mathbb{E}[X_{G^*}]}{\sqrt{\mathrm{Var}[X_{G^*}]}} \geq \sqrt{\frac{3}{2}}\Big(1-\frac{4\lambda^2}{(1+4\lambda)^2}\Big)^{-\frac{1}{2}} f + O(f^2)\,. \tag{3.19}$$

Given this result, the rest of the proof is similar to the proof under the conditions of Proposition 3.2, culminating in

$$\mathrm{P}\big(\sum_{G^*\in\mathscr{G}} w_{G^*}(ab|cd) \leq \sum_{G^*\in\mathscr{G}} w_{G^*}(ac|bd)\big) \leq \Phi\bigg(-\Big(1-\frac{4\lambda^2}{(1+4\lambda)^2}\Big)^{-\frac{1}{2}}\sqrt{\frac{3}{2}}f\sqrt{k}\bigg) + O(f)\,.$$

□

3.D.3 Placement-based Algorithm

In this section, for a node v in tree G, we let $\mathscr{L}_v$ denote the set of leaves under v.

Proof of Theorem 3.3

Theorem 3.3. *Let S be a species tree, i be a species not in $\mathscr{L}_S$, $\mathscr{S}$ be the set of possible species tree topologies by placing i onto S, and S' be the output of Algorithm S3.1. Then, $W(S',\mathscr{G}) = \max_{\hat{S}\in\mathscr{S}} W(\hat{S},\mathscr{G})$.*

Proof. We start with two propositions, proved below.

Proposition 3.5. *After each call to* `ColorLeafSet`$(\mathscr{L}^*,X,T,\mathscr{G},W)$ *with a $T \neq \emptyset$, $W[T] = \sum_{G\in\mathscr{G}} W(T,G)$.*

Proposition 3.6. *Before calling* `OptimalTreeDP` *in line 6 of Algorithm S3.1, lookup table W contains all tripartitions corresponding to internal nodes of all tree topologies in $\mathscr{S}$.*

By Proposition 3.6, all tripartitions corresponding to internal nodes of all tree topologies in $\mathscr{S}$ pre-computed. Then, `OptimalTreeDP` uses a dynamic programming algorithm similar to the one formulated by Mirarab and Warnow 2015 to compute $\arg\max_{\hat{S}\in\mathscr{S}} W(\hat{S},\mathscr{G})$. □

Proposition 3.5. *After each call to* `ColorLeafSet`$(\mathscr{L}^*,X,T,\mathscr{G},W)$ *with a $T \neq \emptyset$, $W[T] = \sum_{G\in\mathscr{G}} W(T,G)$.*

Proof. For a gene tree node w and a color X, let $\mathscr{L}_w^X$ denote the set of leaves in $\mathscr{L}_w$ colored by X. For an internal node w, let u,v be the children of w, p be the parent of w (if w is not the root), and e denote the branch (w,p). For a leaf i and internal node w, let $\mathscr{P}_{i,w}$ denote path between i and w and $s(\mathscr{P}) = 1 - \prod_{\hat{e}\in\mathscr{P}}(1-s(\hat{e}))$. For leaves i,j, let $m(i,j)$ denote MRCA of i and j. Referring back to Table S3.1, we first establish the connection between recursive formulas of the algorithm and counter definitions.

- When $u_X = \sum_{i\in\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,w})}$, $v_X = \sum_{i\in\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,w})}$,

$$w_X := \Big((u_X + v_X)e^{-l(e)}\Big) = \sum_{i\in\mathscr{L}_w^X} e^{-l(\mathscr{P}_{i,w})}e^{-l(e)} = \sum_{i\in\mathscr{L}_w^X} e^{-l(\mathscr{P}_{i,p})}\,.$$

- When $u_{XX}^+ = \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,j})}$, $v_{XX}^+ = \sum_{\{i,j\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})}$,

$$\begin{aligned} w_{XX}^+ := u_{XX}^+ + v_{XX}^+ + u_X v_X &= \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,j})} + \sum_{\{i,j\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})} + \sum_{i\in\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,w})} \sum_{j\in\mathscr{L}_v^X} e^{-l(\mathscr{P}_{j,w})} \\ &= \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,j})} + \sum_{\{i,j\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})} + \sum_{i\in\mathscr{L}_u^X} \sum_{j\in\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})} \\ &= \sum_{\{i,j\}\subseteq\mathscr{L}_w^X} e^{-l(\mathscr{P}_{i,j})}\,. \end{aligned}$$

- For $X \neq Y$, when $u_{XY}^+ = \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})}$, $v_{XY}^+ = \sum_{(i,j)\in\mathscr{L}_v^X\times\mathscr{L}_v^Y} e^{-l(\mathscr{P}_{i,j})}$,

$$\begin{aligned} w_{XY}^+ :=& u_{XY}^+ + v_{XY}^+ + u_X v_Y + u_Y v_X \\ =& \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})} + \sum_{(i,j)\in\mathscr{L}_v^X\times\mathscr{L}_v^Y} e^{-l(\mathscr{P}_{i,j})} + \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_v^Y} e^{-l(\mathscr{P}_{i,j})} + \sum_{(i,j)\in\mathscr{L}_v^X\times\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})} \\ =& \sum_{\{i,j\}\subseteq\mathscr{L}_w^X\times\mathscr{L}_w^X} e^{-l(\mathscr{P}_{i,j})}\,. \end{aligned}$$

- When $u_{XX}^- = \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,j})} \prod_{\hat{e}\in\mathscr{P}_{m(i,j),w}} \big(1-s(\hat{e})\big)$, $v_{XX}^- = \sum_{\{i,j\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})} \prod_{\hat{e}\in\mathscr{P}_{m(i,j),w}} \big(1-s(\hat{e})\big)$,

$$\begin{aligned} w_{XX}^- :=& \big(u_{XX}^- + v_{XX}^- + u_X v_X\big)\big(1-s(e)\big) \\ =& \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} e^{-l(\mathscr{P}_{i,j})} \prod_{\hat{e}\in\mathscr{P}_{m(i,j),p}} \big(1-s(\hat{e})\big) + \sum_{\{i,j\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})} \prod_{\hat{e}\in\mathscr{P}_{m(i,j),p}} \big(1-s(\hat{e})\big) \\ &+ \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_v^X} e^{-l(\mathscr{P}_{i,j})}\big(1-s(e)\big) = \sum_{\{i,j\}\subseteq\mathscr{L}_w^X} e^{-l(\mathscr{P}_{i,j})} \prod_{\hat{e}\in\mathscr{P}_{m(i,j),p}} \big(1-s(\hat{e})\big)\,. \end{aligned}$$

- When $u_{XY}^- = \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})} \big(1-s(\mathscr{P}_{m(i,j),w})\big)$, $v_{XY}^- = \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})} \big(1-s(\mathscr{P}_{m(i,j),w})\big)$, and $X \neq Y$, similarly,

$$w_{XY}^- :=\big(u_{XY}^- + v_{XY}^- + u_X v_Y + u_Y v_X\big)\big(1-s(e)\big) = \sum_{(i,j)\in\mathscr{L}_w^X\times\mathscr{L}_w^Y} e^{-l(\mathscr{P}_{i,j})} \big(1-s(\mathscr{P}_{m(i,j),p})\big)\,.$$

- For $X \neq Y$, when $u_{XX|Y} = \sum_{\{i,j\}\subseteq\mathscr{L}_u^X} \sum_{k\in\left\{k'\in\mathscr{L}_v^Y:\mathscr{L}_{m(i,j)}\subseteq\neq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})} s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)$, $v_{XX|Y} =$

$\sum_{\{i,j\}\subseteq\mathscr{L}_v^X}\sum_{k\in\left\{k'\in\mathscr{L}_v^Y:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)$,

$$w_{XX|Y} := \big(u_{XX|Y}+v_{XX|Y}+(u^+_{XX}-u^-_{XX})v_Y+u_Y(v^+_{XX}-v^-_{XX})\big)e^{-l(e)}\ .$$

Notice that $(u^+_{XX}-u^-_{XX})v_Y = \sum_{\{i,j\}\subseteq\mathscr{L}_u^X}\sum_{k\in\mathscr{L}_v^Y} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),w}\big)$ and $u_Y(v^+_{XX}-v^-_{XX}) = \sum_{\{i,j\}\subseteq\mathscr{L}_v^X}\sum_{k\in\mathscr{L}_u^Y} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),w}\big)$. Thus,

$$\begin{aligned} w_{XX|Y} &= \sum_{\{i,j\}\subseteq\mathscr{L}_w^X}\sum_{k\in\left\{k'\in\mathscr{L}_w^Y:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)e^{-l(e)} \\ &= \sum_{\{i,j\}\subseteq\mathscr{L}_w^X}\sum_{k\in\left\{k'\in\mathscr{L}_w^Y:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,p})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)\ . \end{aligned}$$

- Similarly, when $u_{XY|Z} = \sum_{(i,j)\in\mathscr{L}_u^X\times\mathscr{L}_u^Y}\sum_{k\in\left\{k'\in\mathscr{L}_u^Z:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)$, $v^-_{XY|Z} = \sum_{(i,j)\in\mathscr{L}_v^X\times\mathscr{L}_v^Y}\sum_{k\in\left\{k'\in\mathscr{L}_v^Z:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)$, for distinct X,Y,Z,

$$w_{XY|Z} = \sum_{(i,j)\in\mathscr{L}_w^X\times\mathscr{L}_w^Y}\sum_{k\in\left\{k'\in\mathscr{L}_w^Z:\mathscr{L}_{m(i,j)}\subsetneq\mathscr{L}_{m(i,k')}\right\}} e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,p})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big)\ .$$

- For distinct X,Y,Z,

$$\begin{aligned} w_{XX|YZ} :=& v_X u_{YZ|X}+u_X v_{YZ|X}+u_{XX|Z}v_Y+v_{XX|Z}u_Y+u_{XX|Y}v_Z+v_{XX|Y}u_Z \\ &+(u^+_{YZ}v^+_{XX}-u^-_{YZ}v^-_{XX})+(u^+_{XX}v^+_{YZ}-u^-_{XX}v^-_{YZ})\ . \end{aligned}$$

Notice that,

$$\begin{aligned} v_X u_{YZ|X} &= \sum_{(h,i,j,k)\in\mathscr{L}_v^X\times\mathscr{L}_u^Y\times\mathscr{L}_u^Z\times\mathscr{L}_u^X}\delta_G(hk|ij)e^{-l(\mathscr{P}_{h,w})}e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,w})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big) \\ &= \sum_{(h,i,j,k)\in\mathscr{L}_v^X\times\mathscr{L}_u^Y\times\mathscr{L}_u^Z\times\mathscr{L}_u^X}\delta_G(hk|ij)e^{-l(\mathscr{P}_{i,j})-l(\mathscr{P}_{k,h})}s\big(\mathscr{P}_{m(i,j),m(i,k)}\big) \\ &= \sum_{(h,i,j,k)\in\mathscr{L}_v^X\times\mathscr{L}_u^Y\times\mathscr{L}_u^Z\times\mathscr{L}_u^X} w_G(hk|ij)\ . \end{aligned}$$

Similarly,

$$u_X v_{YZ|X} = \sum_{\substack{h\in\mathscr{L}_u^X\\ i\in\mathscr{L}_v^Y\\ j\in\mathscr{L}_v^Z\\ k\in\mathscr{L}_v^X}} w_G(hk|ij), u_{XX|Z}v_Y = \sum_{\substack{\{h,i\}\subseteq\mathscr{L}_u^X\\ j\in\mathscr{L}_u^Z\\ k\in\mathscr{L}_v^Y}} w_G(hi|jk), v_{XX|Z}u_Y = \sum_{\substack{\{h,i\}\subseteq\mathscr{L}_v^X\\ j\in\mathscr{L}_v^Z\\ k\in\mathscr{L}_u^Y}} w_G(hi|jk),$$

$$u_{XX|Y}v_Z = \sum_{\substack{\{h,i\}\subseteq\mathscr{L}_u^X\\ j\in\mathscr{L}_u^Y\\ k\in\mathscr{L}_v^Z}} w_G(hi|jk), v_{XX|Y}u_Z = \sum_{\substack{\{h,i\}\subseteq\mathscr{L}_v^X\\ j\in\mathscr{L}_v^Y\\ k\in\mathscr{L}_u^Z}} w_G(hi|jk)\ .$$

Also,

$$\begin{aligned}
u^+_{YZ}v^+_{XX} - u^-_{YZ}v^-_{XX} &= \sum_{(h,i)\in\mathscr{L}_u^Y\times\mathscr{L}_u^Z}\ \sum_{\{j,k\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{h,i})-l(\mathscr{P}_{j,k})}\\
&\quad - \sum_{(h,i)\in\mathscr{L}_u^Y\times\mathscr{L}_u^Z}\ \sum_{\{j,k\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{h,i})-l(\mathscr{P}_{j,k})} \prod_{\hat{e}\in\mathscr{P}_{m(h,i),w}} \big(1-s(\hat{e})\big) \prod_{\hat{e}\in\mathscr{P}_{m(j,k),w}} \big(1-s(\hat{e})\big)\\
&= \sum_{(h,i)\in\mathscr{L}_u^Y\times\mathscr{L}_u^Z}\ \sum_{\{j,k\}\subseteq\mathscr{L}_v^X} e^{-l(\mathscr{P}_{h,i})-l(\mathscr{P}_{j,k})} \left(1 - \prod_{\hat{e}\in\mathscr{P}_{m(h,i),m(j,k)}} \big(1-s(\hat{e})\big)\right)\\
&= \sum_{(h,i)\in\mathscr{L}_u^Y\times\mathscr{L}_u^Z}\ \sum_{\{j,k\}\subseteq\mathscr{L}_v^X} w_G(hi|jk)\ .
\end{aligned}$$

Similarly,

$$u^+_{XX}v^+_{YZ} - u^-_{XX}v^-_{YZ} = \sum_{\{h,i\}\subseteq\mathscr{L}_u^X}\ \sum_{(j,k)\in\mathscr{L}_v^Y\times\mathscr{L}_v^Z} w_G(hi|jk)\ .$$

Notice that above cases count exactly once all quartets $hi|jk$ for all leaf nodes h,i colored X, j colored Y, k colored Z such that MRCA of h,i,j,k is w; namely,

$$w_{XX|YZ} = \sum_{\{h,i\}\subseteq\mathscr{L}_w^X}\ \sum_{j\in\mathscr{L}_w^Y}\ \sum_{k\in\{k':k'\in\mathscr{L}_w^Z,\mathrm{MRCA}(h,i,j,k')=w\}} w_G(hi|jk)\ .$$

- We define $I(G)$ to be the set of internal nodes of gene tree G and $\mathscr{L}_G^X$ be the set of leaves of gene tree G with color X. It is trivial to verify that at the

$$Q = \sum_{G\in\mathscr{G}}\ \sum_{w\in I(G)} w_{AA|BC} + \sum_{G\in\mathscr{G}}\ \sum_{w\in I(G)} w_{BB|CA} + \sum_{G\in\mathscr{G}}\ \sum_{w\in I(G)} w_{CC|AB}\ .$$

At the end of procedure `UpdateCounters`, $\sum_{w\in I(G)} w_{XX|YZ} = \sum_{\{h,i\}\subseteq\mathscr{L}_G^X}\sum_{(j,k)\in\mathscr{L}_G^Y\times\mathscr{L}_G^Z} w_G(hi|jk)$. Thus, Q

returned by `UpdateCounters` satisfies:

$$Q = \sum_{G \in \mathcal{G}} \left(\sum_{\substack{\{h,i\} \subseteq \mathcal{L}_G^A \\ (j,k) \in \mathcal{L}_G^B \times \mathcal{L}_G^C}} w_G(hi|jk) + \sum_{\substack{\{h,i\} \subseteq \mathcal{L}_G^B \\ (j,k) \in \mathcal{L}_G^C \times \mathcal{L}_G^A}} w_G(hi|jk) + \sum_{\substack{\{h,i\} \subseteq \mathcal{L}_G^C \\ (j,k) \in \mathcal{L}_G^A \times \mathcal{L}_G^B}} w_G(hi|jk) \right) .$$

For tripartition $T = A|B|C$, note that by assumption, before the call, all the gene tree leaves are colored such that recoloring $\mathcal{L}^*$ by X would produce a coloring that matches T. Thus, at the end of the call to `ColorLeafSet`, for each gene tree G, we have $A \cap \mathcal{L}_G = \mathcal{L}_G^A$, $B \cap \mathcal{L}_G = \mathcal{L}_G^B$, and $C \cap \mathcal{L}_G = \mathcal{L}_G^C$. Then, the value returned by `UpdateCounters` satisfies:

$$Q = \sum_{G \in \mathcal{G}} W(A|B|C, G) . \tag{3.20}$$

It can be easily verified that after each call to `ColorLeafSet`$(\mathcal{L}^*, X, T, \mathcal{G}, W)$, the species tree tripartition T matches the coloring of all gene trees as required by conditions of (3.20), concluding $W[T] = Q = \sum_{G \in \mathcal{G}} W(T, G)$. □

Proposition 3.6. *Before calling* `OptimalTreeDP` *in line 6 of Algorithm S3.1, lookup table W contains all tripartitions corresponding to internal nodes of all tree topologies in* $\mathcal{S}$.

Proof. Each $\hat{S} \in \mathcal{S}$ places i above a different node w of S creating a new node corresponding to tripartition $\mathcal{L}_w|\{i\}|\mathcal{L}_S - \mathcal{L}_w$ covered in line 24. Besides new nodes, each existing internal node w of S will correspond to a different tripartition after placing i onto S depending on the relative location of w and i. Let u, v denote the larger and the smaller child of w. Node w corresponds to $\mathcal{L}_u|\{i\} \cup \mathcal{L}_v|\mathcal{L}_S - \mathcal{L}_w$ if i is under u, corresponds to $\{i\} \cup \mathcal{L}_u|\mathcal{L}_v|\mathcal{L}_S - \mathcal{L}_w$ if i is under v, and corresponds to $\mathcal{L}_u|\mathcal{L}_v|\{i\} \cup \mathcal{L}_S - \mathcal{L}_w$ if i is above w. All three cases for each node w is covered in lines 20–22. □

Proof of Theorem 3.4

Theorem 3.4. *If there exists a species tree topology S^* satisfying that for each quartet subtree $ab|cd$,*

$$\sum_{G\in\mathscr{G}} w(ab|cd) > \max\Big(\sum_{G\in\mathscr{G}} w(ac|bd), \sum_{G\in\mathscr{G}} w(ad|bc)\Big), \tag{3.6}$$

then the output of Algorithm S3.2 will be S^*.

Proof. We start with a Corollary 3.1 of Theorem 3.3

Corollary 3.1. *Assuming* (3.6)*, if S is compatible with the true tree S^*, then S' is compatible with S^*.*

By induction, W_i in line 8 of Algorithm S3.2 should contain all tripartitions of S^*, as at that time $S_i = S^*$ by Corollary 3.1. Consequentially, the output of Algorithm S3.2 must also be S^*. □

Proof of Proposition 3.4

Proposition 3.4. *The time complexity of Algorithm S3.2 is $O(kHn^2\log n)$.*

Proof. We begin by a proposition and a corollary.

Proposition 3.7. *Procedure* `ColorNode` *on any species tree node w takes $O(kH|\mathscr{L}_w|\log|\mathscr{L}_w|)$ time.*

Proof (sketch) of Proposition 3.7. We can prove this proposition by induction. For an internal node w with larger child u and smaller child v, if for some constant $C \geq \frac{6}{\log 2}$, `ColorNode` on u calls `UpdateCounters` at most $Ck|\mathscr{L}_u|(\log|\mathscr{L}_u|+1)$ times and `ColorNode` on u calls `UpdateCounters` at most $Ck|\mathscr{L}_v|(\log|\mathscr{L}_v|+1)$ times, then `ColorNode` on w calls

`UpdateCounters` at most

$$
\begin{aligned}
&Ck|\mathscr{L}_u|(\log|\mathscr{L}_u|+1)+Ck|\mathscr{L}_v|(\log|\mathscr{L}_v|+1)+3k(|\mathscr{L}_v|+1)\\
\leq&Ck|\mathscr{L}_u|(\log|\mathscr{L}_w|+1)+Ck|\mathscr{L}_v|(\log\frac{|\mathscr{L}_w|}{2}+1)+6k|\mathscr{L}_v|\\
\leq&Ck|\mathscr{L}_u|(\log|\mathscr{L}_w|+1)+Ck|\mathscr{L}_v|(\log|\mathscr{L}_w|+1)-Ck|\mathscr{L}_v|\log 2+6k|\mathscr{L}_v|\\
\leq&Ck|\mathscr{L}_w|(\log|\mathscr{L}_w|+1)+(6-C\log 2)k|\mathscr{L}_v|\\
\leq&Ck|\mathscr{L}_w|(\log|\mathscr{L}_w|+1) \text{ times.}
\end{aligned}
$$

It is easy to verify that each `UpdateCounters` takes $O(H_G)$ time where H_G is the height of the gene tree, and thus `ColorNode` on node w takes $O(kH|\mathscr{L}_w|\log|\mathscr{L}_w|)$ time. □

Corollary 3.2 (Corollary of Proposition 3.7)**.** *For any tree topology S with n species, the* `Place` *procedure on S takes $O(kHn\log n)$ time.*

`NaivePlacement` of taxon set T makes $r(|T|-3)$ calls to `Place`, each of which takes $O(kH|T|\log|T|)$ time. Thus, `NaivePlacement` takes $O(rkH|T|^2\log|T|)$ time and when $T=\mathscr{L}_S$ and $r=O(1)$, $O(rkH|T|^2\log|T|)=O(n^2kH\log n)$. □

Proofs of Theorems 3.6 and Theorem 3.5

Theorem 3.6. *Under the conditions of Theorem 3.4, the DAC Algorithm S3.3 will output S^*.*

Proof. By Theorem 3.4, S_i in line 5 of Algorithm S3.3 are compatible with S^*. With Corollary 3.1, by induction, each S_e in line 21 of Algorithm S3.3 is compatible with S^*. Consequentially, W_i in line 26 contain all tripartitions of S^*, as at that time $S_i'=S^*$, and the output of Algorithm S3.3 must also be S^*. □

Theorem 3.5. *When the inequality condition in Theorem 3.4 is satisfied, then the time complexity of the DAC algorithm is $O(n^{1.5+\varepsilon}kH)$ with arbitrarily high probability.*

Proof (sketch). From the inequality (3.6), we can trivially deduct that S^* is the species tree topology that maximizes the weighted quartet score and each S_i in line 5 of Algorithm S3.3 is compatible to S^*. Also, each C_e in line 15 of Algorithm S3.3 equals the set of species under the edges coming off of the internal nodes on the path of S^* corresponding to e.

We now introduce a proposition

Proposition 3.8. *With high probability,* $\max_{e \in E_{S_i}} |C_e| \leq 2\sqrt{n}\log n + O(\sqrt{n})$.

Proof. For each pair of nodes u, v of S^*, let $C_{u,v} := \{x : x \in \mathscr{L}_S, u \text{ is not on } \mathscr{P}_{x,v} \text{ and } v \text{ is not on } \mathscr{P}_{x,u}\}$. It is easy to verify that for every e of S_i, $C_e = C_{u,v}$ for some nodes u, v of S^*. For every u and v that are sufficiently apart so that $C_{u,v}$ has $2\sqrt{n}\log n + \omega(\sqrt{n})$ elements and a random T_i in line 4 of Algorithm S3.3,

$$P(C_{u,v} \cap T_i = \emptyset) = \left(1 - \frac{1}{\sqrt{n}}\right)^{|C_{u,v}|} \leq e^{-\frac{1}{\sqrt{n}}|C_{u,v}|} = \frac{1}{n^2} e^{-\omega(1)} = o\left(\frac{1}{n^2}\right).$$

By union bound, the probability that there exists a pair of nodes u, v of S^* such that $|C_{u,v}| \geq 2\sqrt{n}\log n + \omega(\sqrt{n})$ and $C_{u,v} \cap T_i = \emptyset$ is $o(1)$. Since, by definition, $C_e \cap T_i = \emptyset$ for every C_e, with high probability, there exists no C_e having $2\sqrt{n}\log n + \omega(\sqrt{n})$ elements. □

Since $|T_i| \sim \text{Binomial}(n, \frac{1}{\sqrt{n}})$, with high probability $|T_i| = O(\sqrt{n})$ and calling `NaivePlacement` on line 5 takes $O(n^{1.5}kH\log n)$ time. It is easy to confirm that $C_\emptyset = \emptyset$ and every call to `Place` takes as input a species tree topology of $O(\sqrt{n}\log n)$ species with high probability. Thus, with high probability, each call to `Place` takes $O(\sqrt{n}kH\log^2 n \log\log n)$ time and all $O(n)$ calls to `Place` takes $O(n^{1.5}kH\log^2 n\log\log n)$ time. Therefore, the time complexity of the DAC algorithm is $O(n^{1.5}kH\log^2 n\log\log n) = O(n^{1.5+\varepsilon}kH)$ with high probability. □

Chapter 4

ASTRAL-Pro: Quartet-based Species Tree Inference Despite Paralogy

Phylogenetic inference from genome-wide data (phylogenomics) has revolutionized the study of evolution because it enables accounting for discordance among evolutionary histories across the genome. To this end, summary methods have been developed to allow accurate and scalable inference of species trees from gene trees. However, most of these methods, including the widely-used ASTRAL, can only handle single-copy gene trees and do not attempt to model gene duplication and gene loss. As a result, most phylogenomic studies have focused on single-copy genes and have discarded large parts of the data. Here, we first propose a measure of quartet similarity between single-copy and multi-copy trees that accounts for orthology and paralogy. We then introduce a method called ASTRAL-Pro (ASTRAL for PaRalogs and Orthologs) to find the species tree that optimizes our quartet similarity measure using dynamic programming. By studying its performance on an extensive collection of simulated datasets and on real datasets, we show that ASTRAL-Pro is more accurate than alternative methods.

4.1 Introduction

The evolutionary history of a gene can differ from that of the species containing the gene for several reasons (Maddison, 1997), including incomplete lineage sorting (ILS), duplication and loss (duploss for short), gene transfer, hybridization. Species tree inference is a central question in evolutionary biology and dealing with these sources of discordance is crucial. Many approaches have been proposed for species tree inference, including gene trees-species tree co-estimation (Heled and Drummond, 2010; Boussau et al., 2013; Szöllõsi et al., 2014; Liu, 2008; An et al., 2013) and species tree inference from sequence data (Chifman and Kubatko, 2014; Bryant et al., 2012; De Maio et al., 2013). However, the most scalable approach has remained a two-step process: first infer gene trees independently of sequence data and then combine them using summary methods. The goal of a summary method is to find the species tree best explaining the gene trees according to a model of gene tree discordance. While the ultimate goal is to develop summary methods modelling all sources of discordance, the literature

has mostly focused on separate causes.

A major family of summary methods focuses on duplication and loss processes producing multi-copy gene trees (Wehe et al., 2008; Chaudhary et al., 2010; Bansal et al., 2010; Ma et al., 2000; Hallett and Lagergren, 2000; Bayzid et al., 2013). Most of these summary methods rely on maximum parsimony reconciliation (Goodman et al., 1979) and aim at finding the species tree with the minimum reconciliation cost. Example methods include DupTree (Wehe et al., 2008), its later extension iGTP (Chaudhary et al., 2010; Bansal et al., 2010), DynaDup (Bayzid et al., 2013) and earlier similar dynamic programming algorithms (Hallett and Lagergren, 2000). Other methods take a more agnostic approach and minimize the distance between species trees and the gene trees without necessarily invoking specific reasons for discordance. Example methods of this type include MulRF (Chaudhary et al., 2013) and *guenomu* (De Oliveira Martins et al., 2016). A recent result asserts that the optimal solution to the optimization problem solved by MulRF is indeed a statistically consistent estimate of the species tree under a generic duplication-only model of gene evolution (Molloy and Warnow, 2019). These methods are mostly designed to handle duplication and loss, and although in simulations some have reasonable accuracy under ILS and gene transfer (Chaudhary et al., 2015), they have not been widely adopted by biologists.

Several summary methods target ILS as modelled by the multi-species coalescence (MSC) model (Pamilo and Nei, 1988; Rannala and Yang, 2003), and many of them are statistically consistent (e.g., Liu et al., 2009; Larget et al., 2010; Mossel and Roch, 2010; Liu et al., 2010; Wu, 2012; Sayyari and Mirarab, 2016a; Liu and Yu, 2011; Vachaspati and Warnow, 2015). The most successful summary method for ILS has arguably been ASTRAL (Mirarab et al., 2014), which, due to its high accuracy (Giarla and Esselstyn, 2015; Molloy and Warnow, 2018; Ballesteros and Sharma, 2019) and scalability (Mirarab and Warnow, 2015; Yin et al., 2019), has been used to perform species-tree inference in numerous studies. ASTRAL, like several other methods (e.g., Chifman and Kubatko, 2014; Sayyari and Mirarab, 2016a; Larget et al., 2010), relies on dividing gene trees into unrooted four-taxon trees (called quartets), a feature that allows it to address ILS and may contribute to its high accuracy. ASTRAL, however, was designed to

handle single-copy gene trees reconstructed from sets of orthologous genes. This limitation has restrained its application scope. As an example, two studies on plant transcriptomes had to discard thousands of available multi-copy genes (Wickett et al., 2014; Leebens-Mack et al., 2019) and only use the 400–800 single-copy gene trees. A recent result by Legried et al. (2020) asserts that treating gene copies as alleles of a same gene, a feature ASTRAL supports (Rabiee et al., 2019), is a valid method under a parametric model of gene duplication and loss and *may* lead to accurate results. Du et al. (2019) have shown that random sampling of leaves works well empirically and Markin and Eulenstein (2020) have shown that method to be consistent under a model combining ILS and duplication and loss. Beyond ASTRAL, several methods have focused on dividing multi-copy gene trees into single-copy genes without apparent duplications (e.g., Marcet-Houben and Gabaldón, 2011; Scornavacca et al., 2011; Yang and Smith, 2014; Dunn et al., 2013; Ballesteros and Hormiga, 2016). However, to our knowledge, no quartet-based methods *designed* to handle duplication and loss currently exist. Extending quartet-based methods to multi-copy gene trees while modeling orthology and paralogy is difficult.

We introduce the quartet-based species tree inference method ASTRAL for PaRalogs and Orthologs (ASTRAL-Pro). Given a set of multi-copy gene family trees, ASTRAL-Pro seeks to compute a single-copy tree (the species tree) maximizing the total similarity to the input gene trees. To define the similarity, we introduce a new measure of quartet similarity between single-copy and multi-copy trees accounting for orthology and paralogy. Tests on an extensive set of simulated and real datasets provide evidence of ASTRAL-Pro's robustness and accuracy.

4.2 Results

We start by informally introducing the methodology underlying ASTRAL-Pro, leaving the formal definition and proofs to the Methods section. We will then compare the performances of ASTRAL-Pro to leading alternative methods on simulated and real datasets.

4.2.1 ASTRAL-Pro Algorithm

Per-locus quartet similarity.

ASTRAL-Pro maximizes a measure of quartet similarity between a multi-copy and a single-copy tree. Let us consider a rooted gene family tree where multiple leaves can have the same label (i.e., the species identifier). We need a principled way to compare this tree to a species tree where each species identifier appears once. The measure we define is based on several observations.

(i) Internal nodes of the gene tree correspond to either duplication or speciation events; thus, we can *tag* nodes of the tree as speciation or duplication (Def. 4.1). While the true tagging is unknown, as we will see, it can be partially inferred (Fig. 4.1). (ii) Each quartet of leaves in the gene tree defines two *anchor* nodes, and we refer to the Least Common Ancestor (LCA) of the two anchors as the *anchor LCA* (Fig. 4.1). In a correctly tagged tree, a quartet has information about the speciation events only if it includes four distinct species and if the LCA of any three out of four leaves of the quartet is a speciation node (Fig. 4.1). Thus, to define our measure of quartet similarity, we only include these speciation-driven quartets (SQ) and ignore the rest (Def. 4.2). (iii) All the SQs on the same four species that share the same anchor LCA must also share the same topology (Proposition 4.1). Thus, once we know the topology of one of them, the others do not provide new information. We call these SQs *equivalent* (Def. 4.4); in our quartet measure, we count them as one unit, and we consider them as part of the same quartet equivalence class. Moreover, we show that, for all equivalent quartets, the gene copies present at the current time all share the same ancestral locus at the time of the speciation event corresponding to the anchor LCA (Proposition 4.2). See Methods for formal statements.

Based on these observations, we define the per-locus quartet score of a species tree S with respect to a gene family tree G with tagged internal nodes to be the number of quartet equivalence classes of G agreeing with S (Def. 4.5). We then define the Maximum per-Locus Quartet-score Species Tree (MLQST) for a set $\mathscr{G}$ of gene trees as the tree that has the maximum

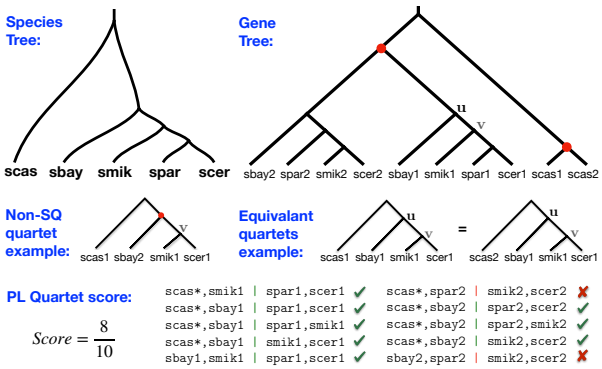

Figure 4.1. Per-locus quartet score. Example gene family tree from the fungi dataset (Butler et al., 2009) restricted to 5 species and a potential species tree. Two nodes of the gene tree are tagged as duplication (red dots) and others as speciation. Quartet `sbay1, smik1 | spar1, scer1` is anchored by nodes u and v, where u is the anchor LCA. Because the LCAs of any three leaves (u or v) are speciation nodes, this quartet is a speciation-driven quartet (SQ). Quartet `scas1, sbay2 | smik1, scer1` is anchored by node v and a duplication (red dot). Since the duplication node is the LCA of three leaves, this quartet is a non-speciation-driven quartet (non-SQ) that does not count toward the per-Locus (PL) Quartet score. Note u is the anchor LCA of both `scas1, sbay1 | smik1, scer1` and `scas2, sbay1 | smik1, scer1`; thus, they form the equivalence class `scas*, sbay1 | smik1, scer1`. In this example, there are 10 equivalence classes of SQ quartets, eight of which match the species tree; thus, the PL quartet similarity is 8. The goal of ASTRAL-Pro is to find the species tree that maximizes this score.

total per-locus quartet score with respect to $\mathscr{G}$ (Def. 4.6).

ASTRAL-Pro

As formalized in Theorem 4.1 in Methods, our new method is based on an efficient dynamic programming algorithm to find the MLQST tree. The ASTRAL-Pro algorithm, like ASTRAL, solves this problem restricted to a large search space X, defined heuristically using Algorithm S4.1. The running time of ASTRAL-Pro grows polynomially with the number of species, the number of genes, and the size of X (Claim 4.3). Finally, note that the per-locus quartet score is only defined for rooted and tagged gene trees. Since, in practice, gene trees are often unrooted and untagged, we also provide Algorithm 4.1 to tag and root gene trees using the parsimony principle.

Statistical consistency and local support.

In the presence of gene duplication and losses only, under the birth-death model called GDL proposed by Arvestad et al. (2009), Theorem 4.2 (Methods) states that ASTRAL-Pro is statistically consistent given correctly tagged and rooted error-free gene trees, even with partially correct rooting (see Claim 4.1). Under the MSC model and in the absence of gene duplication and gene loss, gene trees are single-copy. For single-copy gene trees, ASTRAL-Pro solves the same problem as ASTRAL and thus, like ASTRAL, it is a statistically consistent estimator of the species tree under the MSC model given a random sample of error-free gene trees (Mirarab et al., 2014). However, we do not currently have a proof of consistency of ASTRAL-Pro under models that combine GDL and ILS (see Discussions).

With correctly tagged error-free gene trees, differences in SQ topologies from the species tree must be due to processes other than GDL, such as ILS (Proposition 4.3). We use this observation to extend the local posterior probability (localPP) measure of branch support to multi-copy gene trees (Def. 4.8).

4.2.2 Accuracy of ASTRAL-Pro in simulations

We first test ASTRAL-Pro (A-Pro for short) against two leading summary methods: MulRF (Chaudhary et al., 2013) (optimizing an extension of the RF distance, Robinson and Foulds, 1981, to multi-labelled trees) and DupTree (Wehe et al., 2008) (minimizing the duplication reconciliation cost, Maddison, 1997). We also compare A-Pro to ASTRAL-multi (Rabiee et al., 2019), which is the feature of ASTRAL designed for handling multiple alleles (as opposed to multiple copies); although ASTRAL-multi is not designed for multi-copy data, we include it because of recent theoretical results showing that it is consistent under the GDL model (Legried et al., 2020). We compare the methods in terms of the accuracy of the species tree topology that they produce.

In our tests, we use two simulated datasets, one called S25, which is new to this study,

Table 4.1. Simulation settings for S25 dataset with varying parameters. See Table S4.1 for full parameters and Figures S4.1–S4.6 for full statistics.

Condition	Parameter ranges
Default model	$n = 25$; $k = 1000$; $\tau \sim LN(21.25; 0.2)$ $\lambda_+ = 4.9 \times 10^{-10}$; $\lambda_- = \lambda_+$; $N_e = 4.7 \times 10^8$ $C \approx 5$; ILS $\approx 70\%$ MGTE = 15% (500bp) or 36% (100bp)
Varying λ_+, λ_- (duploss rate)	$\lambda_+ \in \{4.9, 2.7, 1.9, 0.52, 0\} \times 10^{-10}$ $\lambda_- \in \{1, 0.5, 0.1, 0\} \times \lambda_+$; $C \approx \{5, 2, 1, 0.2, 0\}$
Varying λ_+, N_e (dup rate, ILS)	$\lambda_+ \in \{4.9, 1.9, 0\} \times 10^{-10}$; $N_e \in \{4.7, 1.9, 0.48, 0.0001\} \times 10^8$ ILS $\approx \{70, 52, 20, 0\}\%$; $C \approx \{5, 1, 0\}$ MGTE $\approx \{15, 15, 15, 16\}\%$ (500bp) or $\{36, 36, 36, 35\}\%$ (100bp) as N_e changes
Varying n	$n \in \{10, 25, 100, 250, 500\}$ MGTE $\approx \{15, 15, 17, 18, 18\}\%$ (500bp) or $\{34, 36, 40, 43, 43\}\%$ (100bp)
Varying k	$k \in \{25, 100, 250, 1000, 2500, 10000\}$

n: number of ingroup species. k: number of genes. τ: tree height in generations. λ_+: duplication rate. λ_-: loss rate. N_e: Haploid effective population size. We estimated the following empirically. C: mean number of copies per species minus one when $\lambda_- = 0$ and $n = 25$. ILS: mean RF distance between true gene trees and the species tree when $\lambda_+ = 0$. MGTE: mean RF distance between true and estimated gene tree when $\lambda_+ = 0$.

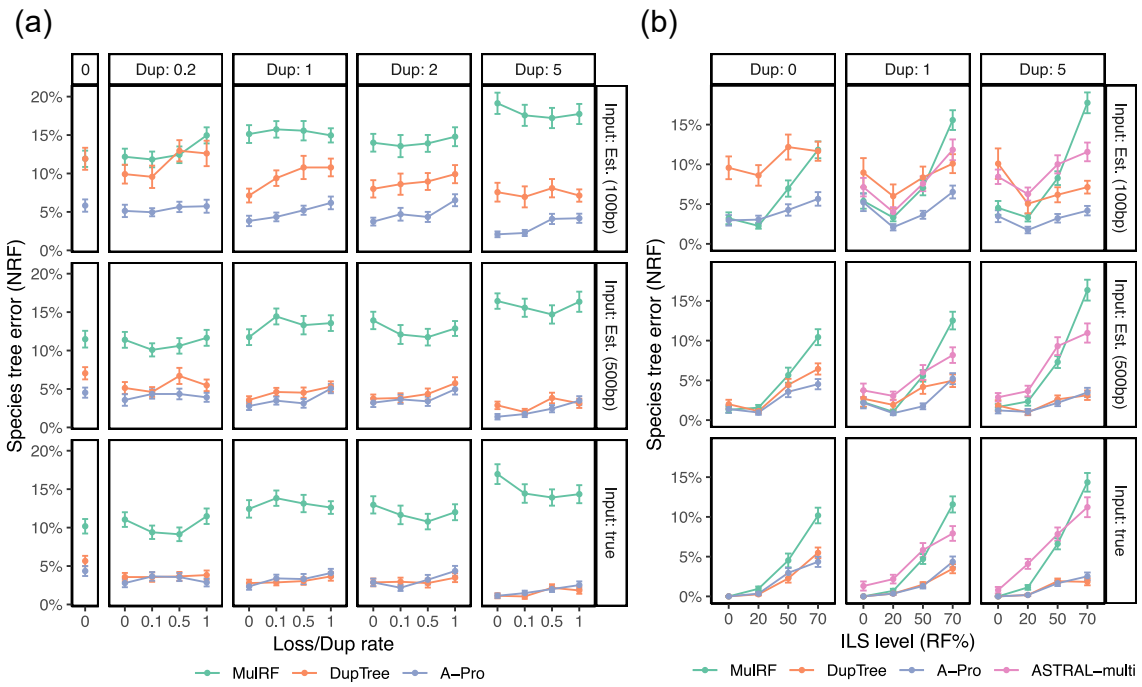

Figure 4.2. Species tree error on the S25 dataset for $n = 25$ in-group species, $k = 1000$ gene trees, and both true and estimated gene trees from 100bp and 500bp alignments. (a) Controlling duplication rate (box columns; labelled by C) and the loss rate (x-axis; ratio of the loss rate to duplication rate). (b) Controlling the duplication rate (columns; labelled by C) and the ILS level (x-axis; NRF between true gene trees and the species tree for $\lambda_+ = 0$). A-Pro and ASTRAL-multi are identical with $\lambda_+ = 0$. See Table 4.1 for parameters and Fig. S4.7 for iGPT-duploss.

and one called S100 from Molloy and Warnow (2019), which is based on a real fungal dataset (Butler et al., 2009; Rasmussen and Kellis, 2012). Both datasets were created by (1) simulating true species and true gene trees under the DLCoal model, which is a unified model of ILS and gene duplication and loss (Rasmussen and Kellis, 2012), (2) simulating a sequence alignment from each true gene family trees, and (3) estimating a gene tree from each gene alignment. In S25, we varied parameters that control the rate of duplication (λ_+), the rate of loss (λ_-), the ILS level, the number of species (n), and the number of genes (k) (Table 4.1). We also varied alignment length, which effectively varied the level of gene tree estimation error. The S100 dataset also varies all these parameters, except n. Thus, we simulate effects of ILS, duplication and loss, and gene tree estimation error. See Methods for details.

S25 dataset

Controlling duplication and loss rates.

We begin by describing the results of experiments that vary the duplication and loss rates (λ_+, λ_-) (Fig. 4.2a). On true gene trees, A-Pro and DupTree are essentially tied in terms of accuracy, except for the case with no duplication and loss where A-Pro is slightly more accurate. Overall, the accuracy of A-Pro and DupTree is statistically indistinguishable under these conditions (p-value = 0.79 according to a multi-variate ANOVA test). Increasing λ_+ *reduces* error ($p < 10^{-5}$), perhaps because additional copies provide more information, akin to increasing the number of loci. Despite statistically significant increases (p = 0.006) in error as λ_- increases, both methods are quite robust to loss rates, losing at most 1.5% accuracy on average when $\lambda_- = \lambda_+$ compared to no losses. MulRF has much higher error than other two methods, with errors that range between 10% and 17% across model conditions (we remind the reader that all these conditions exhibit high ILS, a process that MulRF ignores).

On estimated gene trees, the pattern changes, and the error of DupTree increases dramatically while A-Pro remains relatively accurate. When $\lambda_+ = \lambda_- = 0$, DupTree has on average an 11.5% error, whereas A-Pro has only a 4.5% error for 500bp. Adding duplications helps both methods, but A-Pro remains more accurate. For example, with 100bp input gene trees (i.e., high estimation error), DupTree has an error between 50% to 260% higher than A-Pro. With 500bp input (i.e., low-error gene trees), differences are statistically significant ($p < 10^{-5}$) but are more modest in magnitude (across conditions, DupTree has a median of 28% more error). The relative accuracy of A-Pro and DupTree is not a function of λ_- (p =0.8) but may depend on λ_+ (p =0.05).

In terms of running time, on the default model condition, we observe that A-Pro is the fastest method, taking less than a minute on this dataset, followed closely by DupTree (Fig. S4.8).

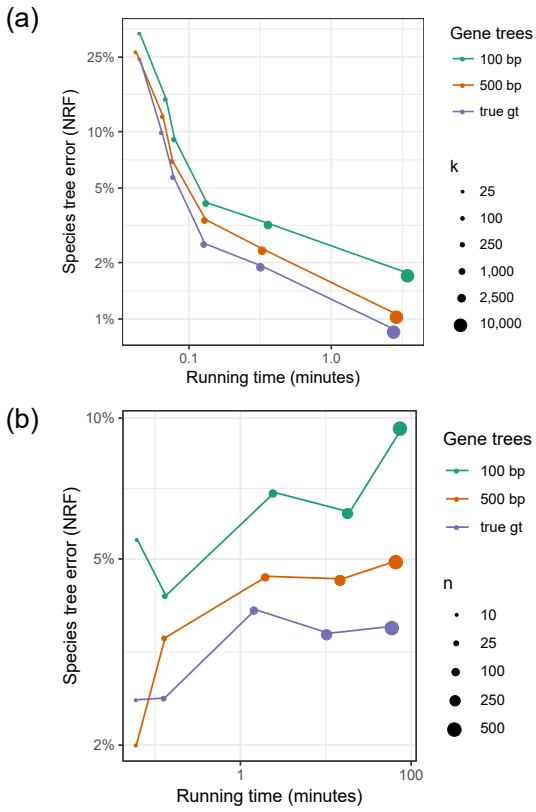

Figure 4.3. Accuracy (y-axis) and running time (x-axis) of A-Pro as the number of genes k (a) or the number of species n (b) changes. Both axis are in log-scale. As k increases, accuracy increases. See also Figure S4.9.

Controlling the level of ILS.

As we change the ILS level (Table 4.1), the reason for the poor performance of MulRF becomes clear (Fig. 4.2b). Without ILS, MulRF has excellent accuracy, often matching A-Pro and beating DupTree on low-error gene trees. As the ILS level increases (especially above 20%), the accuracy of MulRF deteriorates quickly. Overall, ILS has the strongest effect on accuracy ($p \ll 10^{-5}$) but its impact on methods varies ($p \ll 10^{-5}$). DupTree seems as tolerant of ILS as A-Pro, despite the fact that DupTree is not designed specifically for ILS, and both methods are much more tolerant of ILS than MulRF. Nevertheless, once again, DupTree shows extreme sensitivity to gene tree error.

To summarize, DupTree is relatively tolerant of ILS but less tolerant of gene tree error; MulRF is tolerant of gene tree error but not of ILS; A-Pro is quite robust to both.

Controlling the number of genes and species.

Increasing the number of genes k in the most difficult model condition (i.e., high λ_+, λ_-, and ILS) results in continued improvement in accuracy for A-Pro for every value we tested up to $k = 10^4$ (Fig. 4.3a). With true gene trees, the error reduces from 26% with $k = 25$ to below 1% with $k = 10^4$. Even with less accurate gene trees, the error reduces to below 2% with increased numbers of genes. Increasing k increases running time, which empirically grows proportionally with $k^{1.4}$ (Fig. S4.9a). Nevertheless, using 28 cores, the running time was never more than 3.5 minutes even with $k = 10^4$.

Increasing n from 25 to 500 shows that A-Pro is relatively robust to a large number of species (Fig. 4.3b). With true gene trees, the error ranges between 2.5% with 10 species to 3.5% with 500 species. With estimated gene trees, error ranges between 4.1% to 9.5% (for 100bp) and between 2% and 5% (for 500bp). Note that as n increases, the gene tree error also increases (Table 4.1; Fig S4.6). The running time of A-Pro increases roughly quadratically with n (Fig. S4.9b) but is below 2 hours (given 28 cores) even for $n = 500$ ($k = 1000$).

4.2.3 S100 dataset

Patterns of performance on the S100 dataset are consistent with the S25 dataset (Fig. 4.4). DupTree is highly accurate with true gene trees and gene trees with low estimation error but quickly degrades in accuracy as gene tree error increases. MulRF is less sensitive to gene tree error but is more sensitive to the ILS level (which is always moderate or low on this dataset). As in S25, here, we see that using ASTRAL-multi to handle duplication and loss is not beneficial.

A-Pro works the best overall, ranking first in terms of mean error (rounded to two significant digits) in 105 out of 120 test conditions and ranking second in 14 of the 15 remaining cases (Table S4.2). Many of the conditions where A-Pro is ranked second are among those with true gene trees where DupTree works great. The second-best method overall is MulRF, which is not surprising given the low ILS levels in this dataset. As expected, all methods are helped with increased numbers of genes; however, even with 500 genes, differences in accuracy remain,

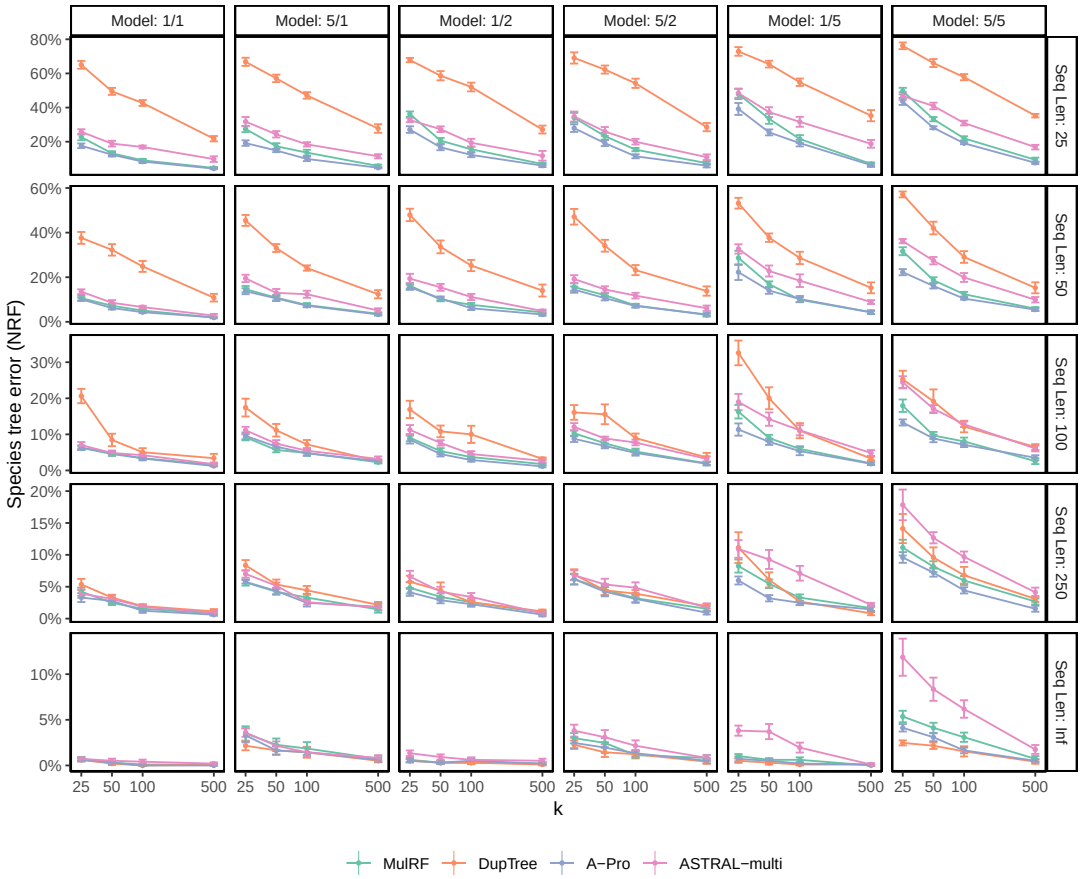

Figure 4.4. Species tree error on S100 dataset. We compare the species tree error of the four methods, showing mean and standard error over 10 replicates for each model condition, with varying numbers of genes (k) and sequence lengths (with Inf signifying true gene trees). Model conditions are labeled as a/b where a is the level of ILS (1 or 5) and b is the duplication/loss rate (1, 2, or 5).

especially with shorter gene sequences.

4.2.4 Accuracy on biological datasets

Plant (1kp) dataset

We reanalyze the transcriptome dataset of 103 plant species, which was previously analyzed by Wickett et al. (2014) using 424 single-copy gene trees using ASTRAL. The original study had also inferred 9683 multi-copy gene trees with up to 2395 leaves for 80 of the 103 species and three other genomes (a total of 83). However, due to a lack of suitable species tree

methods, these gene trees were left unused (Methods). Here, we analyze all 9683 multi-copy gene trees using A-Pro.

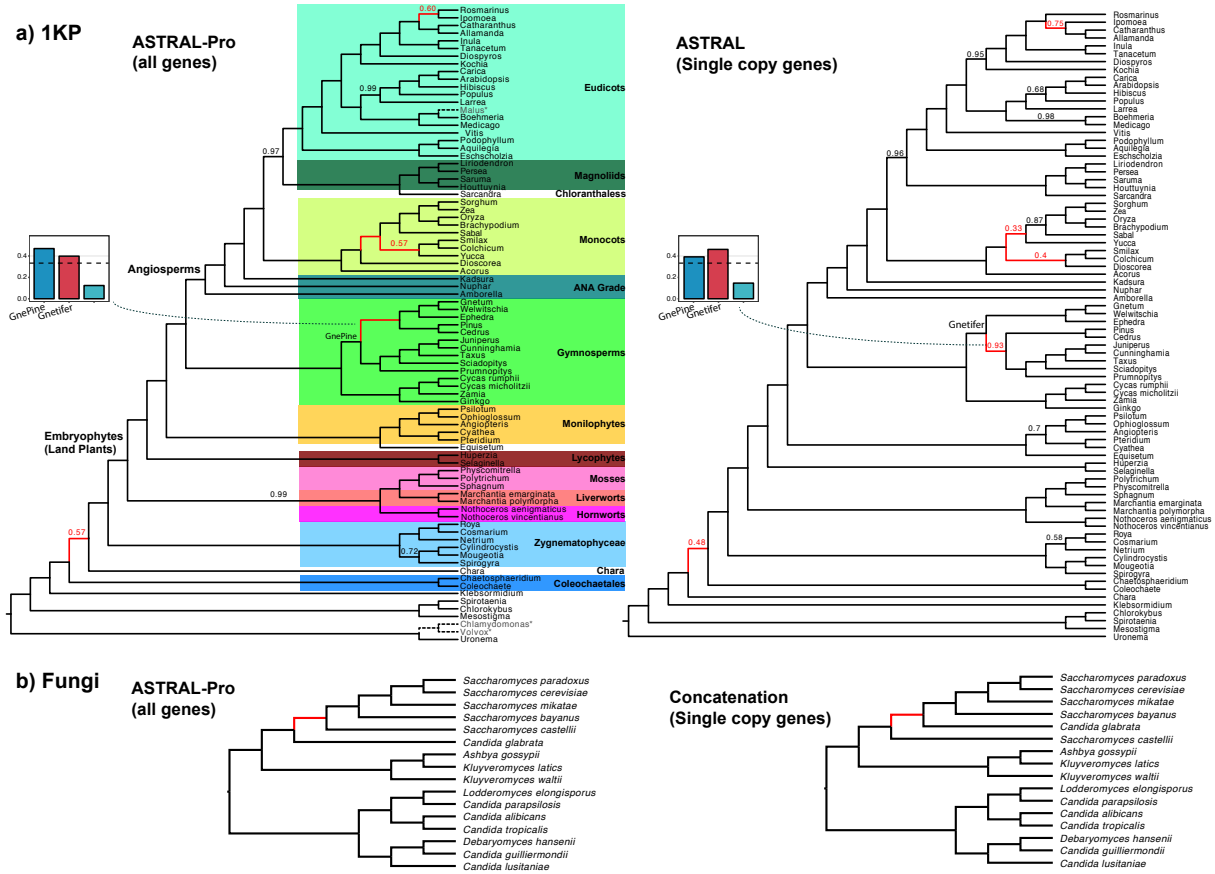

Figure 4.5. Biological dataset. (a) Plant dataset (1kp). Right: ASTRAL on 424 single-copy gene trees. Left: ASTRAL-Pro on 9683 multi-copy gene trees. Three genomes (noted by * and dashed lines) were present in multi-copy dataset but not in the single-copy data. The single-copy tree includes 23 species that were not in the multi-copy data and are pruned from the species tree (localPP support is recomputed using gene trees pruned to the 80 common species). Five branches (red) differ between the two trees. LocalPP support shown except when equal to 1. For the main highly supported conflict (Gnetifer vs Gnepine), we show quartet support of alternative topologies among single-copy gene trees using DiscoVista (Sayyari et al., 2018). (b) Fungi dataset. Right: Concatenation of 706 single-copy gene trees with the red branch enforced as a constraint(Butler et al., 2009). Left: ASTRAL-Pro on 7280 multi-copy gene trees.

A-Pro on multi-copy gene trees returns a species tree (Fig. 4.5a) similar to the single-copy ASTRAL tree reported by the original study but with five differences. In contrast, DupTree differs from the ASTRAL tree in 33 out of 77 branches (21/77 for iGTP-DupLoss) and violates

many known biological relationships (Fig. S4.10). A-Pro has higher localPP than ASTRAL (e.g., four versus eight branches with localPP below 0.95). The A-Pro tree is consistent with ASTRAL for major groups, including placing Zygnematales (not Chara) as sister to all land plants, the placement of Amborella as sister to the rest of angiosperms, and the monophyly of Bryophytes (liverworts, mosses, and hornworts). Some of these consistencies with ASTRAL (e.g., monophyly of Bryophytes) are in contrast to the concatenation analyses of single-copy genes, as reported by Wickett et al. (2014).

Changes between the ASTRAL and A-Pro trees mostly have low support. In A-Pro, unlike ASTRAL, Rosmarinus and Ipomoea are grouped together (albeit, with 0.6 localPP support), which is likely the correct result as these species are in the same order (Lamiales). The ASTRAL tree has only 0.75 localPP for dividing this order. The position of genus Yucca has low support in the ASTRAL tree and has changed in the A-Pro tree. Interestingly, a recent update to this transcriptome analysis using 1124 species (Leebens-Mack et al., 2019) (which samples close genera Asparagales and Liliales) finds Yucca in a position identical to A-Pro. Another change is the relative position of Coleochaetale and Chara which has low localPP in both trees. Most consequentially, the main highly supported change is that A-Pro, unlike ASTRAL, recovers the GnePine hypothesis (i.e., combining Gnetales and Pinaceae) with 1.0 localPP. This hypothesis is supported by several studies (Burleigh and Mathews, 2004; Zhong et al., 2010, 2011; Laurin-Lemay et al., 2012) and all concatenation analyses from Wickett et al. (2014). Examining quartet frequencies for single-copy gene trees around this branch, we see that the second and third most frequent quartets do not match (Fig. 4.5a) and are skewed towards GnePine; this pattern is not consistent with ILS as the main source of discordance, and may suggest other processes such as hybridization. However, multi-copy gene trees also show a similar pattern, with support for GnePine and Gnetifer swapped.

Fungal dataset

We reanalyze a dataset of 16 yeast species with 7,280 multi-copy gene families available from Butler et al. (2009). To obtain the species tree, the original study used only 706 one-to-one orthologs with concatenation and did not use multi-copy gene trees in species tree inference (Methods). We used all amino acid multi-copy gene families as input to A-Pro.

The A-Pro species tree has 1.0 localPP everywhere and matches the published species tree except for one branch (Fig. 4.5b). The position of *Saccharomyces castellii* as sister to *Candida glabrata* and the Saccharomyces group in the original study was enforced by a constraint in the ML search because the unconstrained analyses did not recover the relationship the authors expected. This enforced constraint was justified based on genome rearrangement and syntenic conservation, but was not recovered in the concatenation analyses. In the A-Pro tree, *Candida glabrata* is at the base of this clade, matching the unconstrained concatenation analysis. Salichos and Rokas (2013) also recovered the same topology as A-Pro and used this branch as an example of relationships that challenge phylogenomics. While gene synteny evidence suggests that A-Pro may be finding the wrong resolution, it is worth highlighting that it matches trees inferred using substitution models.

4.3 Discussions

We introduced A-Pro, a summary method for combining multi-copy gene trees. By allowing the use of multi-copy gene trees, A-Pro enables a manyfold increase in the number of genes used in phylogenomic analyses. Note that neither concatenation nor ASTRAL (the dominant methods used by practitioners) are able to use multi-copy genes. The main set of methods available for multi-copy analyses are the co-estimation methods (e.g., Szollosi et al., 2012; Szöllõsi et al., 2013; Boussau et al., 2013). However, these methods, while accurate, are inherently less scalable than summary methods. A-Pro provides a scalable yet accurate alternative to these co-estimation methods.

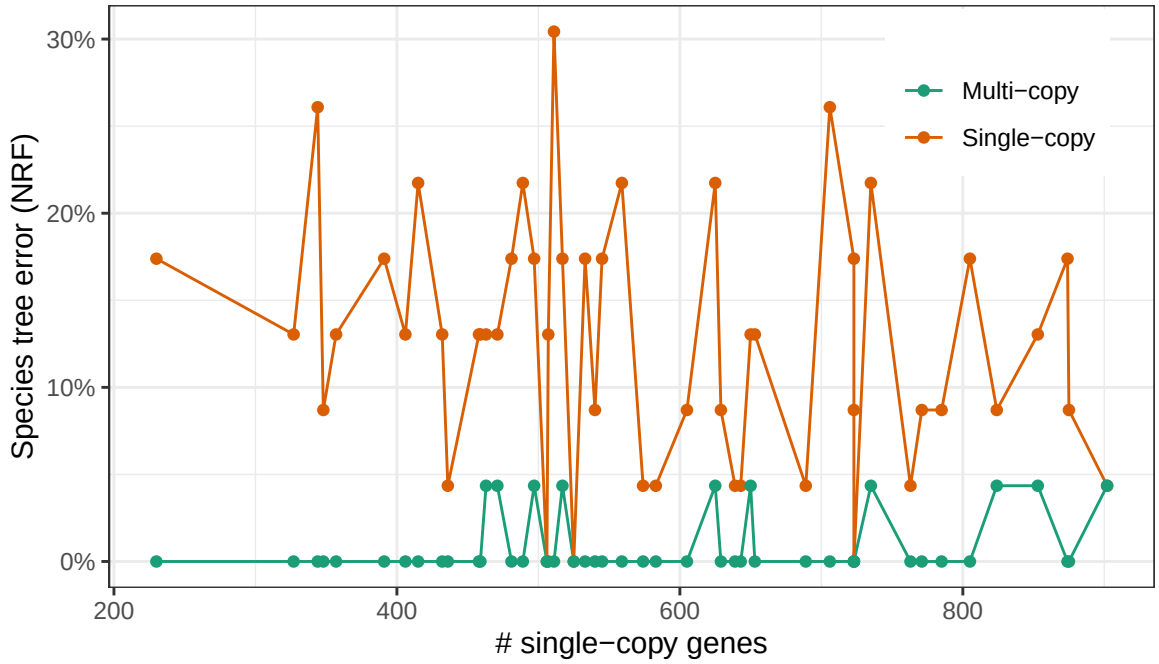

Figure 4.6. Accuracy of the estimated species tree (y-axis) versus the number of single-copy genes (x-axis) across all 50 replicates of the S25 dataset with $k = 10,000$ gene trees (from the experiment varying k). The "Multi-copy" line, representing A-Pro, is using all gene trees while the "Single-copy" line, representing ASTRAL, is only using the single-copy gene trees.

As an example for testing the advantage of using all multi-copy gene trees, we revisit the simulated S25 dataset with $k = 10^4$ multi-copy gene trees. Among the 10^4 gene trees, we have between 200 and 900 single-copy gene trees across our 50 replicates (the variation is due to stochastic differences). An alternative to using ASTRAL-Pro is to use normal ASTRAL on single-copy gene trees. Comparing ASTRAL on single-copy gene trees and ASTRAL-Pro on all 10^4 multi-copy gene trees shows a great loss of accuracy as a result of the filtering (Fig. 4.6). Our simple filtering strategy, keeping all single-copy gene trees, does not consider orthology, but is not dramatically different from the approach used by many (e.g., Wickett et al., 2014; Leebens-Mack et al., 2019). Despite the potential for paralogy in single-copy genes, the example shows the negative impact of gene filtering. This observation is consistent with prior results that have established a close link between the accuracy of summary methods and the number of input genes both in practice (for an overview, see Mirarab, 2019) and in theory (Shekhar et al., 2018).

A-Pro is based on a per-locus quartet-based measure of similarity between multi-copy gene trees and a species tree. The measure relies on internal nodes of gene trees being tagged as speciation or duplication. Somewhat counter-intuitively, despite being a quartet measure, it needs *partially* rooted trees (Claim 4.1). The measure defines an equivalence relationship on quartets and counts each equivalence class only once, avoiding double-counting quartets that are bound to have identical topologies. Avoiding double-counting is at the heart of the approach and likely is a main reason behind its high accuracy on the simulated and empirical data we tested.

Quartet-based methods for handling multi-copy gene trees are not abundant. Besides our method, one can attempt to sample single-copy gene trees, an approach that shows promise but fails to model orthology/paralogy (Du et al., 2019). Legried et al. (2020) recently provided theoretical and empirical evidence that simply treating gene copies as alleles may be sufficient. We showed that this alternative, although attractive in theory, is less accurate and less scalable than A-Pro. We are unaware of other quartet-based species tree inference methods for multi-copy input. Nevertheless, our approach is not the only one that can be imagined and future work should explore other quartet metrics.

To get rooted and tagged gene trees, we used the maximum parsimony principle, with duplication and loss each penalized equally and deep coalescence not penalized at all (Methods). The algorithm we use is not guaranteed to find the correct tags or the root under complex scenarios involving gene duplication and subsequent losses. Thus, the consistency results under the GDL model should be interpreted with this caveat in mind. A-Pro may be statistically consistent even when gene trees are imperfectly rooted and tagged, but we leave this to be determined in future work. Furthermore, there is a large literature on various ways of tagging and rooting gene trees (e.g., Bansal et al., 2013; Durand et al., 2006; Jacox et al., 2016), including other penalties for the duplication and loss events (e.g., there is a suggestion of losses having half the penalty of duplications, David and Alm, 2011). It may also be possible to improve tagging of gene trees using probabilistic orthology inference (Arvestad et al., 2004; Sennblad and Lagergren, 2009) or using synteny information (Bourque et al., 2005; Chauve et al., 2013).

However, these methods often require a species tree. It may be possible to use A-pro in an iterative fashion, where the species tree is inferred, gene trees are re-tagged and re-rooted, and a new species tree is inferred. Future work should explore these approaches.

A-Pro, like other summary methods, depends on accurate input trees. While A-Pro is more robust to gene tree error than alternatives, combining it with co-estimation (Boussau et al., 2013) or gene tree correction (Wu et al., 2013; Lafond et al., 2013, 2014; Scornavacca et al., 2015; Noutahi et al., 2016; El-Mabrouk and Noutahi, 2019) may further improve its accuracy. Future work should also explore extending A-Pro to multifurcating input gene trees because contracting low support branches may help deal with gene tree error (Zhang et al., 2018).

ASTRAL-Pro, which maximizes the per-locus quartet score, is statistically consistent under the MSC model (when given single-copy gene trees as input) and under a GDL model (when given multi-copy gene trees as input). This makes one hope that it may also be consistent under both causes of discordance combined. The DLCoal model (Rasmussen and Kellis, 2012) accounts for ILS, duplication, and loss. Under this model, each duplication immediately creates a daughter locus, which is unlinked from the parent locus; the duplication event gets fixed in all species. Gene trees are seen as generated by first producing a locus tree via a birth-death process that runs on the species tree and then running a MSC process on the locus tree. Because the loci are considered as unlinked, the coalescence processes occur independently between the parent and daughter loci (but the daughter MSC process is "bounded" at the time of duplication). Interestingly, a new paper has suggested that simply selecting one copy of each gene at random and feeding the resulting gene trees to ASTRAL would be consistent under the DLCoal model (Markin and Eulenstein, 2020). Due to the independence of loci, dividing a multi-copy gene family into its constituent loci can give us distributions on gene tree topologies that behave similarly (though not identically) to the MSC model. The per-locus metric *seeks* to count quartet topologies across loci as they existed at the time of speciation events relevant to a quartet (i.e., at the time of the anchor LCA). When successful, it counts only topologies that are drawn from independent coalescent processes. However, complicated scenarios involving a combination of

duplications, losses and ILS can lead to incorrectly tagged gene trees. These scenarios create complications for theoretical proofs. While our simulations were performed under the DLCoal model, we leave it to the future to study whether ASTRAL-Pro is statistically consistent under the DLCoal model.

Our simulations, which all followed the DLCoal model, do not consider some relevant biological scenarios. Examples include whole genome duplication (WGD) events, interlocus gene conversion, and hemiplasy of duplication and loss events (Li et al., 2020). Since ASTRAL-Pro is non-parametric (i.e., does not assume rates of duplication), we predict that WGD events do not impose a major obstacle. The impact of interlocus gene conversion is much harder to predict and needs careful testing. Future work should study ASTRAL-Pro under these more complex scenarios of duplication and loss.

4.4 Methods

4.4.1 The algorithm

Proofs of all propositions, lemmas, and claims can be found in supplementary material (Proofs).

Notations and definitions

Let $\mathscr{S}$ be a set of n species. Let us suppose that we are given a set of binary gene trees $\mathscr{G}$, and, for each tree $G \in \mathscr{G}$ with leaf set $\mathscr{L}_G = \{1 \dots m_G\}$, we have a mapping $\alpha_G : \mathscr{L}_G \to \mathscr{S}$ specifying in which species each gene is sampled. For a rooted tree G, we denote the set of internal nodes in G by $I(G)$, and, for each $u \in I(G)$, we define $\mathscr{L}_G(u)$ as the set of leaves below u. We define two short-hands: $\alpha_G(A) = \{\alpha_G(i) : i \in A\}$ for $A \subset \mathscr{L}_G$ and $\alpha_G(u) = \alpha_G(\mathscr{L}_G(u))$ for a node u (i.e.; all species labels corresponding to a set A of gene tree leaves and all species labels under a gene tree node u, respectively). The notation $G \restriction A$ denotes G restricted to the set A.

We let $\Omega(G)$ be the multi-labelled tree obtained by replacing each leaf $l \in \mathscr{L}_G$ with

$\alpha_G(l)$. Multiple copies of the same species in a gene tree G may be created by gene duplication. Note that we ignore other processes such as transfers, gene conversion, and hybridization. We assume that each duplication creates a new genomic locus (i.e., a position along the genome) and therefore, each locus, except the original one, has a parent locus (which may or may not have survived to the present day). Thus, each element of $\mathscr{L}_G$ can be theoretically mapped to its parent locus, allowing us to "trace" the locus of each leaf to its ancestors.

In each gene tree G, we refer to a subset Q of four distinct elements of $\mathscr{L}_G$ as a quartet. The subtree of a fully resolved tree G restricted to a quartet Q exhibits two degree-three nodes. We refer to these nodes as *anchors of Q on G*. As shown in Fig. 4.7, for a rooted tree G and for a quartet Q, up to label permutations, $G \restriction Q$ can only have two topologies: an *unbalanced* one (when one anchor descends from the other), denoted as $Q \angle G$, and a *balanced* one (otherwise), denoted as $Q \perp G$. We say a tripartition (P_1, P_2, P_3) of $\mathscr{S}$ "can anchor" a quartet Q of G iff $\forall_i : P_i \cap \alpha_G(Q) \neq \emptyset$.

