COMPUTATIONAL FRAMEWORKS FOR INDEL-AWARE EVOLUTIONARY ANALYSIS USING LARGE-SCALE GENOMIC SEQUENCE DATA

By

Wei Wang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science – Doctor of Philosophy

2021

ABSTRACT

COMPUTATIONAL FRAMEWORKS FOR INDEL-AWARE EVOLUTIONARY ANALYSIS USING LARGE-SCALE GENOMIC SEQUENCE DATA

By

Wei Wang

With the development of sequencing techniques, genetic sequencing data has been extensively used in evolutionary studies. The phylogenetic reconstruction problem, which is the reconstruction of evolutionary history from biomolecular sequences, is a fundamental problem. The evolutionary relationship between organisms is often represented by phylogeny, which is a tree or network representation. The most widely-used approach for reconstructing phylogenies from sequencing data involves two phases: multiple sequence alignment and phylogenetic reconstruction from the aligned sequences. As the amount of biomolecular sequence data increases, it has become a major challenge to develop efficient and accurate computational methods for phylogenetic analyses of large-scale sequencing data. Due to the complexity of the phylogenetic reconstruction problem in modern phylogenetic studies, the traditional sequence-based phylogenetic analysis methods involve many over-simplified assumptions. In this thesis, we describe our contribution in relaxing some of these over-simplified assumptions in the phylogenetic analysis.

Insertion and deletion events, referred to as indels, carry much phylogenetic information but are often ignored in the reconstruction process of phylogenies. We take into account the indel uncertainties in multiple phylogenetic analyses by applying resampling and re-estimation. Another over-simplified assumption that we contributed to is adopted by many commonly used non-parametric algorithms for the resampling of biomolecular sequences, all sites in an MSA are evolved independently and identically distributed (i.i.d). Many evolution events, such as recombination and hybridization, may produce intra-sequence and functional dependence in biomolecular sequences that violate this assumption. We introduce SERES, a resampling algorithm for biomolecular sequences that can produce resampled replicates that preserve the intra-sequence dependence. We describe the application of the SERES resampling and re-estimation approach to two classical

problems: the multiple sequence alignment support estimation and recombination-aware local genealogical inference. We show that these two statistical inference problems greatly benefit from the indel-aware resampling and re-estimation approach and the reservation of intra-sequence dependence.

A major drawback of SERES is that it requires parameters to ensure the synchronization of random walks on unaligned sequences. We introduce RAWR, a non-parametric resampling method designed for phylogenetic tree support estimation that does not require extra parameters. We show that the RAWR-based resampling and re-estimation method produces comparable or typically better performance than the traditional bootstrap approach on the phylogenetic tree support estimation problem.

We further relax the commonly used assumption of phylogeny. Evolutionary history is usually considered as a tree structure. Evolutionary events that cause reticulated gene flow are ignored. Previous studies show that alignment uncertainty greatly impacts downstream tree inference and learning. However, there is little discussion about the impact of MSA uncertainties on the phylogenetic network reconstruction. We show evidence that the errors introduced in MSA estimation decrease the accuracy of the inferred phylogenetic network, and an indel-aware reconstruction method is needed for phylogenetic network analysis.

In this dissertation, we introduce our contribution to phylogenetic estimation using biomolecular sequence data involving complex evolutionary histories, such as sequence insertion and deletion processes and non-tree-like evolution.

In the loving memory of my parents, Huaijie Wang and Yuehua Hu.

ACKNOWLEDGEMENTS

It has been a long journey since the start of my Ph.D. study in 2016. I would not have made it to the endpoint without the help and support of many people. First and foremost, I would like to thank my advisor, Professor Kevin Liu, for all his guidance and support. He is always very patient and encouraging. As a research advisor and a career mentor, he opened many new doors for me, which I could not imagine when I first started my Ph.D. study.

I would like to thank my committee members: Professor Arjun Krishnan, Professor Eric Goodman, and Professor Mohammad Ghassemi, for their generous service, valuable comments, and thoughtful questions. It has been a great honor and privilege for me to have them on my doctoral committee.

I would also like to thank Professor Jizhong Lou, who supervised my master's theses at the Institute of Biophysics (IBP), Chinese Academy of Sciences. Professor Lou has excellent advice and unsurpassed knowledge. I am grateful for his support and inspiration.

It's a great pleasure to work with my excellent colleagues, including Hussein, Zhiwei, Jack, Qiqige, Julia, Ahmad, and many others. I must acknowledge the generous help from Hussein in the first year of my Ph.D. Hussein helped me get familiar with different tools and software. He also generously shared his experience on the Ph.D. study and suggested that do not burn out at the beginning of this marathon.

I am grateful to the BEACON, NSF Science and Technology Center for the study of evolution in action for their generous support during my Ph.D. study. I would like to thank the professors and staff at the Computer Science and Engineering department and the Office of International Students and Scholars, especially Professor Eric Torng, Professor Katy Colbry, Professor Sandeep Kulkarni, Erin Dunlop, Brenda Hodge, and Bryce Carlton.

I want to thank my friends from Michigan State University. I thank Xi Yin, a best friend of mine and the best roommate ever, for her support, encouragement, and patience during the first two years of my Ph.D. life. We shared many beautiful memories in East Lansing. I thank Nan Du, a

dear friend of mine, and surprisingly turned out to be a distant cousin of my husband, found by an ancestry test service. He is a living Wikipedia who knows everything and generously offers free service to everyone. I thank Qi Wang for her optimistic attitude that deeply affected me. I thank my friends who lived next door to me, Mengying Sun, Deliang Yang, Boyang Liu, and Yunshi Liang, for bringing me endless joy and comfort during this pandemic. I thank my lifelong friends, Xi (Lucy) Lu, Yue Kang, Mengnan Bai, and Xiaolan Li.