Definition 4.1 (Tagged trees)**.** We say that a rooted tree G is tagged if every internal node is tagged either as duplication or as speciation. A node u with children u_1 and u_2 can be tagged as speciation only if the sets $\alpha_G(u_1)$ and $\alpha_G(u_2)$ are mutually exclusive.

We note that these tags may or may not correspond to real speciation and duplication events. In particular, when loci coalescence before duplication events, a correct tagging corresponding to actual events may not be possible.

Per-locus quartet score

Definition 4.2 (SQ)**.** A quartet Q on a rooted tagged gene tree G is called a speciation-driven quartet (SQ) iff $|\alpha_G(Q)| = 4$ and the LCA of any three out of four leaves of Q is a speciation node. Equivalently, a quartet with topology $ab|cd$ is a SQ if and only if its genes are all contained in different species and the LCA of either a or b with either c or d is tagged as speciation. Let Σ_G denote the set of SQs in G.

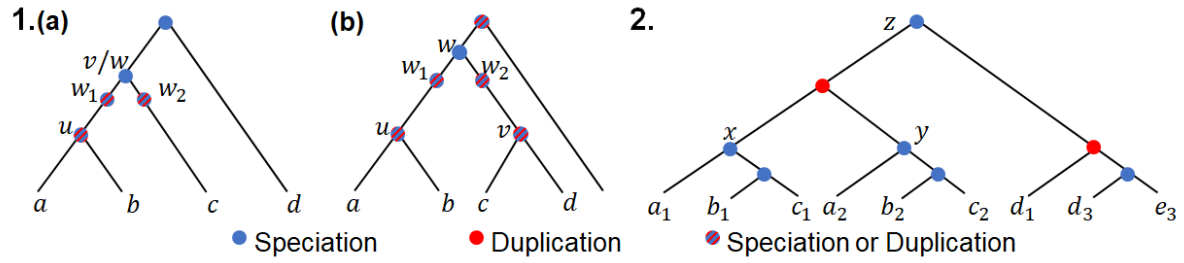

Figure 4.7. 1. An example of a quartet $Q = \{a,b,c,d\}$ with (a) unbalanced topology ($Q \angle G$) and (b) balanced topology ($Q \perp G$). Anchors are u and v, and w is the anchor LCA. While w has to be a speciation for Q to be considered a SQ, u and v (when different from w) can be either speciation or duplication. 2. An example of equivalence classes. Three equivalence classes are anchored on z: all eight quartets of the form $\{a_i,b_j,d_k,e_3\}$, of the form $\{a_i,c_j,d_k,e_3\}$, and of the form $\{b_i,c_j,d_k,e_3\}$, all with balanced topology. Anchored on x: two equivalence classes with unbalanced topology: $\{a_1,b_1,c_1,d_1\} \sim \{a_1,b_1,c_1,d_3\}$ and $\{a_1,b_1,c_1,e_3\}$. Anchored on y: two equivalence classes: $\{a_2,b_2,c_2,d_1\} \sim \{a_2,b_2,c_2,d_3\}$ and $\{a_2,b_2,c_2,e_3\}$.

Definition 4.3 (Quartet anchor LCA)**.** Let u and v be anchors of a quartet Q on a *rooted* tree G. We refer to the LCA of u and v as the *anchor LCA* of Q on G and denote it as $\psi_G(Q)$.

The last definition is central to our approach. Note that anchors of a SQ can be speciations or duplications (Fig. 4.7) and thus SQs are not simply quartets with anchors being speciation nodes. Instead, they are quartets with a topology pre-determined by the speciation event represented by the anchor LCA, regardless of subsequent duplications and losses. Such subsequent duplications and losses may lead to multiple quartets being associated to the same speciation event. Since these events include no new information on the speciation event, we count only SQs towards the quartet score of a species tree and weight them in a non-trivial way to avoid double-counting.

Definition 4.4 (Equivalent SQs)**.** Two SQs on the same 4 species are *equivalent* if they have the same anchor LCA; i.e., for two SQs, $Q_1 \sim Q_2 \iff \alpha_G(Q_1) = \alpha_G(Q_2) \wedge \psi_G(Q_1) = \psi_G(Q_2)$.

Proposition 4.1. *If Q_1 and Q_2 are equivalent SQs on G, then $\Omega(G \restriction Q_1)$ and $\Omega(G \restriction Q_2)$ are isomorphic.*

Thus, equivalent SQs have the same quartet topology when mapped to species. Proposition 4.1 tells us that equivalent SQs do not provide any extra information on the speciation event, and therefore, it is reasonable to count all equivalent SQs as one unit when computing the quartet score of a species tree. This intuition is backed by the following proposition:

Proposition 4.2. *Assuming a correctly rooted tagged tree G, for all equivalent SQs with a shared anchor LCA w, the three (in the unbalanced case) or four (in the balanced case) quartet leaves below w will all share an ancestral locus at the time of the speciation event corresponding to w.*

We can now provide a natural definition of the quartet score. The equivalence relation (Def. 4.4) partitions all quartets in equivalence classes and, by Proposition 4.1, for each equivalence class, we can define a unique quartet tree labelled by $\mathscr{S}$. By Proposition 4.2, each class corresponds to an ancestral locus. We can denote each equivalence class in G as a pair, consisting of the set of species and the anchor node $\big(\alpha_G(Q), \psi_G(Q)\big)$.

Definition 4.5 (Per-locus Quartet Score)**.** The per-locus quartet score of a species tree S with respect to a rooted tagged gene tree G is the number of equivalent quartet classes that match the S topology. More formally, $q(S,G)$ is defined as:

$$\left|\left\{\big(\alpha_G(Q), \psi_G(Q)\big) : Q \in \Sigma_G, \Omega(G \restriction Q) \simeq S \restriction \alpha_G(Q)\right\}\right| .$$

The PL quartet score of S with respect to a set of tagged gene trees $\mathscr{G}$ is $q(S,\mathscr{G}) = \sum_{G\in\mathscr{G}} q(S,G)$.

Note that this definition gracefully handles missing data; gene family trees that do not include a specific species will not contribute quartets that include that species.

Definition 4.6 (Maximum per-Locus Quartet Score Tree (MLQST) problem)**.** Given a set of rooted tagged gene trees $\mathscr{G}$, find the species tree that maximizes the PL quartet score with respected to input gene trees, i.e., $\arg\max_S q(S,\mathscr{G})$.

Finally, note that while the PL quartet score depends on rooting and tagging, it is robust to *some* changes in the root placement; thus, the tree needs to be only partially rooted.

Claim 4.1. *If all nodes on the path between the root r and a node u are tagged as speciations, changing the root to any branch on the path does not alter the PL quartet score.*

4.4.2 Solving the MLQST problem

We start by briefly describing the ASTRAL algorithm to solve a related problem (the MQSST problem), and then describe how we extend this approach to the MLQST problem.

Background: ASTRAL on single-copy gene trees.

Note that, a node in a binary single-copy unrooted species tree forms a tripartition of $\mathscr{S}$ that implies the topology for all quartets anchored at that node, and this observation is at the base of the scoring scheme of ASTRAL. More formally, let $P = P_1|P_2|P_3$ and $M = M_1|M_2|M_3$ be two tripartitions, and let $I_{ij} = |M_i \cap P_j|$. Any species tree that displays P will share a certain number of quartets with any gene tree that displays M, and we call this number $QI(P,M)$ (calculations below extends to multifurcations if M is a d-partition). Defining B_3 as the set of all permutations of $\{1,2,3\}$, Mirarab et al. (2014) showed:

$$\begin{aligned} W(P) &= \frac{1}{2} \sum_{G \in \mathscr{G}} \sum_{M \in \mathscr{P}(G)} QI(P,M) \quad \text{where} \\ QI(P,M) &= \frac{1}{2} \sum_{(i,j,k) \in B_3} I_{i1} I_{j2} I_{k3} (I_{i1} + I_{j2} + I_{k3} - 3) \end{aligned} \tag{4.1}$$

and $\mathscr{P}(G)$ is the set of partitions representing internal nodes of G. The quartet score of a species tree is simply the sum of the weights of its tripartitions. The division by half in $W(P)$ is necessary because the sum counts each shared quartet twice (once at each anchor).

ASTRAL finds the tree S that maximizes the quartet score using dynamic programming. It recursively divides $\mathscr{S}$ into subsets, in each step, choosing the division that maximizes the sum of the weights. To avoid exponential running time, instead of considering all ways of partitioning a set $A \subset \mathscr{S}$ into A' and $A \setminus A'$, we constrain the search space to a given set of bipartitions. Let X

be this set and $X' = \{A : A|(\mathscr{S} \setminus A) \in X\}$ and $Y = \{(C,D) : C \in X', D \in X', C \cap D = \emptyset, C \cup D \in X'\}$. The quartet score of an optimal subtree on the cluster A, denoted as $V(A)$, is

$$V(A) = \max_{(A', A \setminus A') \in Y} V(A') + V(A \setminus A') + W(A'|(A \setminus A')|(\mathscr{S} \setminus A)), \tag{4.2}$$

where $V(\{a\}) = 0$ for all leaves $a \in \mathscr{S}$. This value can be computed recursively, and the optimal tree for $V(\mathscr{S})$ is the ASTRAL tree.

ASTRAL-Pro Algorithm

We extend here ASTRAL to multi-copy gene trees. The input to the new method, called ASTRAL-Pro, is a set of rooted tagged gene trees. This extension involves three changes in the way the weight w is computed: (i) To handle multi-copy gene trees, when computing the tripartition associated to each node, we use α_G to map labels to $\mathscr{S}$. Note that, in a tripartition $M = M_1|M_2|M_3$, the M_i are *sets* and not *multisets*, so multiple copies of the same species are considered only once. (ii) We change the weight calculation $W(P)$ so that each equivalence class of quartets is counted once instead of twice (only at its LCA anchor). (iii) When computing w, we only sum over internal nodes tagged as speciations. In addition, two changes to the algorithm procedure are needed: we need to root and tag gene trees and properly define the set X for multi-copy trees). We now detail these changes.

Weight calculation.

Let G be a rooted tagged gene tree, w an internal node of G tagged as speciation and $P = (P_1|P_2|P_3)$ a tripartition of $\mathscr{S}$.

Definition 4.7. For a species tree tripartition P and a SQ equivalence class that has the LCA anchor w in a gene tree G, we say that the SQ *is mapped from left to P* iff for each quartet Q in the equivalence class (i) P can anchor Q and (ii) the leaves a and b under the anchor of Q that appear first in a post-order traversal of G (e.g., u in Fig. 4.7) both map to the same side of P (that is, $\alpha_G(a) \in P_i, \alpha_G(b) \in P_i$ for some $1 \leq i \leq 3$). We denote such quartets by $Q \xrightarrow{w} P$.

We now state a set of lemmas, followed by the main result.

Lemma 4.1. *If $Q_1 \sim Q_2$ and $Q_1 \xrightarrow{w} P$, then $Q_2 \xrightarrow{w} P$.*

Lemma 4.2. *For a speciation node w with left child w_1 and right child w_2, let $M_1 = \alpha_G(w_1)$, $M_2 = \alpha_G(w_2)$ and $M_3 = \{\alpha_G(z) : z \in \mathscr{L}_G \setminus \mathscr{L}_G(w)$, LCA of w and z is tagged as speciation$\}$. Let $M_w = (M_1|M_2|M_3)$. Recall $I_{ij} = |M_i \cap P_j|$. The number of SQ quartet equivalence classes anchored to w and mapped from left to the species partition P can be counted as follows:*

$$QI_{pro}(P, M_w) = |\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P\}| = \sum_{(i,j,k)\in B_3, j<k} \binom{I_{1i}}{2} I_{2j} I_{2k} + \sum_{(i,j,k)\in B_3} \frac{I_{1i} I_{2j} I_{3k} (I_{1i} + I_{2j} - 2)}{2} \tag{4.3}$$

Lemma 4.3. *If $\Omega(G \restriction Q) \simeq S \restriction \alpha_G(Q)$, there exists a unique $P \in \mathscr{P}(S)$ satisfying $Q \xrightarrow{\psi_G(Q)} P$.*

Lemma 4.4. *Let $\mathbf{1}_{speciation}(w)$ be 1 for speciation nodes and 0 for duplication nodes, and let*

$$w_{pro}(P) = \sum_{G\in\mathscr{G}} \sum_{w\in I(G)} QI_{pro}(P, M_w) \times \mathbf{1}_{speciation}(w) \ .$$

Then: $q(S, \mathscr{G}) = \sum_{P\in\mathscr{P}(S)} w_{pro}(P)$.

Theorem 4.1. *The ASTRAL-Pro algorithm obtained by replacing $W(P)$ function with $w_{pro}(P)$ in the ASTRAL dynamic programming solves the MLQST problem exactly if $X = 2^{\mathscr{S}}$.*

Proof. By Lemma 4.4, $\arg\max_S q(S, \mathscr{G}) = \arg\max_S \sum_{P\in\mathscr{P}(S)} w_{pro}(P)$. Thus, ASTRAL dynamic programming can solve the optimization problem exactly given the full search space (the argument is identical to that of ASTRAL and follows from the additive nature of $q(S, \mathscr{G})$). □

We now make two claims and provide a sketch of proofs in Appendix (Proofs). Note that by Claim 4.3, ASTRAL-Pro has polynomial running time.

Claim 4.2. *For a set of gene trees $\mathscr{G}$ including only speciations, the tree returned by ASTRAL-Pro is the same as the one returned by ASTRAL.*

Algorithm 4.1. Gene tree tagging and rooting.

```
procedure TAGANDROOT(G)
    s ← ∞
    for edge e in G do
        root G at e and let r_e be the new root
        s_e ← TAG(r_e)
        if s_e < s then
            r ← r_e
            s ← s_e
    root at r
    TAG(r)
procedure TAG(u)
    if u is a leaf then
        score(u) ← 0
    else
        u_l, u_r ← children of u
        score(u) ← TAG(u_r) + TAG(u_l)
        if α_G(u_l) ∩ α_G(u_r) = ∅ then
            tag u as Speciation
        else
            tag u as Duplication
            if α_G(u_l) = α_G(u) ∨ α_G(u_r) = α_G(u) then
                if α_G(u_l) = α_G(u_r) then
                    score(u) ← score(u) + 1
                else
                    score(u) ← score(u) + 2
            else
                score(u) ← score(u) + 3
    return score(u)
```

Claim 4.3. *The asymptotic running time of ASTRAL-Pro is* $O(D|X|^{1.73}) = O(D(nN)^{1.73})$ *where* $N = \sum_{G \in \mathscr{G}} |\mathscr{L}_G|$ *and D denotes the number of unique gene tree tripartitions tagged as speciations.*

Tagging and rooting gene trees

Gene trees inferred from sequence data are neither rooted nor tagged. We use the heuristics presented in Algorithm 4.1 to root and tag gene trees, noting that a partially-correct rooting suffices (Claim 4.1). Given a rooted tree, we tag a node as duplication *only if* the node cannot be tagged as speciation by Definition 4.1 (similar to *observable duplication nodes* defined by Scornavacca et al., 2011); other nodes are *assumed* to be speciation.

For rooting, we seek the root position that minimizes the number of duplications and losses while allowing for "free" ILS. In more details, in each gene tree G, for two nodes u and v where $\alpha_G(u) = \alpha_G(v)$, we explain all differences in topologies below u and v by invoking "free" ILS (as opposed to duplication/loss). Then, three scenarios are possible for a node u with children u_l and u_r. (i) When u is duplication and $\alpha_G(u_l) = \alpha_G(u_l)$, we do not need to invoke any loss. One duplication suffices. (ii) If $\alpha_G(u_l) \subset \alpha_G(u_r)$ or vice versa, we need one loss on u_l and an arbitrary amount of ILS. (iii) Else, we need two losses (one in each side) and ILS to describe the differences. Algorithm 4.1 computes the number of duplication and loss events using this strategy, without penalizing ILS and fixing a cost of one for both duplications and losses. As described, it requires quadratic time per rooting and thus cubic time to find an optimal rooting. In our implementation, we used memoization to reduce this time to quadratic (details omitted). The LCA-based linear algorithm of Scornavacca et al. (2011) could also be adapted.

Search Space

We need to constrain the ASTRAL search space to bipartitions in a set X. To define X, we use a heuristic method relying on several strategies (see Algorithm S4.1; supplementary material). First, we use a sampling algorithm (SampleFull procedure) to create single-copy versions of each gene tree, creating a set $\mathscr{F}$. This sampling algorithm prunes the right (or left) subtrees below the highest duplication nodes in the tree, and recurses on each pruned tree, until no species has multiple copies. In addition, per each gene, 2^C (default: $C = 4$) single-copy trees are sampled from $\mathscr{F}$, creating a multiset $\mathscr{I}$. This sampling can be probabilistic (taking each side of a duplication with probability $\frac{1}{2}$) for high numbers of duplications. When the number of input trees is small, $\mathscr{I}$ may become too small; in these cases, $\mathscr{I}$ is augmented using another sampling algorithm (SampleExtra procedure). We provide $\mathscr{I}$ as input to the algorithms implemented in ASTRAL-III for building the set X. Finally, we complete all trees from $\mathscr{F}$ using the tree completion algorithm of ASTRAL-III and add the resulting bipartitions to X. All methods used guarantee that $|X|$ grows polynomially with the number of species, gene trees, and duplication

nodes.

Implementation

We implemented Algorithms 4.1 and S4.1 as part of a native C++ library called from Java. We based on code on the ASTRAL-MP (Yin et al., 2019) code. The code is available for all platforms, and can exploit multi-threading. A-Pro is available at https://github.com/chaoszhang/A-pro.

Statistical Consistency

When the input set $\mathscr{G}$ has only speciation nodes, the MLQST problem reduces to the Maximum Quartet Support Species Tree (MQSST) problem solved by ASTRAL (Mirarab et al., 2014). Thus, like the MQSST, the MLQST is NP-hard (Lafond and Scornavacca, 2019). Moreover, the solution to the MQSST problem is a statistically consistent estimator of the species tree under the MSC model and thus ASTRAL-Pro is also statistically consistent in absence of duplication.

In the presence of gene duplication and losses only, let us consider the birth-death model proposed by Arvestad et al. (2009) and refer to it as the GDL model.

Proposition 4.3. *Under the GDL model, every SQ in every correctly tagged rooted gene tree is isomorphic in topology to the species tree.*

Since all quartets in every equivalence class of SQs match the species tree, the per-locus quartet score will be maximized by the species tree. The following theorem follows.

Theorem 4.2. *Under the GDL model (Arvestad et al., 2009), the solution to the MLQST problem is a statistically consistent estimator of the species tree for correctly rooted and tagged gene trees.*

In fact, we suspect that ASTRAL-Pro is statistically consistent under the GDL model even when gene trees are imperfectly rooted and tagged. We leave the proof to future work.

Finally, note that restricting to X does not impact statistical consistency, as each bipartition of the species tree has a non-zero chance of appearing in the output of this algorithm.

Adopting local posterior probability for A-pro

By Proposition 4.3, assuming no error in the input gene trees or their tagging, differences between topologies of SQs and the species tree are due to processes other than GDL. The main such process is ILS. Thus, we can adopt the same quartet-based metric used for measuring support of ASTRAL trees for A-Pro trees.

For each quadripartition $A|B|C|D$ of $\mathscr{L}_S$, representing an internal branch in the species tree, we define z_1, which is the quartet count of the topology $(A \cup B)|(C \cup D)$, as:

$$\frac{\sum_{G \in \mathscr{G}} \sum_{a \in A, b \in B, c \in C, d \in D} |\{\psi_G(Q) : \alpha_G(Q) = ab|cd, Q \in \Sigma_G\}|}{|A||B||C||D|}.$$

The quartet count for $(A \cup C)|(B \cup D)$ and $(A \cup D)|(B \cup C)$ are similarly defined and are denoted by z_2 and z_3. We use these counts as input the local posterior probability calculation (Sayyari and Mirarab, 2016b). Thus,

Definition 4.8. The localPP support of a branch with counts $z_1 \ldots z_3$ is defined as:

$$\frac{h(z_1)}{h(z_1) + 2^{z_2 - z_1} h(z_2) + 2^{z_3 - z_1} h(z_3)}$$

where $h(x) = \mathbf{B}(x+1, k'-x+2\lambda)(1 - I_{\frac{1}{3}}(x+1, k'-x+2\lambda))$, $\mathbf{B}$ is the beta function, I_x is the regularized incomplete beta function, λ is the Yule prior parameter, set by default to $\frac{1}{2}$, and $k' = z_1 + z_2 + z_3$.

4.4.3 Datasets

We use new and existing simulated datasets as well as a biological dataset to test A-Pro.

New simulated dataset (S25)

We perform a set of simulations using SimPhy (Mallo et al., 2016) starting from a default model condition and adjusting five parameters (Table 4.1). We simulate 50 replicates per condition, and each replicate draws its parameters from prior distributions. Exact commands are given in the supplementary material.

Default model: The species tree, simulated under the Yule process with birth rate 5×10^{-9} and the maximum number of generations of the tree sampled from a log-normal distribution (mean 1.9×10^9), has 25 in-group and an out-group species. Each replicate has 1000 true gene trees simulated under DLCoal with fixed haploid population size $N_e = 4.7 \times 10^8$. Gene trees have mean ILS level in $[60\%, 80\%]$ range (mean 70%) across replicates (Fig. S4.2). The duplication rate $\lambda_+ = 4.9 \times 10^{-10}$; when there is no loss, gene trees on average include 145 leaves (≈ 5 extra copies per species). The loss rate λ_- is set to λ_+; with loss, gene trees have on average 43 leaves. The average number of duplication and loss events are 11 and 9, respectively, but variance is high (Fig. S4.1). For each gene, we use Indelible (Fletcher and Yang, 2009) to simulate gap-free nucleotide sequences along the gene trees using the GTR+Γ model (Tavaré, 1986) with 2 different sequence lengths: 500bp and 100bp. We then use FastTree2 (Price et al., 2010) to estimate maximum likelihood gene trees under the GTR+Γ model. Gene tree estimation error, measured by the FN rate between the true gene trees and the estimated gene trees, depends on the sequence length and fluctuates significantly (from 0–100%) both within and across replicates (Fig. S4.3); mean error is 36% and 15% for 100bp and 500bp, respectively.

Controlling λ_+, λ_-: Here, we consider $5 \times 4 = 20$ conditions, changing duplication and loss rates. Our λ_+ settings result in 0 to 5 extra copies per gene, and the $\frac{\lambda_-}{\lambda_+}$ varies between 0 and 1 (Table 4.1; Fig. S4.4). All other parameters are identical to the default condition.

Controlling λ_+, N_e: Here, we consider $3 \times 5 = 15$ conditions, fixing λ_- to be equal to λ_+, but changing λ_+ and ILS levels (controlled by N_e). Our λ_+ settings result in 0 to 5 extra copies per gene, and the mean ILS level between true and estimated gene trees varies between 0 and 70%

RF. (Table 4.1; Fig. S4.5) All other parameters are identical to the default model.

Controlling n: Fixing all parameters, we vary the number of in-group taxa n from 10 to 500.

Controlling k: Fixing all parameters, we vary the number of gene trees k from 25 to 10,000.

Existing simulations (S100)

We also used an existing dataset that Molloy and Warnow (2019) simulated based on a real fungal dataset (Rasmussen and Kellis, 2012). The simulation protocol of this dataset is similar to that of S25 dataset, with some notable differences. (i) The dataset included 100 species (no out-group); species tree height, speciation rate, and mutation rates all differed from S25. (ii) Shorter gene alignments were also used, resulting in higher MGTE (25bp: 67%, 50bp: 52%, 100bp: 35%, 500bp: 19%). (iii) The duplication rate λ_+ was set to 1×10^{-10}, 2×10^{-10}, or 5×10^{-10} (named 1, 2, and 5, respectively), and the duplication rate equaled the loss rate for all model conditions. (iv) ILS was much lower than S25; two conditions were simulated with N_e set to 1×10^7 and 5×10^7 (named 1 and 5, respectively), which result in 2% and 12% RF between true gene trees and the species tree. (v) Gene trees were estimated using RAxML instead of FastTree2.

Biological data

Wickett et al. (2014) has performed a transcriptome analysis of 103 plant species and 424 single-copy gene trees (out of thousands of genes) using both concatenation and ASTRAL. In preliminary analyses, the authors had inferred multi-copy gene trees using RAxML from 9683 genes for 83 of those species, ranging in size between 5 and 2395 leaves. However, not being able to obtain an accurate species tree from the multi-copy gene trees, they abandoned the strategy in later analyses. The gene trees are available from Matasci et al. (2014). We used RAxML gene trees inferred from the first two codon positions (C12) as the original study.

For the fungal dataset, all the peptide ML gene trees were downloaded from Butler et al. (2009) and used here. We used peptide gene trees because the reference species tree, inferred

through concatenation using MrBayes (Huelsenbeck and Ronquist, 2001), also uses peptide sequences. Authors comment on unreliability of their nucleotide-based analyses due to grouping by GC content.

4.4.4 Methods compared

We compare A-Pro to the following methods, which are the leading methods that can handle multiple copies. Another method, STAG (Emms et al., 2018), is not included because of its poor performance in the study by Molloy and Warnow (2019), including that it fails to run on some model conditions (Fig. S4.11).

DupTree (Wehe et al., 2008) infers a species tree from rooted or unrooted gene trees minimizing the duplication reconciliation cost (Maddison, 1997) under the duplication-only model, but it does not model ILS. We provide DupTree with unrooted gene trees. We also tried iGTP, minimizing Dup-Loss score, but we only show results in supplement (Fig. S4.7) as it was almost universally worse than DupTree.

MulRF (Chaudhary et al., 2013), based on an extension of the RF distance (Robinson and Foulds, 1981) to multi-labelled trees, is a hill-climbing method that aims at finding the tree with the minimum RF distance to the input. We use MulRF because of its advantage over other methods shown in previous studies (Chaudhary et al., 2015).

ASTRAL-multi (Rabiee et al., 2019) is a feature of ASTRAL designed for handling multiple individuals. Legried et al. (2020) proposes to use ASTRAL-multi for multi-copy data. Due to its high memory requirements, we were able to include it in only one experiment of S25.

4.5 Acknowledgments

Chapter 4, in full, is a reprint of the material as it appears in " Zhang, C., Scornavacca, C., Molloy, E. & Mirarab, S. ASTRAL-Pro: quartet-based species-tree inference despite paralogy. *Molecular Biology And Evolution*. **37**, 3292-3307 (2020) ." The dissertation author was the

primary investigator and first author of this paper.

Bibliography

J. An, L. Zhu, Y. Zhang, and H. Tang. Efficient visible light photo-fenton-like degradation of organic pollutants using in situ surface-modified BiFeO3 as a catalyst. *Journal of environmental sciences (China)*, 25(6):1213–25, 6 2013. ISSN 1001-0742. doi: 10.1016/s1001-0742(12)60172-7.

L. Arvestad, A.-C. Berglund, J. Lagergren, and B. Sennblad. Gene tree reconstruction and orthology analysis based on an integrated model for duplications and sequence evolution. In *Proceedings of the eighth annual international conference on Computational molecular biology - RECOMB '04*, pages 326–335, New York, New York, USA, 2004. ACM Press. ISBN 1581137559. doi: 10.1145/974614.974657.

L. Arvestad, J. Lagergren, and B. Sennblad. The gene evolution model and computing its associated probabilities. *Journal of the ACM*, 56(2):1–44, 4 2009. ISSN 00045411. doi: 10.1145/1502793.1502796.

J. A. Ballesteros and G. Hormiga. A New Orthology Assessment Method for Phylogenomic Data: Unrooted Phylogenetic Orthology. *Molecular Biology and Evolution*, 33(8):2117–2134, 8 2016. ISSN 0737-4038. doi: 10.1093/molbev/msw069.

J. A. Ballesteros and P. P. Sharma. A Critical Appraisal of the Placement of Xiphosura (Chelicerata) with Account of Known Sources of Phylogenetic Error. *Systematic Biology*, pages 1–62, 2 2019. ISSN 1063-5157. doi: 10.1093/sysbio/syz011.

M. S. Bansal, J. G. Burleigh, and O. Eulenstein. Efficient genome-scale phylogenetic analysis under the duplication-loss and deep coalescence cost models. *BMC Bioinformatics*, 11(Suppl 1):S42, 2010. ISSN 1471-2105. doi: 10.1186/1471-2105-11-S1-S42.

M. S. Bansal, E. J. Alm, and M. Kellis. Reconciliation revisited: Handling multiple optima when reconciling with duplication, transfer, and loss. *Journal of Computational Biology*, 2013. ISSN 10665277. doi: 10.1089/cmb.2013.0073.

M. S. M. Bayzid, S. Mirarab, and T. Warnow. Inferring optimal species trees under gene duplication and loss. *Pacific Symposium on Biocomputing*, 18:250–261, 2013.

G. Bourque, Y. Yacef, and N. El-Mabrouk. Maximizing Synteny Blocks to Identify Ancestral Homologs. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 21–34. 2005. ISBN 3540289321. doi: 10.1007/11554714{_}3.

B. Boussau, G. J. Szöllősi, and L. Duret. Genome-scale coestimation of species and gene trees. *Genome Research*, 23(2):323–330, 12 2013. ISSN 1549-5469. doi: 10.1101/gr.141978.112.

D. Bryant, R. Bouckaert, J. Felsenstein, N. A. Rosenberg, and A. Roychoudhury. Inferring species trees directly from biallelic genetic markers: Bypassing gene trees in a full coalescent analysis. *Molecular Biology and Evolution*, 29(8):1917–1932, 2012. ISSN 07374038. doi: 10.1093/molbev/mss086.

J. G. Burleigh and S. Mathews. Phylogenetic signal in nucleotide data from seed plants: implications for resolving the seed plant tree of life. *American Journal of Botany*, 91(10): 1599–1613, 10 2004. ISSN 00029122. doi: 10.3732/ajb.91.10.1599.

G. Butler, M. D. Rasmussen, M. F. Lin, M. A. S. Santos, S. Sakthikumar, C. A. Munro, E. Rheinbay, M. Grabherr, A. Forche, J. L. Reedy, I. Agrafioti, M. B. Arnaud, S. Bates, A. J. P. Brown, S. Brunke, M. C. Costanzo, D. A. Fitzpatrick, P. W. J. de Groot, D. Harris, L. L. Hoyer, B. Hube, F. M. Klis, C. Kodira, N. Lennard, M. E. Logue, R. Martin, A. M. Neiman, E. Nikolaou, M. A. Quail, J. Quinn, M. C. Santos, F. F. Schmitzberger, G. Sherlock, P. Shah, K. A. T. Silverstein, M. S. Skrzypek, D. Soll, R. Staggs, I. Stansfield, M. P. H. Stumpf, P. E. Sudbery, T. Srikantha, Q. Zeng, J. Berman, M. Berriman, J. Heitman, N. A. R. Gow, M. C. Lorenz, B. W. Birren, M. Kellis, and C. A. Cuomo. Evolution of pathogenicity and sexual reproduction in eight Candida genomes. *Nature*, 459(7247):657–662, 2009. ISSN 0028-0836. doi: 10.1038/nature08064. URL http://www.nature.com/articles/nature08064.

R. Chaudhary, M. S. Bansal, A. Wehe, D. Fernández-Baca, and O. Eulenstein. iGTP: a software package for large-scale gene tree parsimony analysis. *BMC bioinformatics*, 11(1):574, 1 2010. ISSN 1471-2105. doi: 10.1186/1471-2105-11-574.

R. Chaudhary, J. G. Burleigh, and D. Fernández-Baca. Inferring species trees from incongruent multi-copy gene trees using the Robinson-Foulds distance. *Algorithms for Molecular Biology*, 8:28, 2013. ISSN 1748-7188. doi: 10.1186/1748-7188-8-28.

R. Chaudhary, B. Boussau, J. G. Burleigh, and D. Fernández-Baca. Assessing approaches for inferring species trees from multi-copy genes. *Systematic Biology*, 64(2):325–339, 2015. ISSN 1076836X. doi: 10.1093/sysbio/syu128.

C. Chauve, N. El-Mabrouk, L. Guéguen, M. Semeria, and E. Tannier. Duplication, Rearrangement and Reconciliation: A Follow-Up 13 Years Later. In C. Chauve, N. El-Mabrouk, and E. Tannier, editors, *Models and Algorithms for Genome Evolution*, volume 19 of *Computational Biology*, pages 47–62. Springer London, London, 2013. ISBN 978-1-4471-5297-2. doi:

10.1007/978-1-4471-5298-9{_}4.

J. Chifman and L. S. Kubatko. Quartet Inference from SNP Data Under the Coalescent Model. *Bioinformatics*, 30(23):3317–3324, 8 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu530.

L. A. David and E. J. Alm. Rapid evolutionary innovation during an archaean genetic expansion. *Nature*, 469(7328):93, 2011.

N. De Maio, C. Schlötterer, and C. Kosiol. Linking Great Apes Genome Evolution across Time Scales Using Polymorphism-Aware Phylogenetic Models. *Molecular Biology and Evolution*, 30(10):2249–2262, 10 2013. ISSN 1537-1719. doi: 10.1093/molbev/mst131.

L. De Oliveira Martins, D. Mallo, and D. Posada. A Bayesian Supertree Model for Genome-Wide Species Tree Reconstruction. *Systematic Biology*, 65(3):397–416, 5 2016. ISSN 1063-5157. doi: 10.1093/sysbio/syu082.

P. Du, M. W. Hahn, and L. Nakhleh. Species Tree Inference under the Multispecies Coalescent on Data with Paralogs is Accurate. *bioRxiv*, page 498378, 2019. doi: 10.1101/498378.

C. W. Dunn, M. Howison, and F. Zapata. Agalma: an automated phylogenomics workflow. *BMC bioinformatics*, 14(1):330, 2013. ISSN 1471-2105.

D. Durand, B. V. Halldórsson, and B. Vernot. A hybrid micro-macroevolutionary approach to gene tree reconstruction. *Journal of Computational Biology*, 13(2):320–335, 2006. ISSN 1066-5277.

N. El-Mabrouk and E. Noutahi. Gene Family Evolution—An Algorithmic Framework. In *Bioinformatics and Phylogenetics*, pages 87–119. Springer, 2019.

D. M. Emms, S. Kelly, and S. P. Road. STAG: Species Tree Inference from All Genes. *bioRxiv*, page 267914, 1 2018. doi: 10.1101/267914.

W. Fletcher and Z. Yang. INDELible: A flexible simulator of biological sequence evolution. *Molecular Biology and Evolution*, 26(8):1879–1888, 2009. ISSN 07374038. doi: 10.1093/molbev/msp098.

T. C. Giarla and J. A. Esselstyn. The Challenges of Resolving a Rapid, Recent Radiation: Empirical and Simulated Phylogenomics of Philippine Shrews. *Systematic Biology*, 64(5):727–740, 9 2015. ISSN 1063-5157. doi: 10.1093/sysbio/syv029.

M. Goodman, J. Czelusniak, G. W. Moore, A. E. Romero-Herrera, and G. Matsuda. Fitting the Gene Lineage into its Species Lineage, a Parsimony Strategy Illustrated by Cladograms Constructed from Globin Sequences. *Systematic Biology*, 28(2):132–163, 1979. ISSN 1063-

5157, 1076-836X. doi: 10.1093/sysbio/28.2.132.

M. T. Hallett and J. Lagergren. New algorithms for the duplication-loss model. In *Proceedings of the fourth annual international conference on Computational molecular biology - RECOMB '00*, pages 138–146, New York, New York, USA, 2000. ACM Press. ISBN 1581131860. doi: 10.1145/332306.332359.

J. Heled and A. J. Drummond. Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution*, 27(3):570–580, 3 2010. ISSN 1537-1719. doi: 10.1093/molbev/msp274.

J. P. Huelsenbeck and F. Ronquist. MRBAYES: Bayesian inference of phylogenetic trees. *Bioinformatics*, 17(8):754–755, 2001. ISSN 1367-4803. doi: 10.1093/bioinformatics/17.8.754.

E. Jacox, C. Chauve, G. J. Szöllősi, Y. Ponty, and C. Scornavacca. eccetera: comprehensive gene tree-species tree reconciliation using parsimony. *Bioinformatics*, 32(13):2056–2058, 2016.

D. Kane and T. Tao. A bound on partitioning clusters. *Electr. J. Comb.*, 24:P2.31, 2017.

M. Lafond and C. Scornavacca. On the Weighted Quartet Consensus problem. *Theoretical Computer Science*, 769:1–17, 5 2019. ISSN 03043975. doi: 10.1016/j.tcs.2018.10.005.

M. Lafond, M. Semeria, K. M. Swenson, E. Tannier, and N. El-Mabrouk. Gene tree correction guided by orthology. *BMC Bioinformatics*, 14(S15):S5, 10 2013. ISSN 1471-2105. doi: 10.1186/1471-2105-14-S15-S5.

M. Lafond, C. Chauve, Dondi, and N. El-Mabrouk. Polytomy refinement for the correction of dubious duplications in gene trees. *Bioinformatics*, 30(17):i519–i526, 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu463.

B. R. Larget, S. K. Kotha, C. N. Dewey, and C. Ané. BUCKy: Gene tree/species tree reconciliation with Bayesian concordance analysis. *Bioinformatics*, 26(22):2910–2911, 11 2010. ISSN 13674803. doi: 10.1093/bioinformatics/btq539.

S. Laurin-Lemay, H. Brinkmann, and H. Philippe. Origin of land plants revisited in the light of sequence contamination and missing data. *Current Biology*, 2012. ISSN 09609822. doi: 10.1016/j.cub.2012.06.013.

J. H. Leebens-Mack, M. S. Barker, E. J. Carpenter, M. K. Deyholos, M. A. Gitzendanner, S. W. Graham, I. Grosse, Z. Li, M. Melkonian, S. Mirarab, M. Porsch, M. Quint, S. A. Rensing, D. E. Soltis, P. S. Soltis, D. W. Stevenson, K. K. Ullrich, N. J. Wickett, L. DeGironimo, P. P. Edger, I. E. Jordon-Thaden, S. Joya, T. Liu, B. Melkonian, N. W. Miles, L. Pokorny, C. Quigley, P. Thomas, J. C. Villarreal, M. M. Augustin, M. D. Barrett, R. S. Baucom, D. J. Beerling, R. M. Benstein, E. Biffin, S. F. Brockington, D. O. Burge, J. N. Burris, K. P. Burris,

V. Burtet-Sarramegna, A. L. Caicedo, S. B. Cannon, Z. Çebi, Y. Chang, C. Chater, J. M. Cheeseman, T. Chen, N. D. Clarke, H. Clayton, S. Covshoff, B. J. Crandall-Stotler, H. Cross, C. W. DePamphilis, J. P. Der, R. Determann, R. C. Dickson, V. S. Di Stilio, S. Ellis, E. Fast, N. Feja, K. J. Field, D. A. Filatov, P. M. Finnegan, S. K. Floyd, B. Fogliani, N. García, G. Gâteblé, G. T. Godden, F. Q. Y. Goh, S. Greiner, A. Harkess, J. M. Heaney, K. E. Helliwell, K. Heyduk, J. M. Hibberd, R. G. J. Hodel, P. M. Hollingsworth, M. T. J. Johnson, R. Jost, B. Joyce, M. V. Kapralov, E. Kazamia, E. A. Kellogg, M. A. Koch, M. Von Konrat, K. Könyves, T. M. Kutchan, V. Lam, A. Larsson, A. R. Leitch, R. Lentz, F.-W. Li, A. J. Lowe, M. Ludwig, P. S. Manos, E. Mavrodiev, M. K. McCormick, M. McKain, T. McLellan, J. R. McNeal, R. E. Miller, M. N. Nelson, Y. Peng, P. Ralph, D. Real, C. W. Riggins, M. Ruhsam, R. F. Sage, A. K. Sakai, M. Scascitella, E. E. Schilling, E.-M. Schlösser, H. Sederoff, S. Servick, E. B. Sessa, A. J. Shaw, S. W. Shaw, E. M. Sigel, C. Skema, A. G. Smith, A. Smithson, C. N. Stewart, J. R. Stinchcombe, P. Szövényi, J. A. Tate, H. Tiebel, D. Trapnell, M. Villegente, C.-N. Wang, S. G. Weller, M. Wenzel, S. Weststrand, J. H. Westwood, D. F. Whigham, S. Wu, A. S. Wulff, Y. Yang, D. Zhu, C. Zhuang, J. Zuidof, M. W. Chase, J. C. Pires, C. J. Rothfels, J. Yu, C. Chen, L. Chen, S. Cheng, J. Li, R. Li, X. Li, H. Lu, Y. Ou, X. Sun, X. Tan, J. Tang, Z. Tian, F. Wang, J. Wang, X. Wei, X. Xu, Z. Yan, F. Yang, X. Zhong, F. Zhou, Y. Zhu, Y. Zhang, S. Ayyampalayam, T. J. Barkman, N.-p. Nguyen, N. Matasci, D. R. Nelson, E. Sayyari, E. K. Wafula, R. L. Walls, T. Warnow, H. An, N. Arrigo, A. E. Baniaga, S. Galuska, S. A. Jorgensen, T. I. Kidder, H. Kong, P. Lu-Irving, H. E. Marx, X. Qi, C. R. Reardon, B. L. Sutherland, G. P. Tiley, S. R. Welles, R. Yu, S. Zhan, L. Gramzow, G. Theißen, G. K.-S. Wong, and O. T. P. T. Initiative. One thousand plant transcriptomes and the phylogenomics of green plants. *Nature*, 574(7780):679–685, 10 2019. ISSN 0028-0836. doi: 10.1038/s41586-019-1693-2.

B. Legried, E. K. Molloy, T. Warnow, and S. Roch. Polynomial-Time Statistical Estimation of Species Trees under Gene Duplication and Loss. In *Research in Computational Molecular Biology. RECOMB 2020. Lecture Notes in Computer Science*, volume 12074, pages 120–135. Springer, Cham, Switzerland, 2020. doi: 10.1007/978-3-030-45257-5_8.

Q. Li, N. Galtier, C. Scornavacca, and Y.-B. Chan. The Multilocus Multispecies Coalescent: A FlexibleNew Model of Gene Family Evolution. *bioRxiv*, page 2020.05.07.081836, 2020. doi: 10.1101/2020.05.07.081836. URL http://biorxiv.org/content/early/2020/05/08/2020.05.07.081836.abstract.

L. Liu. BEST: Bayesian estimation of species trees under the coalescent model. *Bioinformatics*, 24(21):2542–2543, 11 2008. ISSN 1367-4803. doi: 10.1093/bioinformatics/btn484.

L. Liu and L. Yu. Estimating Species Trees from Unrooted Gene Trees. *Systematic Biology*, 60 (5):661–667, 10 2011. ISSN 1076-836X. doi: 10.1093/sysbio/syr027.

L. Liu, L. Yu, D. K. Pearl, and S. V. Edwards. Estimating species phylogenies using coalescence times among sequences. *Systematic Biology*, 58(5):468–477, 10 2009. ISSN 10635157. doi: 10.1093/sysbio/syp031.

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

B. Ma, M. Li, and L. Zhang. From gene trees to species trees. *SIAM Journal on Computing*, 30 (3):729–752, 2000. ISSN 0097-5397.

W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 9 1997. ISSN 10635157. doi: 10.2307/2413694.

D. Mallo, L. De Oliveira Martins, and D. Posada. SimPhy: Phylogenomic Simulation of Gene, Locus, and Species Trees. *Systematic biology*, 65(2):334–44, 3 2016. ISSN 1076-836X. doi: 10.1093/sysbio/syv082.

M. Marcet-Houben and T. Gabaldón. TreeKO: a duplication-aware algorithm for the comparison of phylogenetic trees. *Nucleic Acids Research*, 39(10):e66–e66, 5 2011. ISSN 1362-4962. doi: 10.1093/nar/gkr087.

A. Markin and O. Eulenstein. Quartet-Based Inference Methods are Statistically Consistent Under the Unified Duplication-Loss-Coalescence Model, 2020.

N. Matasci, L.-H. L.-H. Hung, Z. Yan, E. E. J. Carpenter, N. N. J. Wickett, S. Mirarab, N. Nguyen, T. Warnow, S. Ayyampalayam, M. S. Barker, J. Burleigh, M. Gitzendanner, E. Wafula, J. Der, C. dePamphilis, B. Roure, H. Philippe, B. Ruhfel, N. Miles, S. Graham, S. Mathews, B. Surek, M. Melkonian, D. Soltis, P. Soltis, C. Rothfels, L. Pokorny, J. Shaw, L. DeGironimo, D. Stevenson, J. Villarreal, T. Chen, T. Kutchan, M. Rolf, R. Baucom, M. Deyholos, R. Samudrala, Z. Tian, X. Wu, X. Sun, Y. Zhang, J. Wang, J. Leebens-Mack, and G. Wong. Data access for the 1,000 Plants (1KP) project. *GigaScience*, 3(1):17, 2014. ISSN 2047-217X. doi: 10.1186/2047-217X-3-17.

S. Mirarab. Species Tree Estimation Using ASTRAL: Practical Considerations. *Arxiv preprint*, 1904.03826, 4 2019.

S. Mirarab and T. Warnow. ASTRAL-II: Coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 6 2015. ISSN 14602059. doi: 10.1093/bioinformatics/btv234.

S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548, 9 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu462.

E. K. Molloy and T. Warnow. To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303, 3 2018. ISSN 1063-5157. doi: 10.1093/sysbio/syx077.

E. K. Molloy and T. Warnow. FastMulRFS : Statistically consistent polynomial time species tree estimation under gene duplication. *bioRxiv*, page 835553, 2019. doi: 10.1101/835553.

E. Mossel and S. Roch. Incomplete lineage sorting: consistent phylogeny estimation from multiple loci. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 7(1):166–171, 1 2010. ISSN 1557-9964. doi: 10.1109/TCBB.2008.66.

E. Noutahi, M. Semeria, M. Lafond, J. Seguin, B. Boussau, L. Guéguen, N. El-Mabrouk, and E. Tannier. Efficient gene tree correction guided by genome evolution. *PLoS ONE*, 11(8), 2016. ISSN 19326203. doi: 10.1371/journal.pone.0159559.

P. Pamilo and M. Nei. Relationships between gene trees and species trees. *Molecular biology and evolution*, 5(5):568–583, 1988. ISSN 0737-4038.

M. N. Price, P. S. Dehal, and A. P. Arkin. FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 3 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0009490.

M. Rabiee, E. Sayyari, and S. Mirarab. Multi-allele species reconstruction using ASTRAL. *Molecular Phylogenetics and Evolution*, 130:286–296, 1 2019. ISSN 10557903. doi: 10.1016/j.ympev.2018.10.033.

B. Rannala and Z. Yang. Bayes estimation of species divergence times and ancestral population sizes using DNA sequences from multiple loci. *Genetics*, 164(4):1645–1656, 2003.

M. Rasmussen and M. Kellis. Unified modeling of gene duplication, loss, and coalescence using a locus tree. *Genome research*, 22(4):755–765, 2012. doi: 10.1101/gr.123901.111.Freely.

D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53 (1-2):131–147, 1981.

L. Salichos and A. Rokas. Inferring ancient divergences requires genes with strong phylogenetic signals. *Nature*, 497(7449):327–31, 2013. ISSN 1476-4687. doi: 10.1038/nature12130.

E. Sayyari and S. Mirarab. Anchoring quartet-based phylogenetic distances and applications to species tree reconstruction. *BMC Genomics*, 17(S10):101–113, 11 2016a. ISSN 1471-2164. doi: 10.1186/s12864-016-3098-z.

E. Sayyari and S. Mirarab. Fast Coalescent-Based Computation of Local Branch Support from Quartet Frequencies. *Molecular Biology and Evolution*, 33(7):1654–1668, 7 2016b. ISSN 0737-4038. doi: 10.1093/molbev/msw079.

E. Sayyari, J. B. Whitfield, and S. Mirarab. DiscoVista: Interpretable visualizations of gene tree discordance. *Molecular Phylogenetics and Evolution*, 122:110–115, 5 2018. ISSN 10557903.

doi: 10.1016/j.ympev.2018.01.019.

C. Scornavacca, V. Berry, and V. Ranwez. Building species trees from larger parts of phylogenomic databases. *Information and Computation*, 209(3):590–605, 3 2011. ISSN 08905401. doi: 10.1016/j.ic.2010.11.022.

C. Scornavacca, E. Jacox, and G. J. Szöllosi. Joint amalgamation of most parsimonious reconciled gene trees. *Bioinformatics*, 31(6):841–848, 2015. ISSN 14602059. doi: 10.1093/bioinformatics/btu728.

B. Sennblad and J. Lagergren. Probabilistic Orthology Analysis. *Systematic Biology*, 58(4): 411–424, 8 2009. ISSN 1076-836X. doi: 10.1093/sysbio/syp046.

S. Shekhar, S. Roch, and S. Mirarab. Species Tree Estimation Using ASTRAL: How Many Genes Are Enough? *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(5):1738–1747, 9 2018. ISSN 1545-5963. doi: 10.1109/TCBB.2017.2757930.

G. J. Szollosi, B. Boussau, S. S. Abby, E. Tannier, and V. Daubin. Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. *Proceedings of the National Academy of Sciences*, 109(43):17513–17518, 2012. ISSN 0027-8424. doi: 10.1073/pnas.1202997109.

G. J. Szöllõsi, W. Rosikiewicz, B. Boussau, E. Tannier, and V. Daubin. Efficient exploration of the space of reconciled gene trees. *Systematic biology*, 62(6):901–12, 11 2013. ISSN 1076-836X. doi: 10.1093/sysbio/syt054.

G. J. Szöllõsi, E. Tannier, V. Daubin, and B. Boussau. The inference of gene trees with species trees. *Systematic Biology*, 64(1):e42–e62, 7 2014. ISSN 1063-5157. doi: 10.1093/sysbio/syu048.

S. Tavaré. Some Probabilistic and Statistical Problems in the Analysis of DNA Sequences. *Lectures on Mathematics in the Life Sciences*, 17:57–86, 1986.

P. Vachaspati and T. Warnow. ASTRID: Accurate Species TRees from Internode Distances. *BMC Genomics*, 16(Suppl 10):S3, 2015. ISSN 1471-2164.

A. Wehe, M. S. Bansal, J. G. Burleigh, and O. Eulenstein. DupTree: a program for large-scale phylogenetic analyses using gene tree parsimony. *Bioinformatics*, 24(13):1540–1541, 2008.

N. J. Wickett, S. Mirarab, N. Nguyen, T. Warnow, E. J. Carpenter, N. Matasci, S. Ayyampalayam, M. S. Barker, J. G. Burleigh, M. A. Gitzendanner, B. R. Ruhfel, E. Wafula, J. P. Der, S. W. Graham, S. Mathews, M. Melkonian, D. E. Soltis, P. S. Soltis, N. W. Miles, C. J. Rothfels, L. Pokorny, A. J. Shaw, L. DeGironimo, D. W. Stevenson, B. Surek, J. C. Villarreal, B. Roure, H. Philippe, C. W. DePamphilis, T. Chen, M. K. Deyholos, R. S. Baucom, T. M. Kutchan,

M. M. Augustin, J. J. Wang, Y. Zhang, Z. Tian, Z. Yan, X. Wu, X. Sun, G. K.-S. Wong, and J. J. Leebens-Mack. Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences*, 111(45):4859–4868, 10 2014. ISSN 0027-8424. doi: 10.1073/pnas.1323926111.

Y. Wu. Coalescent-based species tree inference from gene tree topologies under incomplete lineage sorting by maximum likelihood. *Evolution*, 66(3):763–775, 2012. ISSN 00143820. doi: 10.1111/j.1558-5646.2011.01476.x.

Y.-C. Wu, M. D. Rasmussen, M. S. Bansal, and M. Kellis. TreeFix: Statistically Informed Gene Tree Error Correction Using Species Trees. *Systematic Biology*, 62(1):110–120, 2013. ISSN 1063-5157, 1076-836X. doi: 10.5061/dryad.44cb5.

Y. Yang and S. A. Smith. Orthology Inference in Nonmodel Organisms Using Transcriptomes and Low-Coverage Genomes: Improving Accuracy and Matrix Occupancy for Phylogenomics. *Molecular Biology and Evolution*, 31(11):3081–3092, 11 2014. ISSN 1537-1719. doi: 10.1093/molbev/msu245.

J. Yin, C. Zhang, and S. Mirarab. ASTRAL-MP: scaling ASTRAL to very large datasets using randomization and parallelization. *Bioinformatics*, 35(20):3961–3969, 10 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz211.

C. Zhang, M. Rabiee, E. Sayyari, and S. Mirarab. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6):153, 5 2018. ISSN 1471-2105. doi: 10.1186/s12859-018-2129-y.

B. Zhong, T. Yonezawa, Y. Zhong, and M. Hasegawa. The Position of Gnetales among Seed Plants: Overcoming Pitfalls of Chloroplast Phylogenomics. *Molecular Biology and Evolution*, 27(12):2855–2863, 12 2010. ISSN 0737-4038. doi: 10.1093/molbev/msq170.

B. Zhong, O. Deusch, V. V. Goremykin, D. Penny, P. J. Biggs, R. A. Atherton, S. V. Nikiforova, and P. J. Lockhart. Systematic Error in Seed Plant Phylogenomics. *Genome Biology and Evolution*, 3:1340–1348, 1 2011. ISSN 1759-6653. doi: 10.1093/gbe/evr105.

Appendices

4.A Proofs

Proof of Proposition 4.1. Denote $Q_1 = \{a,b,c,d\}$ and $Q_2 = \{\tilde{a},\tilde{b},\tilde{c},\tilde{d}\}$ (with obvious correspondence of labels). Let w be the anchor LCA and note that anchor LCA is the LCA of three (if $Q_1 \angle G$) or four (if $Q_1 \perp G$) of the quartet leaves; thus, by Definition 4.2, w is a speciation node or otherwise Q_1 would not be a SQ. Let the children of w be denoted by w_1 and w_2; by Definition 4.1, $\alpha_G(w_1)$ and $\alpha_G(w_2)$ must be mutually exclusive. In the unbalanced case, w.l.o.g, assume the topology is $(((a,b),c),d)$; then, let u denote the LCA of w and d and note that u is the LCA of a, b, and d. Thus, by Definition 4.2, u is a speciation node or otherwise Q_1 would not be a SQ. Let the children of u be denoted by u_1 and u_2, and w.l.o.g., let u_1 be the child on the same side as w. By Definition 4.1, $\alpha_G(u_1)$ and $\alpha_G(u_2)$ must be mutually exclusive. Therefore $\alpha_G(w) \subseteq \alpha_G(u_1)$. Consequently, $\alpha_G(w_1)$, $\alpha_G(w_2)$, and $\alpha_G(u_2)$ are mutually exclusive. Given that $a,b \in \mathscr{L}_G(w_1)$, $c \in \mathscr{L}_G(w_2)$, $d \in \mathscr{L}_G(u_2)$, mutual exclusivity is possible only if $\tilde{a},\tilde{b} \in \mathscr{L}_G(w_1)$, $\tilde{c} \in \mathscr{L}_G(w_2)$, $\tilde{d} \in \mathscr{L}_G \setminus \mathscr{L}_G(u_1)$. In the case of balanced topology (w.l.o.g, $((a,b),(c,d))$), mutual exclusivity of $\alpha_G(w_1)$ and $\alpha_G(w_2)$ and the fact that $a,b \in \mathscr{L}_G(w_1)$ and $c,d \in \mathscr{L}_G(w_2)$ implies that $\tilde{a},\tilde{b} \in \mathscr{L}_G(w_1)$, $\tilde{c},\tilde{d} \in \mathscr{L}_G(w_2)$. Thus, in either case, $\Omega(G \restriction Q_1) \simeq \Omega(G \restriction Q_2)$. □

Proof of Proposition 4.2. Each node of a gene tree represents an ancestral or present-day gene and thus belongs to a locus. The children of a speciation node stay in the same locus that their parent, while for a duplication node we have that exactly one of the two children change locus and the other stays in the same locus as its parent. Therefore, all nodes under w, which is a speciation node, belong to the descendants (including itself) of the locus to which w belongs,

and when tracing back to the time of speciation event w, they will lead to the same locus. Since all equivalence classes share the same anchor LCA, the result follows. □

Proof of Lemma 4.1. Note that P can anchor Q_1 only if any species tree that includes P must match the gene tree topology for Q_1 By Proposition 4.1, due to equivalence of Q_1 and Q_2, we infer Q_2 must (i) match the same species quartet set as Q_1 and (ii) share the same anchor LCA w. Thus, P can also anchor Q_2. (iii) When $Q_1 \angle G$ as shown in Figure 4.7, $\tilde{a}, \tilde{b} \in \mathscr{L}_G(w_1)$ are the leaves mapped to the quartet tree and thus mapped to the same partition as a, b; similarly, when $Q_1 \perp G$, the pair of leaves under the left subtree of the anchor LCA of both quartets map to the same partition of P. □

Proof of Lemma 4.2. First note that:

$$
\begin{aligned}
&|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P\}| = \\
&|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P, Q \perp G\}| + \\
&|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P, Q \angle G\}|
\end{aligned}
$$

We compute each part separately. Recall here that, since $Q \xrightarrow{w} P$, Q is a SQ quartet and thus $|Q| = |\alpha_G(Q)| = 4$.

When $Q \perp G$, let $Q = \{a, b, c, d\}$ with $\alpha_G(a), \alpha_G(b) \in M_1, \alpha_G(c), \alpha_G(d) \in M_2$. Since $Q \xrightarrow{w} P$, leaves $\alpha_G(a)$ and $\alpha_G(b)$ must be in the same partition of P. When $\alpha_G(a), \alpha_G(b) \in P_1$, leaves $\alpha_G(c)$ and $\alpha_G(d)$ must be in partition P_2 and P_3 respectively since P can anchor Q. W.l.o.g., we can assume $\alpha_G(c) \in P_2$. Therefore, $\alpha_G(a), \alpha_G(b) \in M_1 \cap P_1$, $\alpha_G(c) \in M_2 \cap P_2$, $\alpha_G(d) \in M_2 \cap P_3$. The number of such $\alpha_G(Q)$ is $\binom{|M_1 \cap P_1|}{2} |M_2 \cap P_2||M_2 \cap P_3| = \binom{I_{11}}{2} I_{22} I_{23}$. Similarly, when $\alpha_G(a), \alpha_G(b) \in M_1 \cap P_2$ and $\alpha_G(a), \alpha_G(b) \in M_1 \cap P_3$, the number of such $\alpha_G(Q)$ is $\binom{I_{12}}{2} I_{21} I_{23}$ and $\binom{I_{13}}{2} I_{21} I_{22}$ respectively. Thus,

$$
|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P, Q \perp G\}| = \binom{I_{11}}{2} I_{22} I_{23} + \binom{I_{12}}{2} I_{21} I_{23} + \binom{I_{13}}{2} I_{21} I_{22}
$$

Similarly, when $Q \angle G$, let $Q = \{a,b,c,d\}$ with $\alpha_G(a)$ and $\alpha_G(b)$ in the same partition of P. Notice that, in the unbalanced case, $\alpha_G(a)$ and $\alpha_G(b)$ can be both either in M_1 or either in M_2, and since c and d are not interchangeable as in the balanced case, we can have $\alpha_G(a), \alpha_G(b) \in P_i, \alpha_G(c) \in P_j, \alpha_G(d) \in P_k$ for (i,j,k) with any permutation of $(1,2,3)$, from the definition of P anchoring Q. All together, we have 12 cases.

In the case that $\alpha_G(a), \alpha_G(b) \in P_1, \alpha_G(c) \in P_2, \alpha_G(d) \in P_3$, and $\alpha_G(a), \alpha_G(b) \in M_1$, we have $\alpha_G(a), \alpha_G(b) \in M_1 \cap P_1$, $\alpha_G(c) \in M_2 \cap P_2$, and $\alpha_G(d) \in M_3 \cap P_3$. The number of such $\alpha_G(Q)$ is $\binom{|M_1 \cap P_1|}{2} |M_2 \cap P_2||M_3 \cap P_3| = \binom{I_{11}}{2} I_{22} I_{33}$. The other 11 permutations are similar. In total,

$$\begin{aligned}
&|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P, Q \angle G\}| \\
=&\binom{I_{11}}{2}(I_{22}I_{33} + I_{32}I_{23}) + \binom{I_{12}}{2}(I_{21}I_{33} + I_{31}I_{23}) \\
&+ \binom{I_{13}}{2}(I_{21}I_{32} + I_{31}I_{22}) + \binom{I_{21}}{2}(I_{12}I_{33} + I_{32}I_{13}) \\
&+ \binom{I_{22}}{2}(I_{11}I_{33} + I_{31}I_{13}) + \binom{I_{23}}{2}(I_{11}I_{32} + I_{31}I_{12})
\end{aligned} \tag{4.4}$$

Thus,

$$\begin{aligned}
QI_{pro}(P, M_w) =& |\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P\}| = \\
&\binom{I_{11}}{2} I_{22} I_{23} + \binom{I_{12}}{2} I_{21} I_{23} + \binom{I_{13}}{2} I_{21} I_{22} \\
&+ \binom{I_{11}}{2}(I_{22}I_{33} + I_{32}I_{23}) + \binom{I_{12}}{2}(I_{21}I_{33} + I_{31}I_{23}) + \binom{I_{13}}{2}(I_{21}I_{32} + I_{31}I_{22}) \\
&+ \binom{I_{21}}{2}(I_{12}I_{33} + I_{32}I_{13}) + \binom{I_{22}}{2}(I_{11}I_{33} + I_{31}I_{13}) + \binom{I_{23}}{2}(I_{11}I_{32} + I_{31}I_{12})
\end{aligned} \tag{4.5}$$

With simple manipulations, it can be shown that the right-hand side of this equation can be rewritten as:

$$\sum_{(i,j,k) \in B_3, j<k} \binom{I_{1i}}{2} I_{2j} I_{2k} + \sum_{(i,j,k) \in B_3} \frac{I_{1i} I_{2j} I_{3k} (I_{1i} + I_{2j} - 2)}{2}$$

□

Proof of Lemma 4.3. Let $\Omega(G \restriction Q)$ be designated by $ab|cd$ and assume w.l.o.g that the anchor corresponding to a and b is the first anchor observed on the post-order traverse of G. It is easy to show (see Mirarab et al., 2014) that if $\Omega(G \restriction Q) \simeq S \restriction \alpha_G(Q)$ there exist exactly two tripartitions P^1 and P^2 in $\mathscr{P}(S)$ that imply a quartet topology that matches $\Omega(G \restriction Q)$ (condition (ii) of Definition 4.7). Each of the two tripartitions has two leaves of $\alpha_G(Q)$ in one of its parts and the other two leaves fall on two different parts. Also, the two leaves that are together can only be a and b or c and d and thus, only one of P^1 and P^2 would include both a and b in the same part. Therefore, by condition (iii) of Definition 4.7, exactly one of $Q \xrightarrow{\psi_G(Q)} P^1$ and $Q \xrightarrow{\psi_G(Q)} P^2$ can be true. □

Proof of Lemma 4.4.

$$
\begin{aligned}
q(S,\mathscr{G}) &= \sum_{G\in\mathscr{G}} q(S,G) && (4.6)\\
&= \sum_{G\in\mathscr{G}} |\{\big(\alpha_G(Q),\psi_G(Q)\big) : Q\in\Sigma_G, \Omega(G\restriction Q)\simeq S\restriction\alpha_G(Q)\}| && (4.7)\\
&= \sum_{P\in\mathscr{P}(S)}\sum_{G\in\mathscr{G}} |\{\big(\alpha_G(Q),\psi_G(Q)\big) : Q\subset\mathscr{L}_G, Q\xrightarrow{\psi_G(Q)} P\}| && (4.8)\\
&= \sum_{P\in\mathscr{P}(S)}\sum_{G\in\mathscr{G}}\sum_{w\in I(G)} |\{\alpha_G(Q) : Q\subset\mathscr{L}_G, Q\xrightarrow{w} P\}| && (4.9)\\
&= \sum_{P\in\mathscr{P}(S)}\sum_{G\in\mathscr{G}}\sum_{w\in I(G)} |\{\alpha_G(Q) : Q\subset\mathscr{L}_G, Q\xrightarrow{w} P\}|\times\mathbf{1}_{speciation}(w) && (4.10)\\
&= \sum_{P\in\mathscr{P}(S)}\sum_{G\in\mathscr{G}}\sum_{w\in I(G)} QI_{pro}(P,M_w)\times\mathbf{1}_{speciation}(w) && (4.11)\\
&= \sum_{P\in\mathscr{P}(S)} w_{pro}(P) && (4.12)
\end{aligned}
$$

The first two lines are implied by Definition 4.5. Equation (4.8) follows from Lemma 4.1 and Lemma 4.3 that together establish that each equivalence class of quartets maps to exactly one P. Equation (4.9) follows from Definition 4.4 combined with a simple rearrangement obtained by counting unique tuples once. Equation (4.10) follows from the fact that when w is a duplication

node, $|\{\alpha_G(Q) : Q \subset \mathscr{L}_G, Q \xrightarrow{w} P\}| = 0$. Equation (4.11) follows from Lemma 4.2. □

Proof sketch of Claim 4.1. The rooting that minimizes the number of duplications and losses (#duploss for short) in Alg. 4.1 may not be unique. In particular, if a rooted tree G minimizes #duploss, then rooting it at any branch such that the path between the parent node of the branch and the current root (including the two end nodes) does not contain any duplication node will also minimize #duploss. We call a correctly-tagged gene tree partially-correctly-rooted if the path between the parent node of the branch where it is rooted and the root in the correctly-rooted tree does not contain any duplication node. In particular, when gene trees do not have duplications, then any rooting of a gene tree is partially-correctly. We observe that the equivalence classes of quartets in all partially-correctly-rooted trees stay the same (although *all* quartet trees in the same equivalence class may change from balanced to unbalanced or vice versa), and thus any partially-correct-rooting of gene trees will result in the same species tree. □

Sketch of proof of Claim 4.2. When $\mathscr{G}$ only includes speciation nodes, regardless of rooting, each quartet is a SQ. Since each leaf corresponds to distinct taxa in the species tree, each quartet equivalence class contains only one quartet. Therefore, each quartet is counted exactly once and thus $\sum_{P \in \mathscr{P}(S)} w_{pro}(P) = \sum_{P \in \mathscr{P}(S)} W(P)$ regardless of rooting. □

Sketch of proof of Claim 4.3 (Running time of ASTRAL-Pro). Let

$$N = \sum_{G \in \mathscr{G}} |\mathscr{L}_G|$$

denote the sum of the number of leaves in the gene trees. Then the number of anchor LCAs in all gene trees is $O(N)$. Let D denote the number of unique gene tree tripartitions tagged as speciations and note $D = O(N)$. By only counting each unique gene tree tripartition once against each species tree tripartition, the running time of ASTRAL-Pro becomes $O(D|X|^{1.73})$ (by an argument that is identical to that provided for ASTRAL-III (Mirarab et al., 2014) and follows from results of Kane and Tao (2017). However, while ASTRAL-III guarantees $|X| = O(nk)$ with

$k = |\mathscr{G}|$, in ASTRAL-Pro, in the presence of duplications, $|X|$ can be large; in particular with our sampling algorithm (Algorithm S4.1), $|X| = O(nN)$. Thus, the running time of A-Pro is $O(D(nN)^{1.73})$. Note that this analysis is not tight and can be made more precise in the future. Also, in the future, we will explore sub-sampling a constant number of trees from the output of Algorithm S4.1 per gene tree, which will limit the $|X| = nk$ and thus limit the running time of ASTRAL-pro to $O(D(nk)^{1.73})$. □

Proof of Proposition 4.3. Under GDL, besides leaves, each internal node $u_G \in I(G)$ in a gene tree G corresponds to an internal node $u_S \in I(S)$; if u_G is a duplication node, u_S is the node down the branch in S where the duplication event happened, and if u_G is a speciation node, u_S is the respective speciation node. It is easy to see that $\alpha_G(u_G) \subset \mathscr{L}_S(u_S)$. For each SQ quartet $Q = \{a,b,c,d\}$, assuming w.o.l.g that $G \restriction Q$ has unrooted topology $ab|cd$, let $w_G = \psi_G(Q)$, and u_G and v_G be the children of w_G. Let u_G, v_G, and w_G correspond to u_S, v_S, and w_S in S, respectively. Since w_G is a correctly tagged speciation node, u_S and v_S are descendants from different children of w_S.

When $Q \perp G$, assuming w.o.l.g. $a,b \in \mathscr{L}_G(u_G)$ and $c,d \in \mathscr{L}_G(v_G)$, we get $\alpha_G(a), \alpha_G(b) \in \mathscr{L}_S(u_S)$ and $\alpha_G(c), \alpha_G(d) \in \mathscr{L}_S(v_S)$ and thus $\alpha_G(a)\alpha_G(b)|\alpha_G(c)\alpha_G(d)$ is induced by S.

When $Q \angle G$, assuming w.o.l.g. $a,b \in \mathscr{L}_G(u_G)$, $c \in \mathscr{L}_G(v_G)$, and $d \notin \mathscr{L}_G(w_G)$, we get $\alpha_G(a), \alpha_G(b) \in \mathscr{L}_S(u_S)$ and $\alpha_G(c) \in \mathscr{L}_S(v_S)$. Since d is not under w_G, $\alpha_G(d)$ and w_S are under different children of the species tree node to which the LCA of d and w_G corresponds. Therefore, $\alpha_G(d) \notin \mathscr{L}_S(w_S)$ and thus $\alpha_G(d) \notin \mathscr{L}_S(u_S)$; since $\alpha_G(a) \in \mathscr{L}_S(u_S)$ and $\alpha_G(b) \in \mathscr{L}_S(u_S)$, it follows that $\alpha_G(a)\alpha_G(b)|\alpha_G(c)\alpha_G(d)$ in S. □

4.B Supplementary Algorithms

Algorithm S4.1. Building set X. Default constant parameters: $C = 4$, $E_m = 500$, $E_s = 4$. The algorithm uses the (arbitrary) left/right orientation of children of a node as given in the input.

procedure BUILDX($\mathscr{G}$)
  $\mathscr{F} = \emptyset$ and $\mathscr{I} = \emptyset$
  for $G \leftarrow \mathscr{G}$ **do**
    $(M,S) \leftarrow$ SAMPLEFULL$(G, \mathscr{L}_G, C)$
    $\mathscr{F} \leftarrow \mathscr{F} \cup S$
    $\mathscr{I} \leftarrow \mathscr{I} \uplus M$
  for $G \in \{$randomly sample $\max(0, \min(|\mathscr{G}|, \frac{E_m - |\mathscr{G}|}{E_s}))$ trees from $\mathscr{G}\}$ **do**
    $\mathscr{I} \leftarrow \mathscr{I} \uplus$ SAMPLEEXTRA$(G, \mathscr{L}_G)$
  $X \leftarrow$ run all ASTRAL-III methods for building X with $\mathscr{I}$ as input (i.e., `-i` $\mathscr{I}$)
  $X \leftarrow X \cup \Big($all bipartitions of $\{G$ completed via the ASTRAL-III tree-completion method $\forall G \in \mathscr{F}\}\Big)$
procedure SAMPLEFULL(G, A, c)
  if $|\alpha_G(A)| = |A|$ **then**
    return (multiset: $[\Omega(G \restriction A)$ repeated 2^c times], set: $\{\Omega(G \restriction A)\}$)
  else
    $A_l \leftarrow \emptyset$ and $A_r \leftarrow \emptyset$
    $G_A \leftarrow G \restriction A$ (degree-2 nodes removed)
    for $a \in A$ **do**
      $p \leftarrow$ the highest ancestor of a in G_A tagged as a duplication node (or $\emptyset$ if it doesn't exist)
      if $(p = \emptyset) \vee (a$ is to the left of $p)$ **then**
        $A_l \leftarrow A_l \cup \{a\}$
      if $(p = \emptyset) \vee (a$ is to the right of $p)$ **then**
        $A_r \leftarrow A_r \cup \{a\}$
    $(L.m, L.s) \leftarrow$ SAMPLEFULL$(G, A_l, \max(c-1, 0))$
    $(R.m, R.s) \leftarrow$ SAMPLEFULL$(G, A_r, \max(c-1, 0))$
    if $c = 0$ **then**
      return (multiset: randomly select $L.m$ or $R.m$ with equal probabilities, set: $L.s \cup R.s$)
    else
      return (multiset: $L.m \uplus R.m$, set: $L.s \cup R.s$)
procedure SAMPLEEXTRA(G, A)
  if $|\alpha_G(A)| = |A|$ **then**
    return multiset $[\Omega(G)$ repeated once]
  else
    $A_l \leftarrow \emptyset$ and $A_r \leftarrow \emptyset$
    $G_A \leftarrow G \restriction A$ (degree-2 nodes removed)
    for $a \in A$ **do**
      $p \leftarrow$ the highest ancestor of a in G_A tagged as a duplication node (or $\emptyset$ if it doesn't exist)
      if $(p \neq \emptyset) \wedge (a$ is to the left of $p)$ **then**
        $A_l \leftarrow A_l \cup \{a\}$
      if $(p \neq \emptyset) \wedge (a$ is to the right of $p)$ **then**
        $A_r \leftarrow A_r \cup \{a\}$
    $B_l \leftarrow \{x : x \in A_l, \alpha_G(x) \in \alpha_G(A_l) \setminus \alpha_G(A_r)\}$
    $B_r \leftarrow \{x : x \in A_r, \alpha_G(x) \in \alpha_G(A_r) \setminus \alpha_G(A_l)\}$
    $G_L \leftarrow G \restriction ((\mathscr{L}_G \setminus A_r) \cup B_r)$ (degree-2 nodes removed)
    $G_R \leftarrow G \restriction ((\mathscr{L}_G \setminus A_l) \cup B_l)$ (degree-2 nodes removed)
    for internal node u of G_L where $\mathscr{L}_G(u) \subset B_r$ **do**
      $B_u \leftarrow \{$one leaf node arbitrarily chosen from $\{x : \alpha_G(x) = s, x \in \mathscr{L}_G(u)\} : s \in \alpha_G(u)\}$
      replace u with a star tree consisting of leaves from the set B_u
    for internal node u of G_R where $\mathscr{L}_G(u) \subset B_l$ **do**
      $B_u \leftarrow \{$one leaf node arbitrarily chosen from $\{x : \alpha_G(x) = s, x \in \mathscr{L}_G(u)\} : s \in \alpha_G(u)\}$
      replace u with a star tree consisting of leaves from the set B_u
    $R =$ SAMPLEEXTRA$(G_L, A_l) \uplus$ SAMPLEEXTRA(G_R, A_r)
    return R

4.C Simulation details

Simphy command for default parameters:

```
simphy -sl f:25 -rs 50 -rl f:1000 -rg 1 -sb f:0.000000005 -sd f:0
-st ln:21.25,0.2 -so f:1 -si f:1 -sp f:470000000 -su ln:-21.9,0.1
-hh f:1 -hs ln:1.5,1 -hl ln:1.551533,0.6931472 -hg ln:1.5,1 -cs 9644
-v 3 -o default -ot 0 -op 1 -lb f:0.00000000049 -ld f:0.00000000049
-lt f:0
```

Other settings use a similar command with parameters changed according to the table below.

Table S4.1. Simphy parameters for all experiments

Parameter name	**Parameter value**
Default Parameters	
Speciation rate	5e-9
Extinction rate	0
Locus trees	1000
Gene trees	1
Number of leaves	25 + an outgroup
Ingroup divergence to the ingroup ratio	1.0
Generations	LogN(21.25,0.2)
Haploid effective population size	4.7e+8
Global substitution rate	LogN(-21.9,0.1)
Lineage specific rate gamma shape	LogN(1.5,1)
Gene family specific rate gamma shape	LogN(1.551533,0.6931472)
Gene tree branch specific rate gamma shape	LogN(1.5,1)
Duplication rate	4.9e-10
Loss rate to duplication rate ratio	1
Seed	9644
Sequence length	500, 100
Sequence base frequencies	Dirichlet(A=36,C=26,G=28,T=32)
Sequence transition rates	Dirichlet(TC=16,TA=3,TG=5,CA=5,CG=6,AG=15)
Controlling Duplication and Loss Rates (5×4 conditions)	
Duplication rate	4.9e-10, 2.7e-10, 1.9e-10, 5.2e-11, 0
Loss rate to duplication rate ratio	1, 0.5, 0.1, 0
Controlling Duplicatoin and ILS Rate (3×4 conditions)	
Duplication rate	4.9e-10, 1.9e-10, 0
Haploid effective population size	4.7e+8, 1.9e+8, 4.8e+7, 1e+4
Controlling n	
Number of leaves	10, 25, 100, 250, 500 + an outgroup
Controlling k	
Locus trees	25, 100, 250, 1000, 2500, 10000

4.D Supplementary Figures and Tables

	1st	2nd	3rd	4th
MulRF	42	67	10	1
DupTree	28	8	15	69
A-Pro	105	14	1	0
ASTRAL-multi	12	14	71	23

Table S4.2. Rank of methods on S100 dataset over all 120 test conditions. Ranks are obtained using mean species tree error, rounded to two significant digits to create tie for cases where error values are extremely close.

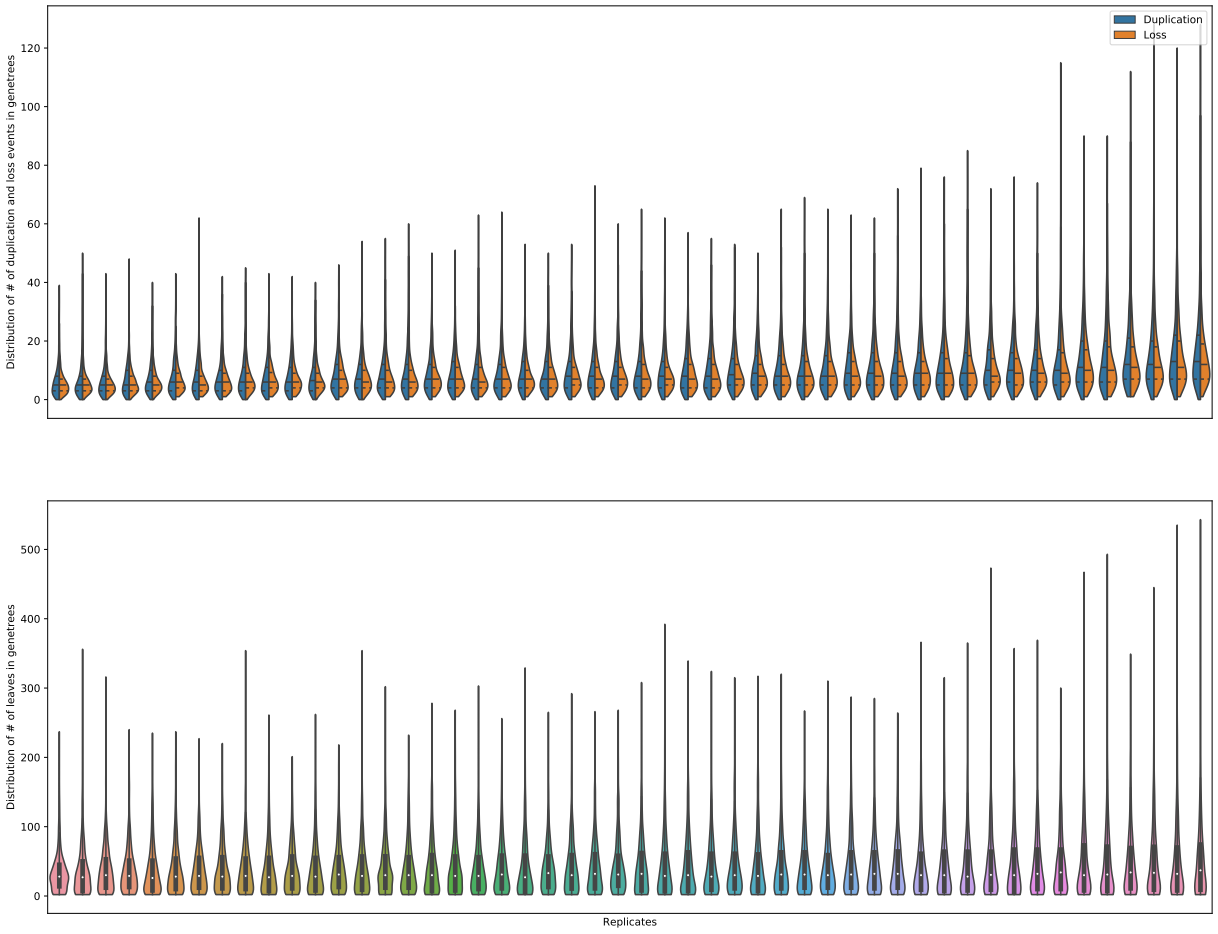

Figure S4.1. Distribution of the number of duplication events, loss events and sizes of leaf set for gene trees in the default condition by replicates. The figure on the top is sorted by the mean number of duplication events, and the figure on the bottom is sorted by mean leaf set size.

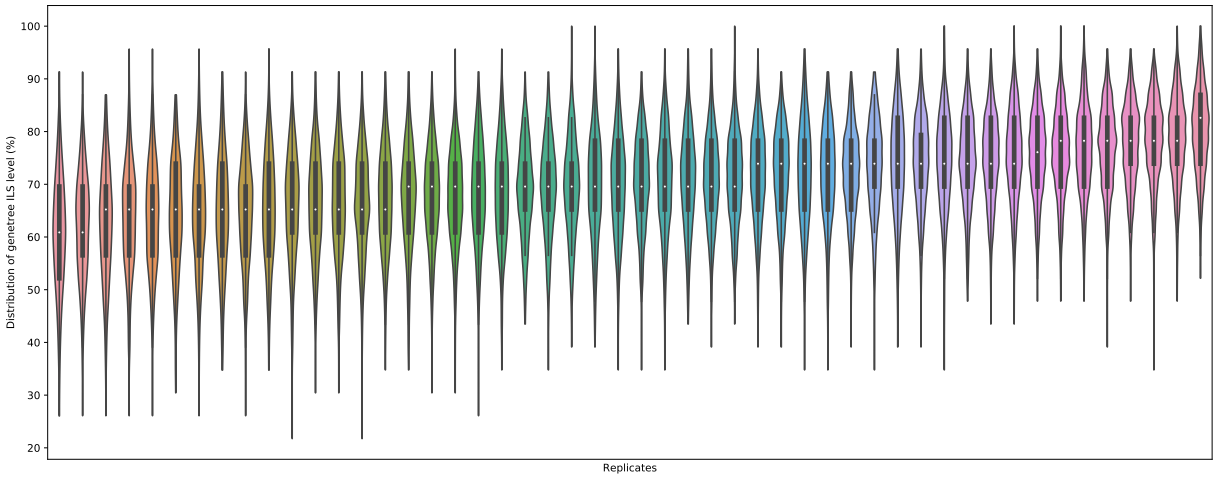

Figure S4.2. Distribution of gene tree ILS, as measured by the normalized RF distance between true gene trees and the true species, in the condition with all default parameters but $\lambda_+ = \lambda_- = 0$. Results are divided by replicates, sorted by mean ILS level.

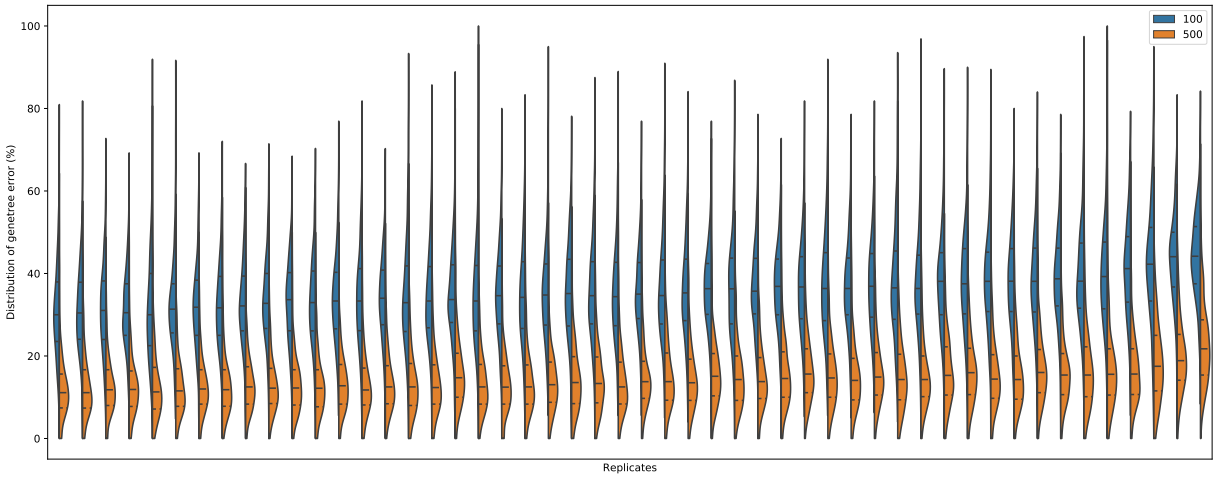

Figure S4.3. Distribution of the gene tree errors (normalized RF distance between true gene trees and the estimated gene tree) for inferred trees with at least 14 leaves in the default condition. Results are divided by sequence length (100bps or 500bps) and by replicates, sorted by mean gene tree error of the 100bps condition.

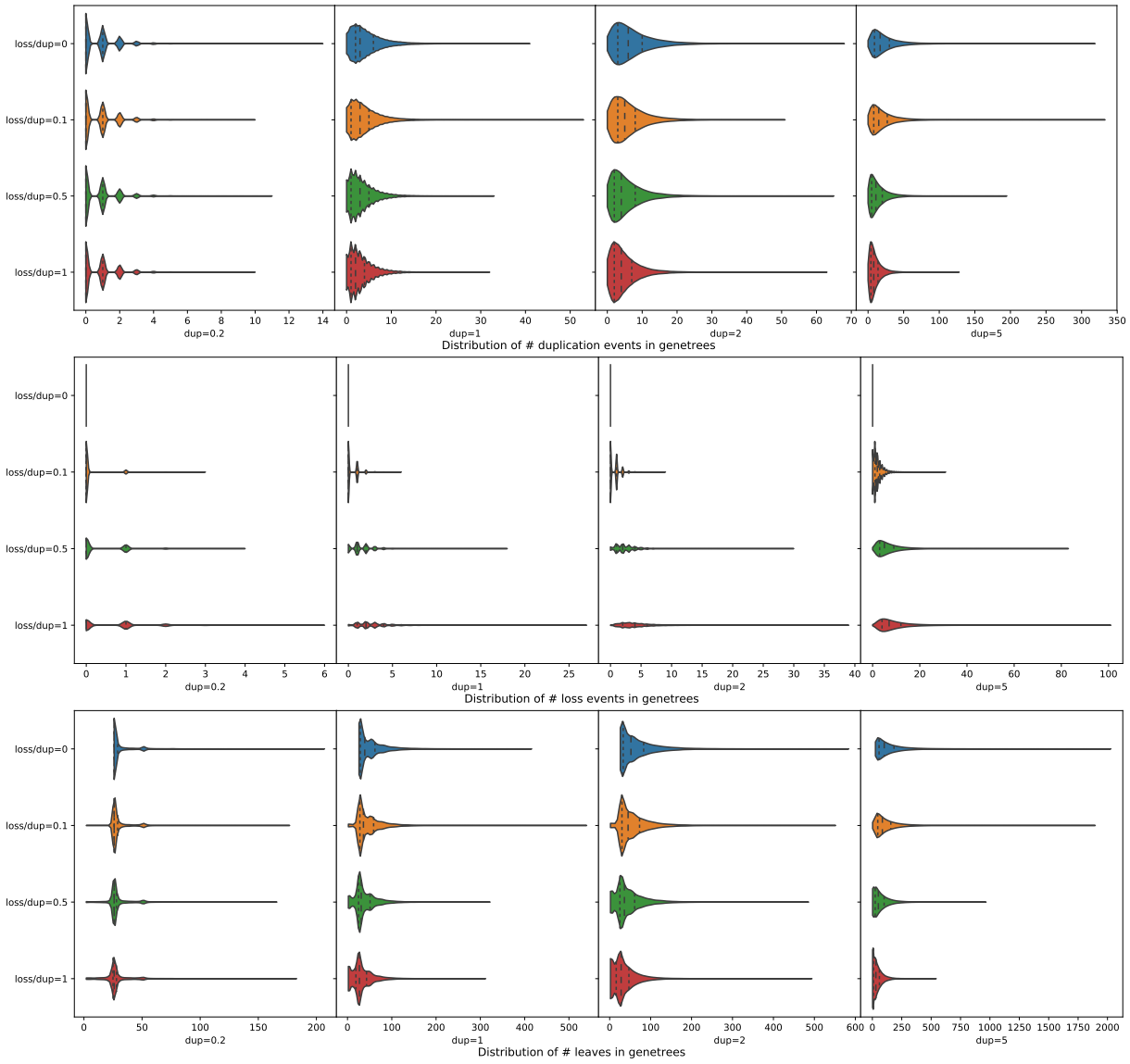

Figure S4.4. Distribution of the number of duplication events, loss events and sizes of leaf set for gene trees of each replicate sorted by duplication and loss rate.

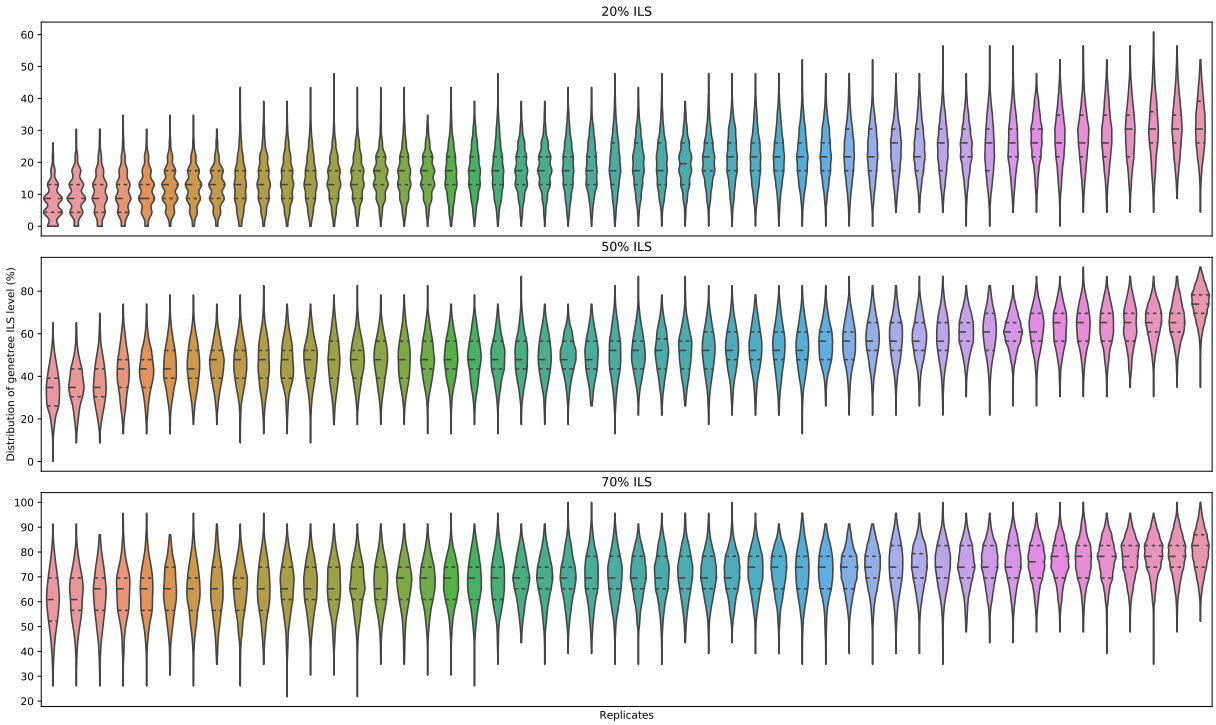

Figure S4.5. Distribution of gene tree ILS levels by replicates and expected ILS level, sorted by mean ILS level.

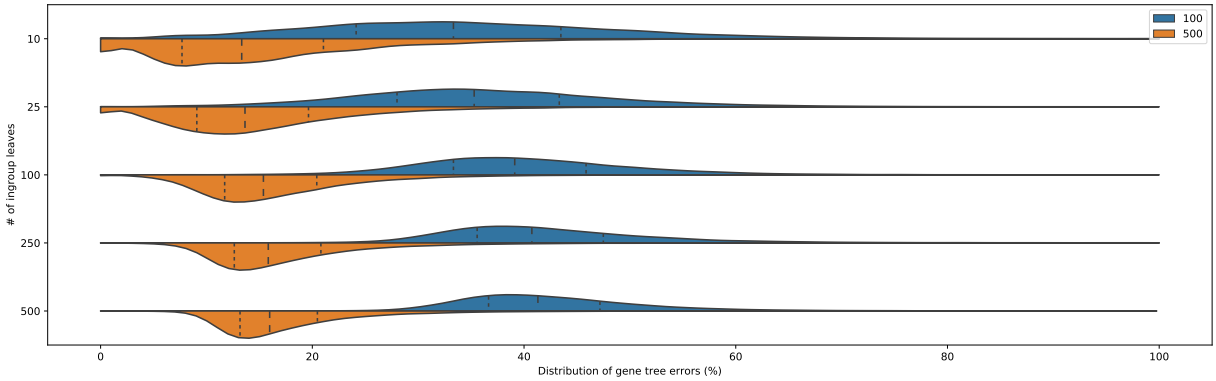

Figure S4.6. Distribution of gene tree errors by the number of in-group species *n*.

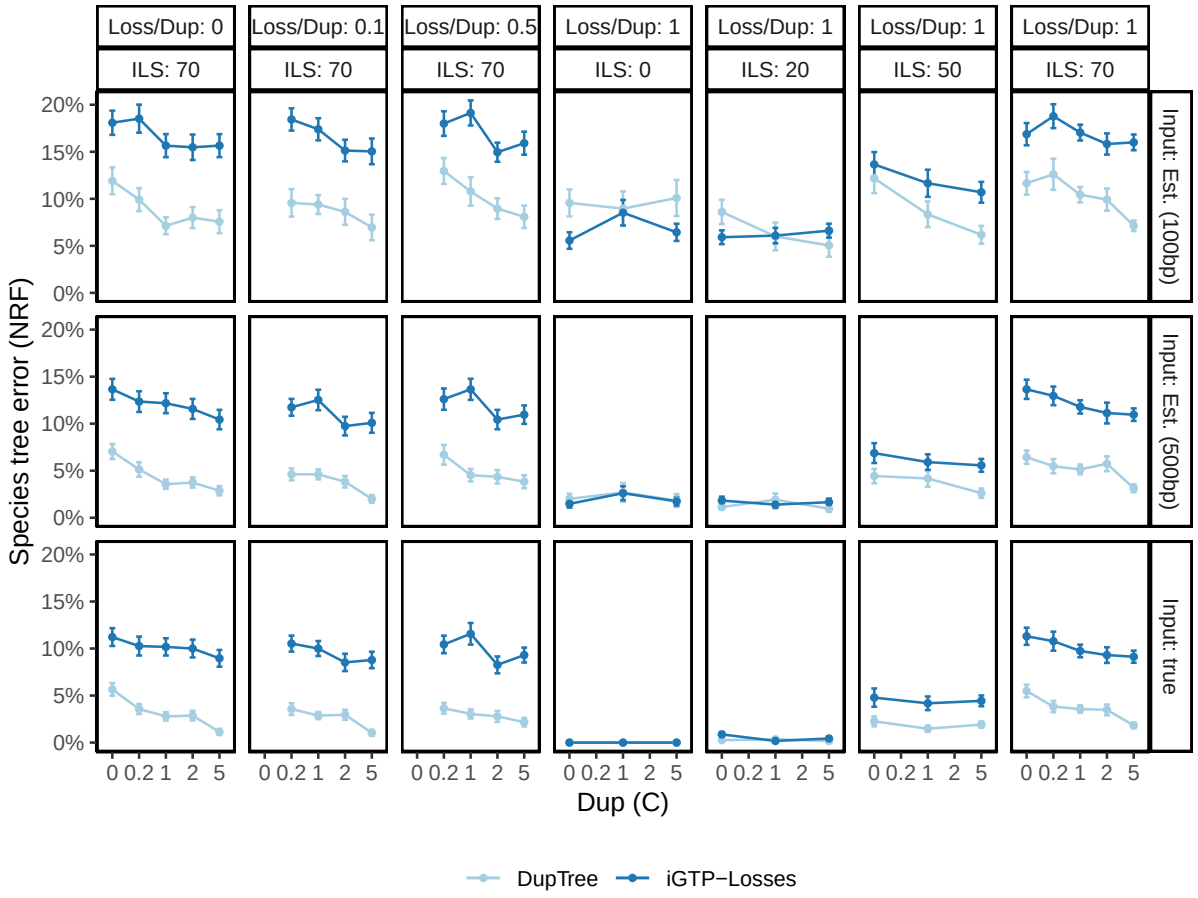

Figure S4.7. Comparison of DupTree and iGTP-DupLoss methods on all the datasets with $n = 25$ and $k = 1000$. DupTree dominates iGTP-DupLoss in most conditions.

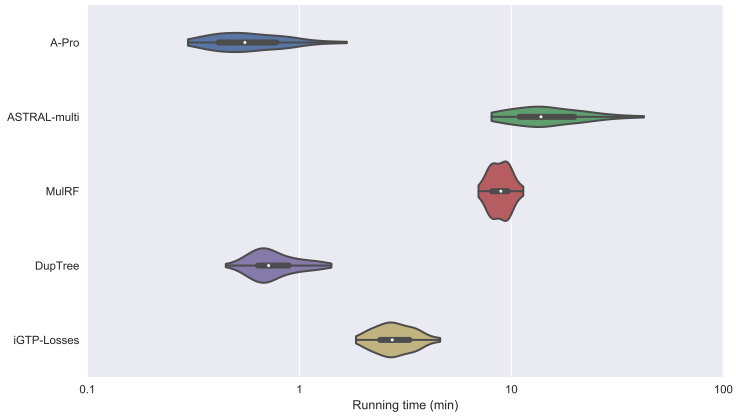

Figure S4.8. Comparing running times, measured on the default model condition, with estimated gene trees (100bp). All methods are run in the single-threaded mode, on the same machine with Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz.

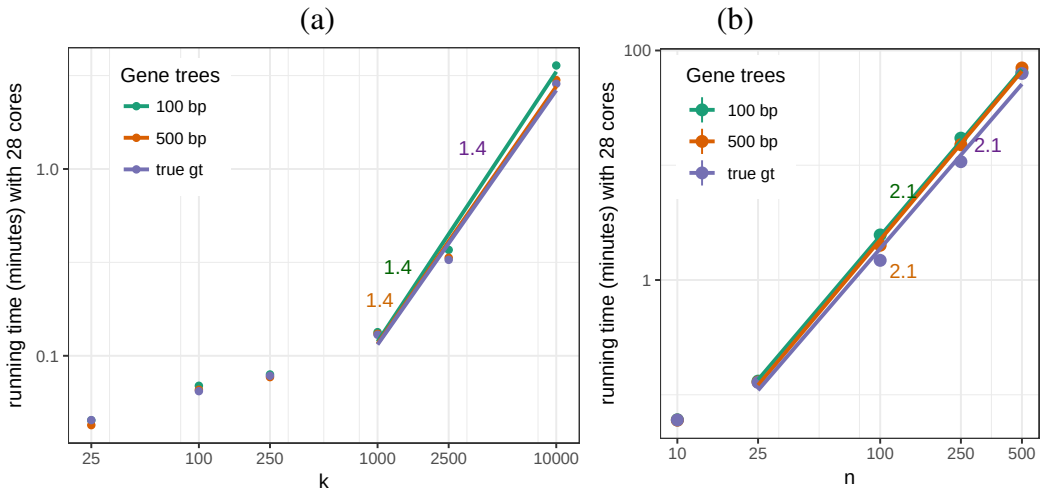

Figure S4.9. The running time of A-Pro versus k (a) and n (b). We fit a line to the log-log plot of the running time only for $k \geq 1000$ and $n \geq 25$ as smaller runs are too fast to be reliable. We empirically estimate the A-Pro running time to grow roughly proportionally with $k^{3/2}$ and n^2.

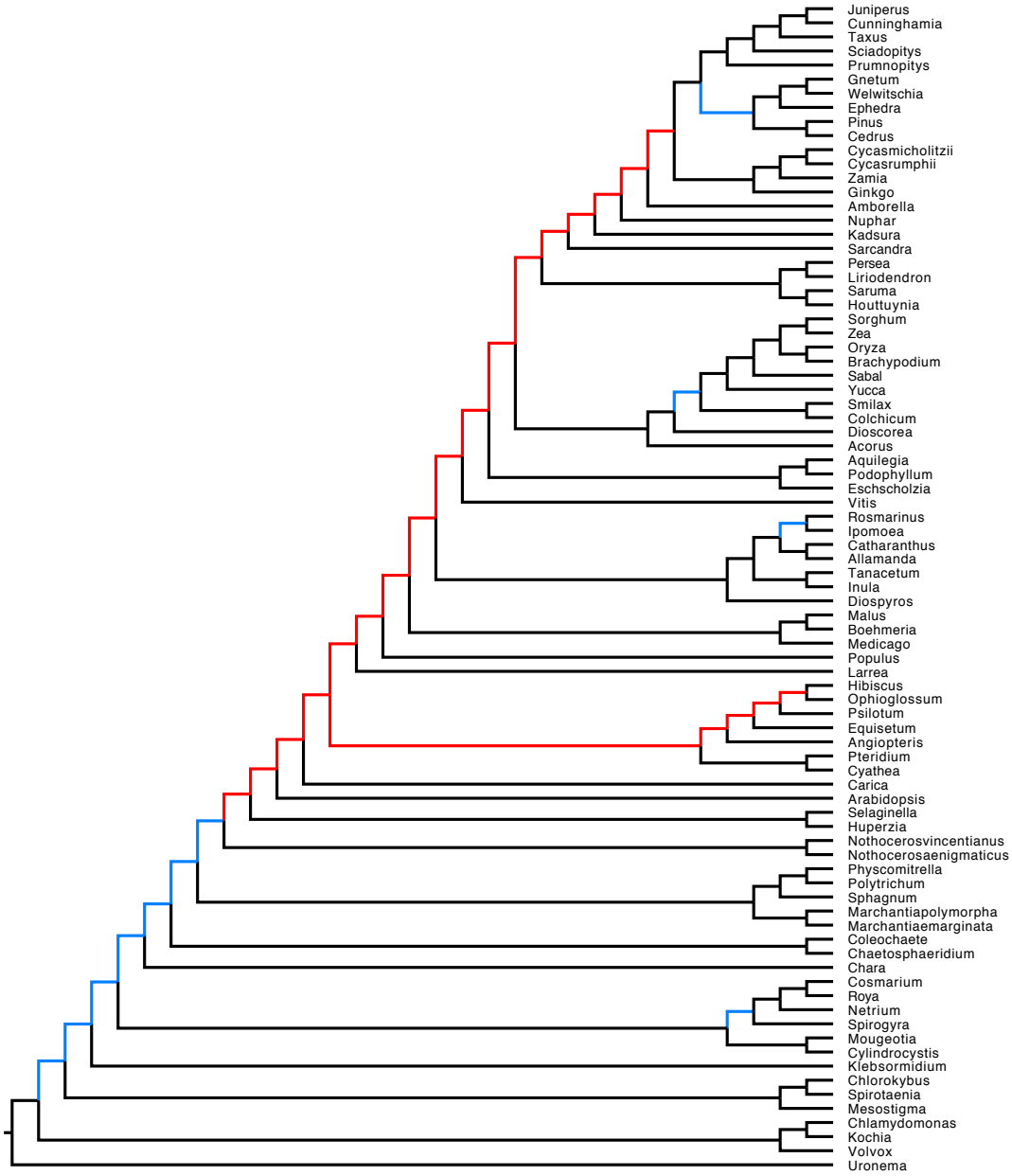

Figure S4.10. DupTree on biological plant dataset. DupTree is run on 9683 multi-copy gene trees available online (Matasci et al., 2014) for the plant dataset. Red: Branches that are obviously wrong, because these branches contradict basic biological categorization. Blue: Branches that contradict ASTRAL on single-copy genes that are not so obviously wrong.

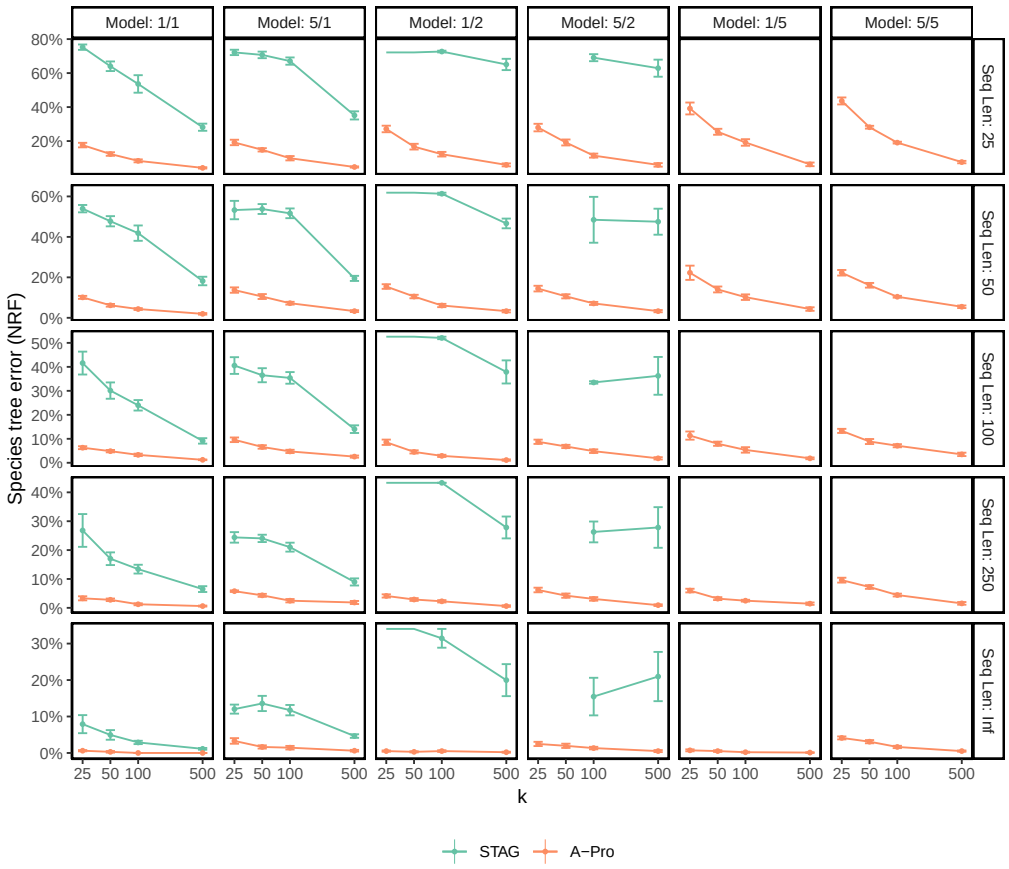

Figure S4.11. Species tree error on S100 dataset. We compare the species tree error of the STAG method to A-Pro, showing mean and standard error over 10 replicates for each model condition, with varying numbers of genes (k) and sequence lengths (with Inf signifying true gene trees). Model conditions are labeled as a/b where a is the level of ILS (1 or 5) and b is the duplication/loss rate (1, 2, or 5). Cases with missing STAG results are due to STAG failing to run on those model conditions. Note that STAG infers a species tree from the input gene trees that have at least one leaf representing each species of interest; if none of the input gene trees satisfy this requirement, then STAG fails to return a tree.

Chapter 5

ASTERISK: Species Tree Inference from Site Patterns under the Multispecies Coalescent Despite Molecular Clock

5.1 Introduction

Species tree inference is difficult because the evolutionary histories of different parts of the genome may not be the same (Maddison, 1997). One cause of the discordance is incomplete lineage sorting (ILS), often modelled by the multi-species coalescent (MSC) process (Kingman, 1982). The standard approach for species tree inference used to be concatenation – concatenating the multiple sequence alignments (MSA) from different parts of the genome and estimating a tree from the concatenated MSA, ignoring ILS. However, concatenation can return incorrect trees with high confidence under MSC model, which is demonstrated by Kubatko and Degnan (2007) and proven by Roch and Steel (2015). Since then, many ILS-aware species tree inference methods have been developed, and they generally follow one of two approaches – the two-step approach and the direct approach.

The two-step approach first infers a gene tree from each part of the genome and uses a summary method – such as MP-EST (Liu et al., 2010), ASTRAL (Mirarab et al., 2014), and ASTRID (Vachaspati and Warnow, 2015) – to obtain a species tree from inferred gene trees. The two-step approach has a major shortcoming: error in inferred gene trees can be carried over to the output species tree (Degiorgio and Degnan, 2014; Molloy and Warnow, 2018; Lanier and Knowles, 2015). Roch et al. (2019) prove that, in theory, the two-step approach returns incorrect trees with high confidence under pathological examples even in the absence of much true gene tree discordance; Jarvis et al. (2014) demonstrate that, in practice, the two-step approach produces species trees contradicting the well-established relationships on an order-level avian phylogenomic dataset. Recently, we proposed weighted ASTRAL (Zhang and Mirarab, 2022), a summary method utilizing the branch lengths and supports from input gene trees, to reduce the impact of gene tree errors. However, there is still a gap in accuracy between weighted ASTRAL and concatenation under low levels of ILS and high levels of gene tree errors.

Alternatively, the direct approach – including BEST (Liu, 2008), *BEAST (Heled and Drummond, 2010), SNAPP (Bryant et al., 2012), SVDQuartet (Chifman and Kubatko, 2014),

QuCo (Rabiee and Mirarab, 2022) – directly infers the species tree from MSA, either sidestepping gene trees or co-estimating species tree and gene trees at the same time. However, these direct methods have not been scalable enough to handle the quickly growing data sizes, despite some more recent efforts in improving their scalability (Ogilvie et al., 2017; Vachaspati and Warnow, 2018; Zhang et al., 2020). Some of these methods are slow because they rely on MCMC sampling under complex models with many parameters. However, others, such as SVDQuartets and QuCo depend on statistics at the quartet level. For these quartet-based site-based methods, a main reason for their lack of scalability is that they rely on first optimizing each quartet and then summarizing quartets to get the final tree. Even though they can sub-sample quartets, such subsampling can create trade-offs between accuracy and running time.

A quartet-based method, however, does not have to examine quartets separately. Each site in an alignment partitions taxa into multiple groups based on what letter they include. We can count all the quartet topologies implied that by partition at the same time using simple combinatorics instead of iterating through all quartets. With this simple technique (which underlines the algorithms such as ASTRAL), an optimization problem defined over all quartets can be solved very efficiently without iterating over all quartets.

Here, we introduce Accurate Species Tree EstimatoR from Individual Site Kernels (ASTERISK) for direction species tree estimation from multiple sequence alignment. The present work has several new contributions: i) We introduce a new optimization objective computed based on DNA site patterns for a quartet of species; we show that optimizing this objective is a statistically consistent estimator under the MSC+GTR model, even allowing for changes in rate across sites (with some limitations) and no assumption about species tree branch lengths (including no assumption of ultrametricity). We call this optimization function for each quartet a site kernel. ii) We design a scalable algorithm to optimize the total quartet site kernels for all quartets across all sites. iii) We propose various modifications to the definition of the quartet site kernel suitable for different models of evolution. The simplest We tested ASTERISK on a simulated dataset. It shows that ASTERISK is on average significantly better than concatenation,

and in all experimental conditions ASTERISK is no worse than concatenation. With abundant genes, the performance of ASTERISK on average levels that of weighted ASTRAL, and which method is more accurate depends highly on experimental conditions.

5.2 Method

We start with describing the model we assume for generating sequence data. We then describe the objective function used in ASTERISK and prove it consistent under the generator model described. We then briefly discuss how the optimization score is optimized in ASTERISK. We end by describing the settings used in our experimental analyses.

5.2.1 Models

We assume a model that generates gene trees under the MSC model and generates sequences under the general time-reversible (GTR) model. We use MSC in the standard fashion. However, our use of the sequence evolution model has an uncommon feature: we allow most of the GTR parameters to change across sites of the same gene, requiring only that pairs of sites share the same parameter.

Coalescent Model

Let T denote the set of species and $n = |T|$.

Definition 5.1. The true species tree is a binary tree $\mathbf{S} = (V_{\mathbf{S}}, E_{\mathbf{S}}, \phi_{\mathbf{S}}, \tau_{\mathbf{S}})$, where $V_{\mathbf{S}}$ is the vertex set of $\mathbf{S}$, $E_{\mathbf{S}}$ is the edge set of $\mathbf{S}$, $\phi_{\mathbf{S}}$ maps T to leaf vertices of $\mathbf{S}$, and $\tau_{\mathbf{S}}$ maps $E_{\mathbf{S}}$ to their branch lengths in coalescent units (CU).

We assume the root branch has branch length $+\infty$. For each species $a \in T$, for simplicity, we may use a to refer to the leaf node $\phi_S(a)$.

Definition 5.2. A gene tree is a binary tree $\mathbf{G} = (V_{\mathbf{G}}, E_{\mathbf{G}}, \phi_{\mathbf{G}}, \tau_{\mathbf{G}})$, where $V_{\mathbf{G}}$ is the vertex set of $\mathbf{G}$, $E_{\mathbf{G}}$ is the edge set of $\mathbf{S}$, $\phi_{\mathbf{S}}$ maps T to leaf vertices of $\mathbf{G}$, and $\tau_{\mathbf{G}}$ maps $E_{\mathbf{G}}$ to their branch lengths in CU.

Let $\mathscr{G}$ be the set of gene trees and $|\mathscr{G}| = k$. Each gene tree $\mathbf{G}_i \in \mathscr{G}$ is generated from the true species tree $\mathbf{S}$ under the multi-species coalescent (MSC) process. Let $\Sigma_i = \{(e,\tau) : \tau \in [0, \tau_{\mathbf{S}}(e)], e \in E_{\mathbf{S}}\}$ denote a set of $|T|-1$ coalescent events generated under MSC. Each (e,τ) denotes a coalescent event happening at τ CU above the child node (target vertex) of e. Coalescence events among any two out of i lineages happen with a rate of $\binom{i}{2}$ per CU. Given that a coalescent event (e,τ) happens, every one of $\binom{i}{2}$ pairs of lineages coalesces with an equal probability $\frac{1}{\binom{i}{2}}$ independent of (e,τ). Since $(\mathbf{S},\mathbf{G}_i)$ uniquely determines Σ_i (but not the other way around), the probability density function (PDF) of $\mathbf{G}_i$ is

$$f(\mathbf{G}_i) = P(\mathbf{G}_i|\Sigma_i) f(\Sigma_i) \; . \tag{5.1}$$

Sequence Evolution Model

In our model, *pairs* of sites evolve on the gene tree. Let us assume the number of site pairs L_i in each gene $\mathbf{G}_i$ is a known parameter. All sites follow the GTR substitution model and all sites in each gene share the same equilibrium frequencies $(\pi^i_A, \pi^i_C, \pi^i_G, \pi^i_T)$. However, substitution rates can vary among pairs. Each site pair ζ^i_j in $\mathbf{G}_i$ maps from T to $\{A,C,G,T\}^2$. The two sites in ζ^i_j evolve independently but share the same GTR transition rate matrix M^i_j and mutation rate function μ^i_j. The function μ^i_j translates each edge e on $\mathbf{S}$ from CU to SU and thus captures the combined effects of the effective population size and mutation rates; nevertheless, we refer to μ^i_j as the mutation rate for simplicity. All lineages of $\mathbf{G}_i$ passing through branch e of the species tree share the same mutation rate $\mu^i_j(e)$ for their length overlapping with e. For example, a branch segment from (e,τ_0) to (e,τ_1), where $\tau_0 < \tau_1$, has SU branch length $(\tau_1 - \tau_0)\mu^i_j(e)$.

5.2.2 Objective Function

We assume that we are given data generated according to the model described before and we assume that it is known what sites belong to what genes and what pairs of sites in each gene

follow the same substitution model. In practice, determining pair of sites will need a heuristics algorithm, which we will return to later. We assume $\pi^i_A, \pi^i_C, \pi^i_G, \pi^i_T$ are given. Since the stationary base frequencies are fixed across the gene and across species, it is trivial to estimate them by simply counting the occurrences of letters across each gene.

Let $N = \{A,C,G,T\}$, $X = \{A,G\}$, and $Y = \{C,T\}$. For a gene tree $\mathbf{G}_i$, let $\pi^i_X = \pi^i_A + \pi^i_G$ and $\pi^i_Y = \pi^i_C + \pi^i_T$. Let $\mathbf{1}_S(s)$ be the 0-1 indicator function indicating whether $s \in S$. For example, $\mathbf{1}_{X\times Y}\left(\zeta^5_2(a)\right)$ is 1 if and only if the site pair indexed 2 of gene 5 of species $a \in T$ is either A or G in the first site and either T or C in the second site. We define the weight of a quartet tree topology $ab|cd$ according to a site pair ζ^i_j in gene tree $\mathbf{G}$ as

$$w^i_j(ab|cd) = \frac{1}{8} \sum_{(p,q)\in\{(a,b),(b,a)\}} \sum_{(r,s)\in\{(c,d),(d,c)\}} w^i_j(p,q,r,s) + w^i_j(r,s,p,q)\ , \tag{5.2}$$

where

$$w^i_j(p,q,r,s) = \left(\pi^i_X\pi^i_Y - \mathbf{1}_{X\times N}\left(\zeta^i_j(p)\right)\mathbf{1}_{Y\times N}\left(\zeta^i_j(q)\right)\right)\left(\pi^i_X\pi^i_Y - \mathbf{1}_{N\times X}\left(\zeta^i_j(r)\right)\mathbf{1}_{N\times Y}\left(\zeta^i_j(s)\right)\right)\ . \tag{5.3}$$

Let a quartet Q be a subset of T where $|Q| = 4$. Let $\mathscr{Q} = \{Q : |Q| = 4, Q \subseteq T\}$ denote the set of all quartets. For any tree (or topology) $\mathbf{S}^*$ with taxon set T, let $\mathbf{S}^* \restriction Q$ denote the tree (or topology) $\mathbf{S}^*$ restrict to quartet Q. We define the score of $\mathbf{S}^*$ as

$$W(\mathbf{S}^*) = \sum_{Q\in\mathscr{Q}} \sum_{i=1}^{k} \sum_{j=1}^{L_i} w^i_j(\mathbf{S}^* \restriction Q)\ . \tag{5.4}$$

Theorem 5.1. *The function* $\arg\max_{\mathbf{S}^*} W(\mathbf{S}^*)$ *is a statistically consistent estimator for the unrooted topology of the true species tree* $\mathbf{S}$.

The proof is provided in the Proof section of the appendix.

5.2.3 Remarks

Remark. *Theorem 5.1 is still correct when SU/CU ratio varies along a species tree branch.*

We can redefine the mutation rate model as the following: The function μ_j^i translates each point (e,τ) on $\mathbf{S}$ to substitution-per-site units (SU). All lineages of $\mathbf{G}$ at (e,τ) share the same mutation rate $\mu_j^i(e,\tau)$ SU per CU. For example, a branch segment from (e,τ_0) to (e,τ_1), where $\tau_0 < \tau_1$, has SU branch length

$$\int_{\tau_0}^{\tau_1} \mu_j^i(e,\tau)\mathrm{d}\tau\ .$$

Remark. *Theorem 5.1 can be adopted to not rely on equilibrium frequencies using a modified definition of $w_j^i(ab|cd)$.*

In (5.2) and (5.3), $w_j^i(ab|cd)$ relies on a known $\pi_X^i\pi_Y^i$. If equilibrium frequencies are unknown, we need to redefine each ζ_j^i as a quadruple of sites in $\mathbf{G}_i$ which maps T to $\{A,C,G,T\}^4$, and modify (5.2) as

$$\begin{aligned} w_j^i(ab|cd) =& \mathbf{1}_{X\times N\times N\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{N\times Y\times N\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{N\times N\times X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{N\times N\times N\times Y}\left(\zeta_j^i(a)\right)\\ &-\mathbf{1}_{X\times N\times N\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{Y\times N\times N\times N}\left(\zeta_j^i(b)\right)\mathbf{1}_{N\times N\times X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{N\times N\times N\times Y}\left(\zeta_j^i(a)\right)\\ &-\mathbf{1}_{N\times X\times N\times N}\left(\zeta_j^i(c)\right)\mathbf{1}_{N\times Y\times N\times N}\left(\zeta_j^i(d)\right)\mathbf{1}_{N\times N\times X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{N\times N\times N\times Y}\left(\zeta_j^i(a)\right)\\ &+\mathbf{1}_{X\times N\times N\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{Y\times N\times N\times N}\left(\zeta_j^i(b)\right)\mathbf{1}_{N\times X\times N\times N}\left(\zeta_j^i(c)\right)\mathbf{1}_{N\times Y\times N\times N}\left(\zeta_j^i(d)\right)\ . \end{aligned}$$

In general, equilibrium frequencies can be calculated by simply counting them. However, this new equation is useful when it is difficult to estimate equilibrium frequencies, for example, when the length of genes is very short.

Remark. *Theorem 5.1 can be adopted to a more general case when each site has its own mutation rate and transition matrix, but only under HKY85 model (Hasegawa et al., 1985).*

In this case, each ζ_j^i represents one site in $\mathbf{G}_i$ which maps T to $\{A,C,G,T\}$. We redefine

$$
\begin{aligned}
&w_j^i(ab|cd)\\
&=\Big(\mathbf{1}_X\left(\zeta_j^i(a)\right)\mathbf{1}_X\left(\zeta_j^i(b)\right)\mathbf{1}_Y\left(\zeta_j^i(c)\right)\mathbf{1}_Y\left(\zeta_j^i(d)\right)+\mathbf{1}_Y\left(\zeta_j^i(a)\right)\mathbf{1}_Y\left(\zeta_j^i(b)\right)\mathbf{1}_X\left(\zeta_j^i(c)\right)\mathbf{1}_X\left(\zeta_j^i(d)\right)\Big)\left(\pi_A^i\pi_C^i\right)^2\\
&+\Big(\mathbf{1}_A\left(\zeta_j^i(a)\right)\mathbf{1}_A\left(\zeta_j^i(b)\right)\mathbf{1}_C\left(\zeta_j^i(c)\right)\mathbf{1}_C\left(\zeta_j^i(d)\right)+\mathbf{1}_C\left(\zeta_j^i(a)\right)\mathbf{1}_C\left(\zeta_j^i(b)\right)\mathbf{1}_A\left(\zeta_j^i(c)\right)\mathbf{1}_A\left(\zeta_j^i(d)\right)\Big)\left(\pi_X^i\pi_Y^i\right)^2\\
&-\Big(\mathbf{1}_A\left(\zeta_j^i(a)\right)\mathbf{1}_A\left(\zeta_j^i(b)\right)\mathbf{1}_Y\left(\zeta_j^i(c)\right)\mathbf{1}_Y\left(\zeta_j^i(d)\right)+\mathbf{1}_Y\left(\zeta_j^i(a)\right)\mathbf{1}_Y\left(\zeta_j^i(b)\right)\mathbf{1}_A\left(\zeta_j^i(c)\right)\mathbf{1}_A\left(\zeta_j^i(d)\right)\Big)\left(\pi_X^i\pi_C^i\right)^2\\
&-\Big(\mathbf{1}_X\left(\zeta_j^i(a)\right)\mathbf{1}_X\left(\zeta_j^i(b)\right)\mathbf{1}_C\left(\zeta_j^i(c)\right)\mathbf{1}_C\left(\zeta_j^i(d)\right)+\mathbf{1}_C\left(\zeta_j^i(a)\right)\mathbf{1}_C\left(\zeta_j^i(b)\right)\mathbf{1}_X\left(\zeta_j^i(c)\right)\mathbf{1}_X\left(\zeta_j^i(d)\right)\Big)\left(\pi_A^i\pi_Y^i\right)^2.
\end{aligned}
\tag{5.5}
$$

Note that the same equation can be used for all models embedded withing HKY85, including JC69 (Jukes and Cantor, 1969), K80 (Kimura, 1980), and F81 (Felsenstein, 1981).

Remark. *Theorem 5.1 is still correct when $X|Y$ does not correspond to purines vs. pyrimidines.*

We can let $X|Y$ be any non-trivial bipartition of N (e.g., $X=\{A\}, Y=\{C,G,T\}$). For a gene tree $\mathbf{G}_i$, let $\pi_X^i=\sum_{c\in X}\pi_c^i$ and $\pi_Y^i=1-\pi_X^i$. In fact, we can define the score of $\mathbf{S}^*$ as

$$
W(\mathbf{S}^*)=\sum_{X|Y}\sum_{Q\in\mathscr{Q}}\sum_{i=1}^{k}\sum_{j=1}^{L_i}w_j^i(\mathbf{S}^*\restriction Q;X|Y)\,,
$$

where $w_j^i(\mathbf{S}^*\restriction Q;X|Y)$ is $w_j^i(\mathbf{S}^*\restriction Q)$ given a specific $X|Y$, and $X|Y$ is summed over all seven non-trivial bipartitions of N.

In fact, this objective function can also be applied to amino acid alignments under amino acid GTR model. In such case, $X|Y$ can be any non-trivial bipartition of the 20 amino acids, and (5.4) only sums over a few bipartitions $X|Y$.

5.2.4 Optimization algorithm

ASTERISK adopts the optimization algorithm from weighted ASTRAL (Zhang and Mirarab, 2022). In its simplest "naive" form, the algorithm works as follows: *i*) Starting from an empty tree, add species consecutively in a random order, each at a place maximizing a pre-defined

objective function, to obtain a full tree. ii) Perform several rounds of NNI moves to improve the full tree from step (i). iii) Repeat the previous step for r rounds to obtain a set of tripartitions, each corresponding to an internal node of some tree from step (ii). iv) Using a final dynamic programming (DP) to find an optimal output tree where each internal node of the output tree is constrained to be corresponding to elements of tripartition in step (iii).

For the placement algorithm we have previously presented (Zhang and Mirarab, 2022) for step (i) to work correctly, the objective function $W(\mathbf{S}^*)$ of a species tree topology $\mathbf{S}^*$ needs to satisfy

$$W(\mathbf{S}^*) = \frac{1}{2} \sum_{A|B|C \in \mathscr{T}(\mathbf{S}^*)} W(A|B|C) \ ,$$

where $\mathscr{T}(\mathbf{S}^*)$ denote the set of all tripartitions of T corresponding to internal nodes of $\mathbf{S}^*$. In ASTERISK, from (5.4), for a tripartition $A|B|C$ corresponding to an internal node of $\mathbf{S}^*$,

$$\begin{aligned}
&W(A|B|C) \\
&= \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{L_i} \sum_{a \in A} \sum_{b \in B} \sum_{c \in C} \Big(\sum_{d \in A - \{a\}} w_j^i(ad|bd) + \sum_{d \in B - \{b\}} w_j^i(ac|bd) + \sum_{d \in C - \{c\}} w_j^i(ab|cd) \Big) \\
&= \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{L_i} \Big(w_j^i(A,B,C) + w_j^i(B,C,A) + w_j^i(C,A,B) \Big) \ , \text{ where} \\
&w_j^i(A,B,C) = \sum_{a \in A} \sum_{b \in B} \sum_{c \in C} \sum_{d \in A - \{a\}} w_j^i(ad|bd) \ .
\end{aligned}$$

For each site, ASTERISK defines four counters for partition A:

$$\begin{aligned}
A^{i,j}_{X \times N} = \sum_{a \in A} \mathbf{1}_{X \times N} \left(\zeta_j^i(a) \right), A^{i,j}_{Y \times N} = \sum_{a \in A} \mathbf{1}_{Y \times N} \left(\zeta_j^i(a) \right), \\
A^{i,j}_{N \times X} = \sum_{a \in A} \mathbf{1}_{N \times X} \left(\zeta_j^i(a) \right), A^{i,j}_{N \times Y} = \sum_{a \in A} \mathbf{1}_{N \times Y} \left(\zeta_j^i(s) \right) \ ,
\end{aligned}$$

and similarly, $B^{i,j}_{X \times N}$, $B^{i,j}_{Y \times N}$, $B^{i,j}_{N \times X}$, $B^{i,j}_{N \times Y}$, $C^{i,j}_{X \times N}$, $C^{i,j}_{Y \times N}$, $C^{i,j}_{N \times X}$, $C^{i,j}_{N \times Y}$ for partition B and C. With

these twelve counters computed, $w^i_j(A,B,C)$ can be computed in constant time:

$$
\begin{aligned}
&w^i_j(A,B,C)\\
&=\frac{1}{8}\left(\pi^i_X\pi^i_Y\left(A^{i,j}_{N\times N}\right)^2-A^{i,j}_{X\times N}A^{i,j}_{Y\times N}\right)\left(2\pi^i_X\pi^i_Y B^{i,j}_{N\times N}C^{i,j}_{N\times N}-B^{i,j}_{N\times X}C^{i,j}_{N\times Y}-B^{i,j}_{N\times Y}C^{i,j}_{N\times X}\right)\\
&+\frac{1}{8}\left(\pi^i_X\pi^i_Y\left(A^{i,j}_{N\times N}\right)^2-A^{i,j}_{N\times X}A^{i,j}_{N\times Y}\right)\left(2\pi^i_X\pi^i_Y B^{i,j}_{N\times N}C^{i,j}_{N\times N}-B^{i,j}_{X\times N}C^{i,j}_{Y\times N}-B^{i,j}_{Y\times N}C^{i,j}_{X\times N}\right),
\end{aligned}
$$

where $A^{i,j}_{N\times N}=A^{i,j}_{X\times N}+A^{i,j}_{Y\times N}$, $B^{i,j}_{N\times N}=B^{i,j}_{X\times N}+B^{i,j}_{Y\times N}$, and $C^{i,j}_{N\times N}=C^{i,j}_{X\times N}+C^{i,j}_{Y\times N}$.

The counters are kept and updated during successive steps of the greedy algorithm. During the placement of each species onto the species tree in step (i), each counter is updated $O(n\log n)$ times, and each takes a constant time (see Zhang and Mirarab, 2022, for details) . Thus, the total running time of the optimization algorithm is $O(n^2L\log n)$ where $L=\sum_{i=1}^{k}L_i$. In comparison, for wASTRAL, this was $O(n^2kH\log(n))$, where H is the average height of the gene tree. Zhang and Mirarab (2022) also proposed a two-step optimization algorithm which reduces the time complexity of the ASTERISK optimization algorithm to $O(n^{1.5}L\log^2(n))$ under some additional conditions (which are automatically satisfied with high probability as $k\to\infty$).

5.2.5 Experimental setup

We use the S200 dataset simulated by Mirarab and Warnow (2015) for benchmarking ASTERISK. This dataset has 201 species (200 in-group + 1 out group). The species trees are generated under two different birth rates $(10^{-6},10^{-7})$ and three different tree heights $(10^7, 2\times 10^6, 5\times 10^5$ generations), corresponding to low ILS ($\approx 10\%$), medium ILS ($\approx 35\%$), and high ILS ($\approx 70\%$), respectively. For each species tree, 1000 genes are simulated with gene lengths uniformly drawn between 300 and 1500 bps replicates. 50 replicates are simulated per each condition, but in the results presented here, we use 10 replicates.

The following species tree reconstruction methods are benchmarked: ASTRAL-III (v5.7.4) takes as input estimated gene trees provided by Mirarab and Warnow (2015), which are reconstructed from genes using Fasttree-2. Weighted ASTRAL by hybrid weighting (v1.8.2.3),

wASTRAL-h for short, takes as input estimated gene trees with branch lengths and supports computed using IQ-TREE (v1.6.12) aBayes option (`--abayes`) under GTR+Γ model. CA-ML trees are provided by Mirarab and Warnow (2015), inferred using Fasttree-2 from concatenated genes. ASTERISK-HKY, a variant of ASTERISK using (5.5) as the quartet site kernel, is directly applied to concatenated genes. ASTERISK also uses estimated mutation rates of all sites to pair sites with similar mutation rates to form site-pairs. The estimated mutation rates for sites of each gene are estimated using IqTree `-wsr` option under GTR+Γ model with inferred species tree from ASTERISK-HKY as the topological constraint. All the sites within a gene are sorted according to the rate and each two consecutive sites are paired in that order, leaving the last site out when an odd number of sites are available.

5.3 Results

In this section, we compare the accuracy of ASTERISK against various species tree inference methods on S200 dataset (Figure 5.1). ASTERISK has lower species tree error rates than ASTERISK-HKY in all conditions. The species tree error rate on average is 7.8% RF for ASTERISK and 10.1% RF for ASTERISK-HKY. This difference is statistically significant ($p < 10^{-5}$ according to an ANOVA test with the method, number of genes, and ILS level as independent variables) but likely shrinks with more genes ($p = 0.08$). When focusing on conditions with 1000 genes, on average ASTERISK and ASTERISK-HKY have species tree error rate 3.2% and 4.1%, respectively, and the difference is still significant ($p = 0.02$). Thus, the advantage of using a more complex model does not disappear due to increased dataset size.

ASTERISK, on average, has a lower species tree error rate (7.8% RF) than CA-ML (9.1% RF) does. The difference is significant with $p = 0.02$ and depends on ILS levels ($p = 0.01$) but not on the birth rates ($p = 0.72$), which control whether speciations are closer to the base or tips of the tree. The improvements of ASTERISK over CA-ML are the most pronounced for conditions with 200 genes and high ILS, where the RF error goes down from 16% with CA-ML

to 10% with ASTERISK). In contrast, CA-ML is in no condition better than ASTERISK for more than 1.3%. When focusing on conditions with 1000 genes, on average ASTERISK and CA-ML have species tree error rate 3.2% and 4.5%, respectively. The difference is significant ($p = 0.006$) and probably depends on ILS levels ($p = 0.06$). With low level of ILS, the difference in species tree error rate between ASTERISK (4.5% RF) and CA-ML (4.1% RF) is not significant ($p = 0.50$) and does not depend on the number of genes ($p = 0.52$) or the birth rate ($p = 0.58$).

ASTERISK on average has a higher species tree error rate (7.8% RF) than wASTRAL-h (6.7% RF) does. The difference is significant with $p = 0.01$ but varies significantly with the number of genes ($p = 0.03$), ILS levels ($p = 0.007$), and birth rates ($p = 0.05$). Particularly, with low level of ILS and birth rate$= 10^{-7}$, ASTERISK on average has a lower species tree error rate (4.4% RF) than wASTRAL-h (6.8% RF) does and the difference is significant ($p = 0.05$). When limited to conditions with 1000 genes, ASTERISK and wASTRAL-h have very similar accuracy (3.2% RF vs. 3.3% RF). The performance seems to differ with birth rate ($p = 0.06$). Note that with birth rate$= 10^{-7}$ and 1000 genes, the gene tree error rate of wASTRAL-h decreases with increasing ILS but due to high variance across replicates; however, the pattern is not statistically significant ($p = 0.21$) and may be an artifact of having fewer replicates. ASTRAL-III in general exhibits a similar pattern to wASTRAL-h but with a higher species tree error rate (8.1% RF).

5.4 Discussion

Our simulation result in S200 dataset shows that the accuracy of ASTERISK dominates ASTERISK-HKY in all conditions. Therefore, there is no incentive to prefer ASTERISK-HKY when resources allow the use of ASTERISK, at least under the conditions we simulated. However, we note that the kernel used for ASTERISK-HKY, unlike the ASTERISK kernel, relies on one site, and thus, less strong assumptions. When the boundaries between genes or regions where GTR parameters can be assumed mostly unchanged are hard to predict, the use of the simpler ASTERISK-HKY method may prove to be more robust as it makes fewer assumptions.

Recently, it has become a common practice to infer species tree using both CA-ML and coalescent method (e.g., ASTRAL) and compare the inferred species trees from the two methods, as neither method can dominate the other in all conditions (e.g., population size, mutation rate, and species tree shape) and the inferred trees from the two methods usually contradict to each other in many branches. We suggest replacing CA-ML with ASTERISK – to infer species tree using both ASTERISK and wASTRAL-h and compare the inferred species tree, because the accuracy of ASTERISK is better than CA-ML on average, and in its worst case (low ILS), there is no significant evidence showing ASTERISK has lower accuracy compared to CA-ML. Future work should run ASTERISK vs. CA-ML on more dataset and more replicates to confirm that ASTERISK is no worse than CA-ML in all conditions. Although ASTERISK and wASTRAL-h have very similar accuracy with abundant genes, we do not recommend replacing wASTRAL-h universally because it is not dominated by ASTERISK. We observe that wASTRAL tends to have a better accuracy as ILS level increases with 1000 genes and deep speciation; this unexpected pattern (not shared by other conditions) is not significant due to high variance across 10 replicates tested here. Future work should investigate this condition using more replicates to reduce the artifact of high variance among replicates.

5.5 Acknowledgements

Chapter 5, in full, is currently being prepared for submission for publication of the material. " Zhang, C. & Mirarab, S. Scalable Coalescence-aware Ancestry Reconstruction from Aligned Genomes ." The dissertation author was the primary investigator and author of this material.

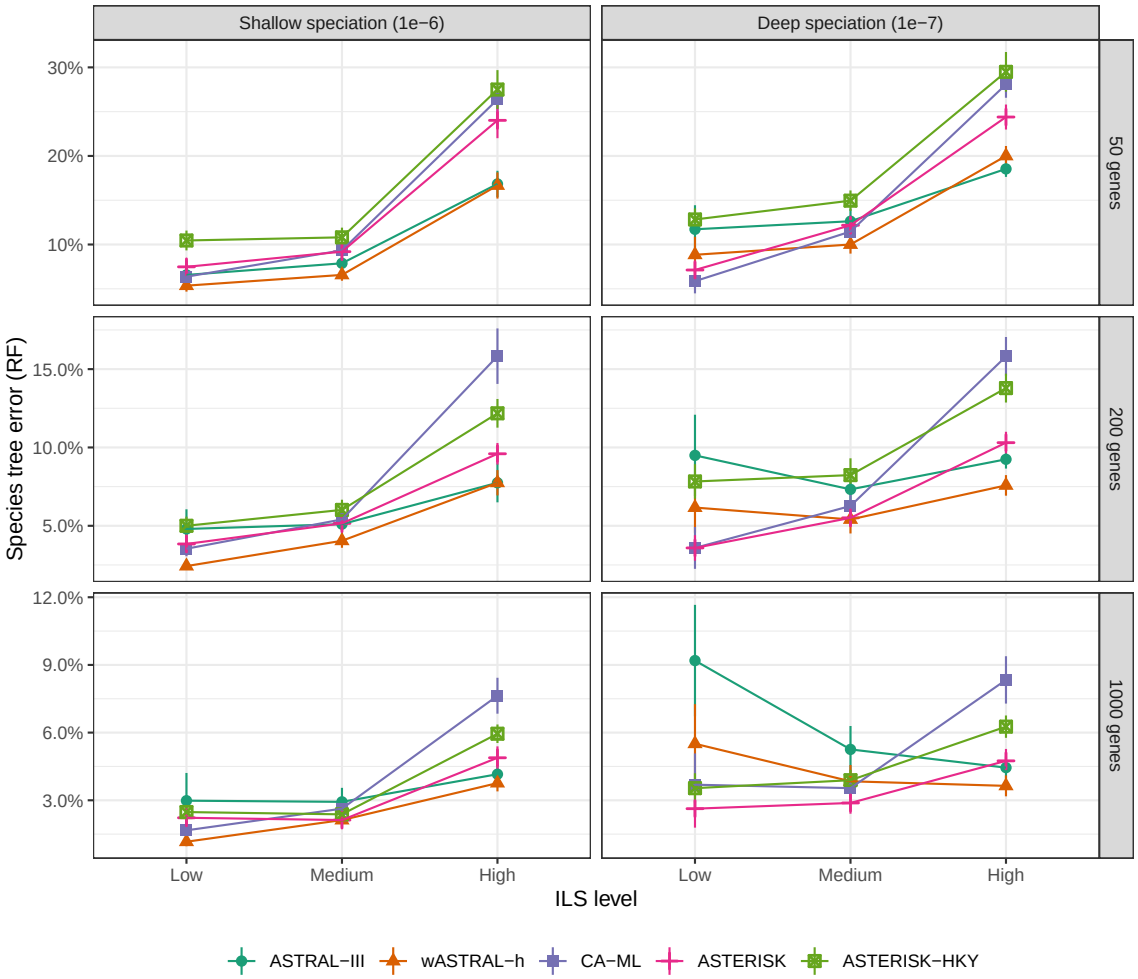

Figure 5.1. A comparison of species tree error (FN) of various reconstruction methods on S200 dataset with $k = \{50, 200, 1000\}$, rates of speciation 1E-6 and 1E-7, and three levels of ILS (Low: 10% RF, Medium, 35% RF, High: 70% RF). Gene trees (for ASTRAL-III and wASTRAL-h) and CA-ML trees are both inferred using FastTree-2. Gene tree branch lengths and supports (for wASTRAL-h) are inferred using IqTree.

Bibliography

D. Bryant, R. Bouckaert, J. Felsenstein, N. A. Rosenberg, and A. Roychoudhury. Inferring Species Trees Directly from Biallelic Genetic Markers: Bypassing Gene Trees in a Full Coalescent Analysis. *Molecular Biology and Evolution*, 29(8):1917–1932, 8 2012. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSS086. URL https://academic.oup.com/mbe/article/29/8/1917/1045283.

J. Chifman and L. Kubatko. Quartet Inference from SNP Data Under the Coalescent Model. *Bioinformatics*, 30(23):3317–3324, 12 2014. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTU530. URL https://academic.oup.com/bioinformatics/article/30/23/3317/206559.

M. Degiorgio and J. H. Degnan. Robustness to Divergence Time Underestimation When Inferring Species Trees from Estimated Gene Trees. *Systematic Biology*, 63(1):66–82, 1 2014. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYT059. URL https://academic.oup.com/sysbio/article/63/1/66/1688532.

J. Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution 1981 17:6*, 17(6):368–376, 11 1981. ISSN 1432-1432. doi: 10.1007/BF01734359. URL https://link.springer.com/article/10.1007/BF01734359.

M. Hasegawa, H. Kishino, and T. a. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution 1985 22:2*, 22(2):160–174, 10 1985. ISSN 1432-1432. doi: 10.1007/BF02101694. URL https://link.springer.com/article/10.1007/BF02101694.

J. Heled and A. J. Drummond. Bayesian Inference of Species Trees from Multilocus Data. *Molecular Biology and Evolution*, 27(3):570–580, 3 2010. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSP274. URL https://academic.oup.com/mbe/article/27/3/570/999753.

E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. Ho, B. C. Faircloth, B. Nabholz, J. T. Howard, A. Suh, C. C. Weber, R. R. Da Fonseca, J. Li, F. Zhang, H. Li, L. Zhou, N. Narula, L. Liu, G. Ganapathy, B. Boussau, M. S. Bayzid, V. Zavidovych, S. Subramanian, T. Gabaldón, S. Capella-Gutiérrez, J. Huerta-Cepas, B. Rekepalli, K. Munch, M. Schierup, B. Lindow, W. C. Warren, D. Ray, R. E. Green, M. W. Bruford, X. Zhan, A. Dixon, S. Li, N. Li, Y. Huang, E. P. Derryberry, M. F. Bertelsen, F. H. Sheldon, R. T. Brumfield, C. V.