I want to express my utmost gratitude to my husband, Teng Cao, for his unconditional love. After losing my parents, I couldn't find a connection with this world until I met him. He saved me from my severe depression and insomnia. He gave me a home where I could go back to and a future to look forward to. I am grateful to my cat, Popcorn. She always gazed at me lovingly. Seeing her every day is a source of mental happiness that relieves me from stress and depression.

Finally, I would like to thank my parents from the very bottom of my heart. My father was my best friend who understood me the most. He had a good sense of humor and brought me a lot of joy and happiness when I was young. My personality was largely inherited from him. My mother was the most beautiful and strong woman in the world. She gave me endless love and encouragement. Her optimistic attitude influenced me deeply. Those good memories never fade away. Their love and companionship made me what I am today.

TABLE OF CONTENTS

LIST OF TABLES x

LIST OF FIGURES xvi

LIST OF ALGORITHMS xxii

CHAPTER 1 INTRODUCTION 1

CHAPTER 2 BACKGROUND 6
2.1 Multiple Sequence Alignment 6
2.1.1 Sequence evolution 7
2.1.2 Multiple Sequence Alignment Estimation 10
2.2 Phylogenies 12
2.2.1 Phylogenetic Trees 13
2.2.2 Phylogenetic Network 15
2.3 Gene Trees and Species Trees 16
2.3.1 Coalescence 18
2.3.2 Multi-Species Coalescent (MSC) Model 19
2.4 Phylogenetic Reconstruction 22
2.4.1 Phylogenetic Tree Reconstruction of Single Gene 22
2.4.2 Branch support 25
2.4.3 Phylogenetic Tree Reconstruction of Multiple Genes 25
2.4.4 Phylogenetic Network Reconstruction 28
2.4.5 Evaluation Metrics 30
2.4.5.1 Comparison of Phylogenetic Trees 30
2.4.5.2 Comparison of Phylogenetic Networks 31

CHAPTER 3 SERES: THE SEQUENTIAL RESAMPLING AND ITS APPLICATION ON MULTIPLE SEQUENCE ALIGNMENT SUPPORT ESTIMATION . . 33
3.1 Introduction 33
3.2 Methods 34
3.2.1 SERES walks on aligned sequences 35
3.2.2 SERES walks on unaligned sequences 36
3.2.3 Performance study 40
3.2.4 Simulated Data 41
3.2.5 Empirical data 43
3.2.6 Performance Measure 43
3.3 Result 44
3.3.1 Simulation study 44
3.3.2 Empirical study 51
3.4 Discussion 54
3.5 Conclusions 56

CHAPTER 4 APPLICATION OF SERES RESAMPLING APPROACH TO ALIGNED SEQUENCES: PHYLOGENETIC HMM INFERENCE AND LEARNING . 57
4.1 Introduction . . . 57
4.2 Methods . . . 59
4.2.1 Standalone recHMM analysis . . . 59
4.2.2 The SERES+recHMM pipeline . . . 59
4.2.3 Simulated datasets . . . 60
4.2.4 Empirical datasets . . . 61
4.3 Results . . . 61
4.3.1 Simulation study . . . 61
4.3.2 Empirical study . . . 68
4.4 Discussion . . . 71
4.5 Conclusions . . . 74

CHAPTER 5 PHYLOGENETIC SUPPORT ESTIMATION WITH THE RANDOM WALK RESAMPLING APPROACH . . . 75
5.1 Introduction . . . 75
5.2 Methods . . . 76
5.2.1 RAWR-based Phylogenetic Support Estimation . . . 76
5.2.2 Bootstrap Phylogenetic Support Estimation. . . . 78
5.2.3 Additional Performance Study . . . 79
5.2.4 Simulated datasets . . . 83
5.2.5 Empirical datasets . . . 83
5.2.6 Performance measurement. . . . 84
5.3 Results . . . 86
5.3.1 Simulation Study . . . 86
5.3.1.1 Performance comparison of RAWR versus bootstrap. . . . 86
5.3.1.2 RAWR support estimation using reduced resampling replication. . 89
5.3.1.3 Results of Additional Performance Study . . . 89
5.3.2 Empirical Study . . . 96
5.3.2.1 Performance comparison of RAWR versus bootstrap. . . . 96
5.3.2.2 Results of Additional Performance Study . . . 96
5.4 Discussion . . . 98
5.5 Conclusion . . . 101

CHAPTER 6 AN APPLICATION OF RANDOM WALK RESAMPLING TO PHYLOGENOMIC ANALYSIS OF DARWIN'S FINCHES . . . 102
6.1 Introduction . . . 102
6.2 Methods . . . 103
6.2.1 Dataset . . . 103
6.2.2 Process of the raw sequencing data . . . 104
6.2.3 Concatenated MLE phylogenetic tree inference . . . 105
6.2.4 Phylogenetic support estimation using bootstrap resampling. . . . 106
6.2.5 Phylogenetic support estimation using RAWR resampling. . . . 106
6.3 Results and Discussion . . . 107

6.4 Conclusion 108

CHAPTER 7 IMPACT OF MULTIPLE SEQUENCE ALIGNMENT ERROR ON THE SUMMARY-BASED PHYLOGENETIC NETWORK RECONSTRUCTION 113
7.1 Introduction 113
7.2 Methods 114
7.2.1 Simulated Dataset 114
7.2.2 Simulation Experiments 115
7.2.3 Empirical Datasets and Experiments 118
7.3 Results 119
7.3.1 Simulation Study 119
7.3.1.1 D-statistics for gene flow detection 119
7.3.1.2 Phylogenetic network inference 120
7.3.2 Empirical Study 123
7.4 Discussion 124
7.5 Conclusion 127

CHAPTER 8 CONCLUSIONS AND FUTURE WORK 129

BIBLIOGRAPHY 132

LIST OF TABLES

Table 3.1: Simulated datasets: parameter values and summary statistics. The simulation model condition parameters consist of the number of taxa, model tree height, and insertion/deletion probability. Each model condition corresponds to a distinct set of model parameter values. The following table columns list average summary statistics for each model condition (n = 20). "NHD" is the average normalized Hamming distance of a pair of aligned sequences in the true alignment. "Gappiness" is the percentage of true alignment cells which consists of indels. "True align length" is the length of the true alignment. "Est align length" is the length of the MAFFT-estimated alignment [64] which was provided as input to the support estimation methods. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the MAFFT-estimated alignment and vice versa, respectively. 42