Mello, P. V. Lovell, M. Wirthlin, M. P. C. Schneider, F. Prosdocimi, J. A. Samaniego, A. M. V. Velazquez, A. Alfaro-Núñez, P. F. Campos, B. Petersen, T. Sicheritz-Ponten, A. Pas, T. Bailey, P. Scofield, M. Bunce, D. M. Lambert, Q. Zhou, P. Perelman, A. C. Driskell, B. Shapiro, Z. Xiong, Y. Zeng, S. Liu, Z. Li, B. Liu, K. Wu, J. Xiao, X. Yinqi, Q. Zheng, Y. Zhang, H. Yang, J. Wang, L. Smeds, F. E. Rheindt, M. Braun, J. Fjeldsa, L. Orlando, F. K. Barker, K. A. Jønsson, W. Johnson, K. P. Koepfli, S. O'Brien, D. Haussler, O. A. Ryder, C. Rahbek, E. Willerslev, G. R. Graves, T. C. Glenn, J. McCormack, D. Burt, H. Ellegren, P. Alström, S. V. Edwards, A. Stamatakis, D. P. Mindell, J. Cracraft, E. L. Braun, T. Warnow, W. Jun, M. T. P. Gilbert, and G. Zhang. Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331, 12 2014. ISSN 10959203. doi: 10.1126/SCIENCE.1253451/SUPPL{_}FILE/JARVIS.SM.PDF. URL https://www.science.org/doi/10.1126/science.1253451.

T. H. Jukes and C. R. Cantor. Evolution of Protein Molecules. *Mammalian Protein Metabolism*, pages 21–132, 1 1969. doi: 10.1016/B978-1-4832-3211-9.50009-7. URL https://linkinghub.elsevier.com/retrieve/pii/B9781483232119500097.

M. Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution 1980 16:2*, 16 (2):111–120, 6 1980. ISSN 1432-1432. doi: 10.1007/BF01731581. URL https://link.springer.com/article/10.1007/BF01731581.

J. F. Kingman. The coalescent. *Stochastic Processes and their Applications*, 13(3):235–248, 9 1982. ISSN 0304-4149. doi: 10.1016/0304-4149(82)90011-4.

L. S. Kubatko and J. H. Degnan. Inconsistency of Phylogenetic Estimates from Concatenated Data under Coalescence. *Systematic Biology*, 56(1):17–24, 2 2007. ISSN 1063-5157. doi: 10.1080/10635150601146041. URL https://academic.oup.com/sysbio/article/56/1/17/1658327.

H. C. Lanier and L. L. Knowles. 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, 2 2015. ISSN 1095-9513. doi: 10.1016/J.YMPEV.2014.10.022. URL https://pubmed.ncbi.nlm.nih.gov/25450097/.

L. Liu. BEST: Bayesian estimation of species trees under the coalescent model. *Bioinformatics*, 24(21):2542–2543, 11 2008. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTN484. URL https://academic.oup.com/bioinformatics/article/24/21/2542/192785.

L. Liu, L. Yu, and S. V. Edwards. A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC Evolutionary Biology*, 10(1):1–18, 10 2010. ISSN 14712148. doi: 10.1186/1471-2148-10-302/TABLES/2. URL https://bmcecolevol.biomedcentral.com/articles/10.1186/1471-2148-10-302.

W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 9 1997. ISSN

1063-5157. doi: 10.1093/SYSBIO/46.3.523. URL https://academic.oup.com/sysbio/article/46/3/523/1651369.

S. Mirarab and T. Warnow. ASTRAL-II: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12):i44–i52, 6 2015. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTV234. URL https://academic.oup.com/bioinformatics/article/31/12/i44/215524.

S. Mirarab, R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. ASTRAL: genome-scale coalescent-based species tree estimation. *Bioinformatics*, 30(17):i541–i548, 9 2014. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTU462. URL https://academic.oup.com/bioinformatics/article/30/17/i541/200803.

E. K. Molloy and T. Warnow. To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303, 3 2018. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYX077. URL https://academic.oup.com/sysbio/article/67/2/285/4159193.

H. A. Ogilvie, R. R. Bouckaert, and A. J. Drummond. StarBEAST2 Brings Faster Species Tree Inference and Accurate Estimates of Substitution Rates. *Molecular Biology and Evolution*, 34(8):2101–2114, 8 2017. ISSN 0737-4038. doi: 10.1093/MOLBEV/MSX126. URL https://academic.oup.com/mbe/article/34/8/2101/3738283.

M. Rabiee and S. Mirarab. QuCo: quartet-based co-estimation of species trees and gene trees. *Bioinformatics*, 38(Supplement_1):i413–i421, 6 2022. ISSN 1367-4803. doi: 10.1093/BIOINFORMATICS/BTAC265. URL https://academic.oup.com/bioinformatics/article/38/Supplement_1/i413/6617531.

S. Roch and M. Steel. Likelihood-based tree reconstruction on a concatenation of aligned sequence data sets can be statistically inconsistent. *Theoretical population biology*, 100C: 56–62, 3 2015. ISSN 1096-0325. doi: 10.1016/J.TPB.2014.12.005. URL https://pubmed.ncbi.nlm.nih.gov/25545843/.

S. Roch, M. Nute, and T. Warnow. Long-Branch Attraction in Species Tree Estimation: Inconsistency of Partitioned Likelihood and Topology-Based Summary Methods. *Systematic Biology*, 68(2):281–297, 3 2019. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYY061. URL https://academic.oup.com/sysbio/article/68/2/281/5104882.

P. Vachaspati and T. Warnow. ASTRID: Accurate species TRees from internode distances. *BMC Genomics*, 16(10):1–13, 10 2015. ISSN 14712164. doi: 10.1186/1471-2164-16-S10-S3/FIGURES/14. URL https://bmcgenomics.biomedcentral.com/articles/10.1186/1471-2164-16-S10-S3.

P. Vachaspati and T. Warnow. SVDquest: Improving SVDquartets species tree estimation using

exact optimization within a constrained search space. *Molecular Phylogenetics and Evolution*, 124:122–136, 7 2018. ISSN 1055-7903. doi: 10.1016/J.YMPEV.2018.03.006.

C. Zhang and S. Mirarab. Weighting by Gene Tree Uncertainty Improves Accuracy of Quartet-based Species Trees. *bioRxiv*, page 2022.02.19.481132, 7 2022. doi: 10.1101/2022.02.19.481132. URL https://www.biorxiv.org/content/10.1101/2022.02.19.481132v2https://www.biorxiv.org/content/10.1101/2022.02.19.481132v2.abstract.

C. Zhang, J. P. Huelsenbeck, and F. Ronquist. Using Parsimony-Guided Tree Proposals to Accelerate Convergence in Bayesian Phylogenetic Inference. *Systematic Biology*, 69(5): 1016–1032, 9 2020. ISSN 1063-5157. doi: 10.1093/SYSBIO/SYAA002. URL https://academic.oup.com/sysbio/article/69/5/1016/5716338.

Appendices

5.A Proof

Theorem 5.1. *The function* $\arg\max_{\mathbf{S}^*} W(\mathbf{S}^*)$ *is a statistically consistent estimator for the unrooted topology of the true species tree* $\mathbf{S}$.

Proof. We start with introducing a lemma:

Lemma 5.1. *If the SU distance between species a and b on site* ζ_j^i *of gene tree* $\mathbf{G}_i$ *is t, and*

$$M_j^i = \begin{bmatrix} -(a_j^i\pi_C^i + b_j^i\pi_G^i + c_j^i\pi_T^i) & a_j^i\pi_C^i & b_j^i\pi_G^i & c_j^i\pi_T^i \\ a_j^i\pi_A^i & -(a_j^i\pi_A^i + d_j^i\pi_G^i + e_j^i\pi_T^i) & d_j^i\pi_G^i & e_j^i\pi_T^i \\ b_j^i\pi_A^i & d_j^i\pi_C^i & -(b_j^i\pi_A^i + d_j^i\pi_C^i + f_j^i\pi_T^i) & f_j^i\pi_T^i \\ c_j^i\pi_A^i & e_j^i\pi_C^i & f_j^i\pi_G^i & -(c_j^i\pi_A^i + e_j^i\pi_C^i + f_j^i\pi_G^i) \end{bmatrix},$$

then

$$\mathbb{E}\left[\mathbf{1}_{X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{Y\times N}\left(\zeta_j^i(b)\right)\middle|\mathbf{G}_i\right] = \pi_X^i\pi_Y^i(1-e^{-C_j^i t}),$$

where $C_j^i = \frac{a_j^i\pi_A^i\pi_C^i + c_j^i\pi_A^i\pi_T^i + d_j^i\pi_G^i\pi_C^i + f_j^i\pi_G^i\pi_T^i}{\pi_X^i\pi_Y^i}$.

Proof. Recall $X = \{A, G\}$, and $Y = \{C, T\}$. Notice that we can reduce M_j^i to a two-by-two

matrix of X and Y, and thus

$$\mathbb{E}\Big[\mathbf{1}_{X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{Y\times N}\left(\zeta_j^i(b)\right)\Big|\mathbf{G}_i\Big] = \begin{bmatrix}\pi_A^i & 0 & \pi_G^i & 0\end{bmatrix} e^{M_j^i t}\begin{bmatrix}0\\1\\0\\1\end{bmatrix}$$

$$= \begin{bmatrix}\pi_X^i & 0\end{bmatrix} e^{\begin{bmatrix}-\pi_Y^i C_j^i & \pi_Y^i C_j^i\\ \pi_X^i C_j^i & -\pi_X^i C_j^i\end{bmatrix} t}\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}\pi_X^i & 0\end{bmatrix}\begin{bmatrix}\pi_X^i + \pi_Y^i e^{-C_j^i t} & \pi_Y^i - \pi_Y^i e^{-C_j^i t}\\ \pi_X^i - \pi_X^i e^{-C_j^i t} & \pi_Y^i + \pi_X^i e^{-C_j^i t}\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix}$$

$$= \pi_X^i \pi_Y^i (1 - e^{-C_j^i t}) .$$

□

A corollary follows:

Corollary 5.1. *Let $Q = \{a,b,c,d\}$ and $\mathbf{G}_i \upharpoonright Q$ has topology $ab|cd$. Let t_j^i be the SU length of the internal branch of $\mathbf{G}_i \upharpoonright Q$ on site ζ_j^i and T_j^i be the total SU length of the terminal branches of $\mathbf{G}_i \upharpoonright Q$ on site ζ_j^i, then*

$$\mathbb{E}\big[w_j^i(ab|cd)\big|\mathbf{G}_i\big] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i}, \mathbb{E}\big[w_j^i(ac|bd)\big|\mathbf{G}_i\big] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i (T_j^i + 2t_j^i)} .$$

Proof. Let t_a, t_b, t_c, t_d be the lengths of terminal branches leading to leaves a,b,c,d, respectively. From Lemma 5.1,

$$\mathbb{E}\Big[\pi_X^i \pi_Y^i - \mathbf{1}_{X\times N}\left(\zeta_j^i(a)\right)\mathbf{1}_{Y\times N}\left(\zeta_j^i(b)\right)\Big|\mathbf{G}_i\Big] = \pi_X^i \pi_Y^i e^{-C_j^i (t_a + t_b)} ,$$

and similarly,

$$\mathbb{E}\Big[\pi_X^i \pi_Y^i - \mathbf{1}_{N\times X}\left(\zeta_j^i(c)\right)\mathbf{1}_{N\times Y}\left(\zeta_j^i(d)\right)\Big|\mathbf{G}_i\Big] = \pi_X^i \pi_Y^i e^{-C_j^i (t_c + t_d)} .$$

Recall that

$$w_j^i(p,q,r,s) = \left(\pi_X^i \pi_Y^i - \mathbf{1}_{X\times N}\left(\zeta_j^i(p)\right)\mathbf{1}_{Y\times N}\left(\zeta_j^i(q)\right)\right)\left(\pi_X^i \pi_Y^i - \mathbf{1}_{N\times X}\left(\zeta_j^i(r)\right)\mathbf{1}_{N\times Y}\left(\zeta_j^i(s)\right)\right). \tag{5.6}$$

Since conditional on $\mathbf{G}_i$, the two sites of ζ_j^i are independent, from (5.6), we get

$$\mathbb{E}\left[w_j^i(a,b,c,d)\middle|\mathbf{G}_i\right] = (\pi_X^i \pi_Y^i e^{-C_j^i(t_a+t_b)})(\pi_X^i \pi_Y^i e^{-C_j^i(t_c+t_d)}) = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i},$$

and similarly,

$$\mathbb{E}\left[w_j^i(a,c,b,d)\middle|\mathbf{G}_i\right] = (\pi_X^i \pi_Y^i e^{-C_j^i(t_a+t_j^i+t_c)})(\pi_X^i \pi_Y^i e^{-C_j^i(t_b+t_j^i+t_d)}) = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i(T_j^i+2t_j^i)}.$$

Recall now that

$$w_j^i(ab|cd) = \frac{1}{8}\sum_{(p,q)\in\{(a,b),(b,a)\}}\ \sum_{(r,s)\in\{(c,d),(d,c)\}} w_j^i(p,q,r,s) + w_j^i(r,s,p,q), \tag{5.7}$$

By symmetry in this equation, we have

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\mathbf{G}_i\right] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i} \text{ and } \mathbb{E}\left[w_j^i(ac|bd)\middle|\mathbf{G}_i\right] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i(T_j^i+2t_j^i)}.$$

□

The corollary above leads to the following lemma:

Lemma 5.2. *Let $Q = \{a,b,c,d\}$, $\mathbf{S} \restriction Q$ has topology $ab|cd$, and let $\Sigma_i^* = \{(e_1,\tau_1),(e_2,\tau_2),(e_3,\tau_3)\}$ be the three coalescent events among a,b,c,d in gene tree $\mathbf{G}_i$; then, for each site pair ζ_j^i,*

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*\right] \geq \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*\right].$$

Proof sketch. Let $p_1 = (e_1, \tau_1), p_2 = (e_2, \tau_2), p_3 = (e_3, \tau_3)$ and w.o.l.g. assume p_1 is above p_2 and p_3. Let $\delta_a, \delta_b, \delta_c, \delta_d, \delta_2, \delta_3$ be the SU distance of a, b, c, d, p_2, p_3 to p_1 in $\mathbf{G}_i$ under μ_j^i, respectively. W.o.l.g., assume $\delta_2 < \delta_3$. Let E^* denote the event that p_1 is on the internal branch of the unrooted version of $\mathbf{G}_i \restriction Q$, and let E_1, E_2, E_3 denote the event that $\mathbf{G}_i \restriction Q$ has topology $ab|cd, ac|bc, ad|bc$, respectively.

Case 1: The event E^* happens.

It is easy to verify that $T_j^i = \delta_a + \delta_b + \delta_c + \delta_d - 2\delta_2 - 2\delta_3$ and $t_j^i = \delta_2 + \delta_3$, regardless of the topology of $\mathbf{G}_i$. It is easy to confirm that for some p_3 that avoid deep coalescence, we have $P(E_1|\Sigma_i^*, E^*) = 1$ and $P(E_2|\Sigma_i^*, E^*) = P(E_3|\Sigma_i^*, E^*) = 0$; for each of these Σ_i^*,

$$\begin{aligned}(\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i} &= \mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, E^*\right] \\ &> \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, E^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, E^*\right] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i (T_j^i + 2t_j^i)} .\end{aligned}$$

For all other p_3 that have deep coalescence, $P(E_1|\Sigma_i^*, E^*) = P(E_2|\Sigma_i^*, E^*) = P(E_3|\Sigma_i^*, E^*) = \frac{1}{3}$, and thus

$$\begin{aligned}\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, E^*\right] &= \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, E^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, E^*\right] \\ &= \frac{1}{3}(\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i} + \frac{2}{3}(\pi_X^i \pi_Y^i)^2 e^{-C_j^i (T_j^i + 2t_j^i)} .\end{aligned}$$

Therefore, for all Σ_i^*,

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, E^*\right] \geq \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, E^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, E^*\right] .$$

Case 2: The event $\bar{E}^*$ happens.

It is easy to verify that $T_j^i = \delta_a + \delta_b + \delta_c + \delta_d - 2\delta_3$ and $t_j^i = \delta_3 - \delta_2$, regardless of the topology of $\mathbf{G}_i$. It is easy to confirm that for some p_3 that avoid deep coalescence, $P(E_1|\Sigma_i^*, \bar{E}^*) =$

1, and thus for these Σ_i^*,

$$(\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i} = \mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, \bar{E}^*\right]$$
$$> \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, \bar{E}^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, \bar{E}^*\right] = (\pi_X^i \pi_Y^i)^2 e^{-C_j^i (T_j^i + 2t_j^i)} .$$

For all other p_3, $P(E_1|\Sigma_i^*, \bar{E}^*) = P(E_2|\Sigma_i^*, \bar{E}^*) = P(E_3|\Sigma_i^*, \bar{E}^*) = \frac{1}{3}$, and thus

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, \bar{E}^*\right] = \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, \bar{E}^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, \bar{E}^*\right]$$
$$= \frac{1}{3}(\pi_X^i \pi_Y^i)^2 e^{-C_j^i T_j^i} + \frac{2}{3}(\pi_X^i \pi_Y^i)^2 e^{-C_j^i (T_j^i + 2t_j^i)} .$$

Therefore, for all Σ_i^*,

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*, \bar{E}^*\right] \geq \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*, \bar{E}^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*, \bar{E}^*\right] .$$

Finally, by combining the two cases, we have

$$\mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*\right] \geq \mathbb{E}\left[w_j^i(ac|bd)\middle|\Sigma_i^*\right] = \mathbb{E}\left[w_j^i(ad|bc)\middle|\Sigma_i^*\right] .$$

□

The lemma above leads to the following proposition:

Proposition 5.1. *Let $Q = \{a,b,c,d\}$ and $\mathbf{S} \restriction Q$ has topology $ab|cd$, then for each site pair ζ_j^i,*

$$\mathbb{E}\left[w_j^i(ab|cd)\right] > \mathbb{E}\left[w_j^i(ac|bd)\right] = \mathbb{E}\left[w_j^i(ad|bc)\right] .$$

Proof. From Equation 5.1, we have

$$\mathbb{E}\left[w_j^i(ab|cd)\right] = \int \mathbb{E}\left[w_j^i(ab|cd)\middle|\Sigma_i^*\right] f(\Sigma_i^*) \mathrm{d}\Sigma_i^* ,$$

$$\mathbb{E}\big[w_j^i(ac|bd)\big] = \int \mathbb{E}\big[w_j^i(ac|bd)\big|\Sigma_i^*\big] f(\Sigma_i^*)\mathrm{d}\Sigma_i^*, \mathbb{E}\big[w_j^i(ad|bc)\big] = \int \mathbb{E}\big[w_j^i(ad|bd)\big|\Sigma_i^*\big] f(\Sigma_i^*)\mathrm{d}\Sigma_i^* .$$

From Lemma 5.2, we have

$$\mathbb{E}\big[w_j^i(ab|cd)\big] \geq \mathbb{E}\big[w_j^i(ac|bd)\big] = \mathbb{E}\big[w_j^i(ad|bc)\big] .$$

Notice that $\mathbb{E}\big[w_j^i(ab|cd)\big] \neq \mathbb{E}\big[w_j^i(ac|bd)\big]$ as when there is no deep coalescence,

$$\mathbb{E}\big[w_j^i(ab|cd)\big|\Sigma_i^*\big] > \mathbb{E}\big[w_j^i(ac|bd)\big|\Sigma_i^*\big] = \mathbb{E}\big[w_j^i(ad|bc)\big|\Sigma_i^*\big] .$$

□

Recall that

$$W(\mathbf{S}^*) = \sum_{Q\in\mathscr{Q}} \sum_{i=1}^{k} \sum_{j=1}^{L_i} w_j^i(\mathbf{S}^* \upharpoonright Q) . \tag{5.8}$$

Then, for any species tree topology $\mathbf{S}^*$ different from $\mathbf{S}$, for each gene $\mathbf{G}_i$, let

$$\Delta_i = \sum_{Q\in\mathscr{Q}} \sum_{j=1}^{L_i} \big(w_j^i(\mathbf{S} \upharpoonright Q) - w_j^i(\mathbf{S}^* \upharpoonright Q)\big), \text{ and thus } W(\mathbf{S}) - W(\mathbf{S}^*) = \sum_{i=1}^{k} \Delta_i .$$

Since $-4L_i|\mathscr{Q}| \leq \Delta_i \leq 4L_i|\mathscr{Q}|$, by Hoeffding's inequality,

$$P\big(W(\mathbf{S}^*) \geq W(\mathbf{S})\big) = P\Big(\sum_{i=1}^{k} \Delta_i - \sum_{i=1}^{k} \mathbb{E}[\Delta_i] \leq -\sum_{i=1}^{k} \mathbb{E}[\Delta_i]\Big) \leq e^{-\frac{2\left(\sum_{i=1}^{k} \mathbb{E}[\Delta_i]\right)^2}{\sum_{i=1}^{k} (8L_i|\mathscr{Q}|)^2}} .$$

From Proposition 5.1, we get $\mathbb{E}\big[\Delta_i\big] > 0$ for all i, and thus $\mathbb{E}\big[\Delta_i\big] = \Theta_k(1)$.

Let $\bar{L}$ be the mean of the series $\{L_i\}$ and assume $\bar{L} = \Theta_k(1)$, then

$$P\big(W(\mathbf{S}^*) \geq W(\mathbf{S})\big) \leq e^{-\frac{2\left(\sum_{i=1}^{k} \mathbb{E}[\Delta_i]\right)^2}{\sum_{i=1}^{k} (8L_i|\mathscr{Q}|)^2}} \leq e^{-\frac{2\left(\sum_{i=1}^{k} \mathbb{E}[\Delta_i]\right)^2}{(8|\mathscr{Q}|)^2 k\bar{L}^2}} = e^{-\Theta\left(\frac{k^2}{k}\right)} = e^{-\Theta(k)} .$$

Thus, we obtain the true species tree with arbitrary high probability as $k \to \infty$. □

Chapter 6

TAPER: Pinpointing Errors in Multiple Sequence Alignments Despite Varying Rates of Evolution

1. Erroneous data can creep into sequence datasets for reasons ranging from contamination to annotation and alignment mistakes and reduce the accuracy of downstream analyses. As datasets keep getting larger, it has become difficult to check multiple sequence alignments visually for errors, and thus, automatic error detection methods are needed more than ever before. Alignment masking methods, which are widely used, remove entire aligned sites and may reduce signal as much as or more than they reduce the noise.

2. The alternative we propose here is a surprisingly under-explored approach: looking for errors in small species-specific stretches of the multiple sequence alignments. We introduce a method called TAPER that uses a novel two-dimensional outlier detection algorithm. Importantly, TAPER adjusts its null expectations per site and species, and in doing so, it attempts to distinguish the real heterogeneity (signal) from errors (noise).

3. Our results show that TAPER removes very little data yet finds much of the error. The effectiveness of TAPER depends on several properties of the alignment (e.g., evolutionary divergence levels) and the errors (e.g., their length).

4. By enabling data clean up with minimal loss of signal, TAPER can improve downstream analyses such as phylogenetic reconstruction and selection detection. Data errors, small or large, can reduce confidence in the downstream results, and thus, eliminating them can be beneficial even when downstream analyses are not impacted.

6.1 Introduction

Multiple sequence alignments used in phylogenetics and other evolutionary analyses are susceptible to errors. The input to phylogenetic inference is often prepared through a long pipeline of several error-prone steps (Fig. 6.1). Together, these steps can leave datasets riddled with many types of errors, called *data pipeline errors* here. For example, contaminating DNA (Simion et al., 2018; Laurin-Lemay et al., 2012; Olson and Hassanin, 2003) and sequencing errors can persist even after the assembly of sequences (Francois et al., 2020; Breitwieser

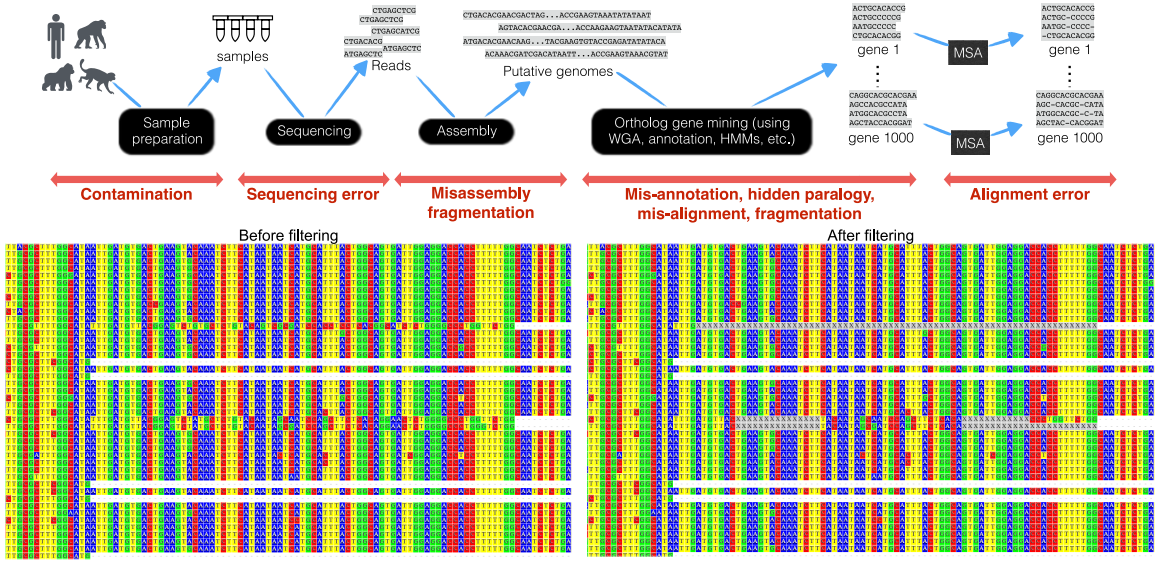

Figure 6.1. Data pipeline errors. Top: Many error-prone steps are needed to produce gene multiple sequence alignments (MSAs) used as input to phylogenomic reconstruction methods. Bottom left: an example data pipeline error in the avian dataset of Jarvis et al. (2014), identified by Springer and Gatesy (2018), where in gene CEPT1, for three species (Phoenicopterus, Mesitornis, Leptosomus) Intron 3 is aligned with exon 4 of other taxa. Bottom right: The result of running TAPER in its default mode. TAPER detects and masks (greyed out and marked by X) most but not all of the mis-aligned parts for these sequences.

et al., 2019). The process for establishing homology using genome annotations, whole genome alignment, or sequence matching involves complex computational problems (Lunter et al., 2008), and thus, errors in homology are not just possible but rampant (Springer and Gatesy, 2018). Most commonly used methods further assume *orthology*, and errors in orthology detection are common (Laurin-Lemay et al., 2012; Salichos and Rokas, 2011). Alignment errors are also ubiquitous and can impact tree accuracy (Li-San Wang et al., 2011; Liu et al., 2009; Ogdenw and Rosenberg, 2006; Fletcher and Yang, 2010; Smirnov and Warnow, 2020). The prevalence of these errors in phylogenomic datasets has been appreciated (Springer and Gatesy, 2018, 2016; Hosner et al., 2016; Sayyari et al., 2017; Philippe et al., 2017; Laurin-Lemay et al., 2012), and several phylogenomics studies have now been criticized (Springer and Gatesy, 2018, 2016; Gatesy and Springer, 2014; Jeffroy et al., 2006; Salichos and Rokas, 2013; Shen et al., 2017). Data pipeline errors represent a major source of that criticism.

Despite the widespread recognition of the challenges associated with data pipeline errors, there are no truly satisfactory ways to address those errors. Researches can not afford to visually curate their phylogenomic datasets. Many authors have mentioned the need for better methods for detecting errors in data (e.g., Philippe et al., 2017). However, detecting and eliminating error comes with its own caveats and issues, and caution is warranted. Aggressive filtering of data upon any suspicion of error can add bias (e.g., by eliminating highly-divergent but genuinely homologous site patterns that should be considered in analyses). Thus, excessive filtering has the potential to eliminate signal. Consistent with this explanation, studies have found that commonly used alignment masking algorithms often have limited impact on accuracy (Portik and Wiens, 2020) and can even reduce the accuracy of phylogeny inference (Tan et al., 2015). Regardless of whether small errors *actually* impact the tree inference, a question on which there is disagreement, the existence of data pipeline errors has the *potential* to impact the results, which can diminish confidence in analyses. Thus, even when errors do not impact the analyses, researchers benefit from detecting and removing them as long as removing the errors does not remove signal. Achieving this objective requires error detection methods that are targeted and find *minimal* portions of the data with putative errors.

The existing methods for data filtering mostly focus on finding entire genes or entire species that should be eliminated (e.g., Hosner et al., 2016; Molloy and Warnow, 2018; Huang et al., 2016). Somewhat more targeted are alignment masking methods that eliminate entire sites from a sequence alignment in order to avoid mis-alignment (e.g., Castresana, 2000; Dress et al., 2008; Capella-Gutiérrez et al., 2009; Rajan, 2012; Steenwyk et al., 2020; Sela et al., 2015). Another form of filtering is to keep all genes and all species but to remove genes from specific species from some gene families because of fragmentation (Sayyari et al., 2017), evidence of unexpected patterns of tree topology and branch length (de Vienne et al., 2012; Wickett et al., 2014; Mai and Mirarab, 2018), or detection of rogue taxa (Westover et al., 2013).

These existing trimming methods operate at coarse levels. Many forms of pipeline errors can be limited to a small stretch of a sequence in a particular species (not all species)

and in a particular set of positions (not an entire gene). For example, the dominant form of error that Springer and Gatesy (2018) found in the avian data of Jarvis et al. (2014) relates to small pieces of introns being annotated as exons (e.g., Fig.6.1b). Such errors are limited to a handful of sequence in a small stretch of sites. Eliminating entire genes, entire sites, or entire species because a small stretch includes errors wastes data. We need methods to find specific stretches of a specific species in a specific gene that appear erroneous. More recently, several authors have started to address the need for finding such small stretches of errors. Whelan et al. (2018) formulated detection of such errors as an step *before* the multiple sequence alignment (MSA) is obtained; their tool, PREQUAL, examines per-position scores in pairwise alignments of sequences to detect non-homologous stretches.

Instead of finding errors before alignment, we can formulate error filtering as outlier detection in a given MSA. Imagine an MSA that is almost fully conserved for all species across all sites except that a small stretch from a single species is close to random with respect to other sequences aligned to it. Such outliers *can* be detected. One would hope alignment methods would leave such sequences unaligned, but most commonly used alignment methods are known to over-align (Löytynoja and Goldman, 2008; Loytynoja and Goldman, 2005). Thus, these stretches are suspect and likely to be erroneous. Avoiding such stretches has motivated some alignment methods to be less aggressive (Loytynoja and Goldman, 2005; Löytynoja et al., 2012; Katoh and Standley, 2016).

Several methods have been recently developed to look for errors in small stretches of MSAs. An early method, DivA (Zepeda Mendoza et al., 2014), used sliding windows and a scoring scheme based on amino acid alignment probabilities. More recently, Spruceup was developed to use pairwise distances in small windows to detect outliers (Borowiec, 2019). Other methods use Hidden Markov Models (HMM) to approach filtering in a probabilistic framework. Divvier (Ali et al., 2019) uses pair-HMMs to quantify the probability of homology between pairs of sequences and Di Franco et al. (2019) use profile HMMs to detect areas of low homology in the entire MSA. Most of these methods tend to be slow and their ability to avoid over-filtering

needs further study. Moreover, these methods are mostly or exclusively designed for amino acids not DNA or RNA data.

Alignment trimming using the outlier detection paradigm needs to contend with two related issues. On the one hand, sequence divergence among species is a function of their phylogenetic relationships and evolutionary rates, and simply being divergent from other sequences cannot be viewed as evidence of error. If one naively looks for species that look unusually divergent compared to the remaining sequences, an outgroup or an ingroup with a highly accelerated rate of evolution would be mistakenly taken as erroneous (a false positive error detection). On the other hand, rates of evolution change across sites, and thus, how much divergence is "normal" depends on the sequence context.

We advocate for two-dimensional (2D) error detection: finding stretches in a sequence that are unusually divergent compared to other sequences, calibrating the *normal* level of divergence based on both genomic positions (columns) *and* the species (rows). An outlier should be detected only if a sequence is unusually divergent along both axes. For tree-based trimming, de Vienne et al. (2012) has pioneered such a two-dimensional approach in their mathematically elegant method Phylo-MCOA, and TreeShrink (Mai and Mirarab, 2018) and TreSpEx (Struck, 2014) follow a similar philosophy.

In this paper, we introduce the Two-dimensional Algorithm for Pinpointing ERrors (TAPER) that takes a multiple sequence alignment as input and outputs outlier sequence positions. Using both simulated and real data, we show that TAPER is able to pinpoint errors in multiple sequence alignments without removing large parts of the alignment.

6.2 Materials and Methods

6.2.1 The TAPER Algorithm

2D Outlier Detection Algorithm

We first describe our general-purpose 2D outlier detection algorithm (Alg. 6.1). The input is a set of n aligned sequences with length L on any arbitrary alphabet Γ (e.g., nucleotide) plus missing data (e.g., gaps). The output is a delineation of each sequence into alternating normal and outlier regions. We use *letter* to refer to a position in a sequence, not counting gaps (or ambiguous letters like `X` for amino acid) as letters.

Step 1.

Compute a divergence score for each letter x in each column i of the alignment. Small scores should indicate agreement with a strong consensus in that site and the largest values should indicate disagreement from an otherwise strong consensus. High scoring letters are candidate outliers. Thus, the drop from large values to small values should be not be gradual; instead, it should have a fast drop as deviation from the consensus weaken. While several such functions can be imagined, here, we use a scoring method that Henikoff and Henikoff (1992) used for sequence weighting. We score a letter x in column i as $\frac{1}{u_i \times p_{i,x}}$ where u_i is the number of unique letters in the column, and $p_{i,x}$ is the fraction of the letters in the column that are x. It can be checked that this score satisfies our criteria (Fig. S6.1).

Step 2.

Since per-column scores are noisy, we combine them along small windows *per each sequence*. We first remove gaps from each sequence. Then, selecting an odd constant value k (e.g., 11), for each overlapping window of size k of each sequence, we assign the median score of the letters in that window as the score of that window. This step produces a distribution of scores for each sequence.

Algorithm 6.1. TAPER algorithm. A: Input alignment on n sequences of length L on alphabet Γ and gap letter -. k, p, q, c: user-provided parameters. $[X]$ denotes $1 \ldots X$.

procedure TAPER(A, k, p, q, c)
  $M \leftarrow$ DIVERGENCESCORES(A)
  for each row $m_i \in M$ **do**
    $\mathbf{v}_i \leftarrow$ WINDOWSCORES(m_i, k)
    $c_i \leftarrow$ JENKSBREAKPOINT($\mathbf{v}_i$)
  $(t_1, \ldots, t_n) \leftarrow$ADJ($(\mathbf{v}_1, \ldots, \mathbf{v}_n)$, $(c_1, \ldots, c_n)$, p, q, c)
  for each row $a_i \in A$ **do**
    $s_i \leftarrow$ REMOVEOUTLIER(a_i, $\mathbf{v}_i$, t_i, k)
  return $(s_1, \ldots, s_n)$

procedure DIVERGENCESCORES(A) ▷ Step 1
  for $i \in [L]$ **do**
    for $x \in \Gamma$ **do**
      $p_x \leftarrow$ relative frequency of x in column i
    $u_i \leftarrow$ number of letters x in Γ with $p_x \neq 0$.
    for $j \in [n]$ **do**

$$M_{j,i} \leftarrow \begin{cases} \emptyset & \text{if } A_{j,i} = - \\ 1/(u_i \times p_{A_{j,i}}) & \text{else} \end{cases}$$

  return M

procedure WINDOWSCORES(m, k) ▷ Step 2
  $m' \leftarrow$ remove $\emptyset$ from m
  $l \leftarrow$ length of m'
  for $i \in [l-k+1]$ **do**
    $\mathbf{v}_i \leftarrow \text{Median}(m'_i, \ldots, m'_{i+k-1})$
  return $\mathbf{v}$

procedure JENKSBREAKPOINT($\mathbf{v}$) ▷ Step 3
  $l \leftarrow$ length of $\mathbf{v}$
  $\mathbf{v}' \leftarrow$ sorted $\mathbf{v}$
  $\mathbf{s}_0 \leftarrow 0$
  for $i \in [l]$ **do**
    $\mathbf{s}_i \leftarrow \mathbf{s}_{i-1} + \mathbf{v}'_i$
  $j \leftarrow \text{argmax}_{i \in [l-1]} \frac{\mathbf{s}_i^2}{i} + \frac{(\mathbf{s}_l - \mathbf{s}_i)^2}{l-i}$
  return $\mathbf{v}'_j$

▷ Step 4

procedure ADJ($(\mathbf{v}_1, \ldots, \mathbf{v}_n)$, $(c_1, \ldots, c_n)$, p, q, c)
  $a \leftarrow \lfloor p \times n \rfloor$-th largest value in $c_1, ..., c_n$
  for $i \in [n]$ **do**
    $l \leftarrow$ length of $\mathbf{v}_i$
    $b_i \leftarrow \lfloor q \times l \rfloor$-th largest value in $\mathbf{v}_i$
    $t_i \leftarrow \max\{a, b_i, c_i, c\}$
  return $(t_1, \ldots, t_n)$

procedure REMOVEOUTLIER(a, $\mathbf{v}$, t, k) ▷ Step 5
  $l \leftarrow$ length of $\mathbf{v}$
  $s \leftarrow$ remove gaps from a
  $N_0 \leftarrow 0$
  $O_0 \leftarrow 0$
  for $i \in [k-1]$ **do**

$$N_i \leftarrow N_{i-1} + \begin{cases} 1, \text{ if } \mathbf{v}_i \leq t \\ 0, \text{ else} \end{cases}$$

$$O_i \leftarrow O_{i-1} + \begin{cases} 0, \text{ if } \mathbf{v}_i \leq t \\ 1, \text{ else} \end{cases}$$

    $B_i^N \leftarrow (N, i-1)$
    $B_i^O \leftarrow (O, i-1)$
  for $i \in \{k, \ldots, l\}$ **do**
    if $N_{i-1} > O_{i-k}$ **then**

$$N_i \leftarrow N_{i-1} + \begin{cases} 1, \text{ if } \mathbf{v}_i \leq t \\ 0, \text{ else} \end{cases}, B_i^N \leftarrow (N, i-1)$$

    else

$$N_i \leftarrow O_{i-k} + \begin{cases} 1, \text{ if } \mathbf{v}_i \leq t \\ 0, \text{ else} \end{cases}, B_i^N \leftarrow (O, i-k)$$

    if $O_{i-1} > N_{i-k}$ **then**

$$O_i \leftarrow O_{i-1} + \begin{cases} 0, \text{ if } \mathbf{v}_i \leq t \\ 1, \text{ else} \end{cases}, B_i^O \leftarrow (O, i-1)$$

    else

$$O_i \leftarrow N_{i-k} + \begin{cases} 0, \text{ if } \mathbf{v}_i \leq t \\ 1, \text{ else} \end{cases}, B_i^O \leftarrow (N, i-k)$$

  if $N_l > O_l$ **then**
    $(S, i) \leftarrow (O, l)$
  else
    $(S, i) \leftarrow (N, l)$
  while $i \neq 0$ **do**
    if $S = O$ **then**
      for $j \in \{i, \ldots, i+k-1\}$ **do**
        $s_j \leftarrow \emptyset$ ▷ Mark position j as outlier
    $(S, i) \leftarrow B_i^S$
  $s' \leftarrow$ add gaps of a back to s
  return s'

Step 3.

For each sequence, we seek to find which of the windows have abnormally high scores; these are considered candidate outliers. To do so, we divide windows of *each sequence* into a low scoring and a high scoring group. We find a cutoff point, t, such that the squared deviations from the mean of the scores below t plus the squared deviations from the mean of points above t is minimized. This approach is the Jenks (1967) natural breaks optimization and is equivalent to

the 2-means clustering.

Step 4.

The initial cutoffs from Step 3 include high-scoring windows for all sequences, regardless of whether any outlier exists. To allow sequences with no outliers, we use three parameters $0 < p, q < 1$ and $c > 1$ to adjust cutoffs. We set the final cutoff of a sequence to the maximum of four quantities: the highest p-quantile of all cutoff values across *all* sequences, the highest q-quantile of all window scores for *that* sequence, the threshold value c, and the initial cutoff t. Thus, the adjusted cutoff will include no high-scoring windows for sequences where the windows have homogeneous scores (controlled by q) or all scores are within the normal range compared to other sequences (controlled by p). User can adjust p, q, and c to control the aggressiveness of the method; p controls how many species can have error while q controls error length; c controls overall aggressiveness. Windows with scores greater than the final cutoff are called red and the rest are called green.

Step 5.

We divide the original sequence without gaps into alternating *normal* and *outlier* sections. Note that a window can span both sections. The sections boundaries are set using dynamic programming, seeking to maximize the number of red windows fully contained in outlier sections and green windows fully contained in normal sections. The point of this step is smoothing of red and green assignments. Since windows that fall on the section boundaries do not count towards the optimization score, frequent switches between normal and outlier regions are eliminated in this step.

The 5-step procedure we described above is called two-dimensional because scores are computed along the columns (step 1), but outliers are detected (Step 2 and 3) and smoothed (Step 4) along the rows. Thus, for a letter to be marked as an outlier, it must be in several windows, all of which have abnormally high scores compared to the rest of their respective columns, when compared to other windows of the same sequence.

TAPER Details

2D outlier detection is expected to be more effective at catching smaller errors with smaller values of k and longer errors with larger values of k (our results confirm this notion; see Fig. S6.2). To be able to catch a wider range of error lengths, we have devised a strategy to combine several k values, each with a different p, q setting. We run the 2D outlier algorithm on multiple k values and report their union. However, to ensure that a specific k value only catches errors of its intended length, we define an upper limit to the length of the detected error, such that only detected errors with lengths less than the upper limit are flagged. In our preliminary analyses, we confirmed that using two or more values of k dramatically improves recall for short errors of length 22 (Fig. S6.3–S6.4). In our default setting, we use k values 5, 9, and 17, with p set respectively to $0.25, 0.25, 0.1$, and q set to $0.1, 0.25, 0.5$; we only keep errors of length up to $6 \times k$ for $k = 5$ or 9. These settings were set based on our understanding of their meaning and are not tuned on any of our data; in fact, they do not seem optimal on a preliminary dataset we have tested (Fig. S6.5). For c, which can be used to control the aggressiveness of the method, we set $c = 3$ by default; this setting is motivated by preliminary analyses on a handful of datasets (Fig. S6.6) but is kept fixed as we study various datasets.

6.2.2 Experiment setup

Datasets

To benchmark TAPER, we inserted random errors into MSAs of three real datasets with different properties (Table6.1) and studied whether TAPER can detect them. The 16S.B dataset is an RNA dataset of 16S, with gold-standard alignments built by Cannone et al. (2002) and used for benchmarking alignment methods (e.g., Liu et al., 2009; Mirarab et al., 2015; Nguyen et al., 2015). Because this dataset includes 27643 sequences, it enables us to sub-sample the alignment to create subsets with controlled divergence levels. We selected subtrees with a range of diameters (i.e., maximum distance between species) from a phylogeny built from

Table 6.1. Datasets used in simulations.

Dataset	Type	# sequences	Alignment Length	Alignment
16S.B (371 sub-clades)	RNA	23–1958 (mean: 544)	854–2940 (mean: 1362)	structure-based+curated
Early-bird (19 genes)	DNA	72–171 (mean: 158)	461–2335 (1168)	Mafft + curated
Small-AA (RV100-BBA0039)	Protein	807	375	Gold-standard

the original MSA. We first find *all* clades in small diameter ranges in increments of 0.025 ($[0, 0.025], [0.025, 0.05], \ldots, [0.975, 1]$) and then select up to 10 largest clades in each diameter range, requiring at least 20 species. This procedure gave us 371 sub-datasets of the 16S.B dataset, ranging in tree diameter from 0.043 to 0.990. The avian early-bird dataset consists of DNA sequences from 19 genes, aligned automatically but also curated manually by Hackett et al. (2008). The RV100-BBA0039 is one of the largest AA alignments available as part of the BAliBASE datasets of gold-standard curated alignments used for benchmarking (Thompson et al., 2005).

Simulating errors

Errors are added to a predefined number of sequences in the alignment (m) and for a predefined length (l). Sequences with errors are selected uniformly at random. For each of the m erroneous sequences, a position to start the error is selected uniformly at random, and l of the original non-gap letters are replaced with a randomly chosen letter. For DNA, we choose randomly among the four possible nucleotides. For proteins, we draw the replaced letter from the set of all amino acids such that the chance of selecting each amino acid is proportional to the number of codons that encode to it (e.g., the chance of flipping to Leucine is six times higher than that of Methionine).

Our experiments explore several error profiles (m and l). First, we fix m to be 5% of the total number of sequences, and vary l between $(2, 3, 5, 8, 16, 32, 64) \times 11$, including 64×11 only for 16S.B as the length is long enough to accommodate such long errors, and excluding 32×11 from early-bird genes with mean sequence length below 704 as the error would be more than half of the length. We then fix l to 8×11 and set m to 1 or to $2\%, 5\%, 10\%$, or 20% of the number

of sequences. To ensure these discrete choices do not impact results, on 16S.B, we also draw l and m from a normal distribution. We determine m by first drawing a value x from a normal distribution centered at 20 and a standard deviation of 2 and set $m = \frac{N}{x} + 1$, where N is the total number of sequences in the alignment (so around 5% of sequences are erroneous). We set the length of the error by drawing from another normal distribution centered at 50 with a standard deviation of 10.

Methods compared

We were able to compare TAPER to two methods: DivA and Divvier. While both methods are mostly designed for AA, Divvier can also be run on DNA and RNA data. We test both methods on the AA data, but also include comparisons to Divvier on RNASim dataset, noting that its design is not optimized for such data. Moreover, while Divvier is mostly focused on dividing columns of an alignment into multiple columns, the same technique can also be used for filtering alignments by simply retaining the largest of the columns obtained from dividing a column (`-partial` mode). In this mode, for every column divided into smaller ones, the most complete column after division is kept and others are removed.

Evaluation criteria

We define a False Positive (FP) as any letter that is not an error, but is marked erroneous; a False Negative (FN) as any letter that is erroneous, but is not marked; a True Positive (TP) as any letter that is erroneous, and is marked erroneous; and, a True Negative (TN) as any a letter that is not erroneous, and is not marked. Each letter in the MSA, excluding gap letters, is categorized into one of the four groups, allowing us to compute recall ($\frac{TP}{TP+FN}$) and FPR ($\frac{FP}{TN+FP}$). We also report the percentage of the alignment made of errors before ($\frac{FN+TP}{FN+FP+TP+TN}$) and after ($\frac{FN}{FN+TN}$) filtering, as well as the percentage of alignment retained ($\frac{FN+TN}{FN+FP+TP+TN}$).

On a subset of datasets, we evaluate the impact of errors on the accuracy of trees inferred using FastTree-II (Price et al., 2010), measured using unweighted and weighted Robinson and

Foulds (1981) (RF) distance. Let R_b be the distance between the tree inferred from the alignment with no added errors (i.e., reference tree) and the tree inferred from the alignment with errors added; let R_a be the distance between the reference tree and the tree inferred from alignment with automatically masked errors. We define relative reduction in error as $\frac{R_b - R_a}{R_b}$ and set $\frac{0}{0} = 0$.

6.3 Results

6.3.1 Simulation Results

16S.B and impact of parameters

On the 16S.B dataset, TAPER effectively finds erroneous sequences (Fig. 6.2a-c) and improves tree accuracy (Fig. 6.2d-e). The FPR is low, never exceeding 0.16% on average across model conditions, and is even lower before adding errors. Note that we assume that the starting alignment is fully correct, but some FPs may in fact be undetected errors. TAPER retains more than 99% in most cases and never less than 97% of alignment letters (Fig. S6.7); thus, the method does not overzealously remove data. Depending on the model condition, 70% to 98% of erroneous letters are detected (Fig. 6.2a). The remaining error is never more than 0.47% of the alignment after filtering, compared to 2.5% before (Fig. 6.2b).

Impact of diameter.

Tree diameter, which is an indicator of the divergence level, has a substantial impact on the effectiveness of TAPER (Fig. 6.2 and S6.8). Over all conditions and using a simple linear model, diameter explains a statistically significant 12% of the variance in the recall (p-value according to an ANOVA test: $\ll 10^{-5}$; see Table S6.1). As the diameter increases, recall reduces gradually, but the largest reductions happen when errors are relatively short (Fig. S6.8). Increasing the diameter also increase FPR, especially when diameter is $>$0.5. The number of sequences in each alignment has a small impact on the recall (0.5% of total variance) and FPR values (Fig. S6.9).

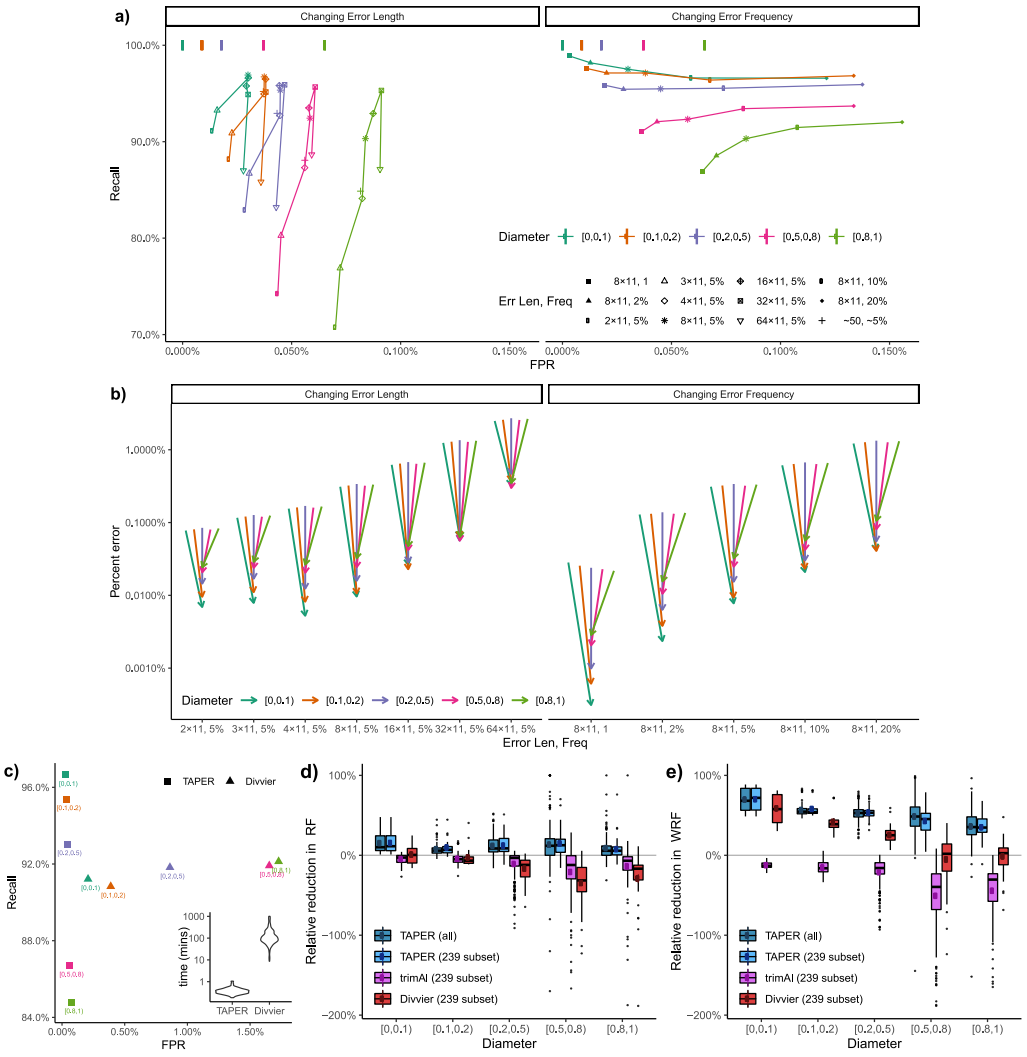

Figure 6.2. (a) ROC (Recall vs FPR) on subclades of 16S.B with varying error length, error frequency, grouped into five diameter categories (colors). Left: Percent erroneous sequences = 5%; the error length varies from 2× – 64× 11nt. We also have normally distributed error length (centered around 50nt) and frequency (around 5%) shown as cross. Right: Error length = 8 × 11nt, the number of erroneous sequence varies (1, 2%, 5%, 10%, or 20% of the size of the subtree). Vertical lines on top show the FPR before adding error (Fig. S6.8). (b) Reduction in the portion of the alignment nucleotides that are in error. Arrows show percent error before and after filtering (log-scale; see also S6.7b). (c) ROC for comparing TAPER and Divvier with normally-distributed length and frequency (subset of 239 cases). Inset: running time comparison. (d) The relative reduction in tree error after filtering. y-axis: the relative reduction in the RF distance of trees inferred from error-prone alignments to the tree inferred from error-free alignment after filtering. Not shown: one case that increased error by 1300% for TAPER and 3500% for Divver, and two (one) cases where Divvier (TAPER) had no error before filtering but some error after. Large dots show mean. Since Divvier is run on a subset of 239 cases, we show TAPER distribution both on all points (780) and the subset. (e) Same as (d) but showing weighted RF (WRF). One Divvier run that increased error by 360% is removed from the figure.

Impact of error profile.

FPR is not substantially impacted by the error length but the recall is. Error length explains 32% of variance in the recall (Table S6.1), which is the lowest with small errors of length 22nt, quickly increases as errors become longer, peaks somewhere between 88 and 352nt, and degrades slightly after that (Fig. 6.2a). The low recall for short errors should be compared with the amount of error left in the alignment after filtering (Fig. 6.2b), which is less than 0.03% on average even for high diameters. The amount of remaining error is the highest (0.47% on average) when inserted errors are long, but even then, error has reduced dramatically (2.7% before filtering). When we vary error frequency, we observe small and inconsistent changes in the recall, decreasing for low diameter and increasing for high diameters. FPR increases substantially as the error frequency goes up but remains below 0.16% even when 20% of sequences are erroneous. Finally, when error length and frequency are drawn from a normal distribution, we see consistent results (Fig. 6.2).

Comparison to Divvier.

We next compare TAPER with Divvier on a subset of 239 sub-clades with error profile drawn from the normal distribution (a subset chosen because of the prohibitive running time of Divvier). Overall, TAPER has far better FPR with a recall that is comparable and can be better or worse depending on the diameter (Fig. 6.2c). As diameter increases, TAPER has only a modest increase in FPR, never exceeding 0.1% on average, and a gradual decrease in recall from 96% to 85%. In contrast, Divvier has a sharp increase in FPR with higher diameter, reaching 1.7% at the highest level, with recall that stays stable around 90%. Thus, when the diameter is less than 0.5, TAPER has higher recall and lower FPR than Divvier, but with higher diameter, it has much lower FPR and a somewhat lower recall. Overall, Divvier is more aggressive in filtering, especially for higher diameter trees. Beyond accuracy, Divvier is much slower. While TAPER takes between 11 to 63 seconds on these data, Divvier takes between 9 minutes to 17 hours.

Impact on tree accuracy.

The improvements in alignment quality lead to substantial improvements in the trees inferred from those alignments (Fig. 6.2de). When considering only topological accuracy, TAPER reduces error in 640 cases, increases it in 79 cases, and leaves it unchanged in the remaining 61 cases among all 780 sub-clades with normally-distributed error profiles. Overall, there is a 10% relative reduction in RF distance. The improvements are much more pronounced when weighted RF distance is used to account for branch lengths: WRF improves in 769 cases, stays the same in five cases, and reduces only in six cases. On average, WRF error is reduced by 50% using TAPER compared to unfiltered alignments, and the improvements are negatively correlated with Diameter ($r = -0.37$). These improvements are despite the fact that TAPER removes no more than 3% (often less than 1%) of the data (Fig. S6.7).

It is tempting to think that a more aggressive filtering would result in even more reduction in accuracy by eliminating more of the error. However, more aggressive filtering runs the risk of also removing signal and *increasing* error. Such an over-filtering is observed for Divvier, especially for trees with high diameter (Fig. 6.2de). When diameter is 0.1 or higher, the mean topological error increases as a result of Divvier filtering, not because errors are not found, but because signal is also removed. The decrease in accuracy becomes more pronounced for larger diameters where FPR of Divvier is high. Thus, for high diameter cases where Divvier has a higher recall and FPR than TAPER, the loss in signal does not seem worth the extra reduction in error.

Early-bird dataset

On the early-bird dataset, which includes 19 genes, TAPER in most cases has FPR below 0.1% and recall above 60% (Fig. 6.3a) and reduces the error to less than 1% of the alignment in all cases (Fig. 6.3b). However, the effectiveness of TAPER varies across genes (Fig. S6.10a). At one end of the spectrum, on EEF and HMG genes, which have high diameter (1.1 ad 0.97), TAPER has FPR close to 0.1% and relatively low recall (as low as 33% when errors are long and

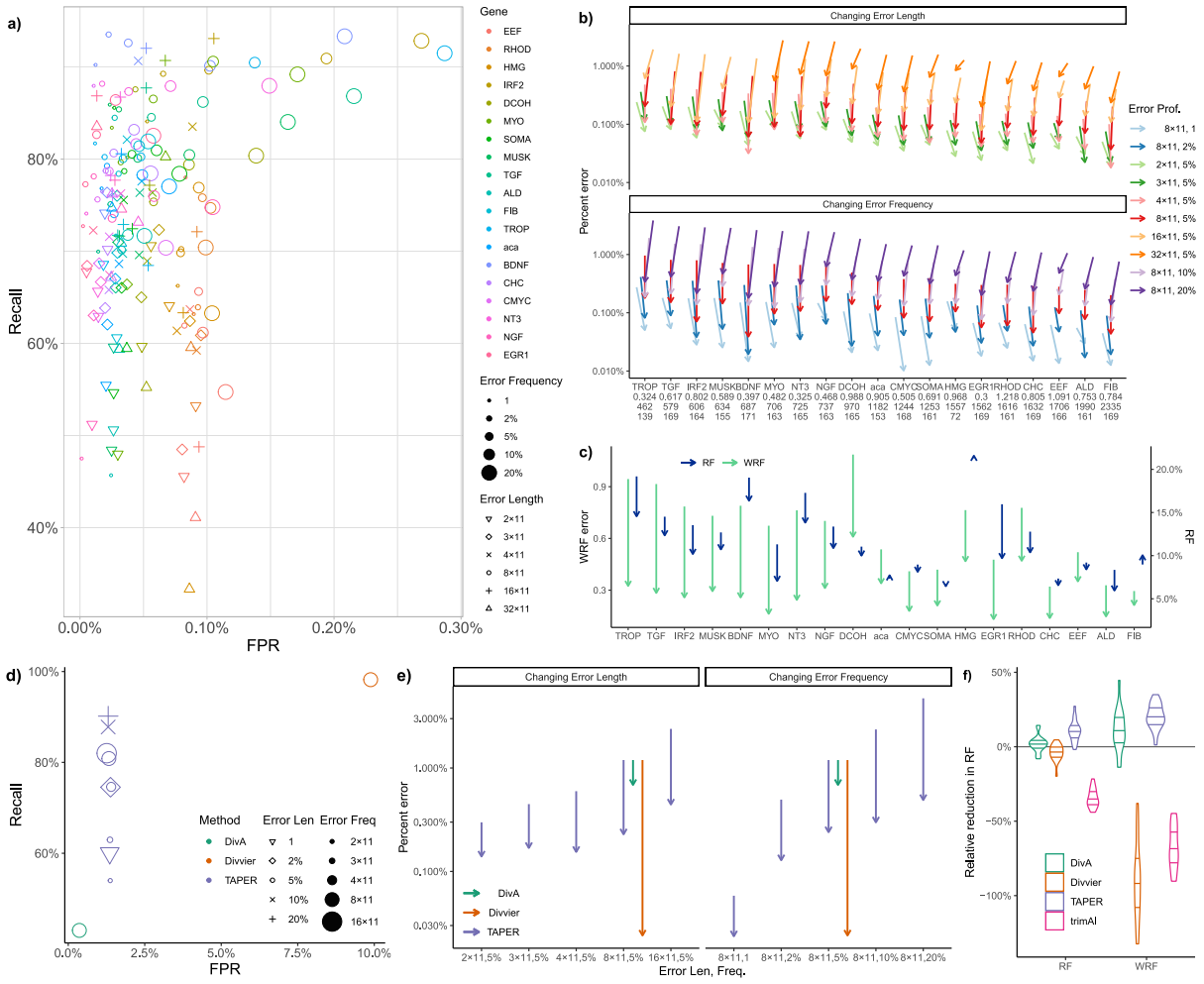

Figure 6.3. (a,d) ROC results with various error profiles for all genes of the early bird (a) and for the AA (d) datasets. (b,e) Reduction in percentage of the nucleotide positions in the alignment that consists of errors for early bird (b) and AA (e) datasets. Arrows show percent error before and after filtering. Note the log-scale. For each gene, below it, we indicate the diameter, mean sequence length, and the number of species present in that gene. See also Fig. S6.10. (c) Reduction in WRF and normalized RF error before and after filtering shown as arrows for each gene. Error profile is fixed to $8 \times 11, 5\%$. (f) Relative reduction in RF and WRF error for TAPER and DivA as a result of filtering for the $8 \times 11, 5\%$ error profile.

no more than 78% under other conditions). In addition to high diameter, the HMG gene includes 72 out of the 171 species. The other extreme is the BDNF gene (diameter: 0.40) where the mean recall is between 74% and 94% and FPR is below 0.2% across all conditions and below 0.05% in most. While BDNF is known to have patterns of branch length variation that differ from the other genes in the early-bird dataset (cf., Fig. 8 of Braun et al., 2019), several other genes such

as NT3 and IRF2 also have high accuracy (Fig. S6.10). Overall, diameter has modest negative impact on the recall of TAPER and the impact is most noticeable for longer errors (Fig. S6.10b). According to a linear model only 9% of variation in recall is explained by diameter (p-value: $\ll 10^{-5}$; see Table S6.2). For example, the IRF2 gene with a moderately high diameter (0.8) has very high recall (ranging from 71% to 93%). The impact of sequence length and number of sequences on the recall was significant (p-values: $0. < 10^{-5}$, 0.003) but modest (1.3% and 1% of total variance).

TAPER does not seem biased towards more divergent species (Fig S6.11). We saw no strong correlation between closeness of a species to the outgroup and the propensity of TAPER to filter it; in fact, there is a weak tendency for *reduced* (not increased) filtering of species that are closer to the outgroup. Also, TAPER does not remove outgroups more often than ingroup species (in fact, it tends to remove them slightly less often).

The error profile matters. The error frequency has small impacts on the recall (Table S6.2), but error length has a large impact (20% of variance; p-value: $\ll 10^{-5}$). In particular, short errors are difficult for all genes, while long errors are difficult for many but not all genes (e.g., EGR1). Overall, the five factors examined and their interactions explain only 42% of the total variance (Table S6.2).

Focusing on the error profile of 5% error and 8×11nt length, we also measured the improvement in tree accuracy (Fig. 6.3c). The topological accuracy increases for all but three genes (HMG, FIB, aca) where it slightly degrades. Improvements in normalized RF can be as high 5%. The WRF metric that considers branch length shows improvements in accuracy for every gene, and the improvements can be dramatic. For example, for BDNF, the error reduces to less than a third of the value before filtering (from 0.79 to 0.26). On average, the WRF before filtering was 2.4 times higher compared to after filtering.

Small-AA dataset

On the AA dataset, where the original alignment includes a set of sequences that are substantially divergent from the rest (Fig. S6.13), TAPER has higher FPR (mean: 1.34%) than the previous datasets (Fig. 6.3d). We attribute the higher FPR to the uncertainty in the alignment, consistent with the observation that even before adding errors, TAPER removes 1.30% of the alignment. This uncertainty makes it harder for TAPER to find inserted errors. Depending on the model condition, the median recall ranges between 54% and 90%. Increasing the frequency of error and the length of the error both improve the recall. A relatively small portion of alignments, typically less than 3%, is removed (Fig. S6.14) and the remaining error after filtering is never more than 0.5% (Fig. 6.3e). This small portion removed leads to a substantial decrease in the tree error reducing the RF and WRF substantially in most cases (Fig. 6.3f).

Compared to TAPER, DivA and Divvier show opposing patterns when tested on the default error profile. DivA is more conservative than TAPER; it has a much lower recall than TAPER (43% vs 80%) but also a lower FPR (0.4% 1.3%). On average, DivA improves phylogenetic accuracy. However, it fails to improve the phylogenetic accuracy in many cases and is far less effective than TAPER in reducing both topological and branch length errors (Fig. 6.3f). In contrast to DivA, Divvier is more aggressive than TAPER and removes more than 11% of the alignment (Fig. S6.14). As a result, it has close to perfect recall but also 10% FPR (Fig. 6.3d). This aggressive filtering results in lowered phylogenetic accuracy, a pattern that is most pronounced for branch lengths (Fig. 6.3f). Beyond accuracy, DivA is the slowest method, taking on average 7.5 hours on these data, followed by Divvier, which takes 58 minutes, and TAPER, which takes only 11 seconds.

6.3.2 Real biological data

To test the effectiveness of TAPER on real data, we revisited 56 genes from the avian dataset of Jarvis et al. (2014) analyzed manually by Springer and Gatesy (2018). Springer and

Gatesy (2018) found errors in all 56 of these genes (some small and others relatively large). Since the ground truth of where *all* errors lie is not known, to evaluate results of TAPER, we rely on a likelihood-based metric. If the nucleotides removed are in fact erroneous, we expect the likelihood of the maximum likelihood gene tree to increase, more than it would if we remove the same number of nucleotides from random positions. Further, since gene trees for this dataset are notoriously difficult to estimate accurately (Mirarab et al., 2014), we also compute the likelihood on the best-estimate of the species tree from Zhang et al. (2018) (we present this tree as Fig. S6.15a) after estimating branch lengths for each specific gene; despite a potential for true gene tree discordance, we expect that removing errors would increase the likelihood of the species tree. However, it must be noted that simply removing the most divergent sequences is also expected to increase the likelihood. Thus, we also test a control method that removes the same number of letters from the alignment as TAPER but simply picks those positions with the highest letter score produced in Step 1 of TAPER. These would be the most divergent positions in otherwise conserved columns but are not necessarily 2D outliers as defined by TAPER.

TAPER marked between 0% and 1.2% (mean: 0.2%) of nucleotides in these genes as erroneous (Fig. 6.4). For both gene trees and the species tree, we see dramatic improvements in the log likelihood (measured using RAxML, Stamatakis, 2014), far exceeding small increases in the likelihood we get by random removal of the same amount of data (Fig. 6.4a). The more data is removed, the more the likelihood increases. Removing as little as 1% of the data can result in 10% improvements in the log likelihood. The control method of removing the most divergent letters increases likelihood even more (roughly twice as much as TAPER). This result is expected as sequences diverging from the consensus decrease the likelihood (whether the difference is real or not). However, simply removing divergent sequences leads to removal of *some* bases from most species in most genes (Fig. 6.4b), which cannot correspond to removal of errors. In contrast, TAPER removes sequences from only a few species for each gene, which is consistent with removal of actual errors. Reassuringly, the two outgroups were not removed more often than other species using TAPER (Fig S6.15b). In contrast, removing high-scoring

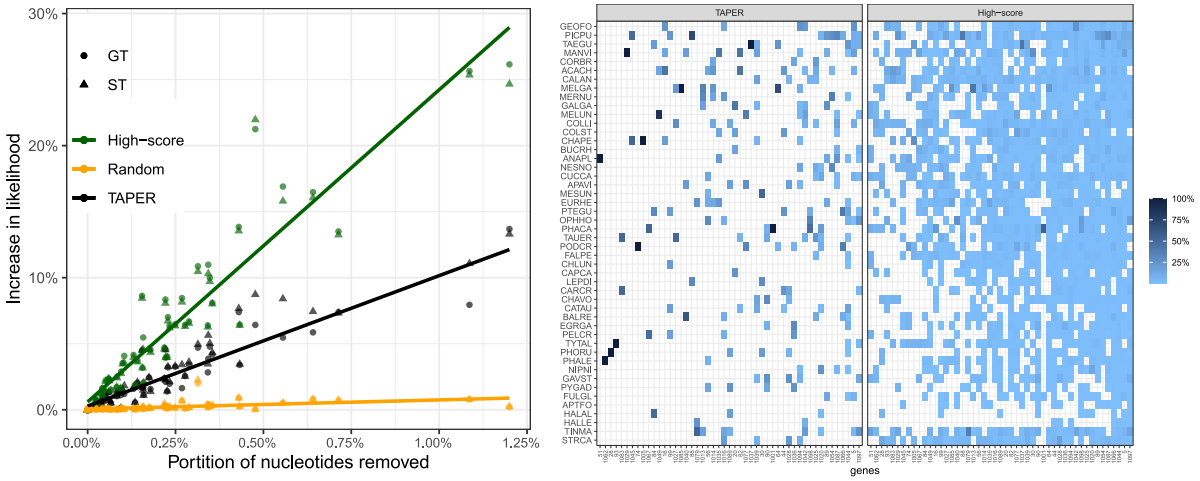

Figure 6.4. (a) Increase in the log likelihood after filtering computed both for the ML gene tree (GT) and the species tree (ST) versus the portion of the nucleotides in an alignment that are removed. For control, we show the results of removing the same amount of data from each gene as TAPER removes but selected either randomly or taking the highest scoring positions in Step 1 of TAPER corresponding to the most divergent letters in their respective columns. y-axis: change in log-likelihood normalized by likelihood before filtering. (b) For each gene (columns), we show the percentage of total sequences removed from a particular species (rows) using TAPER or the control method of filtering the highest scores. Species are sorted from top to the bottom by their average distance to the two outgroups, shown at the bottom. The top seven species are passerines, which are highly divergent from others.

positions removes outgroups and highly divergent passerines frequently.

Visual inspection of genes shows that out of 68 cases of error identified by Springer and Gatesy (2018), 24 of them are fully or mostly found by TAPER, and in eight cases, a minority of erroneous positions are found (Table S6.3). Among the remaining 36 cases (Table S6.4), 21 are errors that are too short ($\leq$ 10bp) or too frequent ($\geq$ 10 out of 48 species) for TAPER to be effective, and in 7 cases, they are somewhat frequent ($\geq$ 5). In another three cases, only a handful of species are present in those sites, making the errors to be a high *portion* of non-gapped species, which TAPER cannot detect (Table S6.4). Overall, TAPER marked 21 out of 29 cases that were not too short or too frequent.

6.4 Discussion

We introduced TAPER, a method for detecting errors in individual species in an alignment. TAPER was able to reduce error dramatically under varied conditions that we studied in this paper while removing relatively little data. By design, TAPER is more conservative than alternative error filtering methods that tend to remove large numbers of sites, species, or both. This conservative design is based on the philosophy that we should strive to eliminate errors without also removing signal. We achieve this goal using the 2D outlier detection algorithm and step 4, which makes sure we remove sequences only if they stand out both compared to other sites of the same sequence and other sequences of the same site. As expected, turning off step 4 would result in increased FP rates (Fig. S6.16).

Like TAPER, Divvier tries to spot false homologies while refraining from excessively removing true homologies. However, TAPER and Divvier use very different algorithmic strategies. Instead of looking for erroneous sequence stretches, Divvier looks for mis-alignments (i.e., aligned columns that should be divided into multiple columns). It first uses pair-HMMs to compute the posterior probability of any two residues in the same column being homologous and then clusters all residues in that column according to these probabilities. By using per-column posteriors, Divvier hopes to avoid splitting a column into multiple clusters simply because it has high divergence levels. In our analyses, Divvier used in the filtering mode was more aggressive than TAPER, leading to higher recall but also much higher FP rates (especially for high diameter datasets). This more aggressive filtering was not beneficial because compared to TAPER, Divvier had less positive impact on the accuracy of inferred trees (and often *reduced* the accuracy). We suspect the simpler design of TAPER is responsible for its higher accuracy and much reduced running time. While TAPER directly looks for outliers based on thresholds adjusted per columns and rows of the alignment (p and q parameters), Divvier's more complex approach relies on the ability to compute homology probabilities using pair-HMMs. Moreover, TAPER works on windows while Divvier works on individual columns. However, we should also note that

while Divvier supports error detection (the feature we tested here), its design seems motivated by splitting alignment columns, which is a sligthly different goal.