Table 3.2: Medium-gap-length model conditions: estimated alignment statistics. The MSA support estimation problem requires an input MSA. Our study included ClustalW [88] and FSA [12] alignments to explore the impact of input alignment quality on downstream support estimation. The following table columns list average statistics for estimated alignments on each model condition (n = 20). "Est align length" is the estimated alignment length. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the estimated alignment and vice versa, respectively. 42

Table 3.3: Empirical dataset summary statistics. The empirical study made use of reference alignments ("Ref align") from the CRW database [14]. The column description is identical to Table 3.1. 43

Table 3.4: Support estimation method performance on simulated datasets. Results are shown for simulated datasets. The top rows show AUC comparisons of GUIDANCE1 ("GUIDANCE1") vs. SERES combined with parametric techniques from GUIDANCE1 ("SERES+GUIDANCE1"), Results AUC comparisons of GUIDANCE2 ("GUIDANCE2") vs. SERES combined with parametric techniques from GUIDANCE2 ("SERES+GUIDANCE2"); the best AUC is shown in bold. Corrected q-values are reported (n = 20) and all were significant (α = 0.05). 46

Table 3.5: Support estimation method performance on long-gap-length model conditions. The performance of GUIDANCE2 and SERES+GUIDANCE2 is compared across model conditions 10.long.A through 10.long.E (named in order of generally increasing sequence divergence). Aggregate PR-AUC and ROC-AUC are reported across all replicate datasets in a model condition (n = 20), and the best AUC for each model condition is shown in bold. Statistical significance of PR-AUC or ROC-AUC differences was assessed using a one-tailed pairwise t-test or DeLong test [25] test, respectively, and multiple test correction was performed using the method of [8]. Corrected q-values are reported (n = 20) and all were significant (α = 0.05). 47

Table 3.6: SERES+GUIDANCE2 performance using alternative methods for estimating an input MSA. Input MSAs in these experiments were estimated using either ClustalW [88] or FSA [12]. (MAFFT was used to estimate input MSAs throughout the rest of our study.) Results are shown for model conditions 10.A through 10.E (named in order of generally increasing sequence divergence). Otherwise, table layout and description are identical to Table 3.5. 48

Table 3.7: Empirical study results. Results are shown for empirical datasets. For each dataset and pairwise method comparison. Table layout, and table description are otherwise identical to Table 3.4. 53

Table 4.1: Simulated dataset statistics. The number of true gene trees and average normalized Hamming distance ("ANHD") are reported for simulated datasets from the simulation study; average ("Avg") and standard error ("SE") are shown for all experimental replicates from each model condition (n = 30). 61

Table 4.2: On 4-taxon model conditions, the posterior probabilities inferred by SERES+recHMM were better correlated with topological accuracy compared with the standalone recHMM. For each method, we calculated the Pearson correlation between the inferred posterior probability of a gene tree g and the topological distance between g and the true gene tree of a site. Average correlation for a method is calculated across all replicates in a model condition (n = 30). 65

Table 4.3: On 5-taxon model conditions, posterior probabilities inferred by the SERES+recHMM had stronger correlation with topological accuracy compared with the posterior probabilities inferred by the standalone recHMM. Otherwise, table layout and description are identical to Table 4.2. 65

Table 4.4: On 6-taxon model conditions, posterior probabilities inferred using SERES+recHMM were more highly correlated with topological accuracy compared to standalone recHMM. Otherwise, table layout and description are identical to Table 4.2. . . . 66

Table 4.5: The comparison among different reversal probabilities γ on 4-, 5- and 6-taxon model conditions. The methods utilize models with $\phi = 3$ to infer a posterior probability distribution over gene tree topologies. For each method's inference, we calculated the Pearson correlation between the inferred posterior probability for a gene tree g and the topological distance between g and the true evolutionary history of a site (i.e., the true local gene tree). The averages are reported across all n replicates in a model condition ($n = 30$). 69

Table 4.6: The comparison among different number of states ϕ on 5-taxon model conditions. The methods utilize models with $\gamma = 0.005$ to infer a posterior probability distribution over gene tree topologies. For each method's inference, we calculated the Pearson correlation between the inferred posterior probability for a gene tree g and the topological distance between g and the true evolutionary history of a site (i.e., the true local gene tree). The averages are reported across all n replicates in a model condition ($n = 30$). 70

Table 4.7: The runtime and memory usage information for standalone recHMM and SERES+recHMM methods on simulation study model conditions. Model conditions were parameterized by the number of sequences, recombination rate ρ, and mutation rate θ. Both methods utilize models with $\phi = 3$ and $\gamma = 0.005$ to infer a posterior probability distribution over gene tree topologies. Average runtime in hours and peak memory usage in GiB are reported across all replicates in a model condition ($n = 30$). 71

Table 5.1: Summary statistics for ClustalW-estimated alignments and RAxML(ClustalW) trees on 10-taxon model conditions. Table layout and description are otherwise identical to Table 5.3. 80

Table 5.2: Long-gap-length model conditions: parameter values and summary statistics. Our simulation study included additional 10-taxon model conditions that utilized the long gap length distribution from the study of 2012 study of Liu et al. [80]. The model parameters consisted of model tree height and insertion/deletion probability, and each model condition corresponds to a distinct set of model parameter values. The long-gap-length model conditions are named 10.long.A through 10.long.E in order of generally increasing sequence divergence. The following table columns list average summary statistics for each model condition ($n = 20$). "NHD" is the average normalized Hamming distance of a pair of aligned sequences in the true alignment. "Gappiness" is the percentage of true alignment cells which consists of indels. "True align length" is the length of the true alignment. "Est align length" is the length of the MAFFT-estimated alignment [64] which was provided as input to the support estimation methods. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the MAFFT-estimated alignment and vice versa, respectively. The table and caption are reproduced from the original study of the SERES resampling algorithm [145]. 80