The immediate impact of TAPER would be in improving phylogenetic analyses that rely on the accuracy of alignments. However, other analyses can also benefit from improved alignments. For example, detection of selection and functional annotation of genes both rely on accurate alignments, and removing errors may improve their accuracy. Many studies in microbiome and ecology also rely on aligning sequences to detect levels of divergence and to characterize diversity of samples; errors in the data can lead to over-estimates of diversity, which may be reduced using filtering methods such as TAPER. Finally, note that inaccuracies in alignments can propagate when they are used as training data for machine learning methods; for example, many research build hidden Markov models (e.g., Pfam) that are then used to recruit and align new sequences. Thus, TAPER has the potential to improve a wide range of downstream analyses.

The limitations of TAPER should also be kept in mind. TAPER is not very effective in finding very short errors. For example, for errors below a length 20nt, it will not be very effective. On the other extreme, since TAPER is looking for outlier regions, it cannot detect very large errors. In the extreme case, if the error is more than half the sequence length or if error appears in many of the sequences, it will not show up as an outlier; instead, it will be taken by the 2D outlier algorithm as natural variation. In between these extremes of short and long is the sweet spot for the error detection by TAPER where there is enough signal to detect oddity of the pattern but the error is not so large that it looks like a real phylogenetic divergence. For shorter errors, changing settings of TAPER (e.g., reducing k) could perhaps make the method more sensitive, but that sensitivity would come at the expense of more FP filtering, which we tried hard to avoid. For longer errors, methods like TreeShrink that look for unexpected patterns in branch lengths provide viable alternatives.

The limitations of TAPER in terms of very short or very long errors do not negate its value on real data. On the Springer and Gatesy (2018) empirical dataset we examined, about one third

of sequences were shorter than 20nt (Fig. S6.17), showing that the range of errors TAPER can detect still covers a substantial portion of errors. More broadly, there are many sources of error in phylogenomic datasets and the expected length distribution of erroneously aligned segments in phylogenomic datasets is likely to be sensitive to many different steps in the assembly, annotation, and alignment pipelines. Although many studies focused on genome assembly have highlighted the challenges associated with the assembly of repeats (Pop, 2009), we do not expect this issue to be a major problem in phylogenomics because failure to assemble individual sequence reads would lead to missing regions that simply cannot be used for phylogenetic estimation. Problems associated with identification of homologous regions in correctly assembled genome sequences are likely to be more common. Indeed, a major source of error that has been noted in previous studies is the alignment of non-homologous exons or the alignment of exons to intronic sequence that has incorrectly been annotated as exonic (e.g., Springer and Gatesy, 2018). Another source of error might be the failure to identify alternatively spliced exons in transcriptome assemblies. Both of these phenomena will lead to errors with a length distribution that resembles the length distribution of exons as a whole (likely with a bias toward the short exons that are more difficult to identify). While some of the shorter exons will fall outside the range detected by TAPER, we expect that a fairly large proportions of the alignment errors will fall in the range of lengths that can be identified using TAPER. Finally, we note that the shorter errors, on average, are expected to have less impact on downstream analyses than longer errors. Thus, we expect TAPER to be a useful addition to the phylogenomics pipeline.

6.5 Acknowledgements

Chapter 6, in full, is a reprint of the material as it appears in " Zhang, C.†, Zhao, Y.†, Braun, E. & Mirarab, S. TAPER: Pinpointing errors in multiple sequence alignments despite varying rates of evolution. *Methods In Ecology And Evolution*. **12**, 2145-2158 (2021) ." The dissertation author was the co-primary investigator and co-first author of this paper.

Bibliography

R. H. Ali, M. Bogusz, and S. Whelan. Identifying Clusters of High Confidence Homologies in Multiple Sequence Alignments. *Molecular Biology and Evolution*, 36(10):2340–2351, 10 2019. ISSN 0737-4038. doi: 10.1093/molbev/msz142. URL https://academic.oup.com/mbe/article/36/10/2340/5519769.

M. Borowiec. Spruceup: fast and flexible identification, visualization, and removal of outliers from large multiple sequence alignments. *Journal of Open Source Software*, 4(42):1635, 10 2019. ISSN 2475-9066. doi: 10.21105/joss.01635. URL https://joss.theoj.org/papers/10.21105/joss.01635.

E. L. Braun, J. Cracraft, and P. Houde. 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, 2019. doi: 10.1007/978-3-030-16477-5{_}6. URL http://link.springer.com/10.1007/978-3-030-16477-5_6.

F. P. Breitwieser, M. Pertea, A. V. Zimin, and S. L. Salzberg. Human contamination in bacterial genomes has created thousands of spurious proteins. *Genome research*, 29(6):954–960, 6 2019. ISSN 1549-5469 (Electronic). doi: 10.1101/gr.245373.118.

J. J. Cannone, S. Subramanian, M. N. Schnare, J. R. Collett, L. M. D'Souza, Y. Du, B. Feng, N. Lin, L. V. Madabusi, K. M. Müller, N. Pande, Z. Shang, N. Yu, and R. R. Gutell. The comparative RNA web (CRW) site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs. *BMC bioinformatics*, 3:2, 2002. ISSN 1471-2105. doi: 10.1186/1471-2105-3-2.

S. Capella-Gutiérrez, J. M. Silla-Martínez, and T. Gabaldón. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 2009. ISSN 1367-4803.

J. Castresana. Selection of Conserved Blocks from Multiple Alignments for Their Use in Phylogenetic Analysis. *Molecular Biology and Evolution*, 17(4):540–552, 2000. ISSN 0737-4038. doi: 10.1093/oxfordjournals.molbev.a026334. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/oxfordjournals.molbev.a026334.

D. M. de Vienne, S. Ollier, and G. Aguileta. Phylo-MCOA: A Fast and Efficient Method to Detect Outlier Genes and Species in Phylogenomics Using Multiple Co-inertia Analysis. *Molecular Biology and Evolution*, 29(6):1587–1598, 6 2012. ISSN 0737-4038. doi: 10.1093/molbev/msr317. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msr317.

A. Di Franco, R. Poujol, D. Baurain, and H. Philippe. Evaluating the usefulness of alignment filtering methods to reduce the impact of errors on evolutionary inferences. *BMC Evolutionary Biology*, 19(1):21, 12 2019. ISSN 1471-2148. doi: 10.1186/s12862-019-1350-2. URL https://bmcevolbiol.biomedcentral.com/articles/10.1186/s12862-019-1350-2.

A. W. Dress, C. Flamm, G. Fritzsch, S. Grünewald, M. Kruspe, S. J. Prohaska, and P. F. Stadler. Noisy: Identification of problematic columns in multiple sequence alignments. *Algorithms for Molecular Biology*, 3(1):7, 2008. ISSN 1748-7188. doi: 10.1186/1748-7188-3-7. URL http://almob.biomedcentral.com/articles/10.1186/1748-7188-3-7.

W. Fletcher and Z. Yang. The Effect of Insertions, Deletions, and Alignment Errors on the Branch-Site Test of Positive Selection. *Molecular Biology and Evolution*, 27(10):2257–2267, 10 2010. ISSN 0737-4038. doi: 10.1093/molbev/msq115. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msq115.

C. M. Francois, F. Durand, E. Figuet, and N. Galtier. Prevalence and Implications of Contamination in Public Genomic Resources: A Case Study of 43 Reference Arthropod Assemblies. *G3*, 10(2):721–730, 2 2020. ISSN 2160-1836. doi: 10.1534/g3.119.400758. URL http://g3journal.org/lookup/doi/10.1534/g3.119.400758.

J. Gatesy and M. S. Springer. Phylogenetic Analysis at Deep Timescales: Unreliable Gene Trees, Bypassed Hidden Support, and the Coalescence/Concatalescence Conundrum. *Molecular phylogenetics and evolution*, 80:231–266, 2014. ISSN 1095-9513. doi: 10.1016/j.ympev.2014.08.013. URL http://www.ncbi.nlm.nih.gov/pubmed/25152276.

S. J. Hackett, R. T. Kimball, S. Reddy, R. C. K. Bowie, E. L. Braun, M. J. Braun, J. L. Chojnowski, W. A. Cox, K.-L. Han, J. Harshman, C. J. Huddleston, B. D. Marks, K. J. Miglia, W. S. Moore, F. H. Sheldon, D. W. Steadman, C. C. Witt, and T. Yuri. A phylogenomic study of birds reveals their evolutionary history. *Science*, 320(5884):1763–1768, 2008. ISSN 0036-8075. doi: 10.1126/science.1157704.

S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 11 1992. ISSN 0027-8424. doi: 10.1073/pnas.89.22.10915. URL http://www.pnas.org/cgi/doi/10.1073/pnas.89.22.10915.

P. A. Hosner, B. C. Faircloth, T. C. Glenn, E. L. Braun, and R. T. Kimball. Avoiding Missing Data Biases in Phylogenomic Inference: An Empirical Study in the Landfowl (Aves: Galliformes). *Molecular Biology and Evolution*, 33(4):1110–1125, 4 2016. ISSN 0737-4038. doi: 10.1093/molbev/msv347. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/

msv347.

C.-H. Huang, R. Sun, Y. Hu, L. Zeng, N. Zhang, L. Cai, Q. Zhang, M. A. Koch, I. Al-Shehbaz, and P. P. Edger. Resolution of Brassicaceae phylogeny using nuclear genes uncovers nested radiations and supports convergent morphological evolution. *Molecular biology and evolution*, 33(2):394–412, 2016. ISSN 0737-4038.

E. D. Jarvis, S. Mirarab, A. J. Aberer, B. Li, P. Houde, C. Li, S. Y. W. Ho, B. C. Faircloth, B. Nabholz, J. T. Howard, A. Suh, C. C. Weber, R. R. da Fonseca, J. Li, F. Zhang, H. Li, L. Zhou, N. Narula, L. Liu, G. Ganapathy, B. Boussau, M. S. Bayzid, V. Zavidovych, S. Subramanian, T. Gabaldon, S. Capella-Gutierrez, J. Huerta-Cepas, B. Rekepalli, K. Munch, M. Schierup, B. Lindow, W. C. Warren, D. Ray, R. E. Green, M. W. Bruford, X. Zhan, A. Dixon, S. Li, N. Li, Y. Huang, E. P. Derryberry, M. F. Bertelsen, F. H. Sheldon, R. T. Brumfield, C. V. Mello, P. V. Lovell, M. Wirthlin, M. P. C. Schneider, F. Prosdocimi, J. A. Samaniego, A. M. V. Velazquez, A. Alfaro-Nunez, P. F. Campos, B. Petersen, T. Sicheritz-Ponten, A. Pas, T. Bailey, P. Scofield, M. Bunce, D. M. Lambert, Q. Zhou, P. Perelman, A. C. Driskell, B. Shapiro, Z. Xiong, Y. Zeng, S. Liu, Z. Li, B. Liu, K. Wu, J. Xiao, X. Yinqi, Q. Zheng, Y. Zhang, H. Yang, J. Wang, L. Smeds, F. E. Rheindt, M. Braun, J. Fjeldsa, L. Orlando, F. K. Barker, K. A. Jonsson, W. Johnson, K.-P. Koepfli, S. O'Brien, D. Haussler, O. A. Ryder, C. Rahbek, E. Willerslev, G. R. Graves, T. C. Glenn, J. McCormack, D. Burt, H. Ellegren, P. Alstrom, S. V. Edwards, A. Stamatakis, D. P. Mindell, J. Cracraft, E. L. Braun, T. Warnow, W. Jun, M. T. P. Gilbert, and G. Zhang. Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331, 12 2014. ISSN 0036-8075. doi: 10.1126/science.1253451. URL http://www.sciencemag.org/content/346/6215/1320.abstracthttp://www.sciencemag.org/cgi/doi/10.1126/science.1253451.

O. Jeffroy, H. Brinkmann, F. Delsuc, and H. Philippe. Phylogenomics: the beginning of incongruence? *Trends in Genetics*, 22(4):225–231, 2006. ISSN 01689525. doi: 10.1016/j.tig.2006.02.003. URL http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&dopt=Citation&list_uids=16490279.

G. F. Jenks. The data model concept in statistical mapping. *International yearbook of cartography*, 7:186–190, 1967.

K. Katoh and D. M. Standley. A simple method to control over-alignment in the MAFFT multiple sequence alignment program. *Bioinformatics*, 32(13):1933–1942, 7 2016. ISSN 1367-4803. doi: 10.1093/bioinformatics/btw108. URL https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btw108.

S. Laurin-Lemay, H. Brinkmann, and H. Philippe. Origin of land plants revisited in the light of sequence contamination and missing data. *Current Biology*, 2012. ISSN 09609822. doi: 10.1016/j.cub.2012.06.013.

Li-San Wang, J. Leebens-Mack, P. K. Wall, K. Beckmann, C. W. de Pamphilis, and T. Warnow.

The Impact of Multiple Protein Sequence Alignment on Phylogenetic Estimation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(4):1108–1119, 7 2011. ISSN 1545-5963. doi: 10.1109/TCBB.2009.68. URL http://www.ncbi.nlm.nih.gov/pubmed/21566256http://doi.ieeecomputersociety.org/10.1109/TCBB.2009.68http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5235137http://ieeexplore.ieee.org/document/5235137/.

K. Liu, S. Raghavan, S. M. Nelesen, C. R. Linder, and T. Warnow. Rapid and Accurate Large-Scale Coestimation of Sequence Alignments and Phylogenetic Trees. *Science*, 324 (5934):1561–1564, 6 2009. ISSN 1095-9203. doi: 10.1126/science.1171243. URL http://www.sciencemag.org/content/324/5934/1561.abstracthttp://www.sciencemag.org/content/324/5934/1561.full.pdfhttp://www.sciencemag.org/cgi/content/abstract/324/5934/1561http://www.ncbi.nlm.nih.gov/pubmed/19541996.

A. Loytynoja and N. Goldman. An algorithm for progressive multiple alignment of sequences with insertions. *Proceedings of the National Academy of Sciences*, 102(30):10557–10562, 7 2005. ISSN 0027-8424. doi: 10.1073/pnas.0409137102. URL http://www.pnas.org/cgi/doi/10.1073/pnas.0409137102.

A. Löytynoja and N. Goldman. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, 320(5883):1632–1635, 2008. ISSN 0036-8075. doi: 10.1126/science.1158395.

A. Löytynoja, A. J. Vilella, and N. Goldman. Accurate extension of multiple sequence alignments using a phylogeny-aware graph algorithm. *Bioinformatics*, 28(13):1684–1691, 7 2012. ISSN 1460-2059. doi: 10.1093/bioinformatics/bts198. URL https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bts198.

G. Lunter, A. Rocco, N. Mimouni, A. Heger, A. Caldeira, and J. Hein. Uncertainty in homology inferences: assessing and improving genomic sequence alignment. *Genome research*, 18(2): 298–309, 2 2008. ISSN 1088-9051. doi: 10.1101/gr.6725608. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2203628&tool=pmcentrez&rendertype=abstract.

U. Mai and S. Mirarab. TreeShrink: fast and accurate detection of outlier long branches in collections of phylogenetic trees. *BMC Genomics*, 19(S5):272, 5 2018. ISSN 1471-2164. doi: 10.1186/s12864-018-4620-2. URL https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-018-4620-2.

S. Mirarab, M. S. Bayzid, B. Boussau, and T. Warnow. Statistical binning enables an accurate coalescent-based estimation of the avian tree. *Science*, 346(6215):1250463–1250463, 12 2014. ISSN 0036-8075. doi: 10.1126/science.1250463. URL http://www.sciencemag.org/cgi/doi/10.1126/science.1250463.

S. Mirarab, N. Nguyen, S. Guo, L.-S. Wang, J. Kim, and T. Warnow. PASTA: Ultra-Large Multiple Sequence Alignment for Nucleotide and Amino-Acid Sequences. *Journal of com-*

putational biology, 22(5):377–86, 5 2015. ISSN 1557-8666. doi: 10.1089/cmb.2014.0156. URL http://online.liebertpub.com/doi/abs/10.1089/cmb.2014.0156http://www.liebertpub.com/doi/10.1089/cmb.2014.0156http://www.ncbi.nlm.nih.gov/pubmed/25549288http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4424971.

E. K. Molloy and T. Warnow. To Include or Not to Include: The Impact of Gene Filtering on Species Tree Estimation Methods. *Systematic Biology*, 67(2):285–303, 3 2018. ISSN 1063-5157. doi: 10.1093/sysbio/syx077. URL https://academic.oup.com/sysbio/article/67/2/285/4159193.

N.-p. D. Nguyen, S. Mirarab, K. Kumar, and T. Warnow. Ultra-large alignments using phylogeny-aware profiles. *Genome Biology*, 16(1):124, 12 2015. ISSN 1474-760X. doi: 10.1186/s13059-015-0688-z. URL http://genomebiology.com/2015/16/1/124https://genomebiology.biomedcentral.com/articles/10.1186/s13059-015-0688-z.

T. H. Ogdenw and M. S. Rosenberg. Multiple sequence alignment accuracy and phylogenetic inference. *Systematic biology*, 55(2):314–328, 2006. ISSN 1063-5157. doi: 10.1080/10635150500541730.

L. E. Olson and A. Hassanin. Contamination and chimerism are perpetuating the legend of the snake-eating cow with twisted horns (Pseudonovibos spiralis). A case study of the pitfalls of ancient DNA. *Molecular Phylogenetics and Evolution*, 27(3):545–548, 2003. ISSN 1055-7903.

H. Philippe, D. M. d. Vienne, V. Ranwez, B. Roure, D. Baurain, and F. Delsuc. Pitfalls in supermatrix phylogenomics. *European Journal of Taxonomy*, 2017. ISSN 2118-9773. doi: 10.5852/ejt.2017.283.

M. Pop. Genome assembly reborn: recent computational challenges. *Briefings in Bioinformatics*, 10(4):354–366, 7 2009. ISSN 1467-5463. doi: 10.1093/bib/bbp026. URL https://academic.oup.com/bib/article-lookup/doi/10.1093/bib/bbp026.

D. M. Portik and J. J. Wiens. Do Alignment and Trimming Methods Matter for Phylogenomic (UCE) Analyses? *Systematic Biology*, 8 2020. ISSN 1063-5157. doi: 10.1093/sysbio/syaa064. URL https://doi.org/10.1093/sysbio/syaa064.

M. N. Price, P. S. Dehal, and A. P. Arkin. FastTree-2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 3 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0009490. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2835736&tool=pmcentrez&rendertype=abstract.

V. Rajan. A Method of Alignment Masking for Refining the Phylogenetic Signal of Multiple Sequence Alignments A Method of Alignment Masking for Refining the Phylogenetic Signal of. *Molecular biology and evolution*, 30(3):689–712, 2012. ISSN 1537-1719.

D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, 1981. URL http://www.sciencedirect.com/science/article/pii/0025556481900432.

L. Salichos and A. Rokas. Evaluating ortholog prediction algorithms in a Yeast Model Clade. *PLoS ONE*, 2011. ISSN 19326203. doi: 10.1371/journal.pone.0018755.

L. Salichos and A. Rokas. Inferring ancient divergences requires genes with strong phylogenetic signals. *Nature*, 497(7449):327–31, 2013. ISSN 1476-4687. doi: 10.1038/nature12130. URL http://www.nature.com/nature/journal/vaop/ncurrent/full/nature12130.html.

E. Sayyari, J. B. Whitfield, and S. Mirarab. Fragmentary Gene Sequences Negatively Impact Gene Tree and Species Tree Reconstruction. *Molecular Biology and Evolution*, 34(12):3279–3291, 12 2017. ISSN 0737-4038. doi: 10.1093/molbev/msx261. URL http://dx.doi.org/10.1093/molbev/msx261http://academic.oup.com/mbe/article/doi/10.1093/molbev/msx261/4344836/Fragmentary-gene-sequences-negatively-impact-genehttps://academic.oup.com/mbe/article/doi/10.1093/molbev/msx261/4344836/Fragmentary-gene-sequenc.

I. Sela, H. Ashkenazy, K. Katoh, and T. Pupko. GUIDANCE2: accurate detection of unreliable alignment regions accounting for the uncertainty of multiple parameters. *Nucleic Acids Research*, 43(W1):W7–W14, 7 2015. ISSN 0305-1048. doi: 10.1093/nar/gkv318. URL https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkv318.

X.-x. Shen, C. T. Hittinger, and A. Rokas. Contentious relationships in phylogenomic studies can be driven by a handful of genes. *Nature Ecology & Evolution*, 1(5):0126, 4 2017. ISSN 2397-334X. doi: 10.1038/s41559-017-0126. URL http://dx.doi.org/10.1038/s41559-017-0126http://www.nature.com/articles/s41559-017-0126.

P. Simion, K. Belkhir, C. François, J. Veyssier, J. C. Rink, M. Manuel, H. Philippe, and M. J. Telford. A software tool 'CroCo' detects pervasive cross-species contamination in next generation sequencing data. *BMC Biology*, 2018. ISSN 17417007. doi: 10.1186/s12915-018-0486-7.

V. Smirnov and T. Warnow. Phylogeny Estimation Given Sequence Length Heterogeneity. *Systematic Biology*, (0):1–47, 7 2020. ISSN 1063-5157. doi: 10.1093/sysbio/syaa058. URL https://academic.oup.com/sysbio/advance-article/doi/10.1093/sysbio/syaa058/5874451.

M. S. Springer and J. Gatesy. The gene tree delusion. *Molecular Phylogenetics and Evolution*, 94(Part A):1–33, 1 2016. ISSN 10557903. doi: 10.1016/j.ympev.2015.07.018. URL http://www.sciencedirect.com/science/article/pii/S1055790315002225http://linkinghub.elsevier.com/retrieve/pii/S1055790315002225http://dx.doi.org/10.1016/j.ympev.2015.07.018https://linkinghub.elsevier.com/retrieve/pii/S1055790315002225.

M. S. Springer and J. Gatesy. On the importance of homology in the age of phylogenomics. *Systematics and Biodiversity*, 16(3):210–228, 4 2018. ISSN 1477-2000. doi: 10.1080/14772000.2017.1401016. URL https://www.tandfonline.com/doi/full/10.1080/14772000.

2017.1401016.

A. Stamatakis. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014. ISSN 14602059. doi: 10.1093/bioinformatics/btu033.

J. L. Steenwyk, T. J. Buida, Y. Li, X.-X. Shen, and A. Rokas. ClipKIT: A multiple sequence alignment trimming software for accurate phylogenomic inference. *PLOS Biology*, 18(12): e3001007, 12 2020. ISSN 1545-7885. doi: 10.1371/journal.pbio.3001007. URL https://dx.plos.org/10.1371/journal.pbio.3001007.

T. H. Struck. TreSpEx—Detection of Misleading Signal in Phylogenetic Reconstructions Based on Tree Information. *Evolutionary Bioinformatics*, 10:EBO.S14239, 1 2014. ISSN 1176-9343. doi: 10.4137/EBO.S14239. URL http://journals.sagepub.com/doi/10.4137/EBO.S14239.

G. Tan, M. Muffato, C. Ledergerber, J. Herrero, N. Goldman, M. Gil, and C. Dessimoz. Current Methods for Automated Filtering of Multiple Sequence Alignments Frequently Worsen Single-Gene Phylogenetic Inference. *Systematic Biology*, 64(5):778–791, 9 2015. ISSN 1063-5157. doi: 10.1093/sysbio/syv033. URL https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/syv033.

J. D. Thompson, P. Koehl, R. Ripp, and O. Poch. BAliBASE 3.0: Latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics*, 61(1):127–136, 7 2005. ISSN 08873585. doi: 10.1002/prot.20527. URL http://doi.wiley.com/10.1002/prot.20527.

K. M. Westover, J. P. Rusinko, J. Hoin, and M. Neal. Rogue taxa phenomenon: A biological companion to simulation analysis. *Molecular Phylogenetics and Evolution*, 69(1):1–3, 2013. ISSN 10557903. doi: 10.1016/j.ympev.2013.05.010.

S. Whelan, I. Irisarri, and F. Burki. PREQUAL: detecting non-homologous characters in sets of unaligned homologous sequences. *Bioinformatics*, 34(22):3929–3930, 6 2018. ISSN 1367-4803. doi: 10.1093/bioinformatics/bty448. URL https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/bty448/5026659.

N. J. Wickett, S. Mirarab, N. Nguyen, T. Warnow, E. J. Carpenter, N. Matasci, S. Ayyampalayam, M. S. Barker, J. G. Burleigh, M. A. Gitzendanner, B. R. Ruhfel, E. Wafula, J. P. Der, S. W. Graham, S. Mathews, M. Melkonian, D. E. Soltis, P. S. Soltis, N. W. Miles, C. J. Rothfels, L. Pokorny, A. J. Shaw, L. DeGironimo, D. W. Stevenson, B. Surek, J. C. Villarreal, B. Roure, H. Philippe, C. W. DePamphilis, T. Chen, M. K. Deyholos, R. S. Baucom, T. M. Kutchan, M. M. Augustin, J. J. Wang, Y. Zhang, Z. Tian, Z. Yan, X. Wu, X. Sun, G. K.-S. Wong, and J. J. Leebens-Mack. Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences*, 111(45):4859–4868, 10 2014. ISSN 0027-8424. doi: 10.1073/pnas.1323926111. URL http://www.pnas.org/lookup/doi/10.

1073/pnas.1323926111http://www.pnas.org/cgi/content/long/111/45/E4859.

M. Zepeda Mendoza, S. Nygaard, and R. R. da Fonseca. DivA: detection of non-homologous and very divergent regions in protein sequence alignments. *BMC Research Notes*, 7(1):806, 2014. ISSN 1756-0500. doi: 10.1186/1756-0500-7-806. URL http://bmcresnotes.biomedcentral.com/articles/10.1186/1756-0500-7-806.

C. Zhang, M. Rabiee, E. Sayyari, and S. Mirarab. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6):153, 5 2018. ISSN 1471-2105. doi: 10.1186/s12859-018-2129-y. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-018-2129-y.

Appendices

Supplementary Materials

6.A Supplementary figures

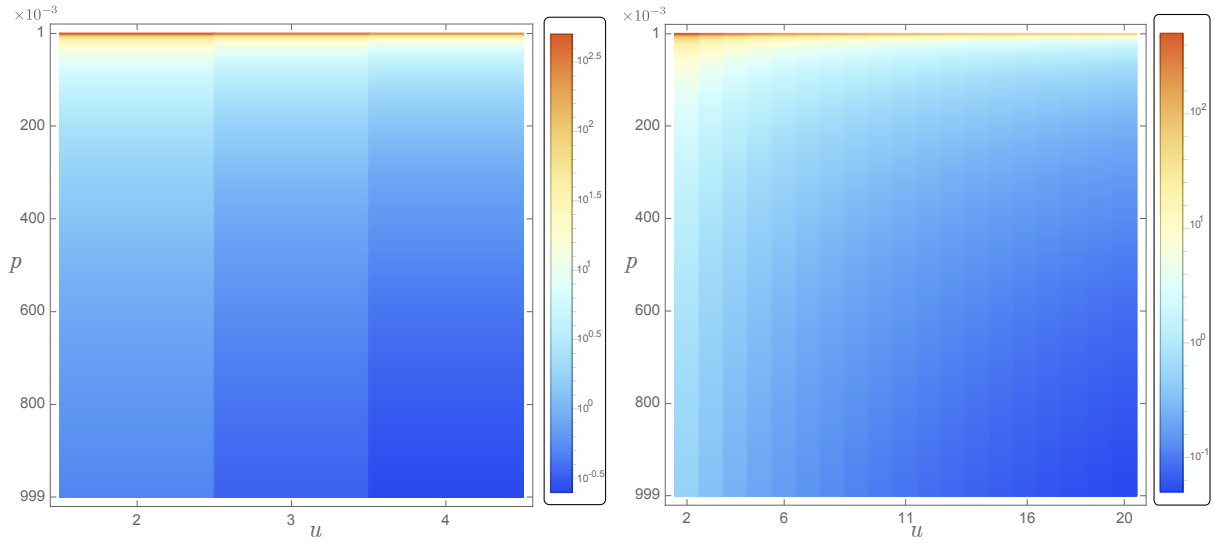

Figure S6.1. Score function. Left: DNA, Right: Protein.

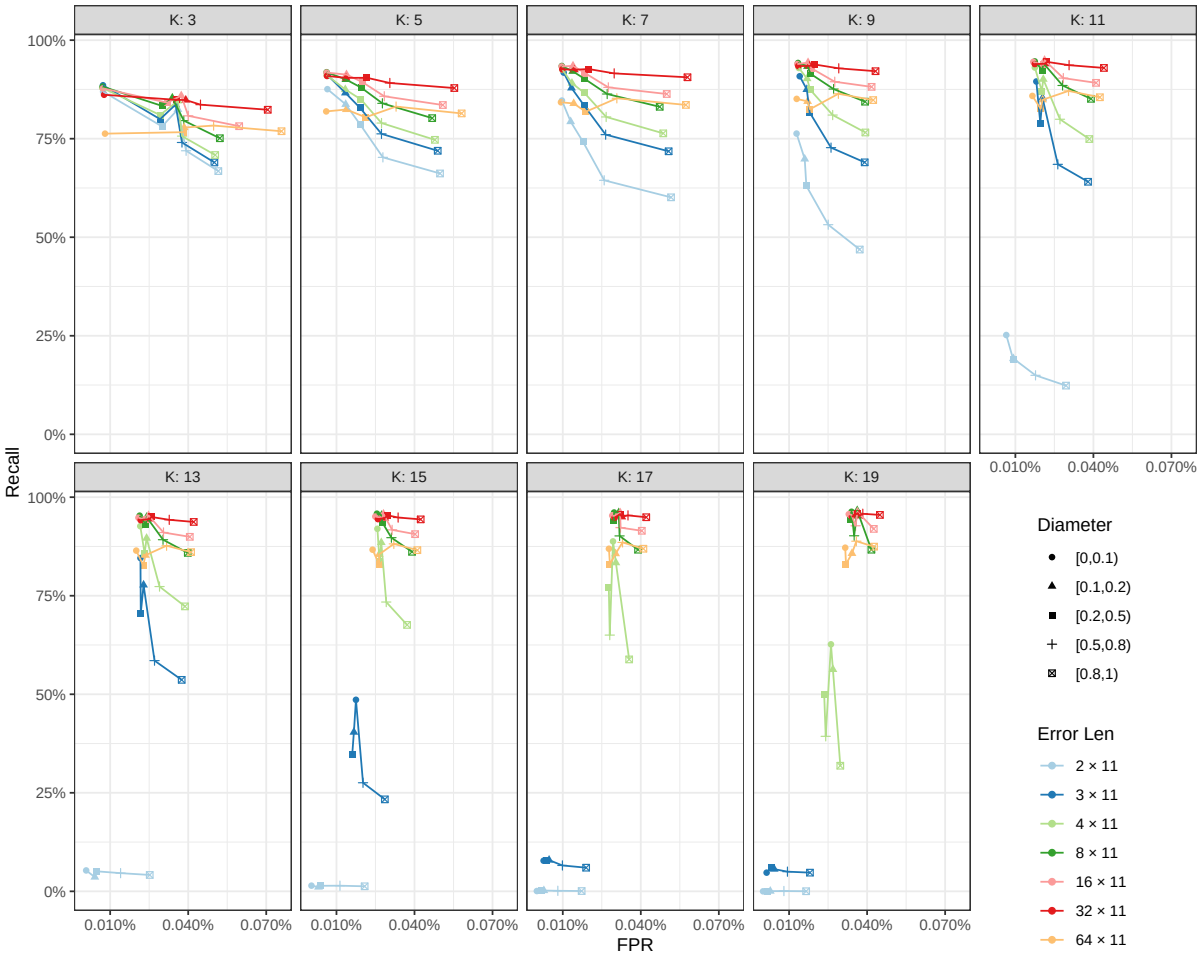

Figure S6.2. Accuracy of TAPER as we change the parameter k (fixing $p = 0.1$ and $q = 0.5$). Smaller k is effective for finding shorter errors (e.g., 2×11) but less so for finding longer errors (e.g., $\geq 8 \times 11$). The false positive rate can increase substantially if k is small and errors are long, and the recall is not ideal in those situations. In contrast, larger k (e.g. $k \geq 9$) is not effective for small errors but can be very effective for longer ones; note how FPR reduces for longer errors when k reaches 9.

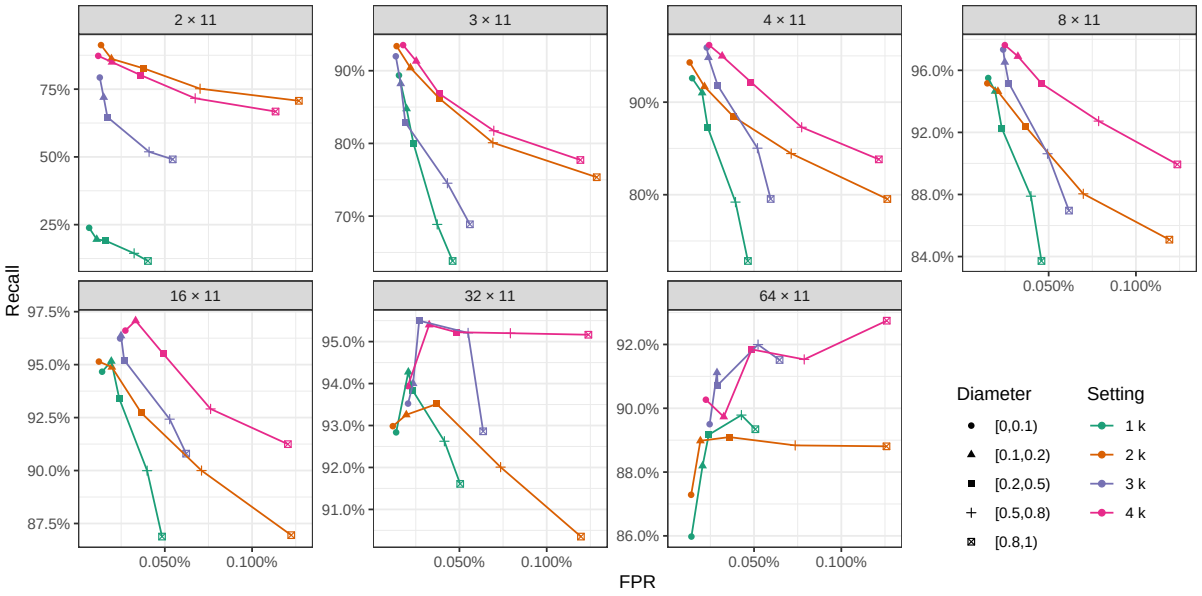

Figure S6.3. A comparison of various strategies for selecting k as the length of error changes (boxes). We either use a single value of $k = 11$ or the union of 2, 3, or 4 values of k. When using unions of multiple ks, we take results for each k only at a certain range of error lengths; for 2k setting: $k = 5$ for error length $[0, 30]$, and $k = 9$ for other lengths; For 3k setting: $k = 5$ for error length $[0, 30]$ $k = 9$ for error length $[0, 54]$ $k = 19$ for other lengths; For 4k setting: $k = 5$ for error length $[0, 20]$ $k = 7$ for error length $[0, 35]$ $k = 11$ for error length $[0, 66]$ $k = 17$ for other lengths. Using one k has very low recall in the case with short error length. The other three settings do not universally dominate each other (there is tradeoff between FPR and recall). Overall, the 2k setting seems to have substantially less recall than 4k, with small advantages in FPR. 3k setting generally has better FPR than the other two methods, but also slightly lower recall. Overall, to protect against FP error filtering, we chose the 3k setting that provides a balance between high recall and low FPR. See Figure S6.4 for more details.

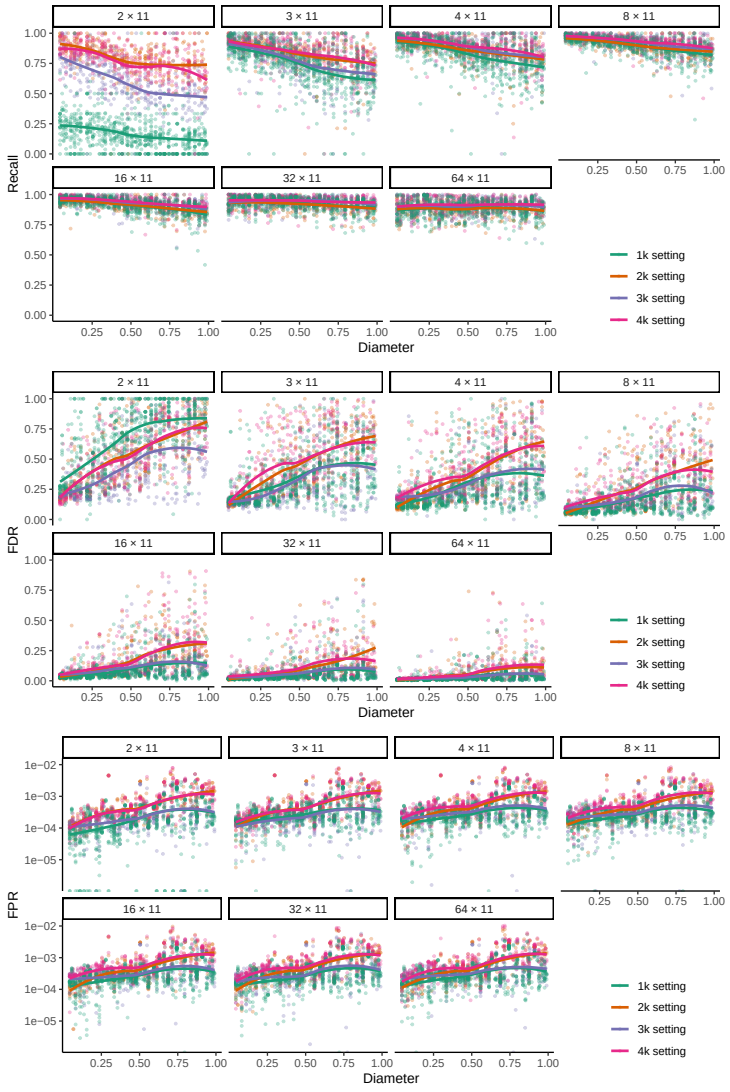

Figure S6.4. A comparison of various strategies for selecting k as the length of error changes (boxes). We either use a single value of $k = 11$ or the union of 2, 3, or 4 values of k. When using unions of multiple ks, we take results for each k only at a certain range of error lengths; for 2k setting: $k = 5$ for error length $[0, 30]$, and $k = 9$ for other lengths; For 3k setting: $k = 5$ for error length $[0, 30]$ $k = 9$ for error length $[0, 54]$ $k = 17$ for other lengths; For 4k setting: $k = 5$ for error length $[0, 20]$ $k = 7$ for error length $[0, 35]$ $k = 11$ for error length $[0, 66]$ $k = 17$ for other lengths. Using one k has very low recall in the case with short error length. The other three settings do not universally dominate each other (there is tradeoff between FPR and recall). Overall, the 2k setting seems to have substantially less recall than 4k, with small advantages in FPR. 3k setting generally has better FPR than the other two methods, but also slightly lower recall. Overall, to protect against FP error filtering, we chose the 3k setting that provides a balance between high recall and low FPR.

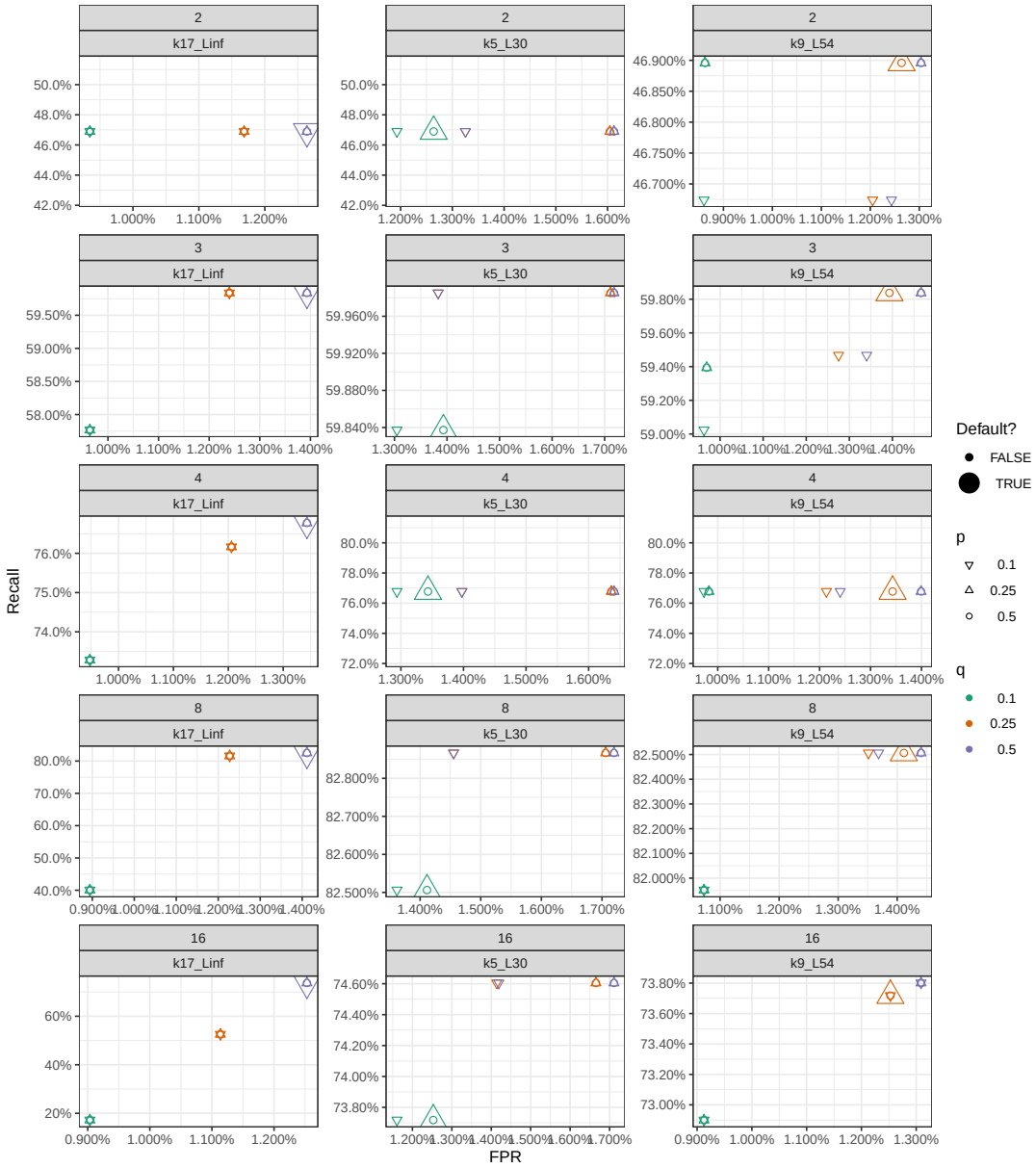

Figure S6.5. On the one AA dataset used in this paper, for only one of the replicates, we tested the impact of changing p and q for each of the three settings of k. Each column of plots corresponds to keeping p and q fixed to the default versions for all but one of the k values, given in the box header. Then, for error of length 2, 3, 4, 8, or 16 $\times$11, we change values of p and q and compute FPR and Recall. The default setting for each k is shown using large symbols and others using smaller symbols. For $k = 17$, it is clear that we need $q = 0.5$ to get good recall with longest errors. With $k = 5$, the default setting is not the best but is not far from having the lowest FPR and all recall values are very close. With $k = 9$, neither recall nor FPR are affected much, but marginally better settings do seem to be available. Note that these settings are not optimized for this dataset (or, in fact, any dataset) since profiles of the error are expected to be very different across datasets.

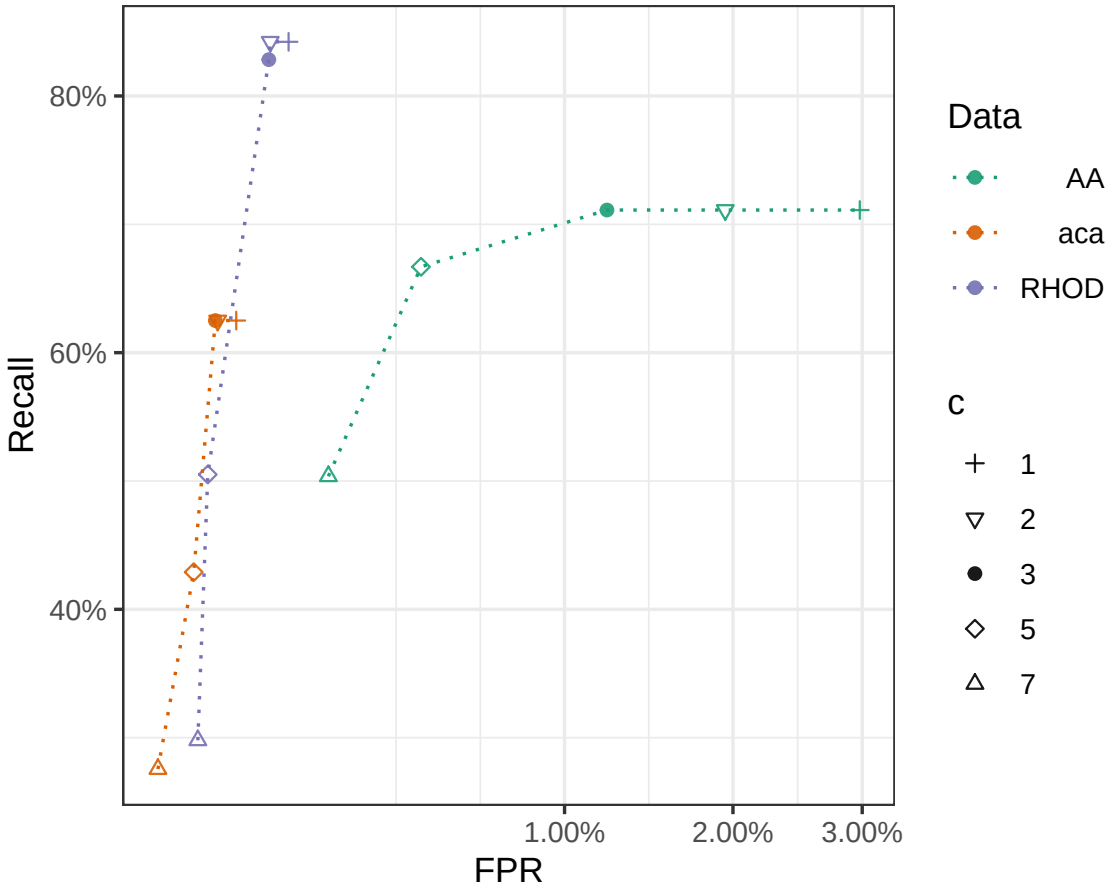

Figure S6.6. On the one AA dataset as well as two genes from the early-bird dataset, for only one of the replicates, we tested the impact of changing c. The default setting $c = 3$ is shown using solid dots and others using hallow symbols. For DNA datasets, using higher c dramatically reduces recall but further decreasing c only increases FPR without increasing recall. Thus, $c = 3$ is a clearly preferred setting. For AA dataset, setting $c < 3$ has no benefits but increases recall. However, values of $c > 3$ do reduce FPR, at some expense to recall. The best choice depends on the level of tolerance for FPR. We believe the default presents a reasonable trade-off. The x-axis is in square root scale.

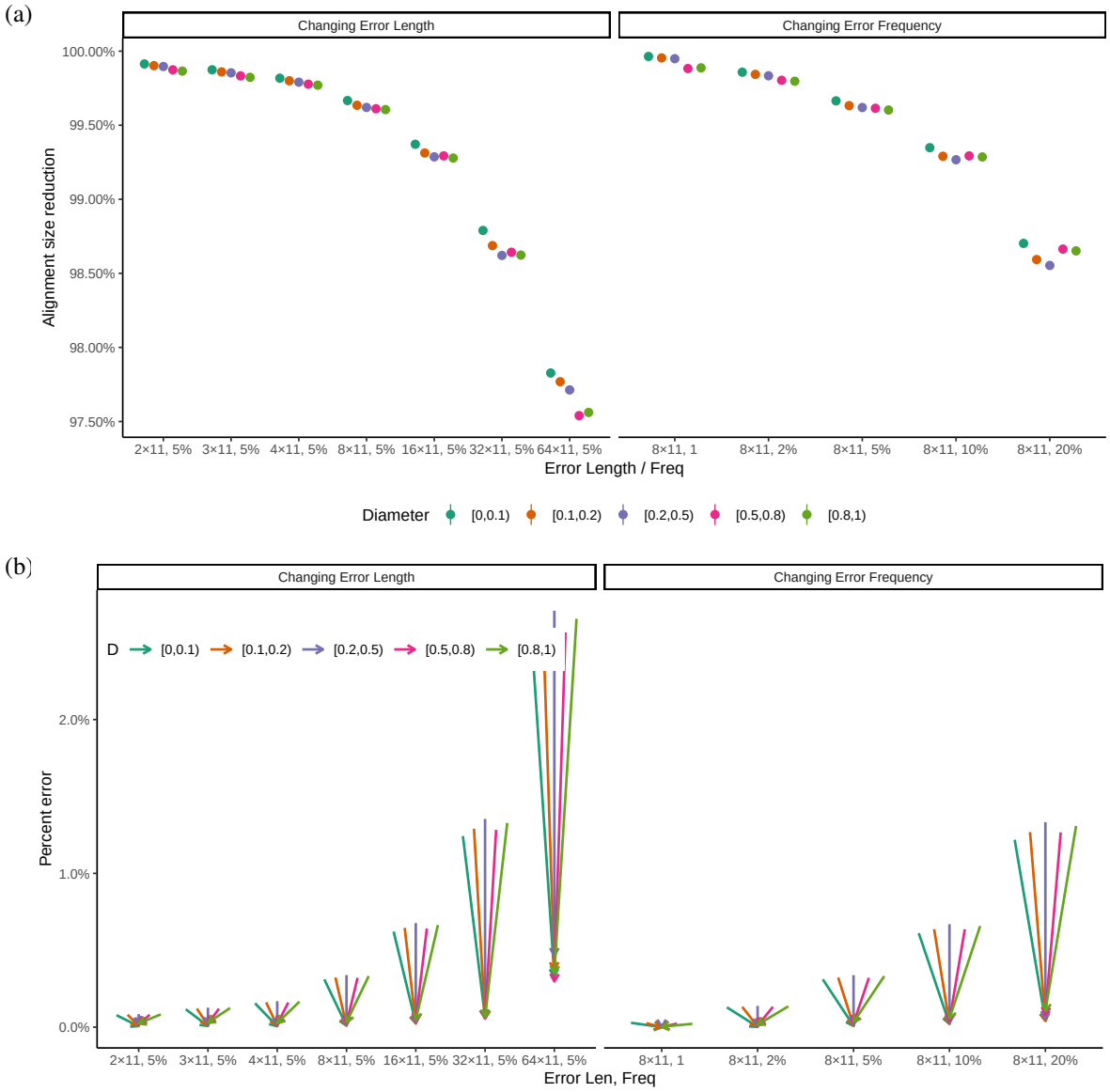

Figure S6.7. (a) Percentage of the alignment remaining after filtering (the total number of non-gap nucleotide positions in the alignment after divided by before filtering) across model conditions as the error length and frequency changes on the 16S.B dataset. (b) Similar to Figure 6.2b, we show change in percent error but without log scale.

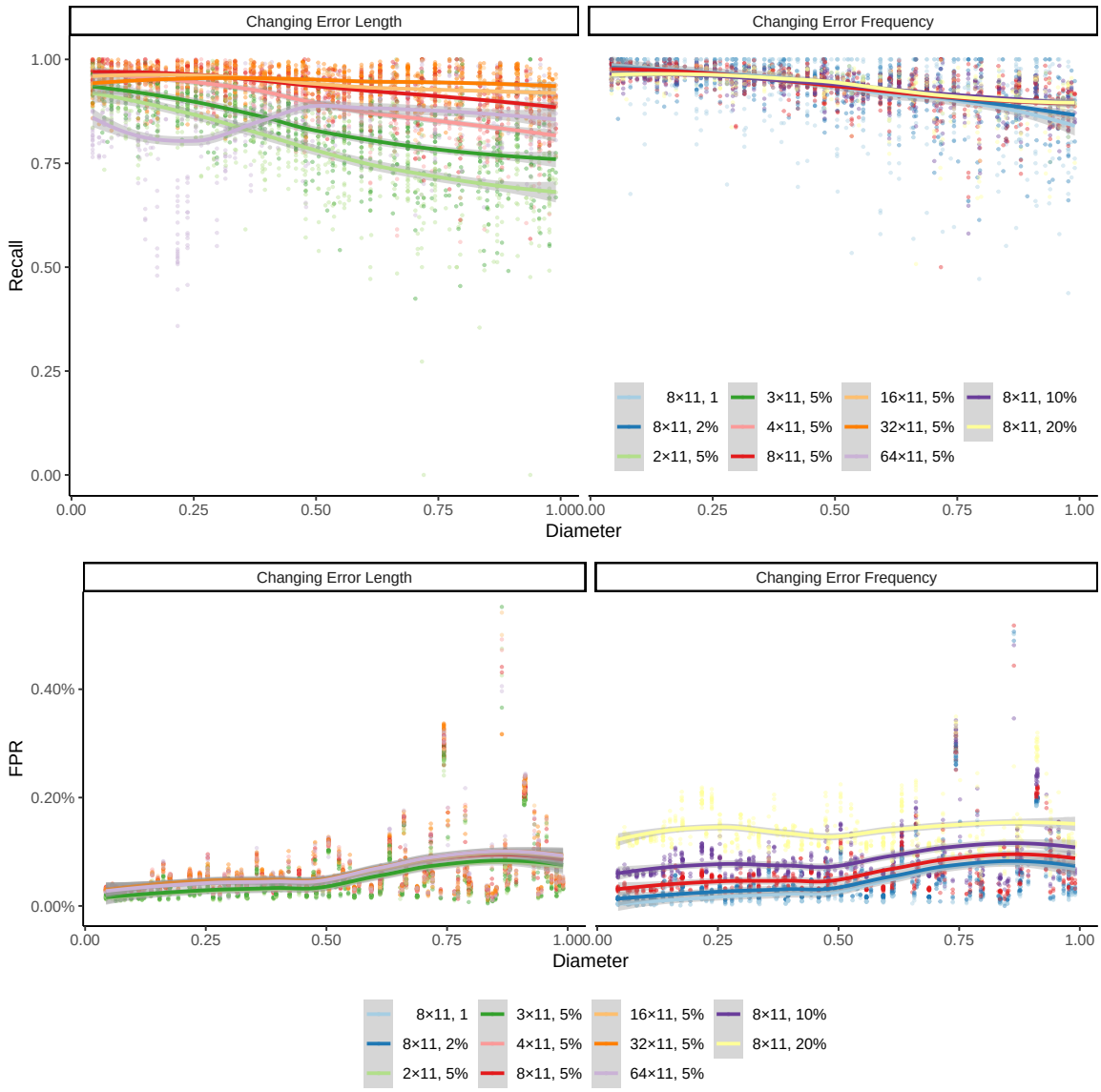

Figure S6.8. Impact of diameter on Recall and FPR on the 16S dataset.

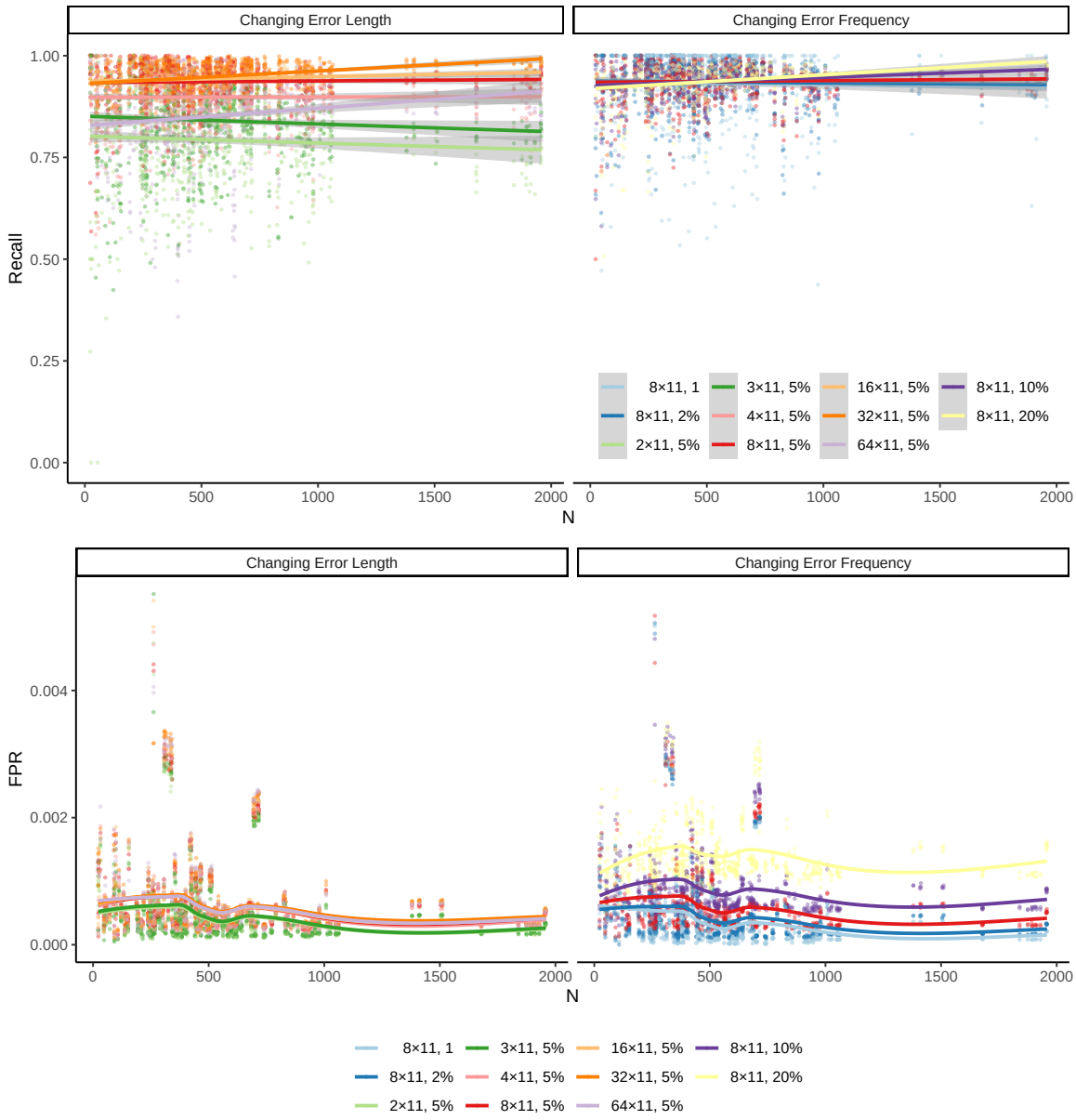

Figure S6.9. Impact of sequence count on the Recall and FPR on the 16S dataset.

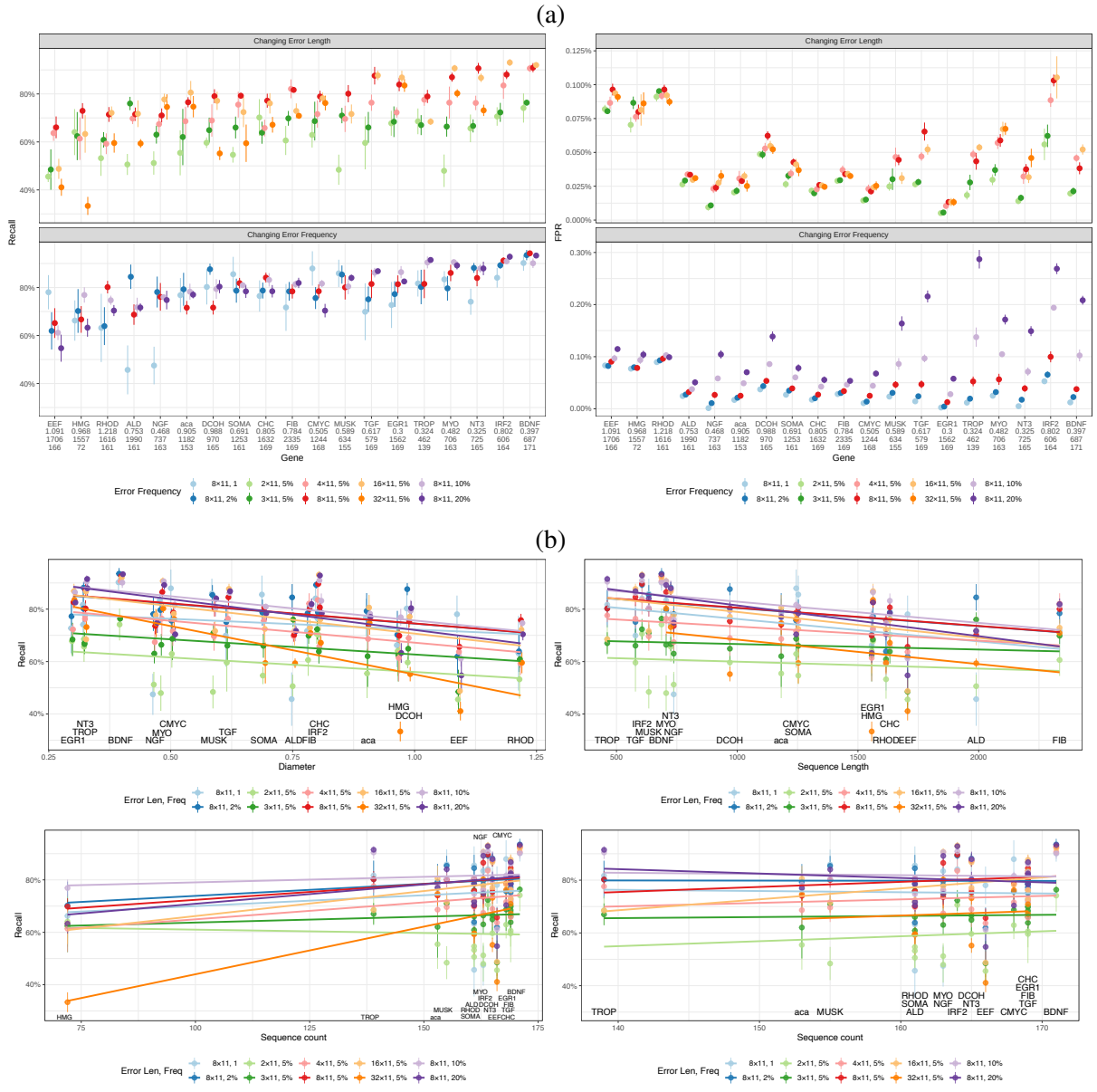

Figure S6.10. a) Recall and FPR of all genes as we change sequence error or error length. Genes are sorted by their average recall. We show diameter, mean sequence length,and the number of species for each gene under its name. b) Impact of Diameter, sequence count (top left), sequence length (top right), and sequence count (bottom) on the recall. For Sequence count, we show results with and without the outlier gene HMG.

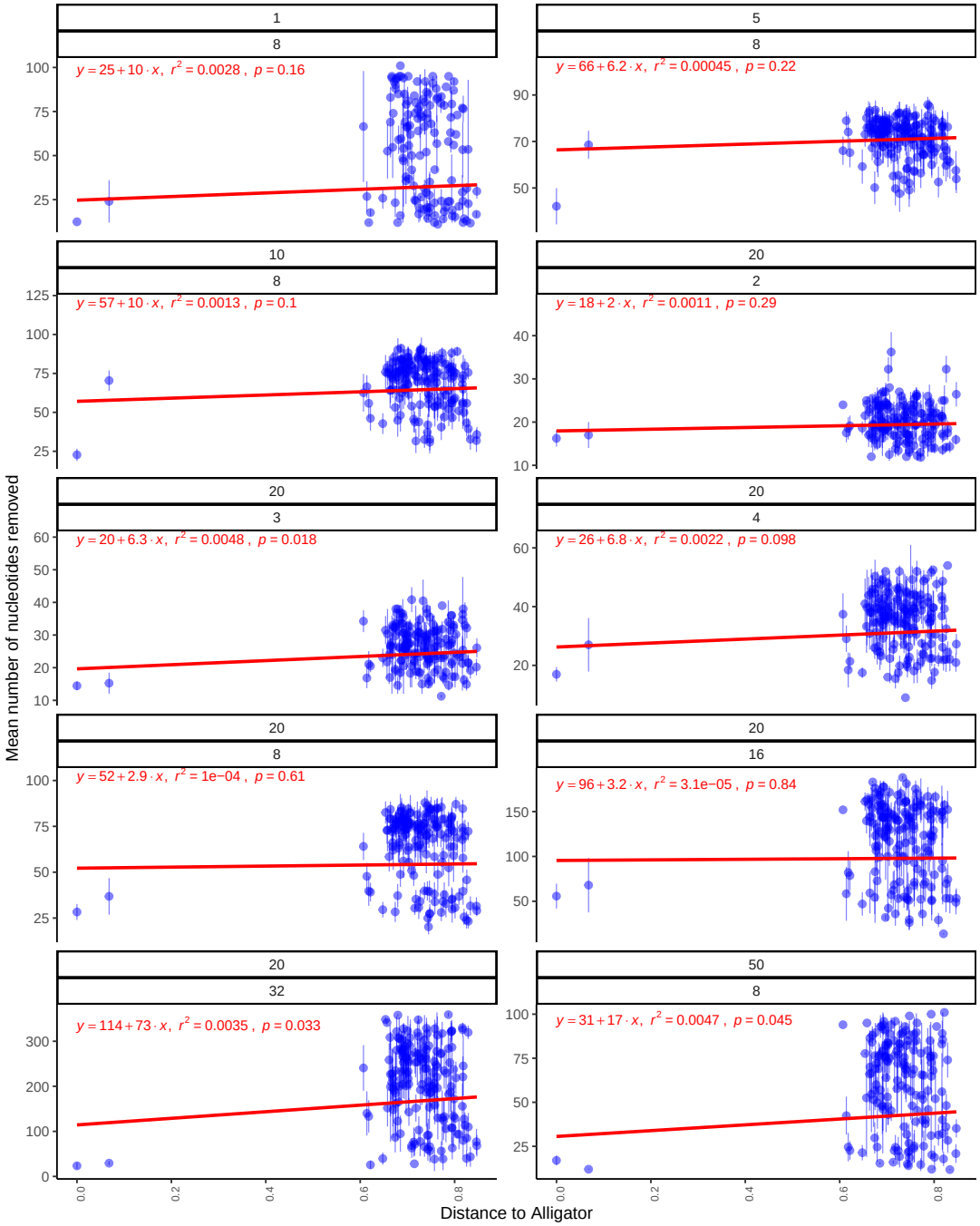

Figure S6.11. Removal of species from the dataset. x-axis: the distance of a species to the outgroup (Alligator) according to the published concatenation species tree. y-axis: the mean number of nucelotides removed from each species across different genes for each of the error profiles (boxes). No discernible impact is observed between the distance to the outgroup and what species are removed. There may be a slight *reduction* in the propensity to remove species that are more divergent from the rest.

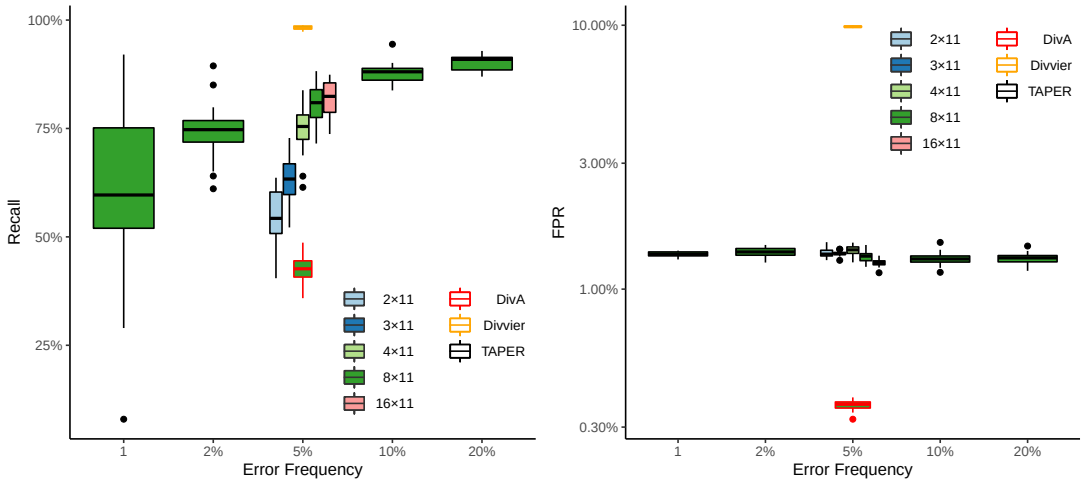

Figure S6.12. Results on AA dataset. Note that DivA and Divvier are only run on the default error profile (5% frequency and 8×11 length).

Figure S6.13. The AA alignment RV100_BBA0039 from the BALIBASE benchmarking dataset. The alignment includes a minority of sites that look very different from the rest.

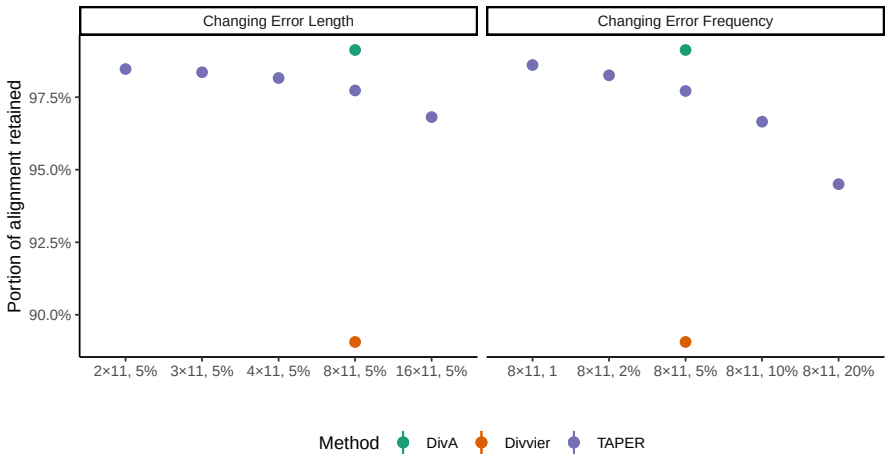

Figure S6.14. The AA alignment RV100_BBA0039 from the BALIBASE benchmarking dataset. The alignment includes a minority of sites that look very different from the rest.