Table 5.3: Model condition parameters and summary statistics of the simulation datasets. Model condition parameters consisted of the number of taxa, tree height, and insertion/deletion probability. The model conditions are named from A to E to represent increasing evolutionary divergence. The average summary statistics are reported for the true alignments, and the MAFFT-estimated alignments over n replicate datasets ($n = 20$). "ANHD" is the average normalized Hamming distance of a pair of aligned sequences in an MSA, "Gappiness" is the proportion of an MSA matrix that consists of indels, "length" is the number of MSA columns, and "SP-FN" and "SP-FP" are the proportions of residue pairs that appear in the true alignment but not in the estimated alignment or vice versa, respectively. The average normalized Robinson-Foulds distance ("nRF") between the model tree and the RAxML(MAFFT)-inferred tree is also reported over n replicate datasets ($n = 20$). 84

Table 5.4: Summary statistics of empirical datasets. Summary statistic calculations and descriptions identical to Table 5.3. 85

Table 5.5: PR-AUC performances on the simulation datasets. The PR-AUC are aggregated over n replicate datasets for a model condition ($n = 20$). Statistical significance of PR-AUC differences between RAWR and bootstrap were evaluated using a one-tailed pairwise t-test and a multiple test correction was performed using the method of [8]. Corrected q-values are reported ($n = 20$). . . 87

Table 5.6: RAWR support estimation using alternative estimation/re-estimation methods. We compared RAWR support estimation using two different estimation/re-estimation methods: either MAFFT and RAxML(MAFFT) or ClustalW and RAxML(ClustalW). For each of the two methods, aggregate PR-AUC is shown across all replicate datasets of each model condition ($n = 20$). 90

Table 5.7: Simulation study: RAWR support estimation using different choices for reversal probability γ. Aggregate PR-AUC is reported across all replicate datasets of each 10-taxon model condition ($n = 20$). 91

Table 5.8: PR-AUC comparison of bootstrap and RAWR methods on 10-taxon long-gap-length model conditions. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. Each method's PR-AUC is reported as an aggregate across all replicate datasets for a model condition ($n = 20$). 91

Table 5.9: PR-AUC comparison of aLRT and RAWR methods for phylogenetic support estimation. We used PhyML [49] to run two types of aLRT analyses: (1) support estimation or a free tree topology that was also estimated as part of the analysis, and (2) support estimation for a RAxML-inferred tree topology. The latter methodology for obtaining an annotation tree is identical to the approach used in all other experiments in our study, and its PR-AUC performance is therefore directly comparable to other simulation study experiments. Table layout and description are otherwise identical to Table 5.8. 92

Table 5.10: PR-AUC comparison of TBE with bootstrap resampling, TBE with RAWR resampling, and RAWR. TBE was used to estimate phylogenetic support using two different resampling approaches: either (1) standard bootstrap resampling, which corresponds to the method originally proposed and studied by Lemoine et al. [74], or (2) RAWR resampling. The former is denoted "TBE with bootstrap resampling", and the latter is denoted "TBE with RAWR resampling". For comparison purposes, RAWR resampling and re-estimation was also run as a third method (denoted "RAWR"), and we used the same methodology as elsewhere in our study (i.e., using a standard branch presence/absence calculation to assess phylogenetic support). 94

Table 5.11: PR-AUC comparison of GUIDANCE2 and RAWR phylogenetic support estimation methods. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. We report each method's aggregate PR-AUC across all replicate datasets for a model condition ($n = 20$). . . 95

Table 5.12: PR-AUC performance of RAWR+teleport on 10-taxon model conditions. 95

Table 5.13: PR-AUC performances on the emprircal datasets. PR-AUC comparison of bootstrap and RAWR methods for phylogenetic support estimation. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. 97

Table 5.14: Empirical study: PR-AUC comparison of aLRT and RAWR methods on CRW benchmarking datasets. Table layout and description are otherwise identical to Table 5.9. 97

Table 5.15: PR-AUC comparison of TBE and RAWR methods on CRW benchmarking datasets. Table layout and description are otherwise identical to Table 5.9. . . . 97

Table 5.16: PR-AUC comparison of GUIDANCE2 and RAWR methods for phylogenetic support estimation. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. 98

Table 7.1: Model parameters and summary statistics of the simulated datasets. The 4-taxon model conditions are named 4.A through 4.E in order of increasing evolutionary divergence; the 8-taxon model conditions are named 8.A through 8.E similarly. Additional model condition parameters include the insertion/deletion rate and the model phylogeny height (see Methods section for details). Average normalized Hamming distance ("ANHD") and the percentage of true MSA cells that consist of indels ("Gappiness") are reported as an average for each model condition. 115

Table 7.2: Topological distance pairwise comparison of species networks inferred by MLE on different MSAs of the mosquito dataset. We obtained the reference alignment from the original study of [103]. The estimated alignments were generated by ClustalW, MAFFT, or FSA. We also compared estimation of species networks with differnt reticulations, which represents different model complexity. The MLE method was used to estimate species networks with at most 1, 2, 3, or 4 reticulations ("ret."). Topological distances [100] of pairwise comparison between estimated networks were measured. Only upper triangular entries in the pairwise distance matrix are shown. 125

Table 7.3: Topological distance pairwise comparison of species networks inferred by MLE on different MSAs of the yeast dataset. The MSAs were estimated using ClustalW, MAFFT, or FSA. Table description and layout are identical to Table 7.2. 126

LIST OF FIGURES

Figure 1.1: Darwin's first evolutionary diagram, drawn in his Notebook in 1837. 2

Figure 2.1: Illustration of character evolution. Characters evolve on a tree. The branching structure of the tree guides the mutation process. The red dots on the branches represent the substitution events. 8