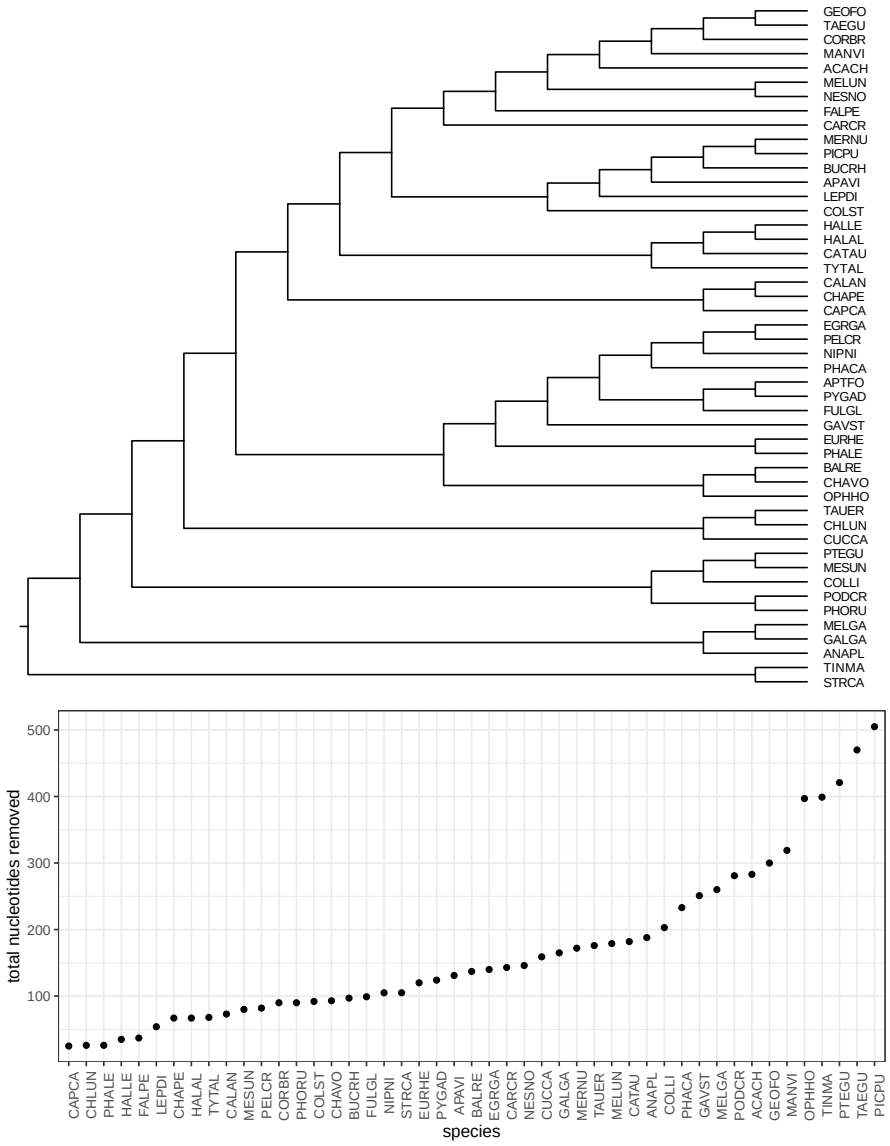

Figure S6.15. The number of nucleotides removed from species (bottom) does not correspond to phylogenetic relationships (top); in particular, the two outgroups, ostrich (STRCA) and tinamu (TINMA) are not removed more often than others. The species tree shown is obtained using ASTRAL-III run on all $>$ 14,000 input gene trees after contracting branches with support no more than 10%; the tree was reported by Zhang et al. (2018).

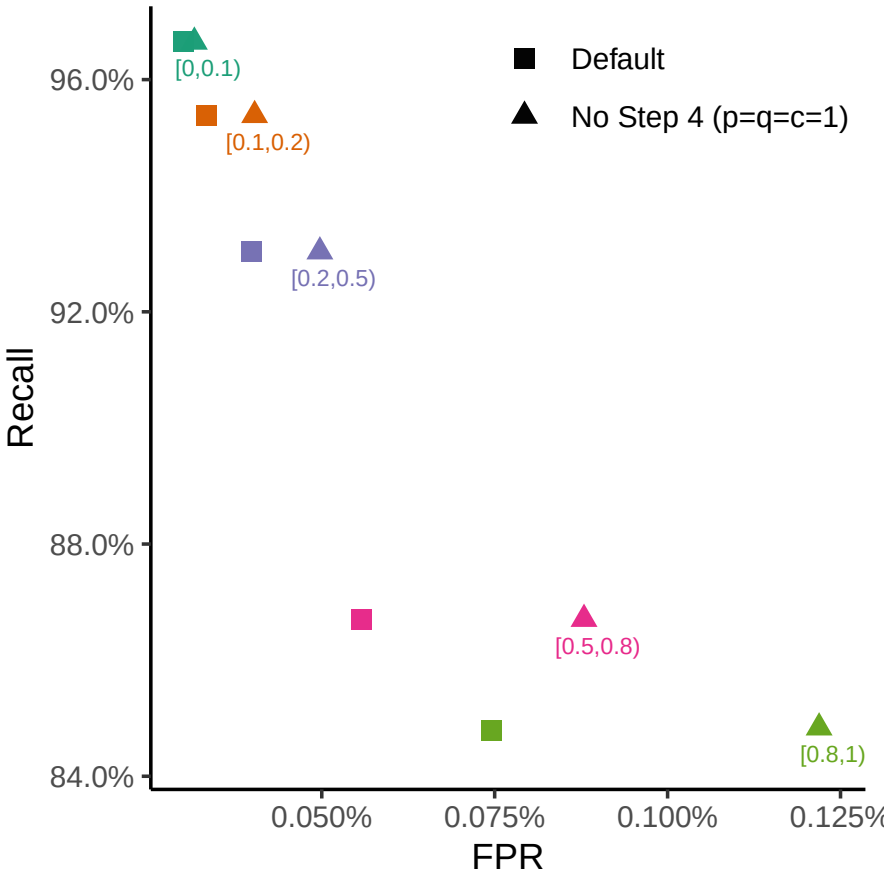

Figure S6.16. Impact of step 4. On the 16SB dataset with normally distributed error profiles, we examine the default setting of TAPER versus a version where step 4 is turned off by setting $p = q = c = 1$. Note that skipping this step keeps the recall fixed but increases the FP rate substantially.

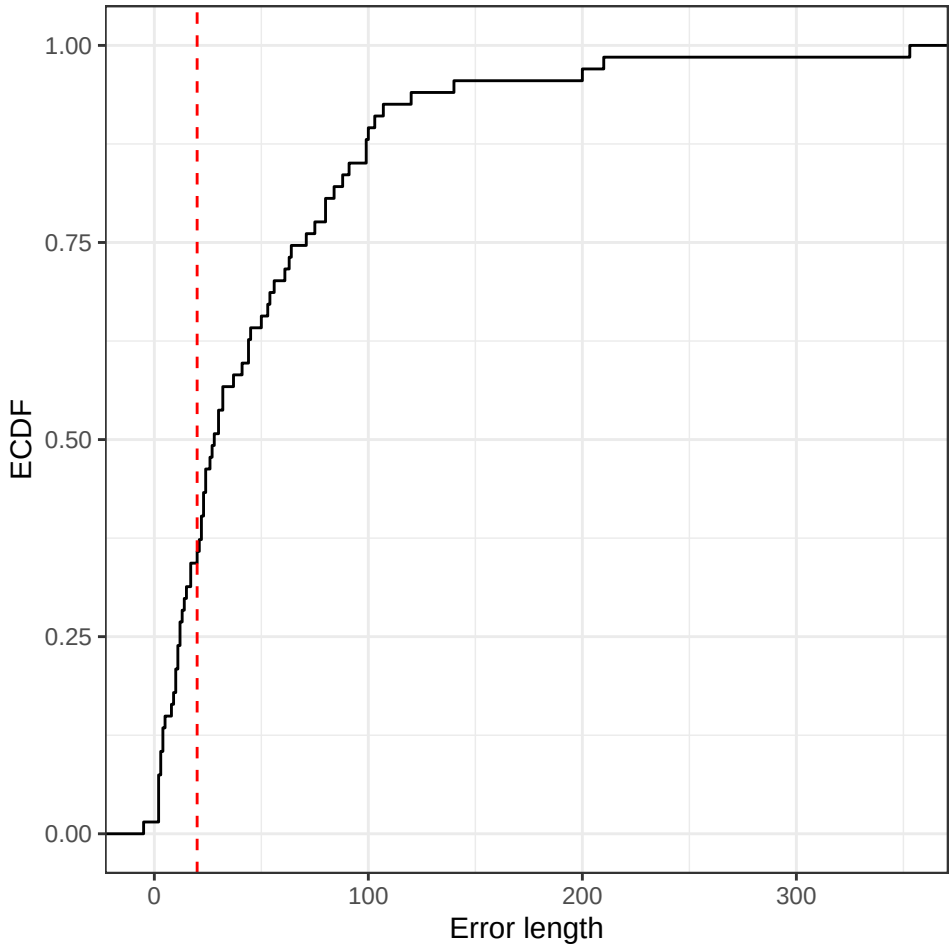

Figure S6.17. Distribution of the error length on the empirical dataset. The dotted line shows the 20nt threshold.

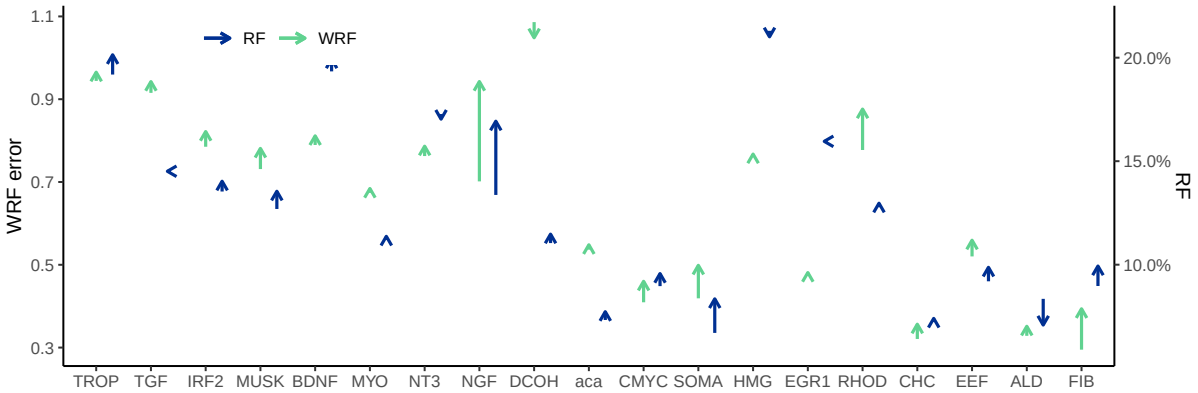

Figure S6.18. Tree Error change by TrimAl. Reduction in WRF and normalized RF error before and after filtering by trimAl shown as arrows for each gene. Error profile is fixed to $8 \times 11,5\%$.

Supplementary Tables

Table S6.1. ANOVA test on the 16S dataset, showing impact of four factors and their interactions: Error Length (ErrLen), Error Frequency (n), Diameter, and Sequence Count (N). X:Y corresponds to interactions of variables X and Y.

	Df	Sum Sq	Mean Sq	F value	Pvalue	PctExp
ErrLen	6	21.17	3.53	991.32	0.000000	31.9
n	4	0.01	0.00	0.99	0.410483	0.0
Diameter	1	7.73	7.73	2171.08	0.000000	11.6
N	1	0.32	0.32	90.54	0.000000	0.5
ErrLen:Diameter	6	4.94	0.82	231.22	0.000000	7.4
n:Diameter	4	0.13	0.03	9.04	0.000000	0.2
ErrLen:N	6	0.13	0.02	6.15	0.000002	0.2
n:N	4	0.11	0.03	7.75	0.000003	0.2
Diameter:N	1	0.03	0.03	9.44	0.002130	0.1
ErrLen:Diameter:N	6	0.24	0.04	11.10	0.000000	0.4
n:Diameter:N	4	0.00	0.00	0.17	0.951870	0.0
Residuals	8860	31.53	0.00			47.5

Table S6.2. ANOVA test on the early-bird dataset, showing impact of five factors and their interactions: Error Length (ErrLen), Error Frequency (n), Diameter, Sequence Length (SL), and Sequence Count (N). X:Y corresponds to interactions of variables X and Y.

	Df	Sum Sq	Mean Sq	F value	*p*-value	% Var Exp
ErrLen	5	4.65	0.93	66.21	0.000000	20.3
n	4	0.23	0.06	4.11	0.002645	1.0
Diameter	1	2.13	2.13	151.73	0.000000	9.3
SL	1	0.29	0.29	20.62	0.000006	1.3
N	1	0.29	0.29	20.77	0.000006	1.3
ErrLen:Diameter	5	0.28	0.06	3.96	0.001478	1.2
n:Diameter	4	0.08	0.02	1.49	0.202735	0.4
ErrLen:SL	5	0.16	0.03	2.23	0.049866	0.7
n:SL	4	0.04	0.01	0.78	0.538640	0.2
Diameter:SL	1	0.17	0.17	12.38	0.000455	0.8
ErrLen:N	5	0.27	0.05	3.81	0.002017	1.2
n:N	4	0.03	0.01	0.51	0.729552	0.1
Diameter:N	1	0.04	0.04	3.14	0.076594	0.2
SL:N	1	0.18	0.18	12.99	0.000330	0.8
ErrLen:Diameter:SL	5	0.08	0.02	1.16	0.327340	0.4
n:Diameter:SL	4	0.02	0.00	0.31	0.868535	0.1
ErrLen:Diameter:N	5	0.09	0.02	1.30	0.260338	0.4
n:Diameter:N	4	0.06	0.01	1.07	0.371634	0.3
ErrLen:SL:N	5	0.15	0.03	2.19	0.052842	0.7
n:SL:N	4	0.14	0.04	2.50	0.041228	0.6
Diameter:SL:N	1	0.03	0.03	2.31	0.128471	0.1
ErrLen:Diameter:SL:N	5	0.14	0.03	2.06	0.067832	0.6
n:Diameter:SL:N	4	0.08	0.02	1.40	0.230659	0.3
Residuals	940	13.20	0.01			57.8

Table S6.3. Errors identified by Springer and Gatesy (2018) that TAPER is able to detect fully (Found), mostly (Majority), or to a lesser degree (Minority). Red: Error is either too short (length ≤ 10) or involves too many sequences (≥ 10). Orange: Error involves somewhat high numbers of sequences (between 5 and 10). †: erroneous homology is restricted to a subset of the region identified by Springer and Gatesy (2018). See the supplementary file `zhang-taper-supplementary-error-pictures.xlsx` for pictures of errors found.

Gene	Positions	L	n	Description of error by Springer and Gatesy (2018)	Found
1066	859-1212	353	2	Partial intron 8 (Pterocles, Podiceps) is aligned with different region of intron 8 in Balearica (no homology with other sequences) and exons 9-11 in remaining taxa	Full
82	413-498	85	2	*Manacus and Acanthisitta exon 6 is combination of intron 5 (5') and exon 6 (3')*	Full
1087	673-753	80	2	Intron 8 (Columba, Anas) aligned against exon 8 in other taxa	Full
1077	1150-1206	56	2	Intron 11 (Melopsittacus, Colius) aligned with exon 10 in other taxa	Full
1028	334-386	53	2	*Gavia and Struthio (intron 3) aligned with exon 3 in other taxa*	Full
1054	577-707	30	2	Part of intron 5 (Merops, Nestor) is aligned against exons 6 and 7 in others	Full
1079	184-201	17	2	3' end of intron 2 (Eurypyga, Haliaeetus leucocephalus) aligned against exon 2 in other taxa	Full
82	136-146	11	2	Intron 2 (Cathartes, Tauraco) is aligned with exon 2 in other taxa	Full
16	2212-2213	2	2	5 end of intron 21 in Geospiza and Acanthisitta is aligned against 5 end of exon 22 in other taxa	Full
1039	1784-1999	210	3	Exon 14 sequences in Gavia, Phalacrocorax, and Opisthocomus are poorly aligned with other sequences and show different splice site boundaries	Full
1016	82-126	44	3	Part of intron 1 and exon 2 (Struthio, Tinamus, Pelecanus) aligned with exon 1 in others	Full
44	634-737	103	4	Exon 9 in Colius, Acanthisitta, Cariama, and Pterocles is a combination of the 5' region of intron 8 and the 3' region of intron 9	Full
1042	289-310	21	5	Exon 1 (Gallus, Haliaeetus leucocephalus, Melopsittacus, Tauraco, Columba) aligned against intron 1 in other taxa	Full
1098	161-309	140	2	Intron 1 (Opisthocomus, Phalacrocorax) aligned with exon 1 in other taxa	Majority
90	733-915	80	2	Intron 8 (Charadrius, Tauraco) aligned with exon 9 in other taxa	Majority
1014	705-768	64	2	Part of intron 7 (Eurypyga and Columba) aligned against exon 8 in others	Majority
1013	250-273	24	2	*Gallus and Tinamus not homologous with other sequences that are present (which are intron 1); problems with Meleagris exon 5*	Majority
93	124-139	15	2	3' end of intron 1 (Tyto, Pelecanus, Melopsittacus, Meleagris) is aligned with 3' end of exon 1 in others	Majority
30	691-762	71	3	Intron 3 (Phoenicopterus, Mesitornis, Leptosomus) aligned with exon 4 of other taxa	Majority
1044	342-441	99	4	Intron 5 (Opisthocomus, Eurypyga, Balearica, Cathartes) aligned against exon 5 in other taxa	Majority
1039	1307-1368	61	4	Intron 12 (Corvus, Columba, Pelecanus, Pygoscelis) aligned against exon 12 in others	Majority
89	740-796	54	4	Four taxa (Gallus, Eurypyga, Fulmarus, Manacus) have partial intron 6 sequence instead of partial exon 6	Majority
1087	1006-1019	13	5	5' end of intron 10 (Cuculus, Apaloderma, Columba, Colius, Anas) aligned against 5' end of exon 10 in others	Majority
1039	1360-1524	164	5?	Exon 11 sequences are poorly aligned across avian tree and have different intron-exon boundaries. The numbers of exons in different taxa also varies.	Majority
1077	448-583	120	2	Regions of intron 5 (?) and 6 (Melopsittacus, Colius) are aligned with exon 7 in others	Minority
1028	1158-1232	80	2	Sequences are not homologous in Merops and Nipponia versus other taxa	Minority
1089	367-421	50	2	Two taxa (Eurypyga, Pygoscelis) have intron 4 aligned against exon 4 in other taxa	Minority
99	401-506	100	7	Intron 3 (Columba, Pycoides, Chlamydotis, Tauraco, Pygoscelis, Cathartes, Tinamus) aligned against exon 3 in other taxa	Minority
1083	127-144	17	7	Seven taxa (Chlamydotis, Anas, Tinamus, Cariama, Tauraco, Acanthisitta, Opisthocomus) with intron 2 aligned against exon 2 in others	Minority
1097	912-1120	200	10	Intron 6 in some taxa (e.g., Tinamus) aligned with exon 6 in other taxa (e.g., Struthio)	Minority
28	1-81† (48-80)	32	13	Intron 2 in 13 taxa (e.g., Haliaeetus albicilla) is aligned with exon 1 and exon 2 in others (e.g., H. leucocephalus)	Minority
20	754-842	88	14	5' region of intron 1 (Gallus and others) aligned against 3' region of intron 1 (e.g., Falco and others); middle portion of intron 1 (Gallus group) aligned against 5' end of exon 2 (Falco group). Also some taxa in Gallus group (palaeognaths, Cathartes, Podiceps, Charadrius) are misaligned with others in group	Minority

Table S6.4. Errors identified by Springer and Gatesy (2018) and missed by TAPER. Notations same as Table S6.3. †: real error boundary. Red: Error is short ($\leq$ 10bp) or frequent ($\geq$ 10); Orange: Error is somewhat frequent ($\geq 5, \leq 10$).

Gene	Positions	L	n	Description of error by Springer and Gatesy (2018)
1040	106-138† (-129)	24	2	Intron 1 (Opisthocomus, Apaloderma) aligned against exon sequences in others
1085[1]	73-97	24	2	Two taxa (Anas, Nipponia) have intron 1 aligned against exon 1
62	93-129† (-105)	12	2	3' end of intron 1 in Cathartes and Cariama is aligned with 3' end of exon 1 in other taxa
51	478-489† (-487)	10	2	3' end of intron 2 in Balearica and Eurypyga is aligned with 5' end of exon 3 in other taxa
1045	415-425	10	2	Intron 3 in Colius and Pelecanus aligned with exon 3 in other taxa
1050[2]	1372-1395	23	3	Intron 12 (Chaetura, Melopsittacus, Acanthisitta) versus exon 13 (Cuculus, Tinamus, Gallus, Nestor, Taeniopygia, Geospiza)
12	1-324† (300-)	20	3	Intron 1 (Eurypyga, Fulmarus, Columba) aligned against exon 1 in other taxa
1078	98-128† (-117)	10	4	3' end of exon 1 in some taxa (Tauraco, Taeniopygia, Eurypyga, Apaloderma) in some taxa aligned against 5' end of exon 2 in various others
1038[3]	230-252	22	4	Part of intron 2 (Fulmarus, Melopsittacus, Nestor, Tyto) is aligned with exon 2 in other taxa
1001[4]	1596-1617	21	4	Intron 14 (Merops, Tauraco, Charadrius, Mesitornis) aligned with exon 14
64[5]	1-147† (133-)	14	4	Exon 1 (Pycoides, Melopsittacus, Gallus, Manacus) is aligned with part of intron 1 (Nestor, Chlamydotis, Mesitornis, Buceros, Merops, Falco, Corvus)
88	1114-1122	8	4	Four taxa (Taeniopygia, Caprimulgus, Eurypyga, Chlamydotis) have intron sequence
1036	838-921	84	5	Sequences in Aptenodytes, Gavia, Phoenicopterus, Chlamydotis, and Phaethon are not homologous to other sequences (exon versus intron sequences are unclear)
1044	25-52	27	5	Intron 1 (Fulmarus, Gavia, Opisthocomus, Colius, Eurypyga) aligned against exon 1 in others
1094	1069-1095	26	5	Five taxa (Gavia, Apaloderma, Chaetura, Pterocles, Pycoides) with intron 7 and five others with exon 7 (Gallus, Meleagris, Pelecanus, Podiceps, Merops)
1062	58-70	12	5	Exon 1 (Cariama, Meleagris, Manacus, Gallus, Colius) versus intron 1 in other taxa
56	106-115	10	5	3' end of exon 1 in Taeniopygia, Columba, Corvus, Geospiza, and Gallus is aligned with 3' end of intron 1 in other taxa
1056	157-180	23	6	Exon 1 (Charadrius, Melopsittacus, Gallus, Cuculus, Taeniopygia, Columba) versus intron 1 in other taxa
1085	187-191	4	6	Six taxa with intron 2 (Charadrius, Buceros, Opisthocomus, Merops, Haliaeetus albicilla, Egreta) aligned against exon 2
1038	76-115	41	7	Part of intron 1 (Egreta, Apaloderma, Podiceps, Balearica, Eurypyga, Phalacrocorax, Caprimulgus) aligned against exon 1 in other taxa (region also missing for many taxa)
84	124-129	5	7	Seven taxa with different intron 1 boundaries (Fulmarus, Anas, Phoenicopterus, Balearica, Gavia, Charadrius, Struthio) versus other taxa
1020	84-174	90	8	Part of intron 2 (Phaethon, Nipponia, Aptenodytes, Phoenicopterus, Cariama, Columba, Mesitornis, Falco) aligned with exon 2 (Pycoides, Calypte, Meleagris, Charadrius, Egreta, Gallus, Haliaeetus leucocephalus, Cuculus)
37	262-312† (-273)	9	8	3' region of exon 1 (or 5' region of intron 1?) in eight taxa (Gallus, Falco, Columba, Balearica, Buceros, Taeniopygia, Geospiza, Cuculus) aligned with 5' region of exon 2 (or 3' region of exon 1?) in other taxa
1083	43-45	2	9	Nine taxa (Leptosomus, Chaetura, Corvus, Taeniopygia, Cuculus, Phaethon, Eurypyga, Balearica, Egreta) share 3 bp from 3' end of intron 1, which is aligned with last 3 bp of exon 1 in many other taxa (also some with missing data)
1089	25-69	44	10	Seventeen taxa with intron 1 sequences aligned against 10 taxa with exon 1 sequences
5	147-161	14	10	Exon 1 (Meleagris, Gallus, Fulmarus, Cariama, Pygoscelis, Taeniopygia, Aptenodytes, Haliaeetus leucocephalus, Colius) aligned against intron 1 (other taxa with sequence)
1029	58-85	28	11	Intron 1 (Opisthocomus, Anas, Podiceps, Eurypyga, Cariama, Aptenodytes, Tauraco, Caprimulgus, Leptosomus, Tinamus, Merops) aligned against exon 1 (Struthio, Falco, Haliaeetus leucocephalus, Gavia, Phaethon, Pygoscelis, Gallus, Meleagris)
74	241-252	3	11	5' end of intron 2 in Eurypyga, Corvus, and Taeniopygia is aligned with 5' end of exon 3 in other taxa
1005	121-124	3	11	Exon 1 (Fulmarus, Podiceps, Melopsittacus, Cathartes, Meleagris, Falco, Tinamus, Gallus, Haliaeetus leucocephalus, Cariama) aligned with intron 1 in other taxa
1014	787-849	75	14	Part of intron 7 (Acanthisitta, Balearica, Colius, Meleagris, Pterocles, Apaloderma, Eurypyga, Columba, Pygoscelis, Falco, Opisthocomus, Caprimulgus, Nipponia, Leptosomus) aligned against exon 8 in others.
1027	1055-1153	99	16	Part of intron 8 in 16 taxa (e.g., Struthio, Pygascelis, Galga) aligned against non-homologous region in other taxa (e.g., Tinamus, Aptenodytes, Meleagris)
1025	412-518	106	16	Part of intron 4 (Calypte, Leptosomus, Phalacrocorax, Balearica, Chlamydotis, Pelecanus, Fulmarus, Cariama, Tyto, Anas, Aptenodytes, Columba, Apaloderma, Tauraco, Corvus, Acanthisitta) aligned with part of exon 5(?) in Haliaeetus leucocephalus, Nipponia, Gallus, Geospiza, Taeniopygia, and Tinamus (and to a lesser extent with other taxa that are largely missing in this region)
1035	58-64	6	17	Intron 1 (Tauraco, Phaethon, Caprimulgus, Fulmarus, Tyto, Pelecanus, Gavia, Aploderma, Haliaeetus albicilla, Leptosomus, Balearica, Podiceps, Charadrius, Pygoscelis, Struthio, Phoenicopterus, Egreta) aligned against exon 1 in other taxa
1067	529-532	4	20	First four bp of intron 5 in 20 taxa (e.g., Gallus, Pelecanus) aligned against first four bp of exon 6 in 28 taxa (e.g., Meleagris, Egretta)
1037	361-408	45	20?	5' and 3' ends of intron 2 are included for some taxa are are misaligned across avian alignment, including misalignment with last 3 bp of exon 2
1049	2725-2762	37	20?	Mix-up of intron 8 and exon 9 sequences

[1] 2 out of 6 aligned have the error; thus, high frequency.

[2] 3 out of 9 aligned have the error; thus, high frequency.

[3] Starts at 247 for 3 of the four species; thus, short for most sequences.

[4] 6 columns in the middle are all gaps, others are not visually clear errors.

[5] 4 out of 11 aligned have the error; thus, high frequency.

Chapter 7

Scalable Models of Antibody Evolution and Benchmarking of Clonal Tree Reconstruction Methods

Affinity maturation (AM) of antibodies through somatic hypermutations (SHMs) enables the immune system to evolve to recognize diverse pathogens. The accumulation of SHMs leads to the formation of clonal lineages of antibodies produced by B cells that have evolved from a common naive B cell. Advances in high-throughput sequencing have enabled deep scans of antibody repertoires, paving the way for reconstructing clonal trees. However, it is not clear if clonal trees, which capture microevolutionary time scales, can be reconstructed using traditional phylogenetic reconstruction methods with adequate accuracy. In fact, several clonal tree reconstruction methods have been developed to fix supposed shortcomings of phylogenetic methods. Nevertheless, no consensus has been reached regarding the relative accuracy of these methods, partially because evaluation is challenging. Benchmarking the performance of existing methods and developing better methods would both benefit from realistic models of clonal lineage evolution specifically designed for emulating B cell evolution. In this paper, we propose a model for modeling B cell clonal lineage evolution and use this model to benchmark several existing clonal tree reconstruction methods. Our model, designed to be extensible, has several features: by evolving the clonal tree and sequences simultaneously, it allows modeling selective pressure due to changes in affinity binding; it enables scalable simulations of large numbers of cells; it enables several rounds of infection by an evolving pathogen; and, it models building of memory. In addition, we also suggest a set of metrics for comparing clonal trees and for measuring their properties. Our benchmarking results show that while maximum likelihood phylogenetic reconstruction methods can fail to capture key features of clonal tree expansion if applied naively, a very simple post-processing of their results, where super short branches are contracted, leads to inferences that are better than alternative methods.

7.1 Introduction

Immune response to new pathogens relies heavily on the *Affinity maturation* (AM) process. AM follows the binding of immunoglobulin (IG) molecules to antigens and improves

the affinity (i.e., binding ability) of B cells to the antigen (Tonegawa, 1983; Neuberger and Milstein, 1995). The AM process involves many aspects, including the activation of naive B cells that have not been exposed to an antigen, *clonal expansion* of cells that increases the pool of antibodies, *somatic hypermutations* (SHMs) (Muramatsu et al., 2000) that alter the structure of antibodies and their ability to bind, and a regulatory mechanism that plays the role of natural selection. The AM process creates *memory* and *plasma* B cells; memory B cells can be reactivated and can undergo the AM process again (Mesin et al., 2020), while plasma B cells can secrete massive levels of neutralizing antibodies. Over time, the AM process leads to the formation of clonal lineages within a given antibody repertoire, where each clonal lineage is formed by descendants of a single naive B cell. The evolutionary history of each of these clonal lineages can be represented by a *clonal tree*, where each vertex corresponds to a B cell, and a directed edge connects each B cell to all its immediate descendants.

New sequencing technologies have enabled high-throughput scanning of antibody repertoires (*Rep-Seq*) and have opened up new avenues for studying adaptive immune systems (Georgiou et al., 2014; Robinson, 2015; Yaari et al., 2015; Watson et al., 2017; Miho et al., 2018). Rep-Seq technologies enabled AM analysis of antibody repertoires responding to antigens of various diseases, such as flu (Laserson et al., 2014; Horns et al., 2019), HIV (Haynes et al., 2012; Sok et al., 2013b), hepatitis (Galson et al., 2016; Eliyahu et al., 2018), multiple sclerosis (Stern et al., 2014; Lossius et al., 2016), rheumatoid arthritis (Elliott et al., 2018). Such analyses allow biologists to identify broadly neutralizing antibodies and reveal antigen-specific and general mutation patterns (Horns et al., 2019; Hsiao et al., 2019).

Due to the short time frame of clonal expansion, inferred clonal trees have unique properties. Some sequenced nodes may belong to the internal nodes of the tree instead of the tips. Also, inferred clonal trees are often not even close to bifurcating. Thus, unlike traditional phylogenetics, perhaps Steiner trees (which can put observations at *some* of the internal nodes) or spanning trees (that put an observation at *all* internal nodes) should be preferred for reconstructing antibody sequences (Fig. 7.1a). Various reconstruction methods have been developed attempting

to recover clonal trees from antibody sequences (e.g., Jiang et al., 2013; Sok et al., 2013a; Lee et al., 2017; Hoehn et al., 2017; Horns et al., 2016; Lees and Shepherd, 2015; DeWitt et al., 2018). Some of these methods use simple clustering methods (e.g., Jiang et al., 2013), while others formulate the problem as a Steiner tree problem (Sok et al., 2013a; Lee et al., 2017; Horns et al., 2016; DeWitt et al., 2018) or maximum-likelihood (ML) phylogenetic tree reconstruction under models of sequence evolution (Hoehn et al., 2017; Lees and Shepherd, 2015).

In order to evaluate methods proposed for reconstructing clonal trees, we need models for antibody sequence evolution and clonal tree expansion that can be used for simulation. This modeling step is challenging for several reasons. (i) Selection, which is an integral part of AM, needs to be modeled directly; otherwise, the shape of the resulting trees will not be realistic. Traditional phylogenetics simulations first simulate a tree of sampled taxa and then evolve sequences down the tree. This two-step approach simplifies simulations but misses the dependency between the clonal tree shape and the antibody sequences. A better approach is to co-evolve the tree and sequences. The challenge in co-evolving is to design a principled model for how sequences impact evolution and to develop a scalable simulation algorithm that can generate large numbers of cells. (ii) Literature suggests that there are hotspots and cold spots of SHMs (e.g., Rogozin and Kolchanov, 1992; Pham et al., 2003). However, traditional models of sequence evolution assume each site evolves independently and will miss the context dependence. (iii) Different antibody cell types(e.g., activated and memory cells) have very different mutational and selection behaviors, and these distinctions need to be modeled.

There have been several attempts at designing statistical models of AM clonal expansion (e.g., Childs et al., 2015; Amitai et al., 2017; Reshetova et al., 2017; Davidsen and Matsen, 2018; Yermanos et al., 2018). As the AM process is complex, these models have taken different routes. For example, determining affinities of sequences to hypothetical antigens is difficult, as affinity binding is a complicated chemical process, and each method models affinity differently. Nevertheless, all these methods have limitations, which we will return to in our discussion section. Two factors worth pointing out are that they do not scale to very large numbers of cells,

and they allow for simulating one round of infection (as opposed to an evolving pathogen and recurring infections); some also avoid differentiating different types of B cells.

In this paper, we propose a scalable and flexible simulation framework that can be instantiated in many ways. We introduce a general birth, death, and transformation (BDT) model and describe how BDT can be instantiated to create a model of AM that simultaneously co-evolves the clonal tree and antibody sequences. We then introduce a scalable sampling algorithm for our model that enables generating large trees. With the simulator (called DimSim) at hand, we note that comparing clonal trees and characterizing their properties require care. We refine existing metrics and define new ones for characterizing properties (e.g., balance) of clonal trees and for comparing them. Finally, we perform extensive simulation studies (Fig. 7.1b) under various parameters using DimSim. We study how the parameters of the AM model impact properties of clonal trees and benchmark the performance of several reconstruction methods.

7.2 Methods

7.2.1 Statistical Models

We first define a general Birth/Death/Transformation (BDT) model and give an efficient algorithm for sampling trees from the BDT model. We then instantiate the general model for simulating AM processes and move on to describe specific choices we made in our simulations.

BDT Model

Forward-time birth-death models are used extensively in macro-evolutionary modeling (Nee, 2006), whereas microevolution simulations often use coalescent models that are easier to sample. We propose a general forward-time model that can allow realistic microevolutionary simulations by ensuring that birth and death rates are not constant and instead change with the properties of evolving units (e.g., cells).

In the BDT model, a set of *entities* continuously undergo birth (B), death (D), and transformation (T) events. Each entity i has a list of properties $\mathbf{x}_i \in \mathbb{R}_+^N$. At each point in time,

the system contains a set S of n active entities, and each active entity $i \in S$ undergoes birth, death, and transformation events according to independent Poisson point processes. In the birth event, a entity i is removed from S and new entities j and k, with properties $\mathbf{x}_j$ and $\mathbf{x}_k$, are added to S; properties $\mathbf{x}_j$ and $\mathbf{x}_k$ are drawn from a distribution determined by $\mathbf{x}_i$ and model parameters. In the event of the death of an entity i, it is removed from S. In the transformation event, a entity i is removed from S and a new entity j with properties $\mathbf{x}_j$, drawn from a distribution determined by $\mathbf{x}_i$, is added to S. Starting from a single node and continuously applied, this process defines a rooted tree where nodes are all entities that ever existed (including those that died); birth events create bifurcations, transformation events create nodes with one child, and death events create leaves. The tree can be subsampled subsequently.

For each entity $i \in S$, the birth, rate, and transformation rates are thoroughly determined by its properties $\mathbf{x}_i$ and the sum of properties over all entities $\mathbf{S} = \sum_{j \in S} \mathbf{x}_j$. We let $\Lambda_B(\mathbf{x}_i, \mathbf{S})$, $\Lambda_D(\mathbf{x}_i, \mathbf{S})$, and $\Lambda_T(\mathbf{x}_i, \mathbf{S})$ denote the birth, death, and transformation rates, respectively. In the time interval between two events for any two entities in the system, we assume a memoryless process. Thus, these rates remain constant between any two events but can change when an event happens. The ratio between the birth rate and the death rate, both of which are functions of the entity properties, can be thought of as the factor controlling the selective pressure, which can be time-variant.

Because of the memoryless property, the time until the next BDT event always follows the exponential distribution with rates $\Lambda_B(\mathbf{x}_i, \mathbf{S}), \Lambda_D(\mathbf{x}_i, \mathbf{S})$, and $\Lambda_T(\mathbf{x}_i, \mathbf{S})$ for each event type. The time until any event for any entity follows an exponential distribution with $\lambda = \sum_{i \in S}(\Lambda_B(\mathbf{x}_i, \mathbf{S}) + \Lambda_D(\mathbf{x}_i, \mathbf{S}) + \Lambda_T(\mathbf{x}_i, \mathbf{S}))$. The probability of the next event being a specific event $E \in \{B, D, T\}$ for a particular entity i is $\Lambda_E(\mathbf{x}_i, \mathbf{S})/\lambda$. Specifying the rate functions and the distribution of properties at the initial state fully specifies the model.

The BDT model can be efficiently sampled if rate functions have certain (very general) properties. We leave all the mathematical details for Appendix 7.B.1. In short, the memoryless properties of the model makes it possible perform efficient sampling despite the fact that rates

change with the tree. The main innovations of the sampling algorithm are: 1) rewriting rate functions as polynomial functions of other parameters, which enable finding the time to the next event in constant time, and 2) using an interval tree data structure to store partial sums needed for normalization. With the algorithm we propose (Algorithm S7.1 in Appendix 7.D), a tree on k nodes drawn from the distribution defined by the BDT process can be sampled in $O(k\log(k))$ time. Thus, the BDT model can be efficiently sampled to create trees with millions of nodes.

Antibody Affinity Maturation (AM) model

We now define a specific instance of the BDT model designed for AM. Simulations according to this AM model are implemented in a `C++` tool called Dynamic IMmuno-SIMulator (DIMSIM). The model has many parameters reflecting immune system properties (Table 7.1), which we define as we progress. The readers are referred to Appendix 7.A for our particular usage of terms commonly used in immunology. The use of birth death models for AM is not new (e.g., Davidsen and Matsen, 2018) but particular choices of our model are different from prior work.

Rounds and stages

We model the evolution of antibody-coding sequences in response to r rounds of infections by an evolving antigen (e.g., flu). Each round consists of two stages, an *infected* stage, where a set of new antigens initiate a response that activates the B cells being modeled, and a *dormant* stage, where the B cells being modeled are not actively involved in an immune response. Both stages used the same BDT model but are parameterized differently. The switch between the two stages happens through user-defined rules (e.g., rules that reflect infection progression as described below). During the infected stage of round i, we assume the existence of a *given* target amino-acid sequence $\zeta_i = (\zeta_i^{(1)}, \ldots, \zeta_i^{(L)})$ of length L without any stop codon, defined as the best possible antibody-coding sequence that can bind to the present antigen. The target can change across rounds, reflecting the evolution of the antigens, a point we will come back to later.

Cell Properties

Since only memory B cells can be repeatedly activated by the encounter with an antigen, we will simulate memory B cells only. Plasma B cells do not undergo SHMs and represent terminal states of the clonal lineage development and thus can be sampled from the leaves of the simulated tree. We will refer to a B cell that has just encountered an antigen and moved to a *germinal center* (GC) as an *activated B cell* (or "activated cell" for short) (Fig. 7.2a).

In the AM model, each entity i represents a B cell with the property vector $\mathbf{x}_i = (g_i, s_i, t_i, {}^{g_i}/_{a_i}, g_i a_i)$ with five values, among which the last three are derived from the first two. We keep derived properties as part of $\mathbf{x}_i$ because they allow us to define $\Lambda_E(\mathbf{x}_i, \mathbf{S})$ as polynomials of saved properties (Table 7.2); this, in turns, enables the use of our fast sampling algorithm. To fit with the BDT model, we assume properties of each cell are fixed in between B/D/T events, ignoring possible temporal changes (Weisel et al., 2016).

- The binary property g_i indicates whether a cell i is an activated B cell (1) or is a memory B cell outside lymph nodes, which we call a "memory cell" for simplicity (0).

- The s_i property stores the DNA sequence of B cell i coding for the variable region of the heavy chain with a fixed length $3L$. We focus on simulating the heavy chain sequences only because most existing Rep-Seq studies focus on sequencing heavy chains only (e.g., Stern et al., 2014; Ellebedy et al., 2016; Magri et al., 2017; Horns et al., 2019). For the sake of simplicity, we assume the fate of the cell depends only on the variable region of the heavy chain. Each cell has a fixed sequence, and mutations occur at the time of a cell birth, which happens only for activated cells in the infected stage. After a birth event for cell i, sequences s_j and s_k of child cells j and k are chosen independently and identically at random (ignoring the G1 origin of mutations Sharbeen et al. (2012).) While any sequence evolution model could be incorporated in the DIMSIM framework, we will later describe a 5-mer-based model used in our analyses.

- Property t_i denotes the rate of transformation, which means the activation of a memory cell

$g_i : 1 \to 0$ in response to an antigen, or the maturation of an activated cell into a memory cell $g_i : 0 \to 1$. Transformations, which only happen during the infected stage, flip g but keep the sequence s intact.

- Property a_i denotes the strength of affinity binding of the Ig receptor of the cell i to the antigen. We let σ denote the total affinity of activated cells and note $\sigma = \sum_{i \in S} g_i a_i$ is the last element of the vector $\mathbf{S}$. Then, a_i/σ is the fraction of total affinity assigned to a cell.

Both t_i and a_i are derived and are set based on the sequence of i and the target.

Sequence affinity and birth, death, and transformation rates

Affinity a_i is only defined and used during the infected stage where the target is available and is function of the cell sequence s_i and the target sequence ζ. The closer the sequence to the target, the higher its affinity should be, a fact that other simulators have also incorporated Davidsen and Matsen (2018). The exact relationships between the sequences and affinity are not known. For the purpose of benchmarking methods, we propose a simple formula. Let $f_\zeta(s_i)$ be a measure of closeness of the sequence to the target in the affinity space, we set

$$a_i \doteq e^{A f_\zeta(s_i)} , \tag{7.1}$$

where A is a constant factor used to calibrate the selective pressure (see below). Note that in this scheme, as sequences get closer to the target, the affinity grows gradually with a speed controlled by A (Fig. 7.2b). We will describe our particular choice of function $f_\zeta(s_i)$ using BLOSUM similarity below.

The event rates are functions of cell properties and the stage (Table 7.2). During the dormant stage, there are no births or transformations; cells only die with a very high uniform rate λ_d for activated cells and a low uniform rate λ'_d for memory cells.

During the infected stage, we adjust the death rates of cells based on their affinities but keep the birth rates constant; this interplay is used to simulate the selective pressure. Note that

we do not claim that a fixed birth rate and changing death rate is biologically realistic (e.g., see Gitlin et al., 2014). However, in terms of dynamics of our model, what matters is the ratio of the birth and death rates, which enable us to make this simplifying choice. In our model, an activated cell can undergo cell division at a uniform rate λ_b, differentiate into a memory cell at a uniform rate $t_i = \rho_m \lambda_b$ or a plasma-like cell at a uniform rate $\rho_p \lambda_b$, and undergo apoptosis (i.e., die). We do not model plasma-like cells; instead, both differentiation into plasma-like cells and apoptosis are treated as death events (Figure 7.2a). The rate of apoptosis of an activated cell i is modelled as inversely proportional to the amount of resources (antigens and FDCs) to which the cell i has access when competing against other activated cells. Thus, the proportion of resources available to the cell i is modeled by the affinity proportion a_i/σ (i.e., the affinity of the cell to the antigen normalized by the current sum of the affinity of all activated cells). This affinity proportion is impacted by the choice of parameter A. The lower the A, the more uniform these proportions become, as expected with low selective pressure; conversely, as A increases, a_i/σ values further diverge between low affinity and high-affinity cells (Fig. 7.2b). Thus, A can be used to control the strength of the selective pressure.

The memory cells undergo apoptosis at a uniform rate λ'_d. They can also be activated by helper T cells to enter the germinal center with the transformation rate

$$t_i \doteq \lambda_t e^{\rho_a A(f_\zeta(s_i)-\Delta_0)} = \lambda_t e^{-\rho_a A \Delta_0} a_i^{\rho_a} \ . \tag{7.2}$$

Note that the activation rate of memory cells increases monotonically with their affinity to the target, according to $a_i^{\rho_a}$ where ρ_a, set by default to $1/2$, is the sensitivity of B cell activation to affinity. This dependency on affinity models the increased propensity of the memory cells to activate when presented by helper T cells with familiar antigen. The default choice $\rho_a = 1/2$ is motivated by the fact that although memory cells with higher binding strengths to the antigen are more likely to be activated, the interaction between a helper T cell and a memory B cell is a one-time event and thus less sensitive to binding strength.

Illustration. As an example, consider a system with two cell types: L and H, each type with its own unique sequence (Fig. 7.2b-d). Assume all cells are activated cells, the number of L and H are the same at one point in time, and H cells have a higher affinity than L cells by a factor of ρ. For ease of exposition, here, we include the mutation rate as part of the death rate because mutation events also decrease cell count. Let's assume the total number of cells equals the carrying capacity C. If L and H have the same affinity (i.e., $\rho = 1$), then the birth and death rates are identical for all cells. As the affinity of H cells is increased (i.e., $\rho > 1$), the death rate of L cells increases linearly whereas the death rate of H cells decreases (Fig. 7.2b). Thus, H cells will have higher birth rates than death, will be selected for, and will expand. If we fix $\rho = 2$ and increase the population size, the death rates of both L and H cells increase, but at different rates (Fig. 7.2d). When the population size is small compared to C, both types of cells have more birth than death. After a threshold ($C/3$ in this example), the death rate of L type surpasses its birth rate (thus, its population starts to shrink) while the population of H cells continues to grow. However, as the population size increases ($2C/3$ here), both sets of cells start to shrink (i.e., higher death rates than birth), because the population size is by definition bounded by C.

Specific (Default) Modeling Choices

Several steps of our simulations are flexible and can be changed by the user. We next describe a particular set of choices we have implemented and used in our experiments below.

Switching between stages. The system enters dormant stage when antigens are neutralized by the antibodies. A simple way to define neutralization is to switch the stage when the total affinity of antibodies produced by plasma-like cells reaches a certain threshold; here, we switch when the sum of affinities of activated cells (σ) reaches a predefined constant M.

Sequence evolution. In our experiments, we use an empirical 5-mer-based model inspired by Yaari et al. (2013). Let $s_i^{(p)}$ be the nucleotide on the p-th position of the nucleotide sequence of the cell i. Each $s_j^{(p)}$ or $s_k^{(p)}$ is independently set to $s \in \{A,C,G,T\}$ with probability: $Pr(s_j^{(p)} = s) = Pr(s_k^{(p)} = s) = f(s, s_i^{(p-2)}, s_i^{(p-1)}, s_i^{(p)}, s_i^{(p+1)}, s_i^{(p+2)})$ where $f : \{A,C,G,T\}^6 \rightarrow$

$[0,1]$ denotes an empirically determined 5-mer frequency model based on the model of Yaari et al. (2013) and recomputed based on newer datasets including non-synonymous mutations (see Appendix 7.B).

Sequence affinity function. While various methods can be imagined for measuring the closeness of the sequence to the target, we used a simple approach: measuring sequence similarity according to the BLOSUM matrix and appropriate scaling of numbers. We assume each amino-acid position contributes to the binding strength to the target and the stability of the structure of the Ig-receptor independently. We model affinity proportionally to the product of the effect of each amino-acid position. This simple model ignores the 3D structure of proteins for the most part but should be sufficient for creating benchmarking datasets as none of the reconstruction methods consider 3D structure either. However, because complementarity-determining regions (CDRs), which include the binding sites, tend to accumulate more SHMs compared to framework regions (FRs) (Tanaka and Nei, 1989; Hsiao et al., 2019), we do differentiate those. When s_i includes a stop codon, we simply set $a_i = 0$. Otherwise, we define the BLOSUM score of an amino acid sequence $\xi = (\xi^{(1)}, \ldots, \xi^{(L)})$ with respect to target ζ as

$$\Delta_\zeta(\xi) = \sum_{p \in \mathbf{CDR}} \left(\delta(\xi^{(p)}, \zeta^{(p)}) - \delta(\zeta^{(p)}, \zeta^{(p)})\right) + w_f \sum_{p \in \{1 \ldots L\} \setminus \mathbf{CDR}} \left(\delta(\xi^{(p)}, \zeta^{(p)}) - \delta(\zeta^{(p)}, \zeta^{(p)})\right) \tag{7.3}$$

where $\delta(.,.)$ gives the BLOSUM score between two amino acids (Table S7.2), and w_f is a constant used to calibrate the importance of CDRs versus FRs in the affinity and transformation processes. We then simply set $f_\zeta(s_i) = \Delta_\zeta(\xi(s_i))$ where $\xi(.)$ translates from DNA to AA.

Choosing targets. One target sequence per round needs to be selected. The extent of the change in targets across rounds impacts the patterns of the immune response and hence the shape of the clonal trees that result. In our experiments, to define targets across rounds, we seek a set of sequences with an evolutionary trajectory that reflects the evolutionary history of a set of real antigen (e.g., influenza virus). Let the known amino-acid sequences of an antigen sampled through time (flu sequence over seasons) be denoted by $\eta_1, \ldots, \eta_r$, and let each sequence have

the fixed length L_η. To choose the targets, we first select an arbitrary naive B cell, here chosen from datasets of Ellebedy et al. (2016), and set $\hat{\Psi}$ to the nucleotide sequence of the variable region of its heavy chain. Then, we simply set ζ_1 to the amino-acid translation of $\hat{\Psi}$. In other words, in the first round, we use the naive cell as the target, and therefore, the first couple of rounds of the simulation should be treated as dummy rounds and should be discarded. Let κ be a positive constant that controls the rate of change in the target relative to the rate of change in the antigen sequences. To define the remaining targets, we seek to find the set of $r-1$ sequences that minimize:

$$\sum_{i,j\in[r]} \left| \kappa \sum_{p\in\mathbf{CDR}} \delta(\zeta_i^{(p)}, \zeta_i^{(p)}) - \delta(\zeta_i^{(p)}, \zeta_j^{(p)}) - \sum_{q=1}^{L_\eta} \left(\delta(\eta_i^{(q)}, \eta_i^{(q)}) - \delta(\eta_i^{(q)}, \eta_j^{(q)}) \right) \right| . \qquad (7.4)$$

Thus, a set of target sequences across r rounds are preferred if their pairwise distance matrix maximally matches the pairwise distance matrix of all antigen sequences over the same rounds (with a scaling). To account for conserved regions, we arbitrarily chose to keep all the non-CDR regions invariable in all target sequences (this choice can be easily changed). Thus, we seek to make the distance between two target sequences from two rounds similar to the distances of antigen sequences, scaled by a factor of κ. We approach this NP-hard problem using a greedy search heuristic (Algorithm S7.3). The heuristic starts with arbitrary $\zeta_2, \ldots, \zeta_r$ and replaces one symbol of one sequence at a time to reduce the objective function; it repeats until reaching a local minimum where no such replacement is possible.

7.2.2 Benchmarking Setup

Flu simulations

We performed several simulations of a series of $r = 56$ seasons of flu, using sequences of hemagglutinin (HA) protein. HA found on the surface of the influenza viruses is the primary target of neutralizing antibodies. High mutation rates of influenza genome changes the sequence of HA and allows the virus to escape from the immune pressure, thus making flu a recurring

seasonal infection. The NCBI Influenza Virus Resource (Bao et al., 2008) contains 961 HA sequences from influenza B virus collected around the world. Each HA sequence is labeled with a year and a location. For simulation purposes, we extracted 59 HA sequences corresponding to flu infections in Hong Kong and selected 56 out of 59 HA sequences that have the same length (584 aa). The selected HA sequences were detected in Hong Kong from 1999 to 2010. Notice that HA sequences could be replaced with other widely available antigen sequences (e.g., Coronavirus).

We used the default settings for the various parameters of Table 7.1, and used the approach described earlier to choose the target amino-acid sequences. Each round corresponds to one season, starts at the infected stage with a given target sequence ζ_l, which ends when $\sigma = M$. At that point, we assume the infection is overcome, and the system switches to dormant, where we stay until the next round starts (times of flu outbreaks are known in our dataset). When the $r = 56$ rounds of infections end, we sample $\varsigma = 200$ antibody-coding nucleotide sequences $\Psi_1, \ldots, \Psi_\varsigma$ from cells in the system (i.e., from the round r) and built their clonal tree. While it may be unrealistic to assume a person gets exposed to flu every season, it is possible, and this procedure allows us to test the impacts of a large number of infections.

To benchmark reconstruction tools, we set up four experiments, varying one or two parameters in each experiment (Table 7.3) and setting the remaining ones to default values (Table 7.1). The central experiment contains 19 conditions, changing the selective pressure (A) and the rate of hypermutation (μ). We vary A from $1/8\times$ of default value (0.1) to $2\times$ and vary μ s from 1.25×10^{-4} to 2×10^{-3} per base-pair per generation. In six combinations, the selective pressure is not high enough to overcome random mutations; in these cases, the affinity values do not increase and as a result, the carrying capacity is never reached. Thus, we exclude these conditions. We also study three other parameters. We vary the weight multiplier of FRs (w_f) from $1/5$ to 2. We vary the carrying capacity (C), which is the germinal center size or the amount of antigens FDCs hold in the context of B cell maturation, from 12500 to 400000. The value of this parameter can impact the speed of novel mutations arising and may change the properties of

simulated trees. We also vary the mean life-time of memory cells from 0.5 year to 16 years, to study the impact of the extent of memory cell activation during recurrent infections.

Compared methods of Clonal Lineage Reconstruction

We compare seven tools: minimum spanning tree, BRILIA (Lee et al., 2017), IgPhyML (Hoehn et al., 2017), RAxML (Stamatakis, 2014), Immunitree (Sok et al., 2013a), and a post-processed version of ML phylogenetic trees (IgPhyML* and RAxML*) with low support branches contracted. We note this is not an exhaustive list, as many other tools also exist (e.g., SAMM, Davidsen and Matsen, 2018, GCtree, DeWitt et al., 2018, IgTree, Barak et al., 2008) that we did not include.

MST(-like) methods. We implemented a simple minimum spanning tree method containing $\Psi_1, \ldots, \Psi_\varsigma$ as well as $\hat{\Psi}$, which is forced to be the root. We compute the nucleotide Hamming distance between all pairs of sequences and construct the minimum spanning tree (MST) using those distances. Besides the simple MST, we also test Immunitree Sok et al. (2013a), a tool that clusters antibody-coding sequences into lineages and builds clonal trees at the same time by optimizing a minimum spanning tree and Steiner tree-like problem. We took as input $\Psi_1, \ldots, \Psi_\varsigma$ and used Immunitree to build a set of clonal trees. We then added vertex $\hat{\Psi}$ as the root and let the roots of the clonal trees to be immediate children of $\hat{\Psi}$.

Brilia clusters antibody-coding sequences into lineages and builds clonal trees at the same time. We took as input $\Psi_1, \ldots, \Psi_\varsigma$ and used Brilia v3.5.4 to build a set of clonal trees. We then added vertex $\hat{\Psi}$ as the root and added roots of the clonal trees as children of $\hat{\Psi}$.

Phylogenetic methods. We tested ML phylogenetic reconstruction tool RAxML v8.2.10 under GTR model and IgPhyML v0.99, a ML method tuned specifically for immune cells. For RAxML, we took as input $\Psi_1, \ldots, \Psi_\varsigma$ and $\hat{\Psi}$ to obtain an unrooted phylogenetic tree and rerooted at $\hat{\Psi}$. For IgPhyML, we took as input $\Psi_1, \ldots, \Psi_\varsigma$ and provided $\hat{\Psi}$ as root to obtain a rooted phylogenetic tree. Both methods produce fully binary trees.

Zero-aware phylogenetic methods. As previously suggested DeWitt et al. (2018); Davidsen and Matsen (2018), contracting short or low support branches is one way of addressing limitations of ML methods. Since the length of each antibody-coding nucleotide sequence < 400, we can assume that both ends of any branch with length less than 10^{-4} would correspond to the same sequence (if it was sampled). Therefore, we contracted branches of length less than 10^{-4} and call the resulting methods RAxML* and IgPhyML*.

Evaluation metrics

The simulated and reconstructed histories of samples $\Psi_1, \ldots, \Psi_\varsigma$ are represented as trees, where samples are uniquely labeled on some nodes and the remaining nodes are left unlabeled. Labeled nodes represent sequences in the samples and unlabeled nodes denote the ancestral sequences not present in the samples. We evaluate results in two ways, described in detail in Appendix 7.B.4. We use a set of metrics for characterizing properties of simulated trees in terms of their topology, branch length, and distribution of labeled nodes (Table 7.4). We also compare the simulated trees to those inferred using each method (Table 7.5).

While metrics for comparing phylogenies exist, these metrics need to be amended for clonal trees that can have sampled ancestral nodes Davidsen and Matsen (2018); DiNardo et al. (2020). Many of the existing metrics can be generalized to compare a simulated tree R and a reconstructed tree E (Table 7.5), both induced down to include all labeled nodes (i.e., removing unlabeled nodes if less than two of their children have any labeled descendants). Unlike traditional phylogenies, here, internal nodes can be labeled, and we define metrics based on rooted trees instead of unrooted trees. We refer to the set of labeled nodes under a node as a cluster – a concept that many of the metrics use. Note that singleton clusters are trivial when all labeled nodes are leaves; however, when there are labeled internal nodes, including or excluding singletons can make a difference. Thus, we also define many of the distances both with and without singleton clusters. Some distances (i.e., FNR and FDR metrics) are already normalized. To normalize other distances, for each experimental condition, we create

a control tree by randomly permuting labels of the true tree. We then normalize distances of a reconstruction method by dividing it by the average score of replicates of the control method.

7.3 Results

7.3.1 Demonstration of the simulation process

Visualizing one replicate of simulation under default condition, we see patterns of average affinity and the number of activated and memory cells that rise and fall as time progress during the infected stage (Fig. 7.3a). During each round of infection, the affinity first decreases and then increases as long as the duration of the infection is long enough. Thus, when the number of activated cells is low and the selective pressure is low, a mutation is likely to lead to reduced affinity, whereas, when the number of activated cells increases, the selective pressure begins to increase and select for higher affinity; these patterns are in concordance with the literature Nakagawa and Calado (2021). The duration of infections, the mean affinity at the end, and the total number of cells also varies widely across different seasons. When the affinity at the start of a season is low, the duration of infection is longer and more activated cells and memory cells are generated (Figs. 7.3a and S7.1a). This pattern is also consistent with the biological expectation: when the immune system already has high affinity to the antigen, it can eradicate the antigen quickly and without much need for further evolution. To further quantify the pattern, we define the novelty of each target ζ_i as the negation of the maximum BLOSUM score between that target and any previous target: $-\max_{j<i}\{\Delta_{\zeta_i}(\zeta_j)\}$. We observe that as novelty of the target increases, the average affinity of activated cells at the end of the infection tends to decrease ($R^2 = 0.242$, $p = 2.5 \times 10^{-4}$), whereas, the number of activated cells at the end of the infection ($R^2 = 0.248$, $p = 2.0 \times 10^{-4}$) and the duration of infection ($R^2 = 0.288$, $p = 4.8 \times 10^{-5}$) both tend to increase (Fig. 7.3b).

Memory cell counts fluctuate. Each season leads to a buildup in memory cells from the start to the end of the infection, and the amount of buildup depends on the duration and correlates

with novelty ($R^2 = 0.264$, $p = 1.2 \times 10^{-4}$). However, the total number of memory cells reduces between seasons due to cell deaths (Fig. S7.1c) and changes across seasons. In particular, a string of short-lived infections and large time spans between the flu seasons between 2002 and 2008 gradually lead to a depletion of the memory cells, which are then built up again in the subsequent seasons (Fig. S7.1c).

7.3.2 Benchmarking reconstruction methods

Default Parameters

Under default parameters, over all evaluation metrics, zero-aware phylogenetic methods (IgPhyML* and RAxML*) clearly have the best accuracy in reconstructing the lineage history (Fig. 7.4). Normal phylogenetic methods (IgPhyML and RAxML), which produce fully binary trees with no samples at leaves, have the lowest FNR error, retrieving more than 90% of the correct clusters. However, their precision is predictably low: close to 35% of their clusters are incorrect. Interestingly, zero-aware phylogenetic methods have only a slight increase in FN rate ($< 2\%$ on average) but enjoy a dramatic improvement in precision. By simply contracting super-short branches, the FDR error reduces to less than 15%, which is better than all other methods. Similarly, normal phylogenetic methods perform poorly according to RF, PD, and MD metrics, which emphasize false positives, but perform well (but not as well as the zero-aware versions) according to triplet-based metrics (TED and TD), which penalize false negatives more than false positives. Among the two phylogenetic reconstruction methods, RAxML is slightly more accurate than IgPhyML.

The MST-like methods have low FDR, coming close to zero-aware phylogeny-aware methods, but also have much higher FNR (25% or more). Immunitree (which uses Steiner trees) is substantially better than a simple MST in terms of FNR, but not in terms of FDR or triplet-based measures. These patterns largely follow the expectations: more resolved trees have lower FNRs whereas less resolved trees have lower FDRs. However, zero-aware phylogeny methods are able to obtain the best FDR and FNR and dominate other methods. BRILIA consistently

has high error in our analyses. These patterns remain largely similar (but are magnified) when singletons are removed from the consideration (Fig. S7.2). The main exception is that when singletons are excluded, Immunitree is no longer the second-best method according to the RF distance.

We next compare properties of the inferred trees and true trees (Figure 7.4c). BRILIA and MST put far too many labels at internal nodes (≈35% instead of ≈8%), while Immunitree and zero-aware phylogenetic trees are very close to the true tree in terms of percent internal samples. BRILIA and Immunitree over-estimate the tree balance, while phylogenetic trees under-estimate balance, especially before contracting low support branches. Conversely, phylogenetic methods over-estimate depth of samples while BRILIA, MST, and Immunitree underestimate the depth; zero-aware phylogenetic methods, however, produce trees that are very close to the true tree in sample depth. Phylogenetic methods, by definition, overestimate bifurcation index as 1; this overestimation is dramatically reduced but not fully eliminated by zero-aware phylogenetic methods and Immunitree. MST is quite close to the correct levels of bifurcation.

Varying selective pressure

The reconstructions methods are all impacted as selective pressure (A) changes, but some methods are more sensitive than others, and they are affected differently (Figs. 7.5ab). Zero-aware phylogenetic methods have the best accuracy across values of A. The ranking among other methods depends on the selective pressure, such that phylogenetic methods become the worst when A is high and become the best when A is low. As A increases, error tends to increase for phylogenetic methods under all evaluation metrics except for the FNR; for example, the FDR of RAxML increases from 27% at the $1/4$x selective pressure to 42% at the 2x level. In contrast, the error of Immunitree, MST, and BRILIA reduces with increased A according to FNR and RF. Zero-aware phylogenetic methods are relatively robust to the A and their error rates change only slightly across conditions. When singletons are removed from the metrics of comparison, patterns remain similar, though the impact of selective pressure becomes less

pronounced (Fig. S7.3a).

The reason behind these patterns becomes more apparent once we consider changes in tree properties (Fig. 7.5c). As A increases, the fraction of internal samples tends to increase. This pattern can be explained: when selective pressure is high, cells with low affinity die off quickly, which results in shorter branch lengths. Since phylogenetic methods put all sequences at leaves, they have reduced accuracy. In contrast, IgPhyML*, RAxML*, and Immunitree are able to successfully assign sequences to internal branches; as a result, their percentage of internal samples match those of the true trees (Figs. 7.5c). Similarly, with increased A, the bifurcation index of the simulated tree tends to decrease, a pattern that is observed also in reconstructed trees from IgPhyML*, RAxML*, Immunitree, MST, and BRILIA. Again, phylogenetic trees, which produce binary trees, are unable to capture these patterns. As A increases, depth of sampled nodes of the simulated tree tends to decrease, a pattern matched by IgPhyML* and RAxML* but not other methods. Finally, when A is high, trees are shorter (i.e., accumulate fewer mutations) and more branches are single mutation (Fig. S7.4), both of which make phylogenetic inference more difficult. The reduced levels of depth, total change, and bifurcation make sense: higher pressure should result in fewer mutations needed to reach M because cells with unfavorable mutations are less likely to survive; this would produce shorter trees.

Varying rate of hypermutation

As the hypermutation rate (μ) increases, error decreases for normal phylogenetic methods (IgPhyML and RAxML) according to most metrics but stays relatively stable for zero-aware methods (Fig. 7.5de). Increasing μ results in simulated trees that are marginally less balanced, are more bifurcating, have fewer internal node samples, and have a higher depth for sampled nodes (Fig. 7.5f). Thus, increasing μ generates trees more similar to what traditional phylogenetic methods assume. Zero-aware phylogenetic methods and Immunitree designate the right percentage of nodes as internal, but both are slightly more bifurcating than true trees (Fig. 7.5f). Overall, zero-aware phylogenetic methods are the most accurate across all values of μ.

Interplay between selective pressure and mutation rate

When we vary both A and μ, we observe that increasing mutation rate has similar effects on the error and tree properties as decreasing the selective pressure (Fig. 7.6). Reassuringly, error patterns observed when fixing one variable and changing the other are consistent with patterns when both variables are changed (Figs. 7.6 and S7.5). The most difficult condition for phylogenetic methods is low mutation rate and high selective pressure, where close to 70% of the branches include only a single mutation and bifurcation index is only 43%. However, zero-aware methods are impacted less in these conditions, and are in fact improved according to the RF metric (Fig. S7.5). In addition, we observe that antibody clonal trees become more phylogenetic-like – that is, more bifurcating (max: 0.74) and fewer internal samples (min: 20%) – with $\mu = 10^{-3}$ and $A = {}^{1}/_{4}$x. Increasing the mutation rate or decreasing the selective pressure beyond these values leads to combinations where the infection could not be overcome.

Other parameters

Beyond the main two parameters, we also studied changing six secondary parameters, most of which had relatively little impact on the results (Fig 7.7). As the weight of FRs regions in computing affinity (w_f) increases, error tends to *slightly* increase for all methods under many evaluation metrics (Fig. S7.6). This pattern can be related to the slight increase in the number of single branch mutations and the reduction in the total number of substitutions across the tree. As germinal center capacity (C) increases, error increases or decreases slightly, depending on what measure is examined (Fig. S7.7). Increasing C tends to reduce the number of internal samples and single mutation branches in the simulated tree, and tends to increase mutations per branch. As memory cell life-time (${}^{1}/_{\lambda'_d}$) increases, error tends to increase for phylogenetic methods (Fig. S7.8), including IgPhyML* and RAxML*, which nevertheless continue to be the best methods. Plasma cells conversion rate (ρ_p) (Fig. S7.9), rate of change in antibody target compared to antigen change (κ) (Fig. S7.10), and the threshold of total affinity for neutralization and stage change (M) (Fig. S7.11) have small and inconsistent impacts on tree inference error.

In all conditions examined, IgPhyML* and RAxML* have the best accuracy (Fig 7.7).

7.4 Discussion

7.4.1 Implications for reconstructing antibody evolution

Our study partially confirms that phylogenetic methods need to change for inferring antibody clonal trees with high accuracy. Depending on the simulation condition, 1% to 20% of sampled sequences belonged to internal nodes, and the true trees are only 60% to 70% bifurcating. We observed that results of phylogenetic inference using ML, taken at face value, can have low accuracy. However, ML phylogenetic methods with the simple adjustment of contracting short branches can outperform the alternative methods based on Steiner trees and spanning trees. In contrast to earlier work Davidson et al. (2018); DeWitt et al. (2018) that used ancestral reconstruction, we used a fixed constant for contraction using a rule-of-thumb based on the length of the sequences. Alternatively, statistical tests of whether a zero branch length null hypothesis can be rejected exist (Jackman et al., 1999; Walsh et al., 1999; Goldman et al., 2000) and are fast (Anisimova et al., 2006) and could be used *in lieu* of our simple heuristic. Moreover, our work implies that phylogenetic methods that try to naturally model zero branch length (e.g., Lewis et al., 2005) are also promising. In particular, the adaptive LASSO method of Zhang et al. (2020) seems suitable for inferring antibody evolution.

Despite the higher accuracy of zero-aware phylogenetic methods compared to the available alternatives, we note that there is still substantial error. Under the default condition, 90% of clusters of the true tree were recovered, but about 15% of the recovered clusters were incorrect. In particular, the discrepancy between FNR and FDR is due to the fact that the inferred trees are somewhat more bifurcating than true trees (e.g., $\approx$70% versus 60% in the default condition). Thus, while contracting some super-short branches has been helpful in increasing accuracy, our zero-aware phylogenetic trees are still biased towards too much resolution. It is possible that better Steiner-based methods that incorporate more advanced models of sequence evolution can

solve this shortcoming.

7.4.2 Implications for evaluation criteria

The ranking of reconstruction methods can change based on which of the ten evaluation criteria we choose, and these rankings only partially correlate (Fig. S7.12). FDR and FNR are weakly *anti*-correlated only when including singletons (mean Spearman's rank correlation coefficient across all tests $\rho = -0.12$). RF distance, which combines both aspects, correlates moderately with both FDR ($\rho = 0.5$) and FNR ($\rho = 0.57$). The triplet-based metrics strongly agree with each other ($\rho = 0.97$) and are mostly compatible with the RF distance ($\rho \approx 0.75$), but are less similar to MD and PD metrics ($\rho \leq 0.52$). Consistent with the observation that triplet metrics penalize false negatives more than false positives, they agree more strongly with FNR than FDR ($\rho = 0.65$ vs 0.26). MD and PD are very similar to each other ($\rho = 0.96$), have no correlation to FNR ($\rho \leq 0.05$), but have moderately high correlation to FDR ($\rho = 0.71$). Finally, we notice that singletons can matter: while FNR and FNR* are highly correlated ($\rho = 0.94$), RF correlates with RF* less strongly ($\rho = 0.71$), and FDR correlates with FDR* only moderately ($\rho = 0.61$).

The choice of the metric should depend on downstream application of the clonal tree. While zero-aware phylogenetic methods are dramatically better than normal phylogenetic methods based on most criteria, they are only slightly better according to the triplet-based criteria. The triplet metrics do not penalize trees heavily if they are more resolved than the true tree or if they move internal nodes to leaves. Thus, when downstream usage is robust to extra resolution and extra terminal edges, triplet metrics offer a good way to measure topological accuracy. On the other extreme, PD and MD are very sensitive to the tree resolution and internal placement, so much so that they often evaluate inferred phylogenetic trees to be much worse than random trees (Fig. S7.5) because these trees generate fully resolved trees and put samples at leaves. Thus, we don't find PD and MD to be reliable metrics of *topological* accuracy. RF distance is in between: it penalizes extra resolution more than triplet metrics but less than path-based metrics.

It does distinguish zero-aware and phylogenetic methods, but rarely evaluates any methods to be worse than random (Fig. S7.5). Overall, dividing the observed error along two (potentially contradictory) axes such as FNR and FDR is recommended because this evaluation provides more insight into reasons behind error.

7.4.3 Comparison to other simulation models

Several simulation tools capable of benchmarking reconstruction methods have been developed. Some of these tools are not comparable to our effort because of various limitations. ImmuneSIM Weber et al. (2020) generates mutations but does not model the clonal tree or the selection process. Methods of Amitai et al. (2017) and Reshetova et al. (2017) are based on the two-step simulation paradigm and only generate clonal trees under selection, leaving sequence generation to other methods. The most relevant method to ours are bcr-phylo Davidsen and Matsen (2018) and gcdynamics Childs et al. (2015), which simulate clonal trees of antibody-coding sequences under AM. Both bcr-phylo and gcdynamics have similarities and differences to our method (Table 7.6). For example, they both support multiple targets but only one round of simulations. Although our model is capable of multiple targets, for simplicity, DIMSIM uses one target per round of infection. However, unlike the two other methods that only simulate activated cells, DIMSIM also simulates memory cells; as a result, it can simulate multiple rounds of infection by an evolving pathogen with changing targets while considering memory built from previous infections. Moreover, DIMSIM simulates in continuous time, whereas the other tools simulate under discrete generations. All three methods use sequences to define affinity, albeit differently: DIMSIM using BLOSUM distance, brc-phylo using hamming distance, and gcdynamics using random energy landscape. A main feature of DIMSIM is that its rates are polynomial fractions of individual and total affinity; this choice enables it to speed up the simulation, allowing it to scale up to large numbers of cells, which makes DIMSIM capable of simulating many lineages at a time.