Figure 2.2: An example of Multiple Sequence Alignment (MSA). (a) A tree shows the evolutionary history of sequences. Evolutionary events make changes to the content of the observable sequences at leaf nodes. (b) Sequences at the leaf nodes can be observed. The unaligned sequences form a character matrix. (c) Multiple sequence alignments can be used to build a character matrix, each column consists of characters that evolve from the same ancestral character. . . 9

Figure 2.3: Progressive alignment algorithm. The progressive alignment algorithms take a set of sequences as input and build a distance matrix for input sequences. Then a guide tree is estimated based on the distance matrix and pairwise alignment is conducted at each internal node on the tree from leaves to the root. This figure is reproduced from http://readiab.org/. 12

Figure 2.4: An example of phylogenetic tree. (a) A rooted phylogenetic tree. V_r is the root node. Taxon B is an ancestor of taxon A. (b) The unrooted version of the same tree by removing the root node V_r. (c) An polytomy tree. Node G has a degree of four, which represents a polytomy node. 14

Figure 2.5: An example phylogenetic network. (a). A phylogenetic network N, r is the root node, A is a tree node, B is a reticulation node, and C, D, E are leaf nodes. The edge in red is a reticulation edge. The tree T_1 and T_2 in (b) and (c) are the inducted trees of N. 16

Figure 2.6: Illustration of coalescence and multi-species coalescence. (a) An illustration of the coalescent process. Each row represents a generation. Generations have no overlap with each other. The new generation is generated by randomly sampling from the previous generation. Each dot represents a gene copy. The blue line represents the inheritance relationship, where one gene is copied from an ancestor gene. Parents are selected at random. When populations are separated, and gene copies only come from samples from the same population. A coalescence event is a point when two lineages merge into a common ancestor. The coalescent process forms a lineage tree. 19

Figure 2.7: Illustration of deep coalescence. (a) Multi-species coalescence results in a gene tree inside a species tree. In this example, the gene tree is concordant with the species tree in terms of topology. During the process of coalescence, genes may separate before the species separate, which is called deep coalescence. The most recent common ancestor (MRCA) of gene copies sampled from species B and C is older than the speciation event of B and C. (b) Deep coalescence with incomplete fixation of gene lineages in species lineages may cause discordance between the gene trees and the species tree. In this example, the gene copies sampled from species A and B coalescent first, which is different from the order speciation events. 21

Figure 2.8: An illustration of the phylogenetic tree pipeline for a single gene. First, homologous sequences are selected and aligned. Then the phylogenetic tree is inferred based on the alignment result. The figure is adapted from [33]. . . . 23

Figure 2.9: An illustration of the phylogenetic tree pipeline for multiple genes. (a) Concatenation methods first concatenate the alignments of genes into a supermatrix. Then the species tree is inferred based on this super matrix. (b) Summary-based methods infer gene trees from the gene alignments first. Then the species tree is inferred from all the gene trees. (c) Co-estimation methods estimate gene trees and species trees simultaneously in a single statistical inference. 26

Figure 3.1: An example of SERES resampling random walk on unaligned sequences. First, we estimate an alignment for the input unaligned sequences. Then, a set of anchors with the highest similarity is estimated using the estimated alignment. The anchors' boundaries and the start and end of the input sequences are used as barriers. Finally, the SERES random walk performs on the input sequences with barriers. The random walk starts at a randomly selected barrier and moves to a randomly selected direction to the next barrier. It reverses at each encountered barrier with probability γ and reverses with certainty at the start and end barriers of the input sequence. During the random walk, the sub-sequences between the previous barrier and the current barrier are sampled for the replicate. The resampling procedure ends when the resampled sequences reach the length criteria. 39

Figure 3.2: SERES+GUIDANCE2 performance using different choices for anchor length. Results are shown for five 10-taxon medium-gap-length model conditions (named 10.A through 10.E in order of generally increasing sequence divergence). We evaluated the performance of SERES+GUIDANCE2 where anchor length in bp was either 3, 5, 10, 30, or 50. We calculated each method's precision-recall (PR) and receiver operating characteristic (ROC) curves. Performance is evaluated based upon aggregate area under curve (AUC) across all replicates for a model condition (n = 20). 49

Figure 3.3: SERES+GUIDANCE2 performance using different choices for the number of anchors. We evaluated the performance of SERES+GUIDANCE2 where the number of anchors used was either 3, 5, 20, 50, or 100. Otherwise, figure layout and description are identical to Figure 3.2. 50

Figure 3.4: Runtime comparison of methods under study. (a) For each method, average runtime (h) across all replicates in each simulation study model condition is reported (n = 20); standard error bars are also shown. The 10-taxon model conditions 10.A through 10.E are shown in order from left to right, followed by the 50-taxon model conditions similarly. (b) Method runtimes are shown for each empirical study dataset. Datasets are arranged from left to right in order of increasing dataset size as measured by number of taxa. 51

Figure 3.5: Memory usage comparison of methods under study. Memory usage is shown in GiB. Otherwise, figure layout and description are identical to Supplementary Figure 3.4. 52

Figure 4.1: The posterior probability distribution inferred by the standalone recHMM method on 4-taxon model conditions. For each site, we split the local gene tree topologies into true class, which contains the true gene tree topologies for the site, and false class', which contains all other gene tree topologies. For each class and each replicate dataset in a model condition, the inferred posterior probabilities for gene trees at any site were binned into deciles; the resulting histogram was normalized over all replicates in a model condition (n = 30). The normalized histograms for the true and false classes are shown in blue and orange, respectively. 63

Figure 4.2: Histogram of posterior probabilities inferred by SERES+recHMM method on 4-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1. 64

Figure 4.3: Distribution of posterior probabilities inferred by standalone recHMM method on 6-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1. 67

Figure 4.4: Distribution of posterior probabilities inferred by SERES+recHMM method on 6-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1. 68