7.4.4 Limitations of the study

Our study has limitations that should be kept in mind.

In our simulations, we did not add errors to sequence data used as input to clonal tree reconstruction methods. Real Rep-Seq samples undergo extensive PCR and thus might contain both sequencing and amplification errors. We assumed that error elimination is already performed (to perfection) prior to reconstruction using existing methods (e.g., Vander Heiden et al., 2014; Safonova et al., 2015; Bolotin et al., 2015; Shlemov et al., 2017). The efficacy of methods that simultaneously filter errors and build clonal trees (e.g., Safonova and Pevzner, 2019; Lee et al., 2017) should be the subject of future research. We also simulated only substitution SHMs but no insertions and deletions, leaving the latter to future work.

In our AM model, we made several simplifying assumptions. For example, absent of a good model of receptor binding, we assumed the affinity grows gradually as the AA sequence becomes more similar to the target sequence. The idea that AM occurs by mutational diffusion along one or more preferred paths in the genotype space has been supported by Kepler et al. (2014). Nevertheless, our i.i.d model is certainly a simplification without a clear empirical support. Moreover, we assumed the existence of a target antibody sequence. The literature has increasingly documented highly convergent immune responses to the same epitope across individuals and conditions (Henry Dunand and Wilson, 2015; Robbiani et al., 2020). This observation gives us reason to think the existence of target sequences is not a bad assumption; nevertheless, the choice of a *single* target may not be realistic. To model the change in the target as the viruses evolve across seasons, we chose targets with evolutionary divergence levels that mimic the divergence levels of the antigen, albeit with some scaling factor. While we believe this choice is sensible, again, we have no evidence to back up this model on empirical grounds. It is conceivable that two antigens with high evolutionary distance are neutralized by similar antibodies, or that, antigens that are very similar require very distant antibodies. We modelled SHMs as affecting daughter cells independently, but it is arguably more realistic to make both

daughter cells carry the same mutation due to the G1 origin of SHMs Sharbeen et al. (2012) (a simple change to the model). Finally, our 5-mer mutation model, while based on the empirical model of Yaari et al. (2013), still fails to capture some complexities of the real antibody evolution. For example, we concentrated substitutions on the CDR region, but other regions are known to also accumulate mutations (Safonova and Pevzner, 2019; Kirik et al., 2017; Ovchinnikov et al., 2018). Other B cell specific models (e.g., Elhanati et al., 2015) including those that seek to tease out the effects of selection from background mutations (e.g., McCoy et al., 2015) and per-position mutability models (Kepler et al., 2014) can be incorporated in the future.

For all these shortcomings, we offer several responses. Due to challenges in modelling antibody repertoire (e.g., Luo and Perelson, 2015), the framework is designed to be flexible and can easily incorporate more complex models. Thus, our work should be considered a first step that will enable better modeling in the future. Also, our objective in simulations was to benchmark reconstruction tools; as long as our modeling choices did not distort the comparison of methods, some model misspecification can be tolerated. We observed that the choice of the best method was not sensitive to many parameter choices.

Beyond model simplifications, we also chose to simulate parts of the complex immune system response, but not others. For example, we simulated one clonal lineage involved in an immune response. As such, we ignored the important *V(D)J recombination* of *IG loci* (Kurosawa and Tonegawa, 1982) and sought to simply simulate a VDJ recombinant that is effective in fighting a specific antigen. Even then, we simulated only one clonal lineage at a time, a limitation that can be easily lifted in the future by starting from multiple root sequences with different VDJ settings and assigning to each a different target sequence. Note that our tool can be easily combined with methods of simulating VDJ recombination, such as IGoR (Marcou et al., 2018). Neither did we simulate light chains, which are often not captured in Rep-Seq sequencing data. Finally, we did not simulate processes such as epitope focusing that produce broadly neutralizing antibodies Bonsignori et al. (2016).

7.4.5 Applications of the framework

Our framework for simulating clonal trees can be extended to other forms of microevolutionary scenarios. While the current implementation is geared towards AM simulations, our proposed algorithm enables forward-time simulation of very large numbers of entities under models that allow dependence between sequences and rates of birth, death, or transformation. The ability to simulate a very large number of entities combined with rates that change with properties of entities give use the necessary ingredients to simulate under complex models of evolution that consider selective pressure. Thus, our framework can be adopted for other forms of microevolutionary simulation, such as the evolution of a virus within a host and accumulation of SHMs in tumor evolution. Such a possibility would become most intriguing if it can also model co-evolution of different types of entities (e.g., antibodies and viruses). While we did not simulate co-evolution here, we believe the framework is capable of performing such simulations by simply creating entity types (just like we had cell types) and making the BDT rates a function of properties across different cell types. Another promising direction for extensions of this work is to integrate the sequence evolutionary models with network-based disease transmissions models (e.g., Ratmann et al., 2017; Moshiri et al., 2019) to enable more accurate simulations of disease spread and evolution.

7.5 Acknowledgements

Chapter 7, in full, has been submitted for publication of the material as it may appear in " Zhang, C., Bzikadze, A., Safonova, Y. & Mirarab, S. Scalable Models of Antibody Evolution and Benchmarking of Clonal Tree Reconstruction Methods. *Frontiers In Immunology*. (2022) ." The dissertation author was the primary investigator and author of this paper.

Figure captions

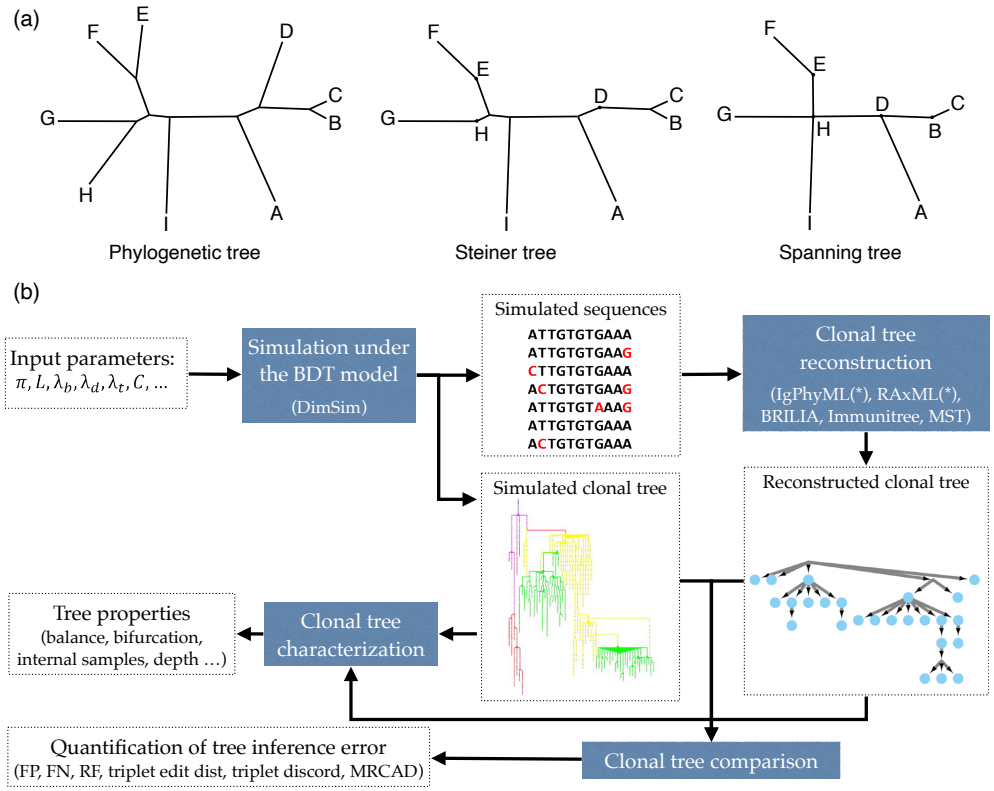

Figure 7.1. (a) Examples of a phylogenetic tree, a Steiner tree, and a spanning tree. Letters indicate sequenced data. Phylogenetic trees put all data points at leaves, and none at internal nodes, spanning trees put data at every node (whether internal or leaf), and Steiner trees are in between (some but not all internal nodes correspond to data). (b) The evaluation framework. The BDT model, parameterized by several values (Table 7.1) is first sampled using the fast algorithm implemented in DIMSIM to create the simulated (i.e., "true") sequence data and clonal trees. These trees are then reconstructed from the simulated sequence data using various methods. The reconstructed clonal tree is compared to the simulated tree using several metrics adopted here to account for internal node sampling and multifurcation. Properties of true and inferred trees are measured using metrics such as balance and resolution.

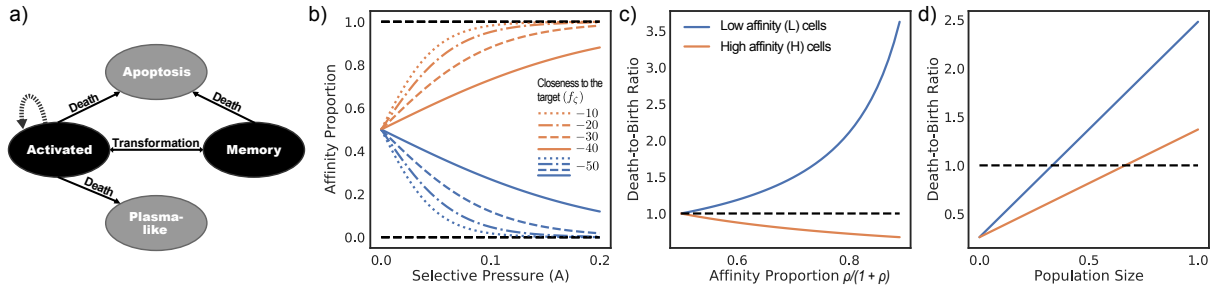

Figure 7.2. a) States of cells and transitions during infected stage. Only states colored black are modeled. Transitions to states colored grey are treated as death events. b-d) Consider a population of activated B cells where all cells have one of two sequences: L (low) or H (high). Let ρ be the ratio of affinity of H-type cells to L-type cells, and let the affinity proportion be the total affinity of a cell type over the total affinity (i.e., $\rho/1+\rho$ for H and $1/1+\rho$ for L). b) The affinity proportion as a function of the selective pressure A when the sequence closeness to the target $f_\zeta(.)$ is kept fixed for L and varies for H. c) the ratio of death rate to birth rate as a function of affinity proportion of H cells, fixing the population size to the carrying capacity. d) ratio of death rate to birth rate as a function of the population size normalized by the carrying capacity, fixing $\rho = 2$. All other parameters set to defaults (Table 7.1). The selective pressure A and the level of binding control the portion of affinity taken up by better sequences (b), which controls the growth of the cell type (c), which is also a function of the total population size (d).

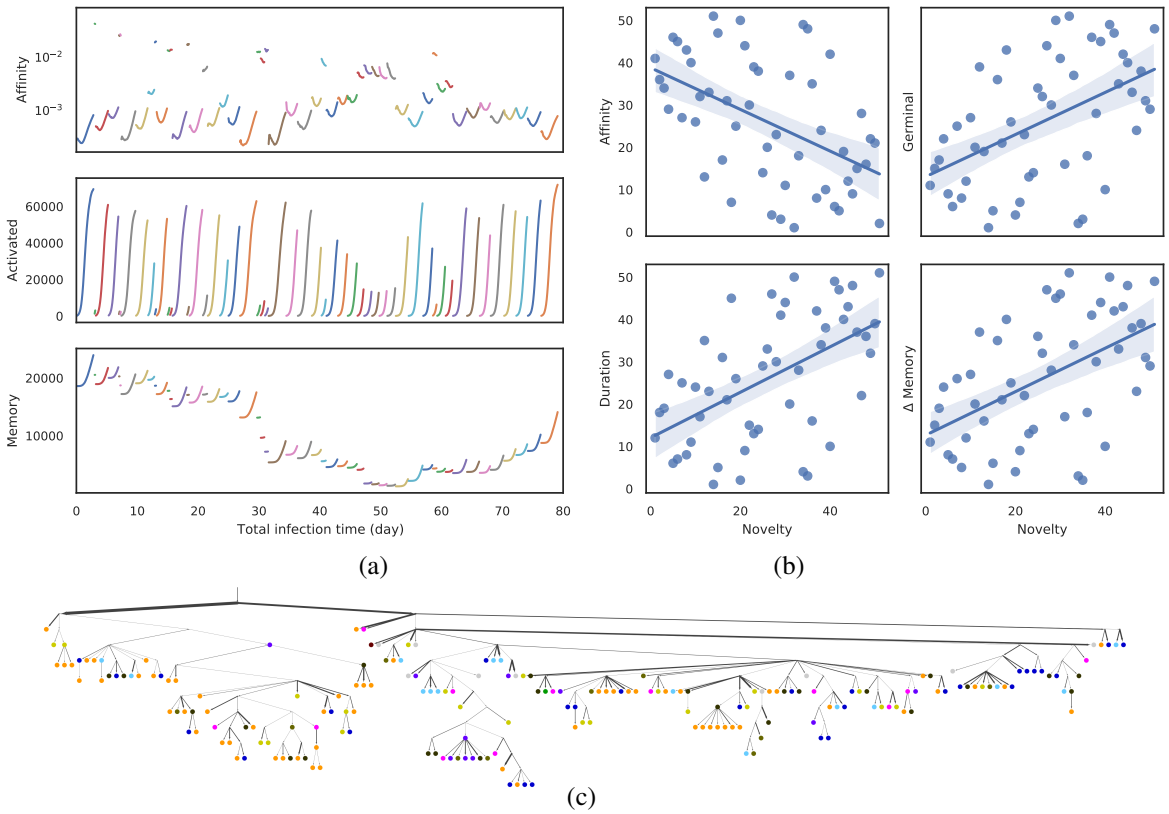

Figure 7.3. a) Average affinity of activated cells to current infection target (log scale), the number of activated cells, and the number of memory cells by total time in infected stage across the last 51 stages of infection (colors) each corresponding to one flu season (discarding the first 5 rounds and dormant stages). b) Impact of the novelty of the antigen on the outcome of the infection across the 56 seasons simulated. The novelty of seasons is measured by $-\max_{j<i}\{\Delta_{\zeta_i}(\zeta_j)\}$ and is ranked from less novel to more novel on the x axis. The y-axis shows ranking (from low to high) of average affinity of activated cells to the current infection target ($R^2 = 0.242$, $p = 2.5 \times 10^{-4}$) at the end of the infection, the number of activated cells ($R^2 = 0.248$, $p = 2.0 \times 10^{-4}$) at the end of the infection, the duration of infection ($R^2 = 0.288$, $p = 4.8 \times 10^{-5}$), and the change in memory cell count ($R^2 = 0.264$, $p = 1.2 \times 10^{-4}$) from the start to the end of the infection. c) Clonal tree of memory cells sampled from one simulation under default condition after all 56 seasons. Nodes are colored by seasons when the memory cells emerge (gray for season 1 through 46; as part (a) for others). Here, 17 internal nodes are sampled and are indicated as circles. Edge weights denote the number of mutations of sequences denoted by adjacent nodes. See Figure S7.1 for more.

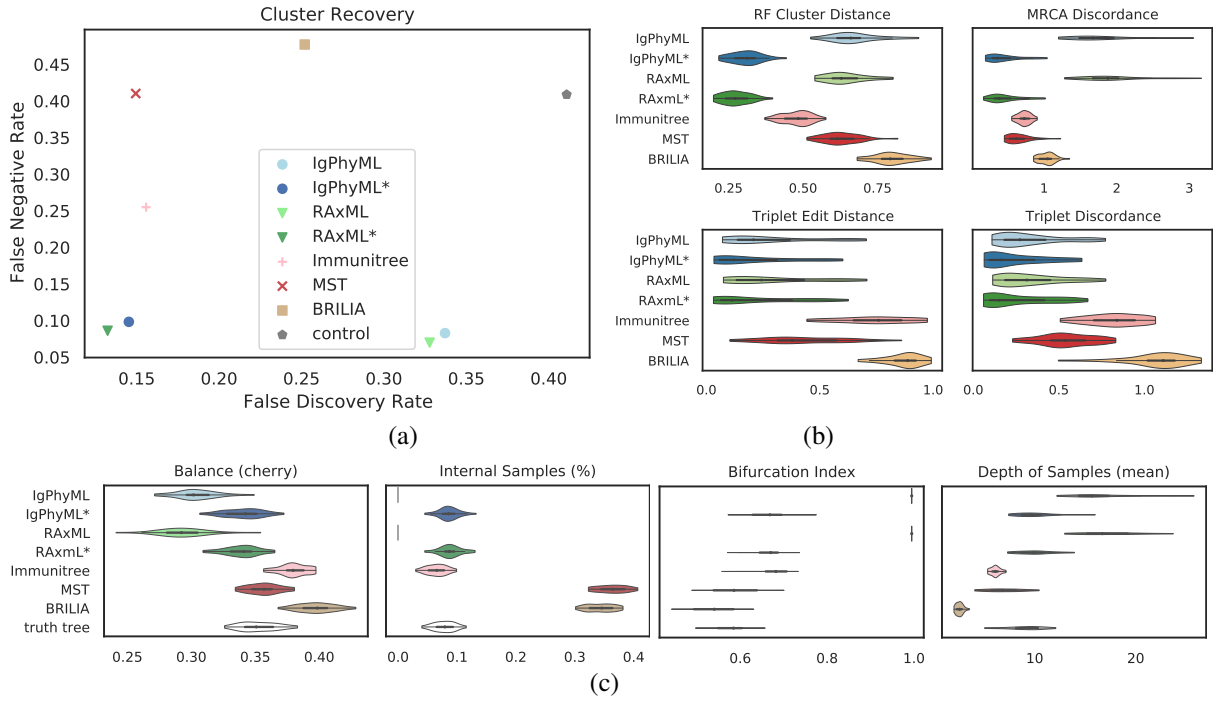

Figure 7.4. (a) False Discovery Rate (FDR) and False Negative Rate (FNR) of various reconstruction methods on simulations under default conditions (30 replicates); (b) Normalized Robinson-Foulds cluster distance (RF), MRCA discordance (MD), triplet edit distance (TED), and triplet discordance (TD). (c) Properties of the estimated and true trees. For results excluding singletons and the PD metric, see Fig. S7.2.

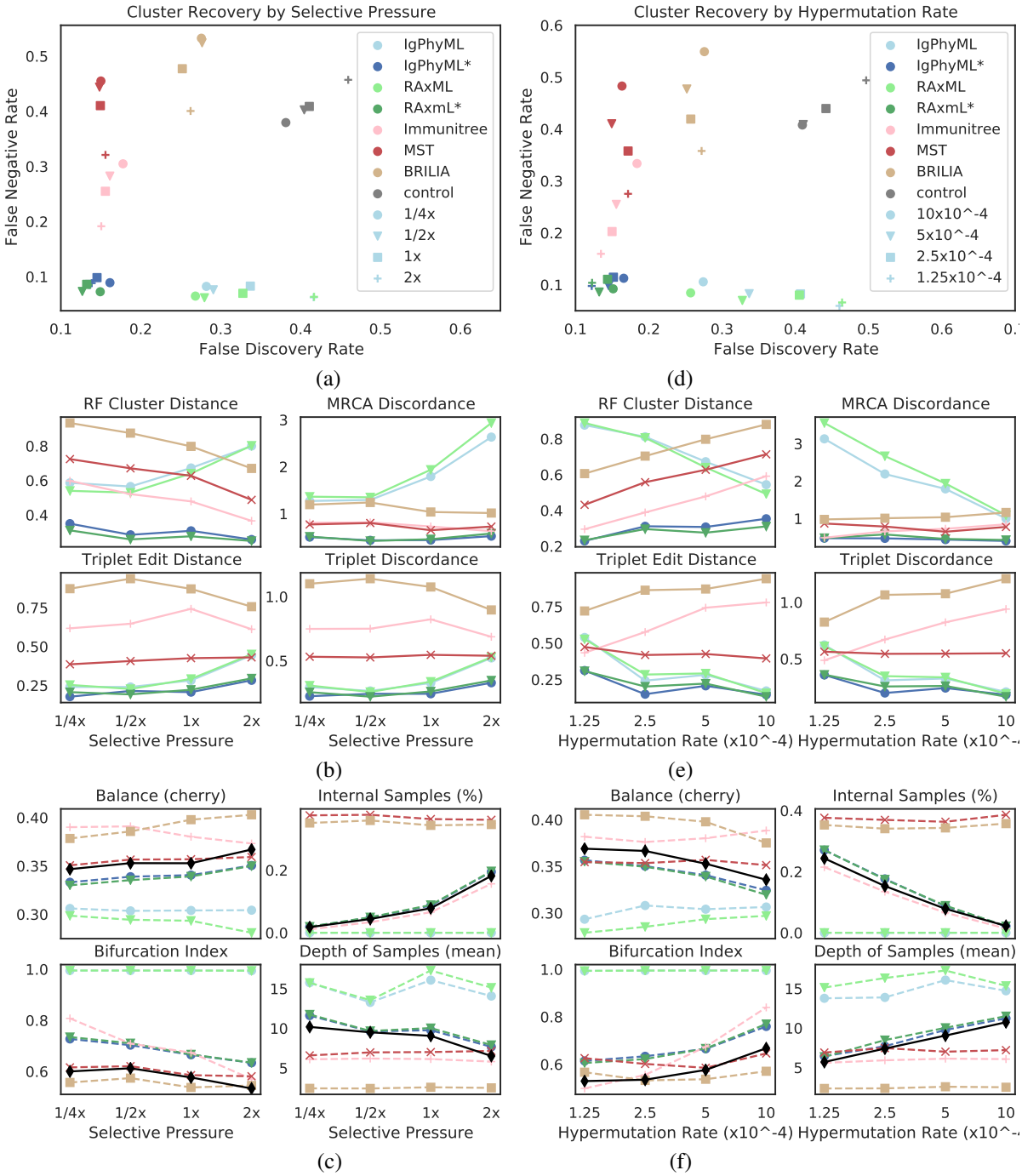

Figure 7.5. Impact of selective pressure A (a-c) and mutation rate μ (d-f) on tree inference error (a,b,d,e) and tree properties (c,f). We measure tree error by FDR and FNR (a,d), Robinson-Foulds cluster distance (RF), MRCA discordance (MD), triplet edit distance (TED), and triplet discordance (TD) (b,e). Tree errors and tree properties are averaged over 30 replicates. We show properties of true (black) and reconstructed trees (c,f). $\mu = 5 \times 10^{-5}$ in (a-c) and $A = 0.1$ in (d-f), which are all default values.

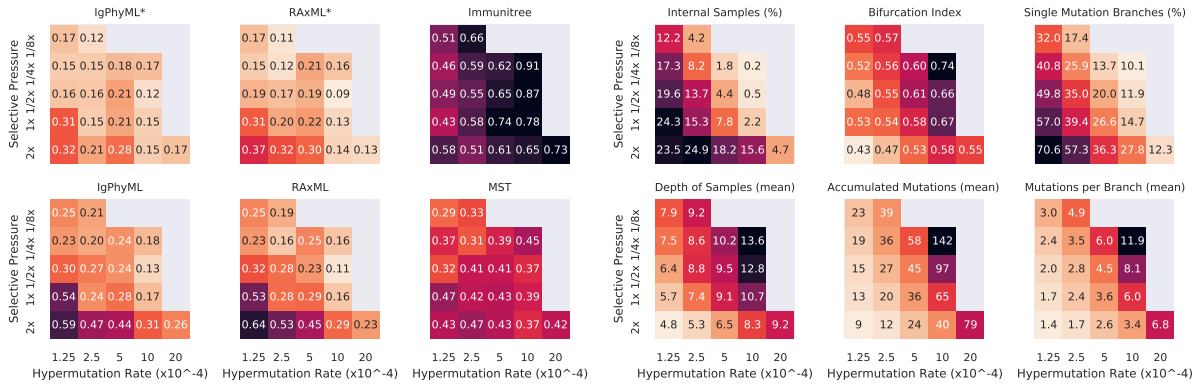

Figure 7.6. For varying levels of selective pressure (A), rate of hypermutation (μ), and all reconstruction methods except BRILIA, we show tree error measured by the triplet edit distance TED (left) and properties of the true tree (right). When the mutation rate is too high and the selective pressure is too low, the simulation never ends, meaning that the total affinity needed to overcome the antigen is never reached; these conditions are missing from the figure. For other evaluation criteria see S7.5.

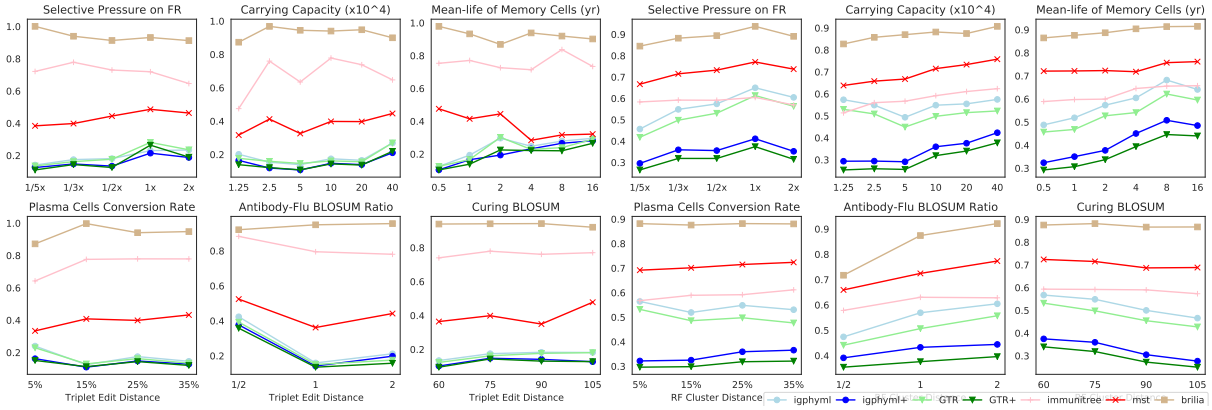

Figure 7.7. a) Triplet edit distances and b) RF cluster distances by selective pressure on framework region, carrying capacity, mean-life of memory cells, plasma cell conversion rate, antibody-flu BLOSUM ratio (MARatio), stage change threshold (M).

Table 7.1. Parameters of the AM model

Param.	Default	Parameter description
λ'_d	$1/402$	Rate (inverse life time) of cell death for memory cells (days^{-1})
λ_b	6	Rate of cell division for activated B cells (days^{-1})
λ_d	10^4	Rate of cell death during dormant stage (day^{-1}).
λ_t	0.01	Rate of activation of a typical responsive memory cell
ρ_p	$1/100$	Portion of activated B cells that turn into plasma cells per cell division
ρ_m	$1/4$	Portion of activated B cells that turn into memory B cells per cell division
μ	5×10^{-4}	Rate of SHMs per base pair per generation
$\mathbf{K}^5$	Appendix 7.B	Empirical 5-mer mutation frequencies per generation
L	125	Length of the amino acid antibody-coding sequence (assuming the length is fixed)
CDR	31–35,50–65, 98–114	Positions of the three CDR regions (amino acid coordinates)
$\delta(i,j)$	Table S7.2	BLOSUM matrix defined on a pair of amino-acids i and j
Δ_0	-120	BLOSUM score of a typical memory B cell antibody-coding sequence to target
Δ'_0	-75	BLOSUM score of activated B cell antibody-coding sequences that leads to cure
w_f	$1/3$	BLOSUM score multiplier of non-CDR positions (i.e., FRs)
κ	2	BLOSUM score ratio of antibody-coding sequences to antigen sequences
A	0.1	Selective pressure: factor connecting sequence similarity and log binding affinity
ρ_a	$1/2$	Factor connecting log affinity and B cell activation (sensitivity to affinity level A)
C	10^5	Carrying capacity limited by total resources (see text for meaning)
M	$Ce^{A\Delta'_0}$	The threshold of the sum of affinity for a stage change
r	56	Rounds of viral infections
$\hat{\Psi}$	Appendix 7.B	Nucleotide sequence of the initial B cell
$\zeta_1 \dots \zeta_r$	Appendix 7.B	Target amino acid sequences for viral infections in each round
$\eta_1 \dots \eta_r$	Appendix 7.B	Flu sequences assumed as antigens in the simulation
$t_1 \dots t_r$	Appendix 7.B	Starting time of each infected stage (day)

Table 7.2. Birth, death, and transformation rates. See Table S7.1 for polynomial forms.

Rate functions	Infected stage	Dormant stage
$\Lambda_B(\mathbf{x}_i,\mathbf{S})$	$g_i\lambda_b + (1-g_i)\times 0$	0
$\Lambda_D(\mathbf{x}_i,\mathbf{S})$	$g_i(\frac{\lambda_b(1-\rho_p-\rho_m)}{C}\frac{\sigma}{a_i}+\rho_p\lambda_b)+(1-g_i)\lambda'_d$	$g_i\lambda_d+(1-g_i)\lambda'_d$
$\Lambda_T(\mathbf{x}_i,\mathbf{S})$	$t_i = g_i\rho_m\lambda_b + e^{-\rho_a A\Delta_0}a_i^{\rho_a}(1-g_i)$	0

Table 7.3. Experiment setup

Experiment	Parameters	Parameter values	Parameter units
Selective pressure vs. rate of hypermutation	$A\times\mu$	$(2,2),(2,1),(2,1/2),(2,1/4),(2,1/8),(1,2),$ $(1,1),(1,1/2),(1,1/4),(1,1/8),(1/2,1),(1/2,1/2),$ $(1/2,1/4),(1/2,1/8),(1/4,1),(1/4,1/2),$ $(1/4,1/4),(1/4,1/8),(1/8,1/4),(1/8,1/8)$	$A:10^{-1}$, $\mu:10^{-3}$
Framework weight	w_f	$2,1,1/2,1/3,1/5$	1
Germinal center size	C	$4,2,1,1/2,1/4,1/8$	10^5
Memory cell life	$1/\lambda'_d$	$16,8,4,2,1,1/2$	year (365 days)

Table 7.4. Properties of a clonal tree

Property	Definition
Internal sample (%)	The percentage of labeled nodes that are internal nodes.
Bifurcation index	Ratio of the number of internal nodes to one less than leaf nodes; equals to 1 for bifurcating trees and ≈ 0 for a star tree.
Sample depth	The average depth of labeled nodes.
Balance (cherry)	Half the sum over all leaves of the fraction of their siblings that are also leaves.
Single mutation branches (%)	The percentage of branches with length one.
Accumulated mutations (avg)	The average depth (path length to the root) of all labeled nodes .
Accumulated mutations (sum)	The summation of branch lengths of all branches.
Mutations per branch	The average branch length.

Table 7.5. Metrics for comparing the reference tree R to estimated tree E. See Table S7.5

Metric (abrv.)	Definition
False Discovery Rate (FDR)	the percentage of clusters in E that are not in R
False Negative Rate (FNR)	the percentage of clusters in R that are not in E
RF cluster distance (RF)	the number of clusters in either but not both trees
FDR* , FNR*, and RF$*$	similar to the previous metrics but with singletons excluded
Triplet discordance (TD)	the number of trees induced by triples of *labeled* nodes (leaf or internal) where the topology in the simulated tree and the reconstructed tree differ
Triplet edit distance TED	the sum of the cluster RF distance induced to each triplet of labeled nodes = the number of branch contractions/resolutions that make every triplet of R match E
MRCA Discordance MD	the summation of MRCA discordance† over all ordered pairs of labeled nodes.
Patristic Distance PD	the summation of the patristic discordance‡ over all pairs of labeled nodes.

† MRCA discordance of two labeled nodes is the difference between the number of branches in the path between each of them and their MRCA.

‡ Patristic discordance for a pair of labeled nodes is the difference between the number of branches in the path between the two nodes on the two trees R and E.

Table 7.6. A comparison of Most relevant tools for AM simulation.

	DIMSIM this paper	bcr-phylo Davidsen and Matsen (2018)	gcdynamics Childs et al. (2015)
Targets	Single-target (per round)	Multi-target (1 round)	Multi-target (1 round)
Rounds	Yes	No	No
Affinity	BLOSUM distance	Hamming distance	Random energy landscape
Mutation	Updated Yaari et al. (2013)	Yaari et al. (2013)	i.i.d
Scalability	Up to millions of cells	Thousands of cells	Thousands of cells
Cell type	Activated & Memory	Activated	Activated
Germinal Centers	Combined (single)	Combined (single)	Multiple (in competition)
Time	Continuous	Discrete generations	Discrete generations
Isotype	No	Yes	No
Birth/Death rate	Polynomial fraction of individual and total affinity	Neutral: independent of total affinity Kinetic: function of affinities	A function of affinity

Bibliography

A. Amitai, L. Mesin, G. D. Victora, M. Kardar, and A. K. Chakraborty. A population dynamics model for clonal diversity in a germinal center. *Frontiers in Microbiology*, 8(SEP):1693, sep 2017. ISSN 1664302X. doi: 10.3389/fmicb.2017.01693. URL /pmc/articles/PMC5600966/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC5600966/.

M. Anisimova, O. Gascuel, and J. Sullivan. Approximate Likelihood-Ratio Test for Branches: A Fast, Accurate, and Powerful Alternative. *Systematic Biology*, 55(4):539–552, 8 2006. ISSN 1063-5157. URL http://dx.doi.org/10.1080/10635150600755453.

Y. Bao, P. Bolotov, D. Dernovoy, B. Kiryutin, L. Zaslavsky, T. Tatusova, J. Ostell, and D. Lipman. The Influenza Virus Resource at the National Center for Biotechnology Information. *Journal of Virology*, 82(2):596–601, jan 2008. ISSN 0022-538X. doi: 10.1128/jvi.02005-07.

M. Barak, N. S. Zuckerman, H. Edelman, R. Unger, and R. Mehr. IgTree©: Creating Immunoglobulin variable region gene lineage trees. *Journal of Immunological Methods*, 338(1-2):67–74, sep 2008. ISSN 0022-1759. doi: 10.1016/J.JIM.2008.06.006. URL https://www.sciencedirect.com/science/article/abs/pii/S0022175908002330?via{%}3Dihub.

D. A. Bolotin, S. Poslavsky, I. Mitrophanov, M. Shugay, I. Z. Mamedov, E. V. Putintseva, and D. M. Chudakov. MiXCR: software for comprehensive adaptive immunity profiling. *Nature Methods*, 12(5):380–381, 5 2015. ISSN 1548-7091. doi: 10.1038/nmeth.3364. URL http://www.nature.com/articles/nmeth.3364.

M. Bonsignori, T. Zhou, Z. Sheng, L. Chen, F. Gao, M. G. Joyce, G. Ozorowski, G.-Y. Chuang, C. A. Schramm, K. Wiehe, S. M. Alam, T. Bradley, M. A. Gladden, K.-K. Hwang, S. Iyengar, A. Kumar, X. Lu, K. Luo, M. C. Mangiapani, R. J. Parks, H. Song, P. Acharya, R. T. Bailer, A. Cao, A. Druz, I. S. Georgiev, Y. D. Kwon, M. K. Louder, B. Zhang, A. Zheng, B. J. Hill, R. Kong, C. Soto, J. C. Mullikin, D. C. Douek, D. C. Montefiori, M. A. Moody, G. M. Shaw, B. H. Hahn, G. Kelsoe, P. T. Hraber, B. T. Korber, S. D. Boyd, A. Z. Fire, T. B. Kepler, L. Shapiro, A. B. Ward, J. R. Mascola, H.-X. Liao, P. D. Kwong, and B. F. Haynes. Maturation Pathway from Germline to Broad HIV-1 Neutralizer of a CD4-Mimic Antibody. *Cell*, 165(2):449–463, 4 2016. ISSN 00928674. doi: 10.1016/j.cell.2016.02.022. URL https://linkinghub.elsevier.com/retrieve/pii/S0092867416301246.

R. Bransteitter, P. Pham, P. Calabrese, and M. F. Goodman. Biochemical analysis of hypermutational targeting by wild type and mutant activation-induced cytidine deaminase. *J. Biol. Chem.*, 279(49):51612–51621, Dec 2004.

L. M. Childs, E. B. Baskerville, and S. Cobey. Trade-offs in antibody repertoires to complex antigens. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 370(1676), sep 2015. ISSN 14712970. doi: 10.1098/rstb.2014.0245. URL /pmc/articles/PMC4528422/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC4528422/.

K. Davidsen and F. A. Matsen. Benchmarking Tree and Ancestral Sequence Inference for B Cell Receptor Sequences. *Frontiers in immunology*, 9:2451, 2018. ISSN 16643224. doi: 10.3389/fimmu.2018.02451. URL /pmc/articles/PMC6220437/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC6220437/.

R. Davidson, M. Lawhorn, J. Rusinko, and N. Weber. Efficient Quartet Representations of Trees and Applications to Supertree and Summary Methods. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(3):1010–1015, 5 2018. ISSN 1545-5963. doi: 10.1109/TCBB.2016.2638911. URL https://ieeexplore.ieee.org/document/7782719/.

W. S. r. DeWitt, L. Mesin, G. D. Victora, V. N. Minin, and F. A. t. Matsen. Using Genotype Abundance to Improve Phylogenetic Inference. *Molecular biology and evolution*, 35(5):1253–1265, may 2018. ISSN 1537-1719 (Electronic). doi: 10.1093/molbev/msy020.

Z. DiNardo, K. Tomlinson, A. Ritz, and L. Oesper. Distance measures for tumor evolutionary trees. *Bioinformatics*, 36(7):2090–2097, 4 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz869. URL https://academic.oup.com/bioinformatics/article/36/7/2090/5637226.

Y. Elhanati, Z. Sethna, Q. Marcou, C. G. Callan, T. Mora, and A. M. Walczak. Inferring processes underlying B-cell repertoire diversity. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370(1676):015115, 2015. ISSN 1471-2970. doi: 10.1101/015115. URL http://jnl-biorxiv.drupal-stage-jnl-web01.highwire.org/content/early/2015/02/11/015115.abstract.

S. Eliyahu, O. Sharabi, S. Elmedvi, R. Timor, A. Davidovich, F. Vigneault, C. Clouser, R. Hope, A. Nimer, M. Braun, Y. Y. Weiss, P. Polak, G. Yaari, and M. Gal-Tanamy. Antibody Repertoire Analysis of Hepatitis C Virus Infections Identifies Immune Signatures Associated With Spontaneous Clearance. *Frontiers in Immunology*, 9, 12 2018. ISSN 1664-3224. doi: 10.3389/fimmu.2018.03004. URL https://www.frontiersin.org/article/10.3389/fimmu.2018.03004/full.

A. H. Ellebedy, K. J. L. Jackson, H. T. Kissick, H. I. Nakaya, C. W. Davis, K. M. Roskin, A. K. McElroy, C. M. Oshansky, R. Elbein, S. Thomas, G. M. Lyon, C. F. Spiropoulou, A. K. Mehta, P. G. Thomas, S. D. Boyd, and R. Ahmed. Defining antigen-specific plasmablast and memory B cell subsets in human blood after viral infection or vaccination. *Nature*

Immunology, 17(10):1226–1234, 10 2016. ISSN 1529-2908. doi: 10.1038/ni.3533. URL http://www.nature.com/articles/ni.3533.

S. E. Elliott, S. Kongpachith, N. Lingampalli, J. Z. Adamska, B. J. Cannon, R. Mao, L. K. Blum, and W. H. Robinson. Affinity Maturation Drives Epitope Spreading and Generation of Proinflammatory Anti–Citrullinated Protein Antibodies in Rheumatoid Arthritis. *Arthritis & Rheumatology*, 70(12):1946–1958, 12 2018. ISSN 2326-5191. doi: 10.1002/art.40587. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/art.40587.

J. D. Galson, J. Trück, E. A. Clutterbuck, A. Fowler, V. Cerundolo, A. J. Pollard, G. Lunter, and D. F. Kelly. B-cell repertoire dynamics after sequential hepatitis B vaccination and evidence for cross-reactive B-cell activation. *Genome Medicine*, 8(1):68, 12 2016. ISSN 1756-994X. doi: 10.1186/s13073-016-0322-z. URL http://genomemedicine.biomedcentral.com/articles/10.1186/s13073-016-0322-z.

G. Georgiou, G. C. Ippolito, J. Beausang, C. E. Busse, H. Wardemann, and S. R. Quake. The promise and challenge of high-throughput sequencing of the antibody repertoire. *Nature Biotechnology*, 32(2):158–168, 2 2014. ISSN 1087-0156. doi: 10.1038/nbt.2782. URL http://www.nature.com/articles/nbt.2782.

A. D. Gitlin, Z. Shulman, and M. C. Nussenzweig. Clonal selection in the germinal centre by regulated proliferation and hypermutation. *Nature*, 509(7502):637–640, 5 2014. ISSN 0028-0836. doi: 10.1038/nature13300. URL http://www.nature.com/articles/nature13300.

N. Goldman, J. P. Anderson, and a. G. Rodrigo. Likelihood-based tests of topologies in phylogenetics. *Systematic biology*, 49(4):652–70, 12 2000. ISSN 1063-5157. URL http://www.ncbi.nlm.nih.gov/pubmed/12116432.

B. F. Haynes, G. Kelsoe, S. C. Harrison, and T. B. Kepler. B-cell–lineage immunogen design in vaccine development with HIV-1 as a case study. *Nature Biotechnology*, 30(5):423–433, 5 2012. ISSN 1087-0156. doi: 10.1038/nbt.2197. URL http://www.nature.com/articles/nbt.2197.

C. J. Henry Dunand and P. C. Wilson. Restricted, canonical, stereotyped and convergent immunoglobulin responses. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370(1676):20140238–, 2015. ISSN 1471-2970. doi: 10.1098/rstb.2014.0238. URL http://rstb.royalsocietypublishing.org/content/370/1676/20140238.

K. B. Hoehn, G. Lunter, and O. G. Pybus. A phylogenetic codon substitution model for antibody lineages. *Genetics*, 206(1):417–427, may 2017. ISSN 19432631. doi: 10.1534/genetics.116.196303.

F. Horns, C. Vollmers, D. Croote, S. F. Mackey, G. E. Swan, C. L. Dekker, M. M. Davis, and S. R. Quake. Lineage tracing of human B cells reveals the in vivo landscape of human antibody class switching. *eLife*, 5, 8 2016. ISSN 2050-084X. doi: 10.7554/eLife.16578. URL

https://elifesciences.org/articles/16578.

F. Horns, C. Vollmers, C. L. Dekker, and S. R. Quake. Signatures of selection in the human antibody repertoire: Selective sweeps, competing subclones, and neutral drift. *Proceedings of the National Academy of Sciences*, 116(4):1261–1266, 1 2019. ISSN 0027-8424. doi: 10.1073/pnas.1814213116. URL http://www.pnas.org/lookup/doi/10.1073/pnas.1814213116.

Y.-C. Hsiao, Y. Shang, D. M. DiCara, A. Yee, J. Lai, S. H. Kim, D. Ellerman, R. Corpuz, Y. Chen, S. Rajan, H. Cai, Y. Wu, D. Seshasayee, and I. Hötzel. Immune repertoire mining for rapid affinity optimization of mouse monoclonal antibodies. *mAbs*, 11(4):735–746, 5 2019. ISSN 1942-0862. doi: 10.1080/19420862.2019.1584517. URL https://www.tandfonline.com/doi/full/10.1080/19420862.2019.1584517.

T. R. Jackman, A. Larson, K. de Queiroz, J. B. Losos, and D. Cannatella. Phylogenetic Relationships and Tempo of Early Diversification in Anolis Lizards. *Systematic Biology*, 48 (2):254–285, 6 1999. ISSN 1063-5157. URL http://dx.doi.org/10.1080/106351599260283.

N. Jiang, J. He, J. a. Weinstein, L. Penland, S. Sasaki, X.-S. He, C. L. Dekker, N.-Y. Zheng, M. Huang, M. Sullivan, P. C. Wilson, H. B. Greenberg, M. M. Davis, D. S. Fisher, and S. R. Quake. Lineage structure of the human antibody repertoire in response to influenza vaccination. *Science translational medicine*, 5(171):171ra19, 2013. ISSN 1946-6242. doi: 10.1126/scitranslmed.3004794. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3699344&tool=pmcentrez&rendertype=abstract.

T. B. Kepler, S. Munshaw, K. Wiehe, R. Zhang, J. S. Yu, C. W. Woods, T. N. Denny, G. D. Tomaras, S. M. Alam, M. A. Moody, G. Kelsoe, H.-X. Liao, and B. F. Haynes. Reconstructing a B-cell clonal lineage. II. Mutation, selection, and affinity maturation. *Frontiers in Immunology*, 5(APR):1–10, 2014. ISSN 16643224. doi: 10.3389/fimmu.2014.00170.

U. Kirik, H. Persson, F. Levander, L. Greiff, and M. Ohlin. Antibody Heavy Chain Variable Domains of Different Germline Gene Origins Diversify through Different Paths. *Frontiers in Immunology*, 8(e33038):1433, 11 2017. ISSN 1664-3224. doi: 10.3389/fimmu.2017.01433. URL http://journal.frontiersin.org/article/10.3389/fimmu.2017.01433/full.

Y. Kurosawa and S. Tonegawa. Organization, structure, and assembly of immunoglobulin heavy chain diversity DNA segments. *The Journal of Experimental Medicine*, 155(1):201–218, 1 1982. ISSN 0022-1007. doi: 10.1084/jem.155.1.201. URL https://rupress.org/jem/article/155/1/201/22892/Organization-structure-and-assembly-of.

U. Laserson, F. Vigneault, D. Gadala-Maria, G. Yaari, M. Uduman, J. A. Vander Heiden, W. Kelton, S. Taek Jung, Y. Liu, J. Laserson, R. Chari, J.-H. Lee, I. Bachelet, B. Hickey, E. Lieberman-Aiden, B. Hanczaruk, B. B. Simen, M. Egholm, D. Koller, G. Georgiou, S. H. Kleinstein, and G. M. Church. High-resolution antibody dynamics of vaccine-induced immune responses. *Proceedings of the National Academy of Sciences*, 111(13):4928–4933, 4 2014.

ISSN 0027-8424. doi: 10.1073/pnas.1323862111. URL http://www.pnas.org/lookup/doi/10.1073/pnas.1323862111.

D. W. Lee, I. V. Khavrutskii, A. Wallqvist, S. Bavari, C. L. Cooper, and S. Chaudhury. BRILIA: Integrated tool for high-throughput annotation and lineage tree assembly of B-cell repertoires. *Frontiers in Immunology*, 7(JAN):681, jan 2017. ISSN 16643224. doi: 10.3389/fimmu.2016.00681.

W. D. Lees and A. J. Shepherd. Utilities for High-Throughput Analysis of B-Cell Clonal Lineages. *Journal of immunology research*, 2015:323506, 2015. ISSN 2314-7156 (Electronic). doi: 10.1155/2015/323506.

P. O. Lewis, M. T. Holder, and K. E. Holsinger. Polytomies and bayesian phylogenetic inference. *Systematic Biology*, 54(2):241–253, 2005. ISSN 10635157. doi: 10.1080/10635150590924208.

A. Lossius, J. N. Johansen, F. Vartdal, and T. Holmøy. High-throughput sequencing of immune repertoires in multiple sclerosis. *Annals of Clinical and Translational Neurology*, 3(4):295–306, 4 2016. ISSN 23289503. doi: 10.1002/acn3.295. URL http://doi.wiley.com/10.1002/acn3.295.

S. Luo and A. S. Perelson. The challenges of modelling antibody repertoire dynamics in HIV infection. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370(1676):20140247–, 2015. ISSN 1471-2970. doi: 10.1098/rstb.2014.0247. URL http://rstb.royalsocietypublishing.org/content/370/1676/20140247.

G. Magri, L. Comerma, M. Pybus, J. Sintes, D. Lligé, D. Segura-Garzón, S. Bascones, A. Yeste, E. K. Grasset, C. Gutzeit, M. Uzzan, M. Ramanujam, M. C. van Zelm, R. Albero-González, I. Vazquez, M. Iglesias, S. Serrano, L. Márquez, E. Mercade, S. Mehandru, and A. Cerutti. Human Secretory IgM Emerges from Plasma Cells Clonally Related to Gut Memory B Cells and Targets Highly Diverse Commensals. *Immunity*, 47(1):118–134, 7 2017. ISSN 10747613. doi: 10.1016/j.immuni.2017.06.013. URL https://linkinghub.elsevier.com/retrieve/pii/S107476131730273X.

Q. Marcou, T. Mora, and A. M. Walczak. High-throughput immune repertoire analysis with IGoR. *Nature Communications*, 9(1):561, 12 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-02832-w. URL http://www.nature.com/articles/s41467-018-02832-w.

C. O. McCoy, T. Bedford, V. N. Minin, P. Bradley, H. Robins, and F. A. Matsen. Quantifying evolutionary constraints on B-cell affinity maturation. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370(1676):20140244, 9 2015. ISSN 0962-8436. doi: 10.1098/rstb.2014.0244. URL http://rstb.royalsocietypublishing.org/lookup/doi/10.1098/rstb.2014.0244.

L. Mesin, A. Schiepers, J. Ersching, A. Barbulescu, C. B. Cavazzoni, A. Angelini, T. Okada,

T. Kurosaki, and G. D. Victora. Restricted clonality and limited germinal center reentry characterize memory b cell reactivation by boosting. *Cell*, 180(1):92–106.e11, 2020. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2019.11.032. URL https://www.sciencedirect.com/science/article/pii/S0092867419313170.

E. Miho, A. Yermanos, C. R. Weber, C. T. Berger, S. T. Reddy, and V. Greiff. Computational Strategies for Dissecting the High-Dimensional Complexity of Adaptive Immune Repertoires. *Frontiers in Immunology*, 9, 2 2018. ISSN 1664-3224. doi: 10.3389/fimmu.2018.00224. URL http://journal.frontiersin.org/article/10.3389/fimmu.2018.00224/full.

N. Moshiri, M. Ragonnet-Cronin, J. O. Wertheim, and S. Mirarab. FAVITES: simultaneous simulation of transmission networks, phylogenetic trees and sequences. *Bioinformatics*, 35(11):1852–1861, 6 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/bty921. URL https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/bty921/5161084https://academic.oup.com/bioinformatics/article/35/11/1852/5161084.

M. Muramatsu, K. Kinoshita, S. Fagarasan, S. Yamada, Y. Shinkai, and T. Honjo. Class Switch Recombination and Hypermutation Require Activation-Induced Cytidine Deaminase (AID), a Potential RNA Editing Enzyme. *Cell*, 102(5):553–563, 9 2000. ISSN 00928674. doi: 10.1016/S0092-8674(00)00078-7. URL https://linkinghub.elsevier.com/retrieve/pii/S0092867400000787.

R. Nakagawa and D. P. Calado. Positive Selection in the Light Zone of Germinal Centers. *Frontiers in Immunology*, 12:1053, 3 2021. ISSN 16643224. doi: 10.3389/FIMMU.2021.661678/BIBTEX.

S. Nee. Birth-Death Models in Macroevolution. *Annual Review of Ecology, Evolution, and Systematics*, 37(1):1–17, 12 2006. ISSN 1543-592X. doi: 10.1146/annurev.ecolsys.37.091305.110035. URL http://www.annualreviews.org/doi/10.1146/annurev.ecolsys.37.091305.110035.

M. S. Neuberger and C. Milstein. Somatic hypermutation. *Current Opinion in Immunology*, 7(2):248–254, 4 1995. ISSN 09527915. doi: 10.1016/0952-7915(95)80010-7. URL https://linkinghub.elsevier.com/retrieve/pii/0952791595800107.

V. Ovchinnikov, J. E. Louveau, J. P. Barton, M. Karplus, and A. K. Chakraborty. Role of framework mutations and antibody flexibility in the evolution of broadly neutralizing antibodies. *eLife*, 7, 2 2018. ISSN 2050-084X. doi: 10.7554/eLife.33038. URL https://elifesciences.org/articles/33038.

J. U. Peled, F. L. Kuang, M. D. Iglesias-Ussel, S. Roa, S. L. Kalis, M. F. Goodman, and M. D. Scharff. The biochemistry of somatic hypermutation. *Annu. Rev. Immunol.*, 26:481–511, 2008.

P. Pham, R. Bransteitter, J. Petruska, and M. F. Goodman. Processive AID-catalysed cytosine

deamination on single-stranded DNA simulates somatic hypermutation. *Nature*, 424(6944): 103–107, jul 2003. ISSN 00280836. doi: 10.1038/nature01760. URL www.nature.com/nature.

O. Ratmann, E. B. Hodcroft, M. Pickles, A. Cori, M. Hall, S. Lycett, C. Colijn, B. Dearlove, X. Didelot, S. Frost, A. M. M. Hossain, J. B. Joy, M. Kendall, D. Kühnert, G. E. Leventhal, R. Liang, G. Plazzotta, A. F. Poon, D. A. Rasmussen, T. Stadler, E. Volz, C. Weis, A. J. Leigh Brown, and C. Fraser. Phylogenetic Tools for Generalized HIV-1 Epidemics: Findings from the PANGEA-HIV Methods Comparison. *Molecular Biology and Evolution*, 34(1): 185–203, 1 2017. ISSN 0737-4038. doi: 10.1093/molbev/msw217. URL https://academic.oup.com/mbe/article-lookup/doi/10.1093/molbev/msw217.

P. Reshetova, B. D. van Schaik, P. L. Klarenbeek, M. E. Doorenspleet, R. E. Esveldt, P. P. Tak, J. E. Guikema, N. de Vries, and A. H. van Kampen. Computational model reveals limited correlation between germinal center B-cell subclone abundancy and affinity: Implications for repertoire sequencing. *Frontiers in Immunology*, 8(MAR):221, mar 2017. ISSN 16643224. doi: 10.3389/fimmu.2017.00221. URL /pmc/articles/PMC5337809/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC5337809/.

D. F. Robbiani, C. Gaebler, F. Muecksch, J. C. C. Lorenzi, Z. Wang, A. Cho, M. Agudelo, C. O. Barnes, A. Gazumyan, S. Finkin, T. Hägglöf, T. Y. Oliveira, C. Viant, A. Hurley, H.-H. Hoffmann, K. G. Millard, R. G. Kost, M. Cipolla, K. Gordon, F. Bianchini, S. T. Chen, V. Ramos, R. Patel, J. Dizon, I. Shimeliovich, P. Mendoza, H. Hartweger, L. Nogueira, M. Pack, J. Horowitz, F. Schmidt, Y. Weisblum, E. Michailidis, A. W. Ashbrook, E. Waltari, J. E. Pak, K. E. Huey-Tubman, N. Koranda, P. R. Hoffman, A. P. West, C. M. Rice, T. Hatziioannou, P. J. Bjorkman, P. D. Bieniasz, M. Caskey, and M. C. Nussenzweig. Convergent antibody responses to SARS-CoV-2 in convalescent individuals. *Nature*, 6 2020. ISSN 0028-0836. doi: 10.1038/s41586-020-2456-9. URL http://www.nature.com/articles/s41586-020-2456-9.

D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, 1981. URL http://www.sciencedirect.com/science/article/pii/0025556481900432.

W. H. Robinson. Sequencing the functional antibody repertoire—diagnostic and therapeutic discovery. *Nature Reviews Rheumatology*, 11(3):171–182, 3 2015. ISSN 1759-4790. doi: 10.1038/nrrheum.2014.220. URL http://dx.doi.org/10.1038/nrrheum.2014.22010.1038/nrrheum.2014.220http://www.nature.com/doifinder/10.1038/nrrheum.2014.220http://www.nature.com/articles/nrrheum.2014.220.

I. Rogozin and N. Kolchanov. Somatic hypermutagenesis in immunoglobulin genes. ii. influence of neighbouring base sequences on mutagenesis. *Biochim Biophys Acta*, 1171(1):11–18, 1992.

I. B. Rogozin and M. Diaz. Cutting edge: DGYW/WRCH is a better predictor of mutability at G:C bases in Ig hypermutation than the widely accepted RGYW/WRCY motif and probably reflects a two-step activation-induced cytidine deaminase-triggered process. *J. Immunol.*, 172 (6):3382–3384, Mar 2004.

Y. Safonova and P. Pevzner. IgEvolution: clonal analysis of antibody repertoires. *BioRxiv*, pages 1–18, 2019. doi: 10.1101/725424.

Y. Safonova, S. Bonissone, E. Kurpilyansky, E. Starostina, A. Lapidus, J. Stinson, L. DePalatis, W. Sandoval, J. Lill, and P. A. Pevzner. IgRepertoireConstructor: a novel algorithm for antibody repertoire construction and immunoproteogenomics analysis. *Bioinformatics*, 31 (12):i53–i61, 6 2015. ISSN 1367-4811. doi: 10.1093/bioinformatics/btv238. URL http://bioinformatics.oxfordjournals.org/cgi/content/long/31/12/i53.

G. S. Shapiro, M. C. Ellison, and L. J. Wysocki. Sequence-specific targeting of two bases on both DNA strands by the somatic hypermutation mechanism. *Mol. Immunol.*, 40(5):287–295, Sep 2003.

G. Sharbeen, C. W. Yee, A. L. Smith, and C. J. Jolly. Ectopic restriction of DNA repair reveals that UNG2 excises AID-induced uracils predominantly or exclusively during G1 phase. *Journal of Experimental Medicine*, 209(5):965–974, 5 2012. ISSN 1540-9538. doi: 10.1084/jem.20112379. URL https://rupress.org/jem/article/209/5/965/41182/Ectopic-restriction-of-DNA-repair-reveals-that.

A. Shlemov, S. Bankevich, A. Bzikadze, M. A. Turchaninova, Y. Safonova, and P. A. Pevzner. Reconstructing Antibody Repertoires from Error-Prone Immunosequencing Reads. *The Journal of Immunology*, 199(9):3369–3380, 11 2017. ISSN 0022-1767. doi: 10.4049/jimmunol.1700485. URL http://www.jimmunol.org/lookup/doi/10.4049/jimmunol.1700485.

D. S. Smith, G. Creadon, P. K. Jena, J. P. Portanova, B. L. Kotzin, and L. J. Wysocki. Di- and trinucleotide target preferences of somatic mutagenesis in normal and autoreactive B cells. *J. Immunol.*, 156(7):2642–2652, Apr 1996.

D. Sok, U. Laserson, J. Laserson, Y. Liu, F. Vigneault, J.-P. Julien, B. Briney, A. Ramos, K. F. Saye, K. Le, A. Mahan, S. Wang, M. Kardar, G. Yaari, L. M. Walker, B. B. Simen, E. P. St John, P.-Y. Chan-Hui, K. Swiderek, S. H. Kleinstein, G. Alter, M. S. Seaman, A. K. Chakraborty, D. Koller, I. A. Wilson, G. M. Church, D. R. Burton, and P. Poignard. The effects of somatic hypermutation on neutralization and binding in the PGT121 family of broadly neutralizing HIV antibodies. *PLoS pathogens*, 9(11):e1003754–e1003754, 2013a. ISSN 1553-7374. doi: 10.1371/journal.ppat.1003754. URL https://pubmed.ncbi.nlm.nih.gov/24278016https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3836729/.

D. Sok, U. Laserson, J. Laserson, Y. Liu, F. Vigneault, J.-P. Julien, B. Briney, A. Ramos, K. F. Saye, K. Le, A. Mahan, S. Wang, M. Kardar, G. Yaari, L. M. Walker, B. B. Simen, E. P. St. John, P.-Y. Chan-Hui, K. Swiderek, S. H. Kleinstein, G. Alter, M. S. Seaman, A. K. Chakraborty, D. Koller, I. A. Wilson, G. M. Church, D. R. Burton, and P. Poignard. The Effects of Somatic Hypermutation on Neutralization and Binding in the PGT121 Family of Broadly Neutralizing HIV Antibodies. *PLoS Pathogens*, 9(11):e1003754, 11 2013b. ISSN 1553-7374. doi: 10.1371/journal.ppat.1003754. URL https://dx.plos.org/10.1371/journal.ppat.1003754.

A. Stamatakis. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014. ISSN 14602059. doi: 10.1093/bioinformatics/btu033.

J. N. H. Stern, G. Yaari, J. A. Vander Heiden, G. Church, W. F. Donahue, R. Q. Hintzen, A. J. Huttner, J. D. Laman, R. M. Nagra, A. Nylander, D. Pitt, S. Ramanan, B. A. Siddiqui, F. Vigneault, S. H. Kleinstein, D. A. Hafler, and K. C. O'Connor. B cells populating the multiple sclerosis brain mature in the draining cervical lymph nodes. *Science Translational Medicine*, 6(248):107–248, 8 2014. ISSN 1946-6234. doi: 10.1126/scitranslmed.3008879. URL https://stm.sciencemag.org/lookup/doi/10.1126/scitranslmed.3008879.

T. Tanaka and M. Nei. Positive darwinian selection observed at the variable-region genes of immunoglobulins. *Molecular Biology and Evolution*, 6(5):447–4, 9 1989. ISSN 1537-1719. doi: 10.1093/oxfordjournals.molbev.a040569. URL https://academic.oup.com/mbe/article/6/5/447/1088568/Positive-darwinian-selection-observed-at-the.

S. Tonegawa. Somatic generation of antibody diversity. *Nature*, 302(5909):575–581, 4 1983. ISSN 0028-0836. doi: 10.1038/302575a0. URL http://www.nature.com/articles/302575a0.

J. A. Vander Heiden, G. Yaari, M. Uduman, J. N. Stern, K. C. O'Connor, D. A. Hafler, F. Vigneault, and S. H. Kleinstein. pRESTO: a toolkit for processing high-throughput sequencing raw reads of lymphocyte receptor repertoires. *Bioinformatics*, 30(13):1930–1932, 7 2014. ISSN 1460-2059. doi: 10.1093/bioinformatics/btu138. URL https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btu138.

H. E. Walsh, M. G. Kidd, T. Moum, and V. L. Friesen. Polytomies and the power of phylogenetic inference. *Evolution*, 53(3):932–937, 1999. doi: 10.1111/j.1558-5646.1999.tb05386.x.

C. T. Watson, J. Glanville, and W. A. Marasco. The Individual and Population Genetics of Antibody Immunity. *Trends in Immunology*, 38(7):459–470, 7 2017. ISSN 14714906. doi: 10.1016/j.it.2017.04.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1471490617300625.

C. R. Weber, R. Akbar, A. Yermanos, M. Pavlović, I. Snapkov, G. K. Sandve, S. T. Reddy, and V. Greiff. immuneSIM: tunable multi-feature simulation of B- and T-cell receptor repertoires for immunoinformatics benchmarking. *Bioinformatics*, 36(11):3594–3596, 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btaa158. URL https://doi.org/10.1093/bioinformatics/btaa158.

F. J. Weisel, G. V. Zuccarino-Catania, M. Chikina, and M. J. Shlomchik. A Temporal Switch in the Germinal Center Determines Differential Output of Memory B and Plasma Cells. *Immunity*, 44(1):116–130, 1 2016. ISSN 10747613. doi: 10.1016/j.immuni.2015.12.004. URL https://linkinghub.elsevier.com/retrieve/pii/S1074761315005051.

G. Yaari, J. A. Vander Heiden, M. Uduman, D. Gadala-Maria, N. Gupta, J. N. Joel, K. C.

O'Connor, D. A. Hafler, U. Laserson, F. Vigneault, and S. H. Kleinstein. Models of somatic hypermutation targeting and substitution based on synonymous mutations from high-throughput immunoglobulin sequencing data. *Frontiers in Immunology*, 4(NOV), 2013. ISSN 16643224. doi: 10.3389/fimmu.2013.00358.

G. Yaari, J. I. C. Benichou, J. A. Vander Heiden, S. H. Kleinstein, and Y. Louzoun. The mutation patterns in B-cell immunoglobulin receptors reflect the influence of selection acting at multiple time-scales. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370(1676):20140242–, 2015. ISSN 1471-2970. doi: 10.1098/rstb.2014.0242. URL http://rstb.royalsocietypublishing.org/content/370/1676/20140242.

A. D. Yermanos, A. K. Dounas, T. Stadler, A. Oxenius, and S. T. Reddy. Tracing Antibody Repertoire Evolution by Systems Phylogeny. *Frontiers in Immunology*, 9, 10 2018. ISSN 1664-3224. doi: 10.3389/fimmu.2018.02149. URL https://www.frontiersin.org/article/10.3389/fimmu.2018.02149/full.

C. Zhang, V. Dinh, and F. A. Matsen. Non-bifurcating phylogenetic tree inference via the adaptive LASSO. *Journal of the American Statistical Association*, pages 1–41, 6 2020. ISSN 0162-1459. doi: 10.1080/01621459.2020.1778481. URL https://www.tandfonline.com/doi/full/10.1080/01621459.2020.1778481.

Appendices

7.A Brief introduction of relevant concepts

Antibodies.

Antibodies are produced by *B cells* and are used by the immune system to recognize, bind, and neutralize pathogens. Antibodies are proteins consisting of immunoglobulin (IG) molecules of identical heavy chains and identical light chains. Immunoglobulins are encoded by B-cell receptor (BCR) sequences. Unlike other proteins, IGs are not encoded in the genome directly but present results of somatic *V(D)J recombination* of *IG loci* (Kurosawa and Tonegawa, 1982). Each chain of each IG is encoded by a concatenation of one of V, D (only for heavy chain), and J genes, known as an *IG gene*. An IG gene contains three *complementarity-determining regions* (CDRs) representing antigen binding sites. CDRs are separated by four *framework regions* (FRs) that form a stable structure displaying CDRs on the antibody surface.

AM process.

After successful binding of an IG to a given pathogen, the corresponding B cell undergoes the *affinity maturation* (AM) process aiming to improve its *affinity* (i.e., binding ability) to the antibody (Tonegawa, 1983; Neuberger and Milstein, 1995). First, the targeting B cell moves to a *germinal center* (GC) of a lymph node, where it undergoes *clonal expansion*: cell divisions that increase the pool of antibodies that bind to the antigen. Then, certain enzymes in the B cell and its clones are activated and introduce *somatic hypermutations* (SHMs) in the utilized IG genes as a means to improve affinity (Muramatsu et al., 2000). SHMs change the three-dimensional structure of an antibody (and thus its ability to bind to an antigen) stochastically. The regulatory

mechanisms of the immune system play the role of natural selection by expanding B cells with high affinity for antigen and killing self-reactive B cells with potentially harmful mutations. The AM process activates naive B cells (i.e., those that have not been exposed to an antigen) and differentiates them into *memory* and *plasma* B cells. Memory B cells can be repeatedly activated and subjected to the AM Mesin et al. (2020), while plasma B cells can secrete massive levels of neutralizing antibodies. Studies show that CDRs, which include the binding sites, accumulate more SHMs compared to FRs (Hsiao et al., 2019; Safonova and Pevzner, 2019).

Clonal expansion.

The AM process leads to the formation of clonal lineages within a given antibody repertoire, where each clonal lineage is formed by descendants of a single naive B cell. The expressed IG transcripts within the same clonal lineage share a common combination of V, D, and J genes and differ by SHMs only. The evolutionary history of each clonal lineage can be represented by a *clonal tree*, where each vertex corresponds to a B cell and each B cell is connected by a directed edge with all its immediate descendants.

7.B Supplementary methods

7.B.1 Efficient sampling from the BDT model

Recall that because of the memoryless property, the time until the next BDT event always follows the exponential distribution with rates $\Lambda_B(\mathbf{x}_i, \mathbf{S}), \Lambda_D(\mathbf{x}_i, \mathbf{S})$, and $\Lambda_T(\mathbf{x}_i, \mathbf{S})$ for each event type. The time until *any* event for *any* entity follows an exponential distribution with rate

$$\lambda = \sum_{i \in S} (\Lambda_B(\mathbf{x}_i, \mathbf{S}) + \Lambda_D(\mathbf{x}_i, \mathbf{S}) + \Lambda_T(\mathbf{x}_i, \mathbf{S})) \ .$$

The probability of the next event being a specific event $E \in \{B, D, T\}$ for a particular entity i is

$$\frac{\Lambda_E(\mathbf{x}_i, \mathbf{S})}{\lambda} \ .$$

We **assume** that we are able to write

$$\Lambda_E(\mathbf{x}_i, \mathbf{S}) = \frac{P_E(\mathbf{x}_i, \mathbf{S})}{Q(\mathbf{S})}$$

where $P_E : \mathbb{R}^N_{\geq 0} \times \mathbb{R}^N_{\geq 0} \to \mathbb{R}_{\geq 0}$ and $Q : \mathbb{R}^N_{\geq 0} \to \mathbb{R}_{>0}$ are polynomial functions with a constant degree, where coefficients of P_E are non-negative. With this assumption, for any entity $i \in S$, the birth rate can be written as

$$\Lambda_B(\mathbf{x}_i, \mathbf{S}) = \frac{\sum_{\alpha,\beta \in \Gamma} \mathscr{B}_{\alpha,\beta} \mathbf{S}^\beta \mathbf{x}_i^\alpha}{\sum_{\beta \in \Gamma} Q_\beta \mathbf{S}^\beta}$$

where $\Gamma = [0 \dots \gamma]^N$ for some integer γ, $\mathscr{B}_{\alpha,\beta}$ and Q_β are coefficients of the polynomials, and $\mathbf{a}^\mathbf{b}$ denotes $\prod_i \mathbf{a}_i^{\mathbf{b}_i}$ for vectors $\mathbf{a}$ and $\mathbf{b}$. We can write $\Lambda_D(\mathbf{x}_i, \mathbf{S})$ and $\Lambda_T(\mathbf{x}_i, \mathbf{S})$ similarly by replacing $\mathscr{B}_{\alpha,\beta}$ with $\mathscr{D}_{\alpha,\beta}$ and $\mathscr{T}_{\alpha,\beta}$. Note that in our specific AM model, rates shown in Table S7.1 follow this assumption.

With this assumption, we can write

$$\lambda = \frac{\sum_{\alpha,\beta\in\Gamma} P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}}{\sum_{\beta\in\Gamma} Q_{\beta}\mathbf{S}^{\beta}}$$

where $P_{\alpha,\beta} = \mathscr{B}_{\alpha,\beta} + \mathscr{D}_{\alpha,\beta} + \mathscr{T}_{\alpha,\beta}$ and $\theta_{\alpha} = \sum_{i\in S}\mathbf{x}_i^{\alpha}$ for all α values (note that $\mathbf{S} = \theta_1$). Thus, to efficiently sample the time till the next event, we only need θ_{α} values which we can simply store and update in constant time after each event. This fast storing and updating allows for a constant time sampling of the next event time (in terms of n) for constants N and γ. Once we sample the time till the next event, we need to sample one of the three possible events. The probability of the next event being birth for an entity i is

$$\begin{aligned}
\frac{\Lambda_B(\mathbf{x}_i,\mathbf{S})}{\lambda} =& \frac{\Lambda_B(\mathbf{x}_i,\mathbf{S})}{\sum_{j\in S}(\Lambda_B(\mathbf{x}_j,\mathbf{S})+\Lambda_D(\mathbf{x}_j,\mathbf{S})+\Lambda_T(\mathbf{x}_j,\mathbf{S}))} \\
=& \frac{\sum_{\alpha,\beta\in\Gamma}\mathscr{B}_{\alpha,\beta}\mathbf{S}^{\beta}\mathbf{x}_i^{\alpha}}{\sum_{\alpha,\beta\in\Gamma}P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}} = \sum_{\alpha,\beta\in\Gamma}\left(\mathscr{B}_{\alpha,\beta}\mathbf{S}^{\beta}\mathbf{x}_i^{\alpha}\frac{1}{\sum_{\bar{\alpha},\bar{\beta}\in\Gamma}P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}}\theta_{\bar{\alpha}}}\right) \\
=& \sum_{\alpha,\beta\in\Gamma}\left((\frac{\mathscr{B}_{\alpha,\beta}\mathbf{S}^{\beta}\mathbf{x}_i^{\alpha}}{P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}})(\frac{P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}}{\sum_{\bar{\alpha},\bar{\beta}\in\Gamma}P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}}\theta_{\bar{\alpha}}})\right) \\
=& \sum_{\alpha,\beta\in\Gamma}\left((\frac{\mathscr{B}_{\alpha,\beta}}{P_{\alpha,\beta}})(\frac{\mathbf{x}_i^{\alpha}}{\theta_{\alpha}})(\frac{P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}}{\sum_{\bar{\alpha},\bar{\beta}\in\Gamma}P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}}\theta_{\bar{\alpha}}})\right).
\end{aligned} \tag{7.5}$$

Also note that probability of each death and transformation event can be written similarly. This equation enables an efficient sampling procedure detailed in Algorithm S7.1 of Appendix 7.D:

1. Sample (α,β) pair (representing one term of the polynomial) from a multinomial distribu-

Table S7.1. Birth, death, and transformation rate functions as polynomials.