Figure 4.5: Posterior probability distribution of local gene tree topologies inferred by standalone recHMM versus SERES+recHMM method on Indian HIV-1 dataset. We re-analyzed a subset of the Indian HIV-1 genome dataset that was published by [84]; [150] re-analyzed the original dataset using recHMM. Our re-analysis compared local gene tree probabilities computed using standalone recHMM posterior decoding (top panel) versus SERES+recHMM posterior decoding (bottom panel). The plots show posterior decoding probabilities (y-axis) versus genome coordinate (x-axis). Local gene tree probabilities are colored based on the three possible unrooted topologies for the four-taxon dataset (shown in either blue, orange, or green). 72

Figure 5.1: An illustrated example of RAWR resampling and re-estimation. The first step of the RAWR-based phylogenetic support estimation is sequence resampling. A random walk is performed on the input MSA. MSA sites are resampled during the random walk. The indels are removed from the resampled sites to produce a resampled replicate. Then, MSA is re-estimated from the resampled sequence. Finally, a phylogenetic tree is re-estimated using the re-estimated MSA as input. The dashed lines in the first and second subplots show the reversal breakpoints. 79

Figure 5.2: Runtime and memory usage of the phylogenetic support estimation methods on simulation datasets. The top row includes average runtime usage for each model condition in the simulated study. The y-axis shows runtime in hours and is in log-scale. The left, middle right subplots represent 10-taxon, 50-taxon, and 100-taxon model conditions respectively. The bottom row includes average memory usage for each model condition in the simulated study. The y-axis shows runtime peak memory usage in GiB. The left, middle right subplots represent 10-taxon, 50-taxon, and 100-taxon model conditions respectively. The average runtime or peak memory usage were calculated across all replicate datasets in the model condition (n = 20). 88

Figure 6.1: Pictures of three Darwin's finch species. (a) The small tree finch *Camarhynchus parvulus*. (b) The medium tree finch *Camarhynchus pauper*. (c) The large tree finch *Camarhynchus psittacula*. All three species live on the Galapagos island of Floreana. Arrows indicate the migration of two populations of *C. psittacula* from Isabela and Santa Cruz. Figure comes from [47]. 103

Figure 6.2: Phylogenetic tree of Darwin's finch species and two close relatives reported in Lamichhaney et al. 2015 study[69]. This phylogenetic tree was reproduced from the Figure 1 panel b in Lamichhaney's 2015 paper. The branch length were ignored and the tree was rescaled to an ultrametric tree, where all the leaves have the same distance to the root. The color of the branches and the species name represents the group that the species belongs to. We used the same color as the original study, the purple, brown, cyan, red, green, blue and black color represent the group of warbler finches, vegetarian finch, cocos finch, sharp-beaked ground finches, tree finches, all other ground finches and the outgroups. 109

Figure 6.3: The re-estimated phylogenetic tree for Darwin's finch species and two close relatives with supports estimated by the standard bootstrap method. We re-estimated a species tree using maximum likelihood estimation (MLE) on the concatenated and partitioned genomic sequence alignment of Darwin's finch species and two close relatives. We estimated supports using the standard bootstrap method, which is implemented by RAxML version 8.2.9. The branch length were ignored and the tree was rescaled to an ultrametric tree. The color mapping is the same as Figure 6.2. 110

Figure 6.4: The re-estimated phylogenetic tree for Darwin's finch species and two close relatives with supports estimated by the RAWR-based support estimation method. We calculated supports for the re-estimated phylogenetic tree using the RAWR-based support estimation method. The annotated tree is the same tree as Figure 6.3. The branch length were ignored and the tree was rescaled to an ultrametric tree. The color mapping is the same as Figure 6.2. 111

Figure 7.1: Alignment error for MSA estimation methods of the simulated datasets. The MSA methods in our study consisted of MAFFT, ClustalW, and FSA. We assessed MSA estimation error based on type I and type II error: the former was assessed based on SP-FP proportion ("SPFP"), which is the proportion of nucleotide-nucleotide homologies that appear in the estimated alignment but not the true alignment, and the latter was assessed based on SP-FN proportion ("SPFN"), which is the proportion of nucleotide-nucleotide homologies that appear in the true alignment but not the estimated alignment. Average SPFN and SPFP are shown for each MSA method on each model condition. 116

Figure 7.2: Gene flow detection using the D-statistic on simulated datasets. The left subplot shows the D-statistic score distribution on simulated datasets with model networks. The right subplot shows the D-statistic score distribution on simulated datasets with model trees. The D-statistic values were calculated using MAFFT-estimated alignments, which refer to "estiAln', and true alignments, which refer to "trueAln". Average D-statistic values are reported, and standard error bars are shown over 20 replicates. 120

Figure 7.3: Topological errors of MLE analysis on simulation datasets. The MLE method was conducted on five different inputs: (1) true MSAs and true gene trees (”trueTree”), (2) true alignments and gene trees estimated using FastTree on the true alignments (”trueAln”), (3) ClustalW-estimated alignments and gene trees estimated using FastTree on the ClustalW-estimated MSAs (”clustalwAln”), (4) MAFFT-estimated alignments and gene trees estimated using FastTree on MAFFT-estimated alignments (”mafftAln”), or (5) FSA-estimated alignments and gene trees estimated using FastTree on FSA-estimated alignments (”fsaAln”). Topological error was measured using the reduced distance [100]. Averages and standard error bars are shown for each model condition in the simulation study (n = 20). 122

Figure 7.4: Topological errors of MPL analysis on simulation datasets. Figure description and layout are otherwise identical to Figure 7.3. 122

Figure 7.5: Computational runtime requirements of summary-based species network inference methods for simulation study. The runtime of the MPL and MLE methods on simulation datasets is shown in hours. Averages and standard error bars are shown for each model condition over 20 replicates in the simulation study. 123

Figure 7.6: Computational memory requirements of summary-based species network inference methods of simulation study. The peak main memory usage of the MPL and MLE method on simulation datasets shows in GiB. Figure description and layout are identical to Figure 7.5. 124

LIST OF ALGORITHMS

Algorithm 3.1: SERES walk on aligned sequences 35

Algorithm 3.2: SERES resampling of unaligned sequences 37

Algorithm 3.3: Obtain anchors 38

Algorithm 3.4: Modified Hamming distance calculation 38

Algorithm 5.1: RAWR phylogenetic support estimation 77

Algorithm 5.2: RAWR+teleport resampling procedure 82

CHAPTER 1

INTRODUCTION

Phylogenetics is the study of evolutionary relationships. The evolutionary relationships are often depicted by phylogeny, a graphical representation of the evolutionary history of a group of organisms (taxa), such as genes or species. One of the most simple and widely used representations is the phylogenetic tree, a directed acyclic graph. However, the true evolutionary history is not always tree-like. A phylogenetic network can better depict the horizontal genetic material flow, such as hybridization and recombination.

Phylogenies show the history of genetic information transmission and thus play an essential role in interpreting information on many aspects of organisms, such as the structure and function of genomics. The reconstructed phylogenies are used in numerous biological studies, such as gene function prediction, protein structure prediction, drug discovery, vaccine development, and many non-biological studies, such as computer security applications and linguistics studies.

Since phylogenetic reconstruction is an important and fundamental problem for biological studies, many methods have been developed for this problem [56, 37, 77, 157]. With the technological advances in molecular biology and genomics, increasing amounts of biomolecular sequencing data, such as DNA, RNA, and amino acid sequences, are available for accurate phylogenetic reconstruction, and many of these computational methods infer evolutionary history from biomolecular sequencing data.

A general phylogenetic reconstruction pipeline using sequence data consists of the following steps. First, collect samples from a group of closely related species. Samples are processed and sequenced by any selected sequencing technique, such as Sanger sequencing and next-generation sequencing techniques. Sequence data is produced for several genes of interest. Then the sequence data needs to go through a series of preprocessing steps, such as quality control, filtering, and assembly. After preprocessing, there are two fundamental steps for the phylogenetic reconstruction, Multiple Sequence Alignment (MSA) estimation, and phylogenetic inference.

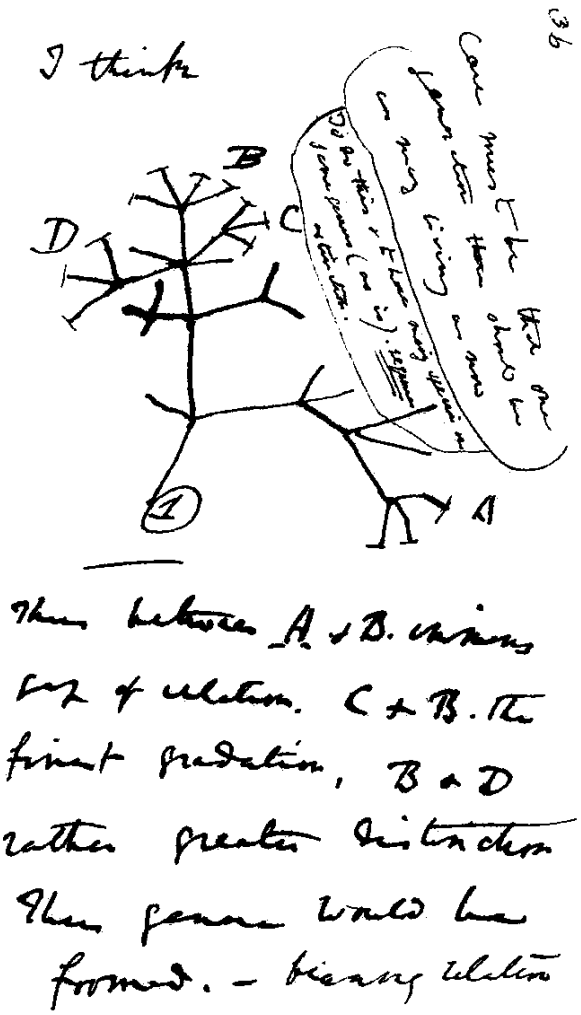


Figure 1.1: Darwin's first evolutionary diagram, drawn in his Notebook in 1837.

The first step is Multiple Sequence Alignment (MSA) estimation. The phylogenetic analysis has to be conducted on homologous sequences, which are characters that evolved from the same ancestor. We discuss the MSA estimation problem in detail in Chapter 2. The MSA obtained in the previous step is used as input for the phylogenetic reconstruction. There are many approaches to phylogenetic tree and network reconstruction, which we discuss in detail in Chapter 2.

Phylogenetic reconstruction is an important step whose results greatly impact downstream biological studies. However, the traditional sequence-based phylogenetic reconstruction methods involve many over-simplified assumptions. In this thesis, we describe our contribution to relaxing some of these over-simplified assumptions in phylogenetic analysis.

Insertions and deletions, known as indels and represented by gaps in MSAs, carry much phylogenetic information but are frequently overlooked during the phylogeny reconstruction process. Many previous studies have shown that phylogenetic information carried by indels helps infer phylogenies [127, 7, 86]. There are studies show that taking into account indels helps resolve some deep branches in phylogenies [115, 119]. Traditional methods only utilize historical substitution events to reconstruct phylogenetic trees. Moreover, these methods usually take the input MSA for granted and ignore uncertainty in the MSA estimation. For the following reasons, indels are either discarded or treated as missing values during the phylogenetic reconstruction process. Some studies considered that indels were unreliable for phylogenetic reconstruction [46]. This viewpoint was later proved to be incorrect [128]. Accurate estimation of indels can be very time-consuming. Unlike substitution events, where sequence length is not affected, insertion and deletion events often involve many sites, resulting in dependence among sites and changes in the sequence length. Furthermore, multiple indels may overlap, which makes it even more challenging to obtain an accurate estimation. Also, there is no uniform opinion on how to deal with indels in the phylogenetic reconstruction yet.