Rate functions	**Infected stage**	**Dormant stage**
$\Lambda_B(\mathbf{x}_i,\mathbf{S})$	$\lambda_b g_i$	0
$\Lambda_D(\mathbf{x}_i,\mathbf{S})$	$\frac{\lambda_b(1-\rho_p-\rho_m)}{C}(\frac{g_i}{a_i})\sigma + (\rho_p\lambda_b - \lambda_d')g_i + \lambda_d'$	$(\lambda_d - \lambda_d')g_i + \lambda_d'$
$\Lambda_T(\mathbf{x}_i,\mathbf{S})$	t_i	0

tion on $\Gamma \times \Gamma$ where each pair has probability $\frac{P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}}{\sum_{\bar{\alpha},\bar{\beta}\in\Gamma}P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}}\theta_{\bar{\alpha}}}$.

2. Sample entity i from a distribution on S where each i has probability $\mathbf{x}_i^{\alpha}/\theta_{\alpha}$.

3. Sample birth, death, or transformation with probabilities $\frac{\mathscr{B}_{\alpha,\beta}}{P_{\alpha,\beta}}$, $\frac{\mathscr{D}_{\alpha,\beta}}{P_{\alpha,\beta}}$, and $\frac{\mathscr{T}_{\alpha,\beta}}{P_{\alpha,\beta}}$.

In this procedure, the probability of selecting the birth event for an entity i is simply $\sum_{\alpha,\beta}\frac{\mathscr{B}_{\alpha,\beta}}{P_{\alpha,\beta}}\frac{\mathbf{x}_i^{\alpha}}{\theta_{\alpha}}\frac{P_{\alpha,\beta}\mathbf{S}^{\beta}\theta_{\alpha}}{\sum_{\bar{\alpha},\bar{\beta}\in\Gamma}P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}}\theta_{\bar{\alpha}}}$, which matches Equation (7.5) (ditto for death and transformation events). In terms of running time:

1. Step 1 takes constant time (in terms of n) given that θ_{α} values (and thus $\mathbf{S}$) are pre-computed for all α.

2. Step 2 can be achieved in $O(\log n)$ time using an interval tree data structure to store partial sums of $\mathbf{x}_j^{\alpha}$'s (see Algorithm S7.1).

3. Step 3 takes constant time.

Thus, a tree on k nodes drawn from the distribution defined by the BDT process can be sampled in $O(k\log(k))$ time by repeated applications of Algorithm S7.1.

7.B.2 Somatic hypermutagenesis frequency models

We next show the model for $\mathbf{K}^5$ and f. Our model is based on an empirical frequency $\mathbf{K}^5(s,s_1,s_2,s_3,s_4,s_5)$ matrix that counts the number of times 5-mer (s_1,s_2,s_3,s_4,s_5) converts to (s_1,s_2,s,s_4,s_5) in one cycle of cell division during hypermutation. Given the matrix, we define

$$f(s,s_1,s_2,s_3,s_4,s_5) = \begin{cases} \mathbf{K}^5(s,s_1,s_2,s_3,s_4,s_5)\frac{\mu}{\text{RateEmp}} & s \neq s_3 \\ 1-\sum_{s'\in\{A,C,G,T\}-\{s\}} \mathbf{K}^5(s',s_1,s_2,s_3,s_4,s_5) & s = s_3 \end{cases} \tag{7.6}$$

where

$$\text{RateEmp} = 1 - \frac{\sum_{s_1,s_2,s_3,s_4,s_5\in\{A,C,G,T\}} \mathbf{K}^5(s_3,s_1,s_2,s_3,s_4,s_5)}{\sum_{s,s_1,s_2,s_3,s_4,s_5\in\{A,C,G,T\}} \mathbf{K}^5(s,s_1,s_2,s_3,s_4,s_5)}\,. \tag{7.7}$$

Somatic hypermutagenesis of antibodies is the result of activation-induced deaminase (AID) enzyme activity that changes a random C:G base into a U:G base in B cell DNA. U:G mismatch can be repaired using UDG (uracil-DNA glycosylase) or MMR (DNA mismatch repair) machinery that forms diversity of hypermutations (Peled et al., 2008). Certain biological mechanisms of SHM occurrences were studied extensively. For example, Rogozin and Kolchanov (1992) observed specific hot/cold-spot DNA motifs for SHMs in immunoglobulin genes. Particularly, WRCY/RGYW where W = {A, T}, Y = {C, T}, R = {G, A} and later predicted more general WRCH/DGYW with H = {A, C, T} and D = {A, G, T} motifs are hot-spots for SHMs caused by weak hydrogen-bounds (Rogozin and Diaz, 2004). SYC/GRS (S = C, G) is a cold-spot motif caused by strong hydrogen-bounds (Bransteitter et al., 2004). The locality of AID enzyme activity has been emphasized. (Smith et al., 1996; Shapiro et al., 2003).

To simulate SHM, we modified a model proposed by Yaari et al. (2013). The model extends the notion of hot/cold-spots and suggests that a certain hierarchy of mutabilities exists following Smith et al. (1996) and Shapiro et al. (2003). The model is based on the mutability of a central base in each 5-mer of an antibody heavy chain and consists of two parts: a targeting model identifying if a mutation occurs in the variable part of an antibody and a substitution

model providing an insight into what is this mutation. In order to avoid selection bias, the authors considered 5-mers where only synonymous substitutions of the central base are possible and inferred probabilities for other 5-mers. Unfortunately, synonymous substitutions constitute only a fraction of possible mutations. To overcome this issue, Yaari et al. (2013) proposed a special inference method to estimate parameters for the rest of 5-mers. Parameters for targeting and substitution models were inferred for 468 and 740 5-mers, respectively. However, the accuracy of this procedure was shown to be suboptimal (Yaari et al., 2013, Table 2). Additionally, some of the datasets that were used to estimate the parameters are derived from an error-prone 454 sequencing technology.

We re-estimated the parameters of this model and considered all 5-mers without limiting our scope to synonymous mutations. We also utilized three up-to-date repertoire sequencing datasets (all data were produced using the Illumina MiSeq platform):

- PRJNA349143. Time series of three individuals during influenza vaccination, both before and after vaccination.
- PRJNA395083. Bulk unsorted PBMC from peripheral blood of several healthy donors.
- A dataset of paired end sequences, added to increase power.

While the last dataset we used is not publicly available, we make the resulting k-mer model available publicly at https://github.com/chaoszhang/immunosimulator/blob/master/kmerFreq.txt.

From each dataset, we obtained a matrix of the size 1024×4, where each row corresponds to a distinct 5-mer and contains *# non-mutated occurrences* of this 5-mer and three possible *# nucleotide substitution occurrences*. To calculate this matrix for a given dataset, we found the closest V gene for every read and record the number of observed 5-mers in the gene and their corresponding mutated copies across the read. For any 5-mer K, the corresponding row of a constructed matrix can be viewed simultaneously as a value of *Binomial* and *Multinomial* distributions. *Binomial* distribution represents the number of occurred mutations among all

occurrences of the 5-mer K, while *Multinomial* distribution indicates the number of mutations to specific bases among all occurred mutations. The parameters of these distributions indicate the mutability and substitution profiles for each 5-mer K. The 5-mer frequencies were combined across all these datasets to obtain the final matrix, available at https://github.com/chaoszhang/immunosimulator/blob/master/kmerFreq.txt.

7.B.3 Default parameters

Here we provide the actual default values used for several parameters that did not fit in Table 1 of the main paper.

BLOSUM.

The BLOSUM matrix table (Table S7.2) is obtained from ftp://ftp.ncbi.nih.gov/blast/matrices/BLOSUM100.

Table S7.2. BLOSUM table

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	8	-3	-4	-5	-2	-2	-3	-1	-4	-4	-4	-2	-3	-5	-2	1	-1	-6	-5	-2
R	-3	10	-2	-5	-8	0	-2	-6	-1	-7	-6	3	-4	-6	-5	-3	-3	-7	-5	-6
N	-4	-2	11	1	-5	-1	-2	-2	0	-7	-7	-1	-5	-7	-5	0	-1	-8	-5	-7
D	-5	-5	1	10	-8	-2	2	-4	-3	-8	-8	-3	-8	-8	-5	-2	-4	-10	-7	-8
C	-2	-8	-5	-8	14	-7	-9	-7	-8	-3	-5	-8	-4	-4	-8	-3	-3	-7	-6	-3
Q	-2	0	-1	-2	-7	11	2	-5	1	-6	-5	2	-2	-6	-4	-2	-3	-5	-4	-5
E	-3	-2	-2	2	-9	2	10	-6	-2	-7	-7	0	-5	-8	-4	-2	-3	-8	-7	-5
G	-1	-6	-2	-4	-7	-5	-6	9	-6	-9	-8	-5	-7	-8	-6	-2	-5	-7	-8	-8
H	-4	-1	0	-3	-8	1	-2	-6	13	-7	-6	-3	-5	-4	-5	-3	-4	-5	1	-7
I	-4	-7	-7	-8	-3	-6	-7	-9	-7	8	2	-6	1	-2	-7	-5	-3	-6	-4	4
L	-4	-6	-7	-8	-5	-5	-7	-8	-6	2	8	-6	3	0	-7	-6	-4	-5	-4	0
K	-2	3	-1	-3	-8	2	0	-5	-3	-6	-6	10	-4	-6	-3	-2	-3	-8	-5	-5
M	-3	-4	-5	-8	-4	-2	-5	-7	-5	1	3	-4	12	-1	-5	-4	-2	-4	-5	0
F	-5	-6	-7	-8	-4	-6	-8	-8	-4	-2	0	-6	-1	11	-7	-5	-5	0	4	-3
P	-2	-5	-5	-5	-8	-4	-4	-6	-5	-7	-7	-3	-5	-7	12	-3	-4	-8	-7	-6
S	1	-3	0	-2	-3	-2	-2	-2	-3	-5	-6	-2	-4	-5	-3	9	2	-7	-5	-4
T	-1	-3	-1	-4	-3	-3	-3	-5	-4	-3	-4	-3	-2	-5	-4	2	9	-7	-5	-1
W	-6	-7	-8	-10	-7	-5	-8	-7	-5	-6	-5	-8	-4	0	-8	-7	-7	17	2	-5
Y	-5	-5	-5	-7	-6	-4	-7	-8	1	-4	-4	-5	-5	4	-7	-5	-5	2	12	-5
V	-2	-6	-7	-8	-3	-5	-5	-8	-7	4	0	-5	0	-3	-6	-4	-1	-5	-5	8

Starting and target sequences.

The starting sequence $\hat{\Psi}$ is set to:

CAGGTGCAGCTGCAGGAGTCGGGCCCAGGACTGGTGAAGCCTTCACAGACCCTGTCCCTCACCTGCA
CTGTCTCTGGTGGCTCCATCAGCAGTGGTGGTTACTACTGGAGCTGGATCCGCCAGCACCCAGGGAAGGGCCT
GGAGTGGATTGGGTACATCTATTACAGTGGGAGCACCTACTACAACCCGTCCCTCAAGAGTCGAGTTACCATA
TCAGTAGACACGTCTAAGAACCAGTTCTCCCTGAAGCTGAGCTCTGTGACTGCCGCGGACACGGCCGTGTATT
ACTGTGCGAGAGCGCGCGTCAATAGGGATATTGCGTACGGCAACTGGTTCGACCCCTGGGGCCAGGGGACCCT
GGTCACCGTCTCCTCA

and thus ζ_0 is

QVQLQESGPGLVKPSQTLSLTCTVSGGSISSGGYYWSWIRQHPGKGLEWIGYIYYSGSTYYNPSLKS

RVTISVDTSKNQFSLKLSSVTAADTAVYYCARARVNRDIAYGNWFDPWGQGTLVTVSS.

η_i, ζ_i, and t_i are given in Table S7.3.

Table S7.3. Flu accession number, CDRs of target sequences, and starting day of infection

i	Accession	Target CDR1	Target CDR2	Target CDR3	Day
1	AAK70482.1	SGGYY	IGYIYYSGSTYYNPSL	ARARVNRDIAYGNWFDP	0
2	AAK70478.1	CWWVP	WWCHCGWCNVXXNIXF	ARARVNREXAYGNWFZA	182
3	ABL76892.1	WWWXX	XGYVYYSGSDYYDPSL	VKVKVNKEVVYGNWFEA	365
4	AFP83103.2	WWWAB	TBYVYYSGSDYYDXSL	VKVKINKEVVYGNWFEA	398
5	AFP83094.2	WWWGX	TGYVYYSGSDYYDXSL	VKVKVNKEVVYGNWFEQ	431
6	AFP83095.2	WWCPP	WWCHCAWXBTXXBISL	ARARVNRELAYGNWFEA	464
7	AFP83197.2	WWCPP	WWCHCZWYZVXXBISF	ARARVNRELAYGNXFEA	497
8	AFP83098.2	WWWAX	AGYVYYSGTDYYDBSL	VKVKINKEVVYGBWFEZ	530
9	AFP83100.2	WWWPK	SXHVYYSGSDYYDXSL	VKVKVNKEVVYGNWFEA	564
10	AAO38870.2	WWCPP	WWCHCCWXBVXYBXSY	ARARVNRELAYGNWFZA	597
11	AFP83199.2	WWLPP	WWCHCEWLHVXXXIXY	ARARVNRELAYGNWFZA	630
12	ABL76881.1	WLWCG	KXYVYYSGSQFYDASL	VKVKLNKEVVYGNWFZL	663
13	AFP83097.2	WCWCG	CRWVYYXXSDYYDIXL	VKVKINKEVVYGDWFEQ	696
14	AFP83202.2	WXYXY	TGYVYYSGSDYYDPSL	VKVKMNKEVVYGNWFEA	730
15	AFP83201.2	WWVPP	WWCNCCWFBTXXXLSF	ARARVNRELAYGNWFEA	763
16	AFP83118.2	WYYXD	TGYVYYSGSDYYBPSL	VKVKLNKEVVYGNWFZK	796
17	AFP83200.2	WWCPP	WWCHCCYIBVXXBXSY	ARARVNRELAYGNWFZA	829
18	AFP83107.2	WWCPP	WWCHCCYVBTXXBXSF	ARARVNRELAYGNWYZA	862
19	AFP83112.2	WFWDG	XKWVYYSGSDYYDXSL	VKVKINKZVVYGNWFEQ	895
20	AFP83115.2	WWCPP	WWCHCCQIBTXXBXSF	ARARVNRELAYGNWFZG	929
21	AFP83114.2	WPWGD	XGYVHYSRSDYYDPSL	VKVKXNKZVVYRNWFEP	962
22	AFP83110.2	WWCPD	WWCHCCWIDWXXBXXY	ARARVNRZLAYRNWFEA	995
23	AFP83105.2	WYWGN	GCXLYYSGSDYYDPSL	IKVKIDKELVYGDWFZV	1028
24	AFP83106.2	WWCPP	WWCHCCWVWWNEGLXB	GXXRXXRDLAYGNWYXA	1061
25	AFP83127.2	WFWBG	TGYLYYSGSDYYDASL	IKVKXNKELVYGNWFET	1095
26	AFP83124.2	WCWCG	BGYLYYSGSDYYBFSL	IKVCIBKEMVYGBWFET	1216
27	AFP83130.2	WWHPP	WWCHCCWRBCXXXXSF	ARARVNRSLAYGNWFEA	1338
28	AFP83134.2	WBYXY	TGYVYYSGSDYYBPSL	VKVKMNKEVVYGNWFEA	1460
29	AFP83131.2	WWHPP	WWCHCCWRBLXXXXSF	ARARVNRZLAYGNWFEA	1581
30	AFP83135.2	PPYGD	PGKVYYSRSDYYDDSL	IKVKXNKYVVYRNWFEK	1703
31	AFP83150.2	HPYGD	PGBVYYSRSDYYDBSL	VKVKINKZVVYRNWFEK	1825
32	AFP83206.2	HPYGD	PPHCYYSRSDYYDBSL	VKVKXNKFVVYRNWFEZ	1946
33	AFP83147.2	HPYGD	PGHVYYSRSDYYDPSL	IKVKINBXVVYRNWFEK	2068
34	AFP83154.2	WXXAY	PGYVYYSGSDYYDPSL	VKVKMNKEVVYGNWFEP	2190
35	AFP83155.2	LPYGD	PGHVYYSRSDYYDDSL	VKVKLBKIVVYRNWFEK	2281
36	AFP83160.2	HPYGD	PGHVYYSRSDYFDDSL	VKVKXNKZVVYRNWFEK	2372
37	AFP83159.2	HPYGD	PGHVYYSHSDYYDDSL	IKVKXNKZVVYRNWFEK	2463
38	AFP83166.2	WEHGY	XGYVYYSGSDYYDPSC	VKVKMNKEVVYGNWFEP	2555
39	AFP83173.2	WBIMY	LGFVYYSGSDYYBPSL	VKVKMNKZVVYGNWFZA	2920
40	AFP83163.2	WPIFY	LGYVYYSGSBYYBPSL	VKVKMNKZIVYGNWFZA	3011
41	AFP83170.2	YZIMY	LGYVYYSASDYYBPSL	VKVKMNKEIVYGNWFEA	3102
42	AFP83174.2	YPIMY	SGYVYYSGSDYYBPSL	VKVKMNKEVVYGBWFEA	3193
43	AFP83184.2	ZSZYY	TDYVYYSGIDYYTPSL	VKVKMNKEVVYDYWFEP	3285
44	AFP83185.2	BBGYY	TDYVYYSGIDYYYPSL	VKVKMTKEVVYDYWFZP	3345
45	AFP83181.2	EBAYY	TDYVYYSGVDYYEPSL	VKVKMNKEVVYDYWFEP	3406
46	AFP83208.2	WDIPY	LGYVYYSASDYYBPSL	VKVKMNKZVVYGNWFZA	3467
47	AFP83178.2	FKIMY	LGYVYYSGSDYYDPSL	VKWKMBKZVYYGNWFZA	3528
48	AFP83177.2	YEIMW	LGFVYYSGSDYYBPSL	VKVKMNKZAVYGNWFZA	3589
49	AJK04689.1	DDGYY	TDYVYYSGIDYYEPSL	VKMKMAKZTVYDYWFZP	3650
50	AJK04818.1	EBFYY	TDYVYYSGVDYYCPSI	VKVKMBKEVVYDYWLEP	3832
51	AJK04119.1	ZDPYY	TDYVYYSGIDYYBPSL	VKVKMRKEVVYDHWFEP	4015
52	AFP83190.2	DDDYF	TDYVYYSGIDYYWPSL	VKVKMTKZVVYDYWFZP	4075
53	AJK05467.1	DDRYY	TDYIYYSGIDYYKPSL	VKVKMSKZVVYDYWFZP	4136
54	AJK05084.1	DDGYY	TDYIFYSGITYYVPXL	VKVKMSKEVIYDHWFZP	4197
55	AJK04964.1	DDGYY	CDYXFYSGIDYYSPSC	VKVKMSKEVVYDYWFEP	4258
56	AJK05278.1	EDFYY	TDYVWYTGIDYYXPXL	VKVKMVKXVVXDYWFZP	4319

7.B.4 Evaluation metrics

Notations.

For a rooted tree T, we let $\mathbf{L}_T$ be the set of leaves and $\mathbf{I}_T$ be the set of internal nodes. For each node v of T, let $\mathscr{C}(v)$ be the set of its children. We define $\phi(v)$ as the set of node labels of labeled nodes below v. Also, for any *set* of nodes V, we define $\phi(V) = \{\phi(v) : \phi(v) \neq \emptyset, v \in V\}$ and $\phi(T) = \phi(\mathbf{I}_T \cup \mathbf{L}_T)$. For a set of nodes V and a set of labels Φ, $\phi(V) \restriction \Phi = \{\Phi' \cap \Phi : \Phi' \cap \Phi \neq \emptyset, \Phi' \in \phi(V)\}$. For labeled nodes Ψ_i and Ψ_j, let $U_T(i,j)$ be the number of edges between the node Ψ_i in T and the MRCA of Ψ_i and Ψ_j in T.

Characterizing a clonal tree

We define a set of metrics for characterizing properties of simulated trees in terms of their topology, branch length, and distribution of labeled nodes (Table S7.4). Some of these metrics are motivated by similar ones on phylogenetic trees, but are adjusted to allow sampled internal nodes and multifurcations. For example, to measure tree balance, we extend the definition of the number of cherries but allow modifications (our definition reduces to the traditional definition when the tree is binary). Other metrics (e.g., percent internal samples) are only meaningful for clonal trees and are meant to quantify the deviation of a clonal tree from phylogenetic trees.

Table S7.4. Properties of a clonal tree T.

Property	Definition
Internal sample (%)	The percentage of labeled nodes in set $\mathbf{I}_T$.
Bifurcation index	Defined as $\frac{\lvert\mathbf{I}_T\rvert}{\lvert\mathbf{L}_T\rvert-1}$ equals 1 for bifurcating trees and ≈ 0 for the star tree.
Sample depth	The average depth of labeled nodes in T.
Balance (cherry)	Half the sum over all leaves of the fraction of their siblings that are leaves. $\sum_{v\in\mathbf{I}_T} \binom{\lvert\mathscr{C}(v)\cap\mathbf{L}_T\rvert}{2} \big/ (\lvert\mathscr{C}(v)\rvert - 1)$ where $0/0 \doteq 1/2$
Single mutation branches (%)	The percentage of branches with length one.
Accumulated mutations (avg)	The average depth (path length to the root) of all labeled nodes of tree T.
Accumulated mutations (sum)	The summation of branch lengths of all branches of tree T.
Mutations per branch	The average branch length of tree T.

The last four metrics require branch length (in mutation unit) on the tree.

Comparing trees

Many metrics exist for comparing phylogenetic trees. However, in the presence of polytomies and sampled ancestral nodes, the classic metrics need to be amended. Here, we generalize several existing metrics and introduce new ones. All metrics are defined over a simulated tree R and a reconstructed tree E, both induced down to include all labeled nodes (i.e., removing unlabeled nodes if less than two of their children have any labeled descendants). See Table S7.5 for precise definitions of metrics.

Table S7.5. Metrics for comparing the reference simulated tree R to estimated tree E.

Metric	AB	Definition
False discovery rate	FDR	$\lvert\phi(E)\setminus\phi(R)\rvert / \lvert\phi(E)\rvert$
FDR no singletons	FDR*	$\lvert\phi(\mathbf{I}_E)\setminus\phi(\mathbf{I}_R)\rvert / \lvert\phi(\mathbf{I}_E)\rvert$
False negative rate	FNR	$\lvert\phi(R)\setminus\phi(E)\rvert / \lvert\phi(R)\rvert$
FNR no singletons	FNR*	$\lvert\phi(\mathbf{I}_R)\setminus\phi(\mathbf{I}_E)\rvert / \lvert\phi(\mathbf{I}_R)\rvert$
RF cluster distance	RF	$\lvert\phi(R)\cup\phi(E)\rvert - \lvert\phi(R)\cap\phi(E)\rvert$
RF cluster distance no singletons	RF*	$\lvert\phi(\mathbf{I}_R)\cup\phi(\mathbf{I}_E)\rvert - \lvert\phi(\mathbf{I}_R)\cap\phi(\mathbf{I}_E)\rvert$
Triplet discordance	TD	$\lvert\{\Phi : \phi(R)\restriction\Phi \neq \phi(E)\restriction\Phi, \Phi\subset\{\Psi_1,\ldots,\Psi_\zeta\}, \lvert\Phi\rvert=3\}\rvert$
Triplet edit distance	TED	$\sum_{\Phi\subset\{\Psi_1,\ldots,\Psi_\zeta\},\lvert\Phi\rvert=3} \lvert(\phi(R)\restriction\Phi)\cup(\phi(E)\restriction\Phi)\rvert - \lvert(\phi(R)\restriction\Phi)\cap(\phi(E)\restriction\Phi)\rvert$
MRCA discordance	MD	$\sum_{i,j\in[\zeta]} \lvert U_R(i,j) - U_E(i,j)\rvert$
Patristic distance	PD	$1/2\sum_{i,j\in[\zeta]} \lvert U_R(i,j) + U_R(j,i) - U_E(i,j) - U_E(j,i)\rvert$

RF-related.

We refer to the set of labeled nodes under some subtree as a cluster. We define False Discovery Rate (FDR) as the percentage of clusters in E that are not in R, False Negative Rate (FNR) as the percentage of clusters in R that are not in E, and Robinson-Foulds cluster distance (RF) as the number of clusters in either but not both trees. Note that unlike traditional Robinson and Foulds (1981) distance, here, internal nodes can also have labels, and we define the metric based on clusters in a rooted tree instead of bipartitions in an unrooted tree. Moreover, the singleton clusters are trivial when all labeled nodes are leaves; however, when there are labeled internal nodes, including or excluding singletons can make a difference. Thus, we also define FPR FNR, and RF distance when excluding singleton clusters.

Triplet-based.

We define *triplet discordance* (TD) as the number of trees induced by triples of *labeled* nodes (leaf or internal) where the topology in the simulated tree and the reconstructed tree differ. We define the *triplet edit distance* (TED) as the summation over all triplets of the labeled nodes of cluster RF distance between the two trees induced to the triplet. Intuitively, it is the sum of the minimum number of branch contractions and resolutions required to covert a triplet in R to a triplet in E, summed over all triplet.

Path discordance.

Patristic discordance for a pair of labeled nodes Ψ_i and Ψ_j is defined as the difference between the number of branches in the path between Ψ_i and Ψ_j on two trees R and E. The patristic discordance (PD) between R and E is the summation of the Patristic discordance over all pairs of labeled nodes (internal or leaf). We define the MRCA discordance for an ordered pair of labeled nodes Ψ_i and Ψ_j as the difference between the number of branches in the path between Ψ_i and its MRCA with Ψ_j when computed from trees R and E. The MRCA discordance (MD) between the two trees is the summation of MRCA discordance over all ordered pairs of labeled nodes.

The FNR and FDR metrics are already normalized. To normalize other metrics, for each experimental condition, we create a control tree by randomly permuting labels of the true tree. We then normalize scores (other than FNR and FDR) of a reconstruction method by dividing it by the average score of replicates of the control method.

Computing FNR, FDR, and RF metrics takes $O(\varsigma)$ time with hashing and randomization (algorithm S7.4). Triplet-based metric can be easily computed in $O(\varsigma^3)$ time with simple preprocessing and iterating over all triplets. Both PD and MD take $O(\varsigma^2)$ time with preprocessing that computes distances to MRCAs.

7.C Supplementary Figures

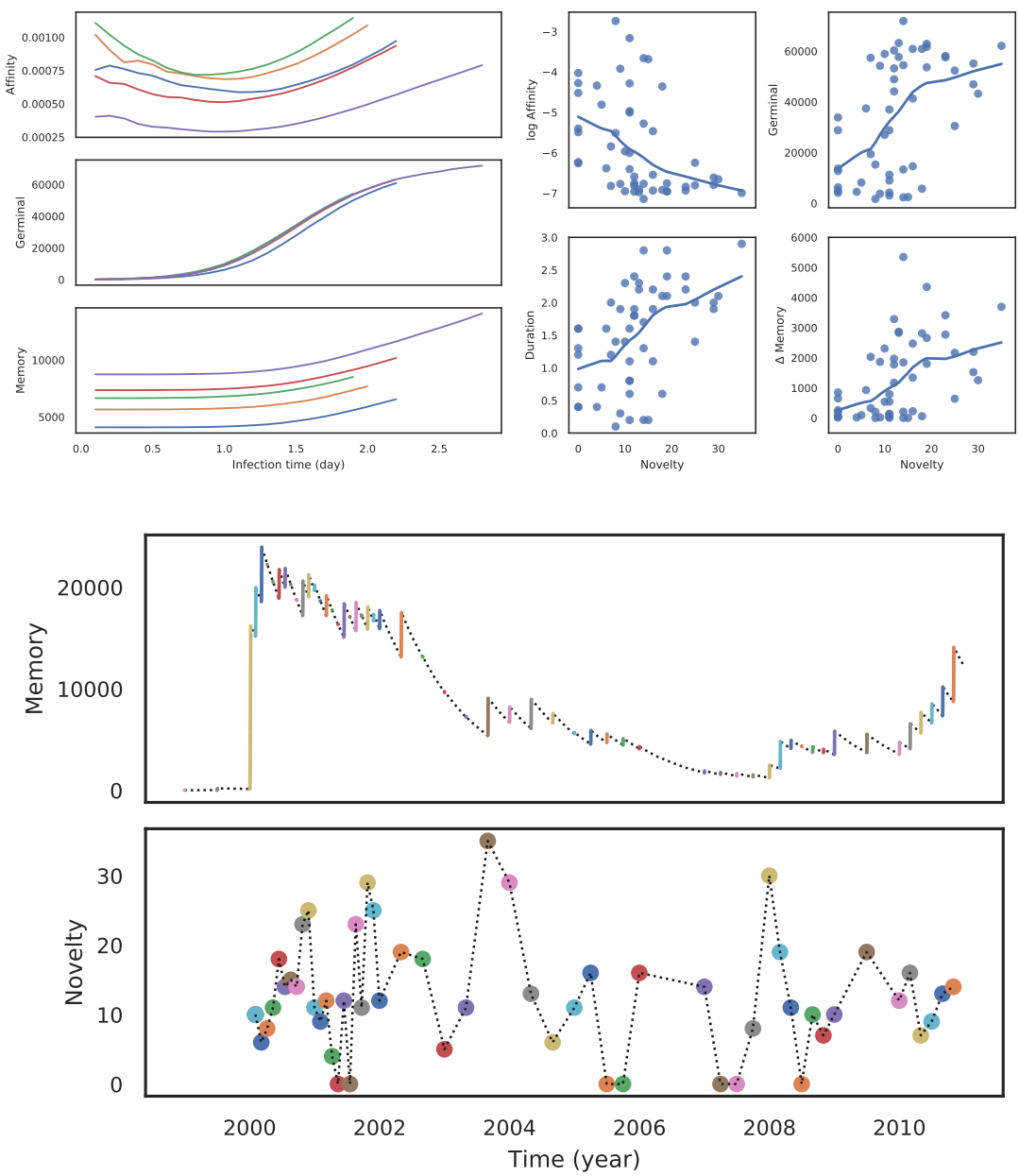

Figure S7.1. a) Log average affinity of activated cells to the current infection target at the end of the infection, the number of activated cells at the end of the infection, and the duration of infection by novelty of the target of one simulation under default conditions, showing the last five rounds as examples. b) Average affinity of activated cells to current infection target, the number of activated cells, and the number of memory cells by time after infection starts for the last five infections of one simulation under default conditions. Lines are fitted using the LOWESS (locally weighted scatter plot smoothing) algorithm. c) Number of memory cells and novelty of infections by time. Dormant stages are indicated by dotted lines.

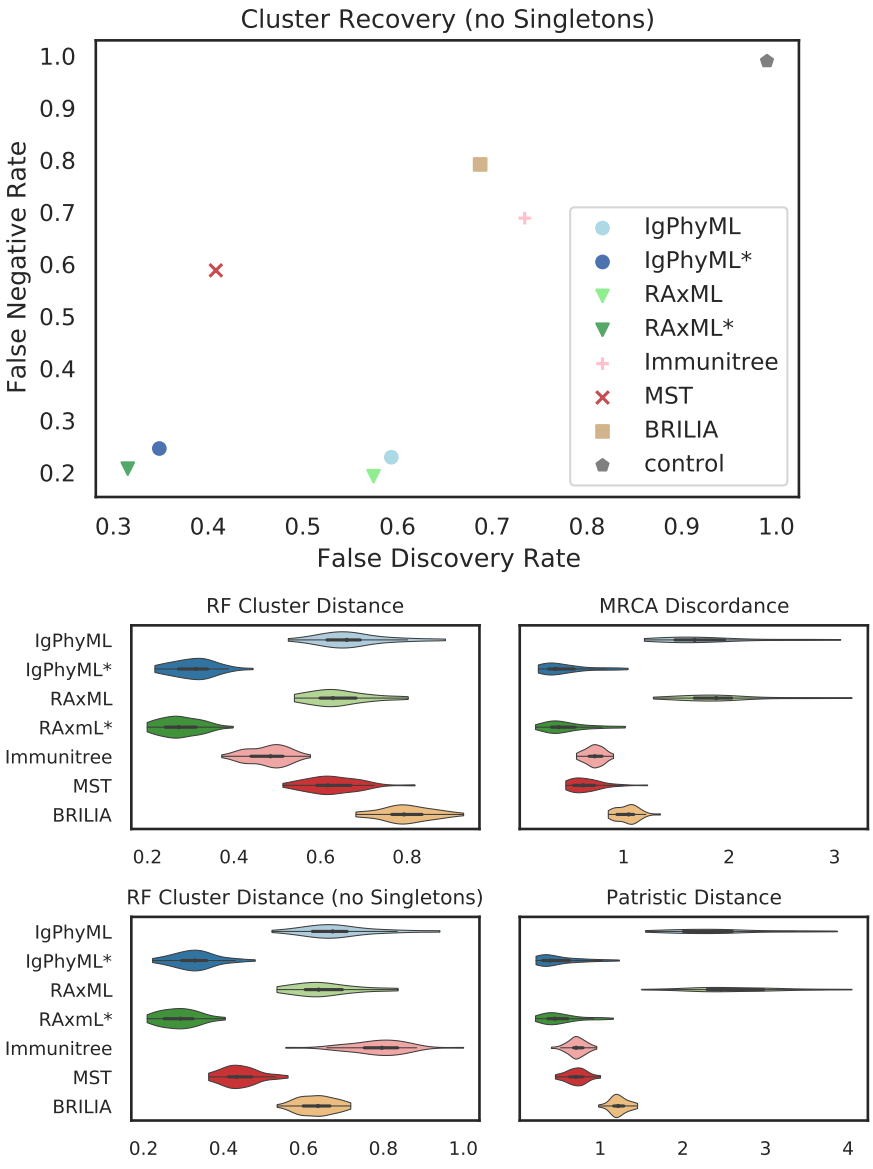

Figure S7.2. Top: FNR* and FPR* rates excluding singletons by reconstruction methods on simulations under default conditions; Bottom: Normalized Robinson-Foulds cluster distance with and without singletons (RF and RF *), MD and PD.

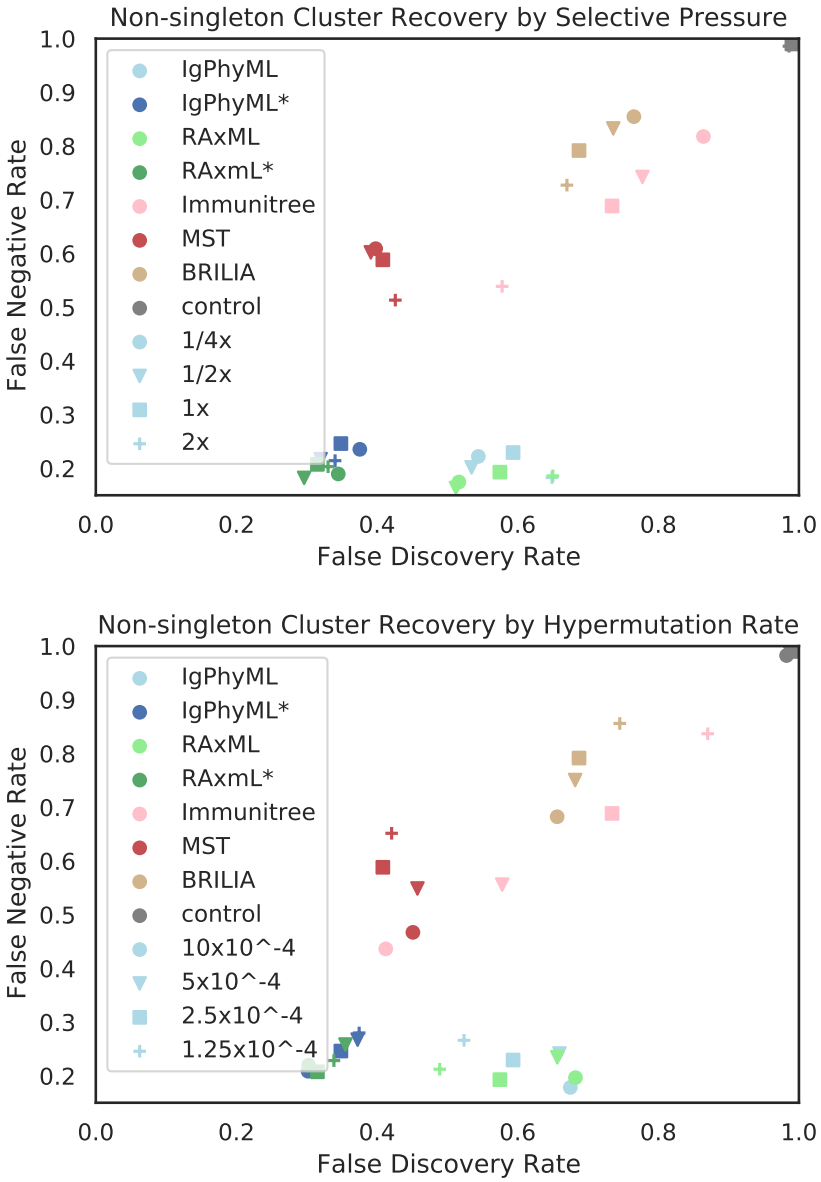

Figure S7.3. Impact of selective pressure A (a) and mutation rate μ (b) on tree inference error by FDR* and FNR*.

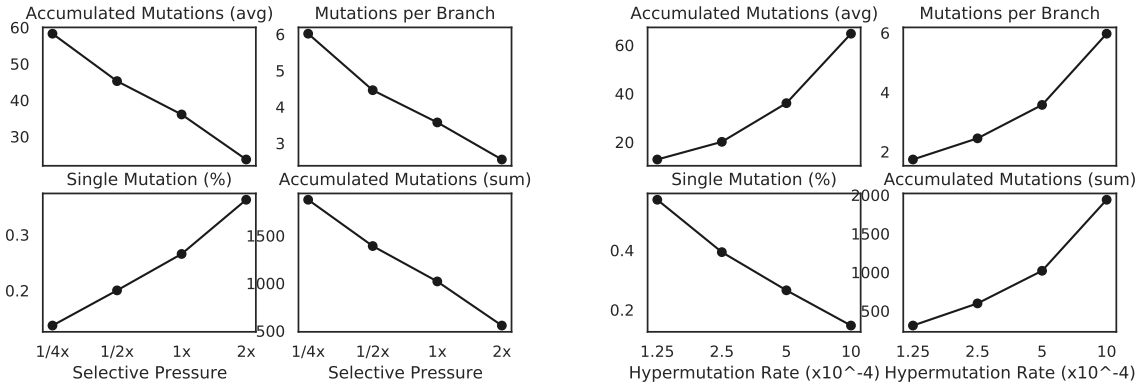

Figure S7.4. Impact of selective pressure A (left) and mutation rate μ (right) on sequence-based branch length properties on true trees. $\mu = 5 \times 10^{-4}$ in (a-d) and $A = 0.1$ in (e-h).

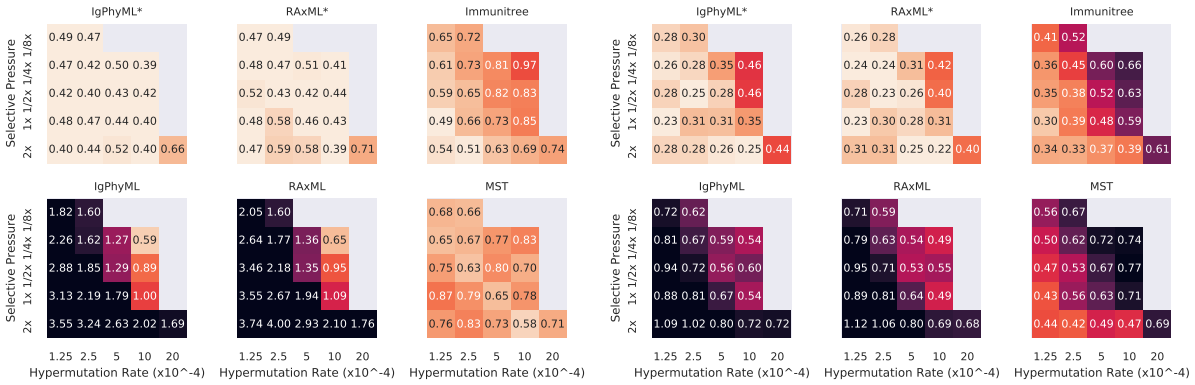

Figure S7.5. For varying levels of selective pressure (A), rate of hypermutation (μ), and reconstruction methods, we show MD error (left), and RF error (right). Under some conditions, reconstructed trees from phylogenetic methods are worse than random permuting labels of true tree because both MD and RF (to a lesser degree) severely penalizes resolution of multifurcated nodes.

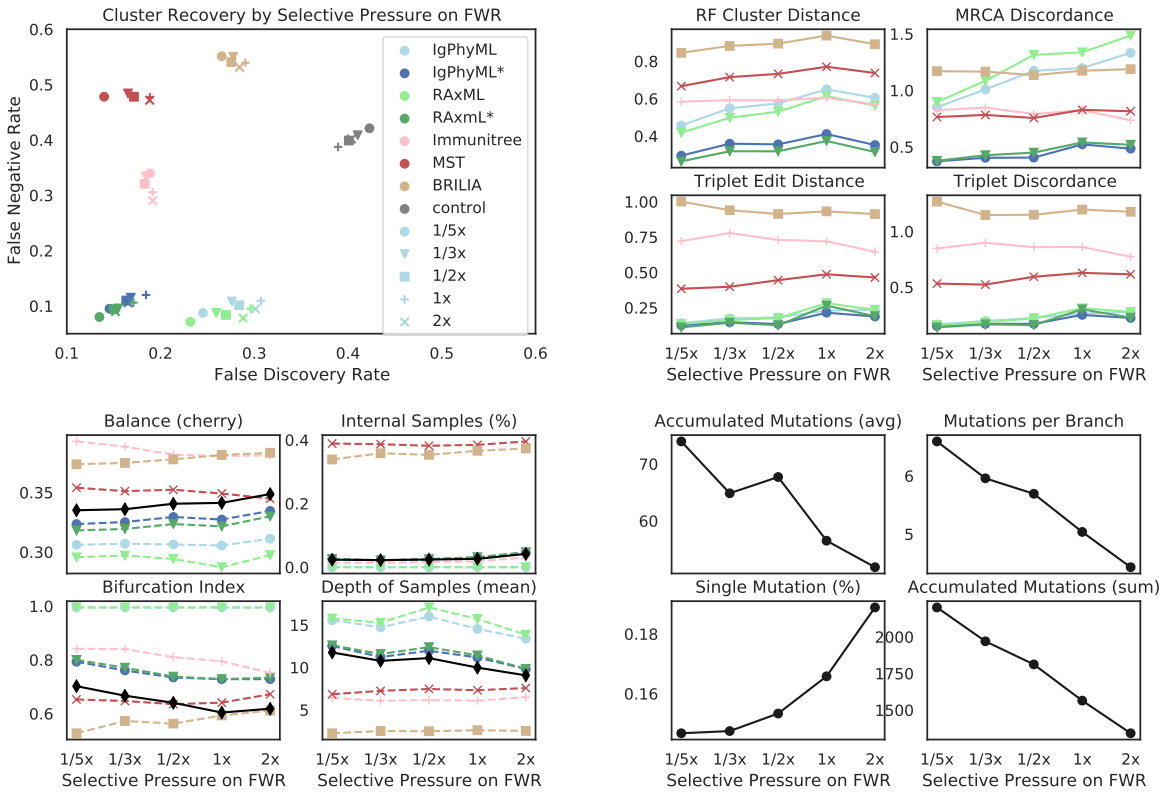

Figure S7.6. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by BLOSUM weight multiplier of framework region (w_f) and reconstruction methods. c) Properties of true (black) and reconstructed trees by BLOSUM weight multiplier of framework region (FR). d) Properties of true trees.

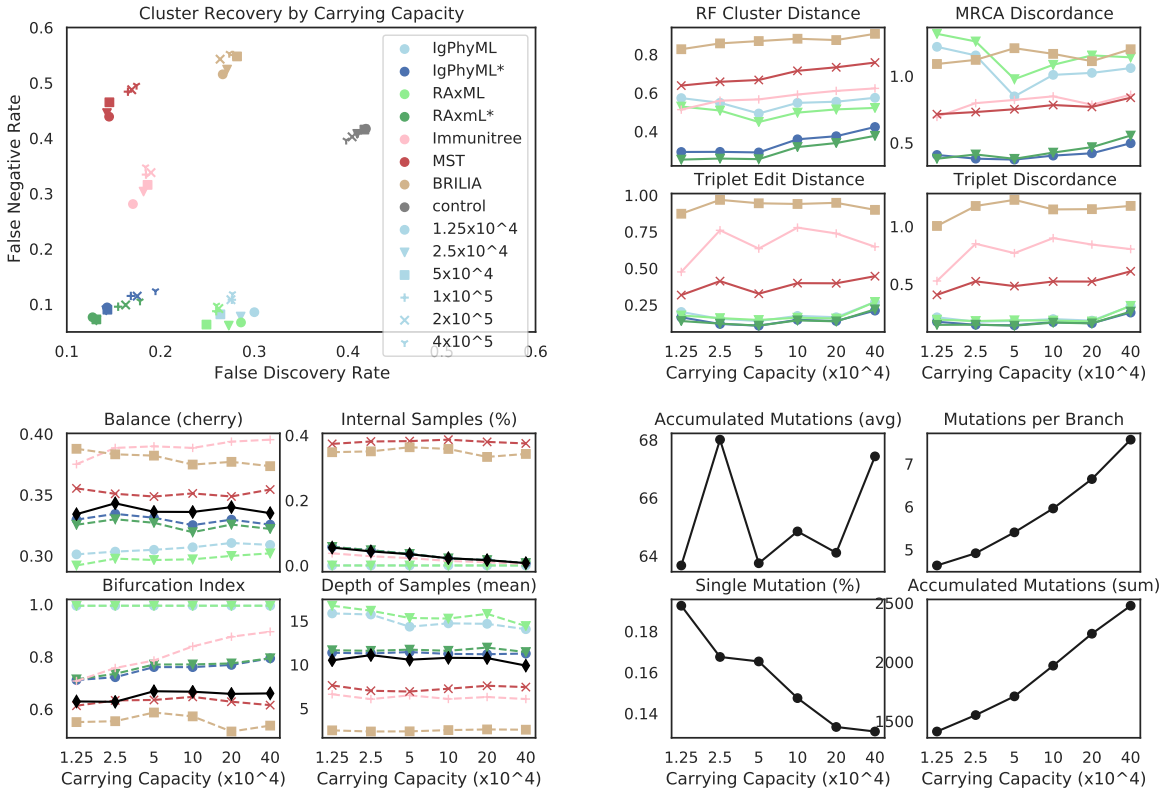

Figure S7.7. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by germinal center capacity (C) and reconstruction methods. c) Properties of true (black) and reconstructed trees by carrying capacity of germinal center of FR. d) Properties of true trees.

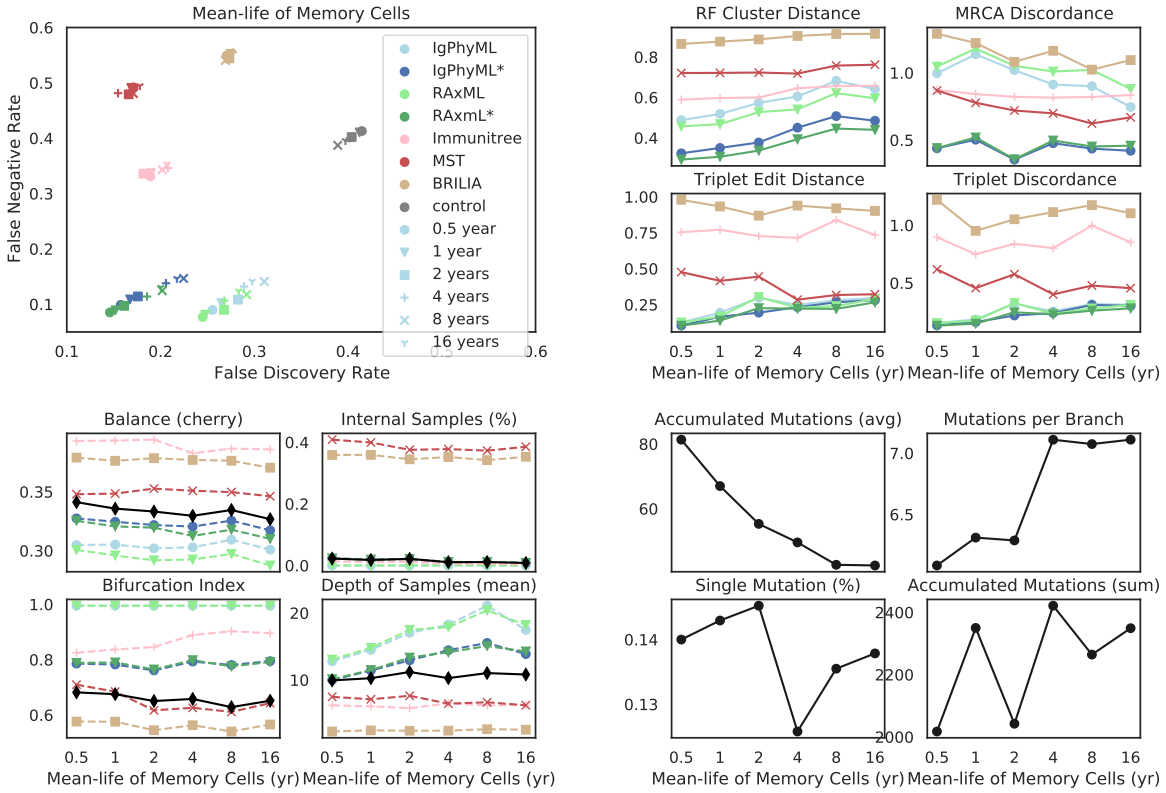

Figure S7.8. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by mean memory cell life-time ($1/\lambda_d'$) and reconstruction methods. c) Properties of true (black) and reconstructed trees by memory cell life (mean). d) Properties of true trees.

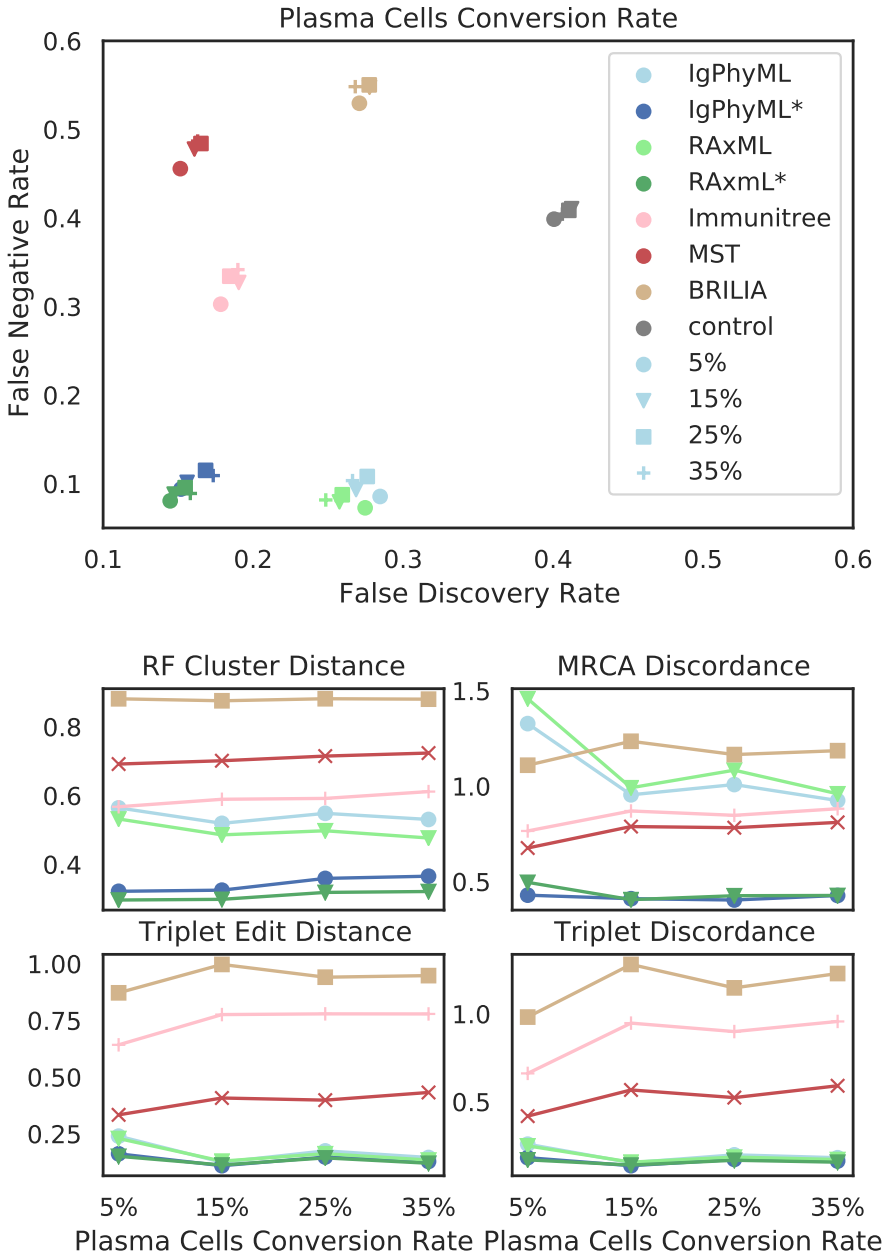

Figure S7.9. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by fraction of activated cells turning into plasma cell per cell division (ρ_p).

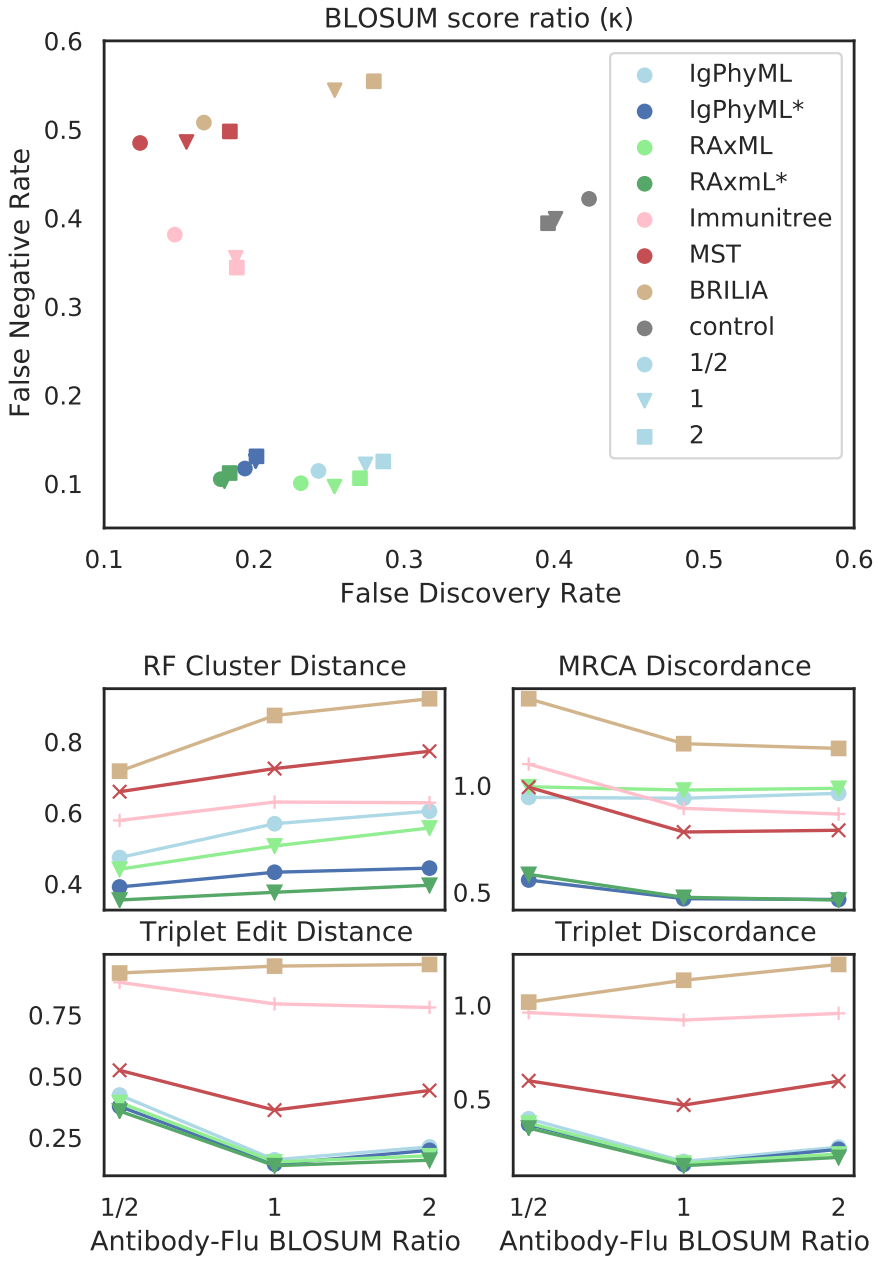

Figure S7.10. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by BLOSUM score ratio of antibody-coding sequences to antigen sequences (κ)

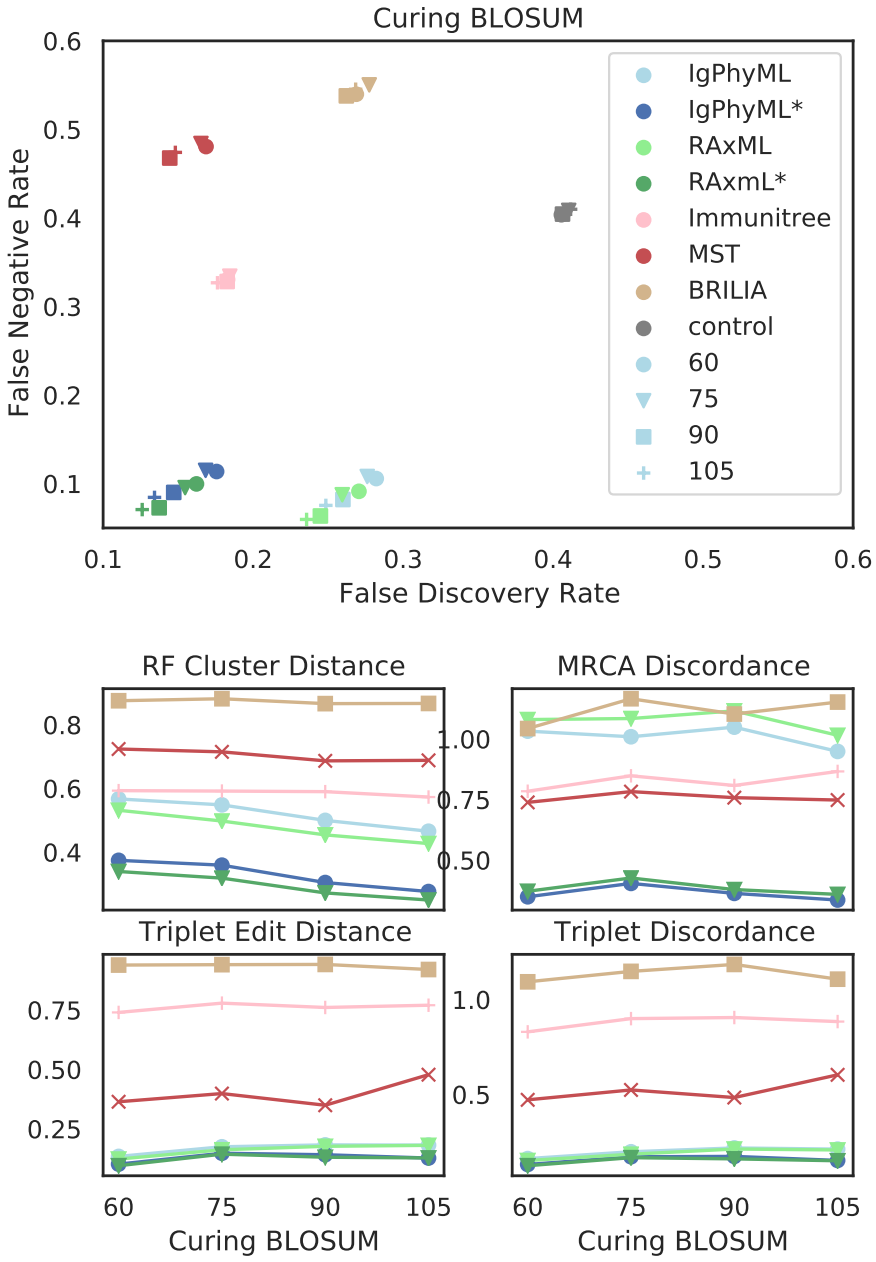

Figure S7.11. a) FNR versus FDR, b) Robinson-Foulds cluster distance (RF), MRCA Discordance (MD), triplet edit distance (TED), and triplet discordance (TD) by BLOSUM score of activated cell antibody-coding sequences that leads to cure (Δ_0').

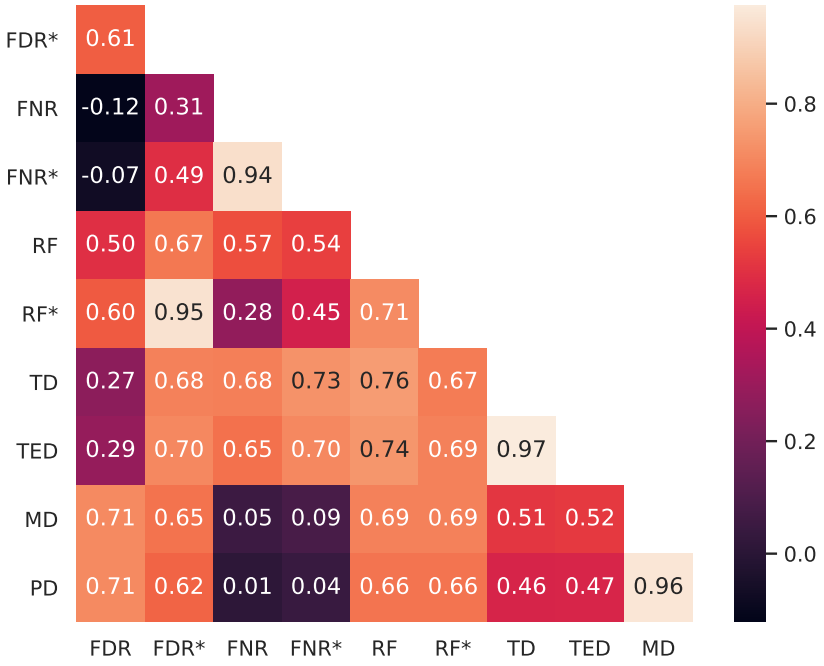

Figure S7.12. Correlations of evaluation metrics. For each replicate of each simulation condition, we compute Spearman's rank correlation coefficient of the reconstruction method for each pair of evaluation metrics. Here, we show the average coefficient over all replicates of all simulation conditions.

7.D Supplementary Algorithms

Recall:

$$\sum_{i,j\in[r]} \left| \kappa \sum_{p\in\mathbf{CDR}} \delta(\zeta_i^{(p)}, \zeta_i^{(p)}) - \delta(\zeta_i^{(p)}, \zeta_j^{(p)}) - \sum_{q=1}^{L_\eta} \left(\delta(\eta_i^{(q)}, \eta_i^{(q)}) - \delta(\eta_i^{(q)}, \eta_j^{(q)}) \right) \right| . \tag{7.8}$$

Algorithm S7.1. Simulating the next event and update time and S accordingly. Before running this procedure, we have computed $\mathbf{S}$ and $\theta_\alpha = \sum_{i \in S} \mathbf{x}_i^\alpha$ for all α from the previous calls to this function (i.e., previous time steps). For each α, we have also built an interval tree T_α on leafset S and each node v storing the summation of $\mathbf{x}_i^\alpha$ for each leaf i under v.

procedure SAMPLETREE(α, v)
  if v is a leaf node **then**
    return v
  else
    $L \leftarrow$ the sum of $\mathbf{x}_i^\alpha$ for each leaf i under left child of v
    $R \leftarrow$ the sum of $\mathbf{x}_i^\alpha$ for each leaf i under right child of v
    $O \leftarrow$ the outcome of a flip of a biased coin with probability of being head $\frac{L}{L+R}$
    if $O =$ Head **then**
      return SAMPLETREE(α, the left child of v)
    else
      return SAMPLETREE(α, the right child of v)
procedure SIMULATINGONEEVENT
  time $\leftarrow$ time $+$ a random sample from exponential distribution where $\lambda = \frac{\sum_{\alpha,\beta \in \Gamma}(P_{\alpha,\beta}\mathbf{S}^\beta \theta_\alpha)}{\sum_{\beta \in \Gamma} Q_\beta \mathbf{S}^\beta}$
  $(\alpha, \beta) \leftarrow$ a random sample from distribution $Pr(\alpha, \beta) = \frac{P_{\alpha,\beta}\mathbf{S}^\beta \theta_\alpha}{\sum_{\bar{\alpha},\bar{\beta} \in \Gamma}(P_{\bar{\alpha},\bar{\beta}}\mathbf{S}^{\bar{\beta}} \theta_{\bar{\alpha}})}$
  $i \leftarrow$ SAMPLETREE(α, the root of $T\alpha$)
  $E \leftarrow$ a sample from $Pr(E = \text{Birth}) = \frac{\mathscr{B}_{\alpha,\beta}}{P_{\alpha,\beta}}, Pr(E = \text{Death}) = \frac{\mathscr{D}_{\alpha,\beta}}{P_{\alpha,\beta}}, Pr(E = \text{Transformation}) = \frac{\mathscr{T}_{\alpha,\beta}}{P_{\alpha,\beta}}$
  if $E =$ Birth **then**
    $(j,k) \leftarrow$ a sample from distribution of outcomes of birth event of i
    $\mathbf{S} \leftarrow \mathbf{S} + \mathbf{x}_j + \mathbf{x}_k$
    $S \leftarrow S \cup \{j,k\}$
    for $\alpha \in \Gamma$ **do**
      $\theta_\alpha \leftarrow \theta_\alpha + \mathbf{x}_j^\alpha + \mathbf{x}_k^\alpha$
      add leaves j and k to T_α while keeping the tree balanced using Algorithm S7.2
  if $E =$ Transformation **then**
    $j \leftarrow$ a sample from distribution of outcomes of transformation event of i
    $\mathbf{S} \leftarrow \mathbf{S} + \mathbf{x}_j$
    $S \leftarrow S \cup \{j\}$
    for $\alpha \in \Gamma$ **do**
      $\theta_\alpha \leftarrow \theta_\alpha + \mathbf{x}_j^\alpha$
      add leaf j to T_α while keeping the tree balanced using Algorithm S7.2
  $\mathbf{S} \leftarrow \mathbf{S} - \mathbf{x}_i$
  $S \leftarrow S - \{i\}$
  for $\alpha \in \Gamma$ **do**
    $\theta_\alpha \leftarrow \theta_\alpha - \mathbf{x}_i^\alpha$
    remove leaf i from T_α, making the leaf ready for future additions using Algorithm S7.2

Algorithm S7.2. Exact algorithm for inserting or removing a leaf from tree T_α keeping it balanced. T_α is represented by a full binary tree where each leaf is labeled with either one entity or $\emptyset$ and each node v has weight w_v equal to the sum of $\mathbf{x}_i^\alpha$ for all leaves under v with label (i) not being $\emptyset$. Assuming a stack S_α keeps all leaves with label $\emptyset$.

procedure ADDWEIGHT(T_α, i, v, u)
  $w_u \leftarrow w_u + \mathbf{x}_i^\alpha$
  if v is under left subtree of u **then**
    ADDWEIGHT(T_α, i, v, the left child of u)
  if v is under right subtree of u **then**
    ADDWEIGHT(T_α, i, v, the right child of u)
procedure INSERTLEAF(T_α, i)
  if S_α is empty **then**
    $H \leftarrow$ the height of T_α
    $T' \leftarrow T_\alpha$
    $T_\alpha \leftarrow$ a full binary tree of height $H+1$, all leaves labeled $\emptyset$, and all nodes having weight 0
    replace the left subtree of the root of T_α with T'
    the weight the root of $T_\alpha \leftarrow$ the weight of the left child of the root of T_α
    push all leaves under right child of the root of T_α into S_α
  $v \leftarrow$ pop one element from S_α
  label of $v \leftarrow i$
  ADDWEIGHT(T_α, i, v, the root of T_α)
procedure REDUCEWEIGHT(T_α, i, v, u)
  $w_u \leftarrow w_u + \mathbf{x}_i^\alpha$
  if v is under left subtree of u **then**
    REDUCEWEIGHT(T_α, i, v, the left child of u)
  if v is under right subtree of u **then**
    REDUCEWEIGHT(T_α, i, v, the right child of u)
procedure REMOVELEAF(T_α, i)
  $v \leftarrow$ leaf of T_α with label i
  label of $v \leftarrow \emptyset$
  push v onto S_α
  REDUCEWEIGHT(T_α, i, v, the root of T_α)

Algorithm S7.3. Heuristics for choosing target sequences to minimize the objective function (7.8).

for $i \leftarrow 2$ to r **do**
  for $q \in$ **CDR do**
    $C_i^{(q)} \leftarrow 0$
    $\zeta_i^{(q)} \leftarrow \zeta_1^{(q)}$
for $p \leftarrow 1$ to L_η **do**
  $t \leftarrow Poisson(\kappa)$
  for $u \leftarrow 1$ to t **do**
    $q \leftarrow$ a uniform random element of **CDR** where $\eta_1^{(p)} = \zeta_1^{(q)}$
    for $i \leftarrow 2$ to r **do**
      if $\eta_i^{(p)} \neq \eta_1^{(p)}$ **then**
        $C_i^{(q)} \leftarrow C_i^{(q)} + 1$
        $\zeta_i^{(q)} \leftarrow \eta_i^{(p)}$ with probability $1/C_i^{(q)}$
$b \leftarrow$ True
while b = True **do**
  $b \leftarrow$ False
  for $i \leftarrow 2$ to r **do**
    for $q \in$ **CDR do**
      for $s \in$ nucleotide alphabet **do**
        if replacing $\zeta_i^{(q)}$ with s reduces the objective function **then**
          $b \leftarrow$ True
          $\zeta_i^{(q)} \leftarrow s$

Algorithm S7.4. The compute set algorithm

Let each label be uniformly randomly assigned to an element in a finite Abelian group with large enough order (e.g., 64-bit integers). To compute FNR, FDR, and RF, we just need to compute $|\phi(R)| = |S_R|$, $|\phi(E)| = |S_E|$, and $|\phi(R) \cap \phi(E)| = |S_R \cap S_E|$, where set S_T for tree T can be computed by calling COMPUTESET(T, the root of T).

procedure COMPUTESET(T, v)
  $w \leftarrow$ the element assigned to the label of v, if v has label; otherwise, $w \leftarrow 0$.
  for u in the children of v **do**
    $w \leftarrow w +$ COMPUTESET(T, u)
  add element w to set S_T
  return w