Based on these challenges, it is important to determine the reliability of indels and take into account the uncertainty of estimated MSAs during the phylogenetic analysis. Resampling and re-estimation are often used in confidence interval estimation. Resampling is the process of drawing samples from the original set of observations. It has been widely used in statistical support estimation, especially those non-parametric approaches, such as the standard bootstrap [32] and jackknife [140]. Non-parametric approaches do not require a particular model. However, these resampling approaches usually assume that the observations are independent and identically distributed (i.i.d), which is another over-simplified assumption that has been widely used in phylogenetic analysis. This assumption does not always hold for biomolecular sequences. Many evolution events produce intra-sequence dependence and functional dependence in the biomolecular sequences that are inconsistent with this assumption, such as recombination and hybridization.

Another over-simplified assumption that we want to address is that the evolutionary relationships

are often simplified to a tree structure, and all reticulate gene flows are ignored. Though the phylogenetic tree is the most widely used representation of the evolutionary history of a group of taxa, the true evolutionary history is not always tree-like. Certain evolutionary processes, such as horizontal gene transfer (HGT), hybridization, and recombination, involve reticulate gene flow between two sibling taxa, which is better represented by a phylogenetic network. Previous research has shown that the uncertainties in MSAs have an effect on downstream tree inference and learning [153, 152]. However, there is little discussion about how the phylogenetic network is influenced by MSA uncertainties.

In this thesis, we relax the widely used over-simplified assumptions introduced above. We introduce new resampling algorithms for the resampling of biomolecular sequences. We take into account the indel uncertainties in several phylogenetic analyses by applying the resampling and re-estimation of MSAs. Furthermore, we show that indel-aware phylogenetic analysis obtains better accuracy than the traditional methods.

The rest of the dissertation is organized as follows. Chapter 2 provides background knowledge about multiple sequence alignment and phylogenetic reconstruction. We introduce the basic concepts of phylogenetic analysis and some of the widely used phylogenetic analysis methods. In Chapter 3, we describe our work on a new sequential resampling algorithm, SERES, which relaxes the over-simplified assumption made by the standard bootstrap approach [32] that all sites in the alignments are evolved independently and identically distributed (i.i.d). We applied the SERES resampling approach together with re-estimation to the MSA support estimation problem and achieved comparable or better performance than the state-of-the-art methods. In Chapter 4, we extend the SERES resampling and re-estimation procedure to another classical problem in computational biology and bioinformatics, recombination-aware local genealogical inference. The SERES resampling approach produces local genealogies that greatly improve the topological accuracy. In Chapter 5, we discuss the simplified version of the sequential resampling algorithm, RAWR, which is designed for phylogenetic tree support estimation by resampling and re-estimation using unaligned sequences. For the phylogenetic support estimation problem, the RAWR-based method

outperformed the bootstrap method on almost all the simulation model conditions and the empirical datasets, which indicates that the RAWR-based support estimate benefits from the indel uncertainties of the input alignment. In Chapter 6, we discuss the application of the RAWR-based support estimation method to the whole-genome sequencing dataset of Darwin's finch from Larmichhaney's study in 2015 [69]. In Chapter 7, we further relax the assumption of phylogenetic tree and introduce a performance study on how MSA uncertainties impact the phylogenetic network inference. We show evidence that the errors introduced in MSA estimation and gene tree estimation steps reduce the accuracy of the inferred phylogenetic network, and an indel-aware reconstruction method is needed for phylogenetic network analysis. Finally, we conclude in Chapter 8 with discussion of our work and future research directions.

CHAPTER 2

BACKGROUND

In this chapter, we briefly introduce the basic concepts that we use in the following chapters. First, we introduce the definition of Multiple Sequence Alignments (MSA) in Section 2.1. We discuss the sequence evolution models and MSA estimation methods in detail. Then we introduce phylogenies in Section 2.2, which includes the definition of phylogenetic tree and phylogenetic network. We further discuss two related conceptions: gene trees and species trees and their discordance in Section 2.3. We also briefly introduce the coalescent model, which integrates the evolutionary process of genes with species. Finally, we discuss the phylogenetic reconstruction methods in Section 2.4 and method performance evaluation in Section 2.4.5.

2.1 Multiple Sequence Alignment

The genetic information encoded in DNA sequences is the biological foundation of life. Evolution and heredity ensure the changes and the continuation of all living species. With the development of sequencing techniques, the cost of genome sequencing has dropped dramatically. Now we can easily read the genomic information using various sequencing technologies [94] and get cheap, large biomolecular sequence datasets. DNA sequences and other sequences derived from the DNA sequences, such as RNA sequences and amino acid sequences, can be represented as one-dimension arrays of characters.

Sequence alignment is a fundamental problem in molecular biology. It is a problem of arranging the sequences to identify regions of similarity. A set of sequences is considered to be unaligned if they have different lengths. Biomolecular sequences are one-dimensional arrays over the alphabet Σ. For DNA sequences, the alphabet $\Sigma = A, T, C, G$. Due to historical insertions and deletions, sequences that share evolutionary history can have different lengths.

The process of sequence alignment is to line up sequences and maximize identical subsequences. The order and the content of the DNA sequences cannot be changed. The only allowed operation

is to add gaps (represented by "-") into the unaligned sequences, which represent gaps that are created by the historical insertion and deletion events, which is referred to as indels. The characters of different sequences are aligned into columns or sites by adding gaps in appropriate positions. All characters in a column are assumed to derive from a common ancestral character. A multiple sequence alignment (MSA) represents the evolutionary relationship among a set of unaligned sequences. The alignment can be considered as a matrix, where each sequence is a row in the matrix, and each column, or site, shows the sequence homology. Homologous are two characters that are evolved from the same ancestral character. Such an evolutionary relationship is called homology. An example is shown in Figure 2.2. Each column in the matrix corresponds to a different character and all the characters evolve from the same tree.

There are two widely used summary statistics that measure the sequence divergence, the gappiness and the Normalized Hamming Distance (NHD). Insertion and deletion events in the evolutionary history create gaps in the multiple sequence alignment. Gappiness is calculated by the percentage gap characters in the multiple sequence alignment matrix. For a pairwise sequence alignment, the classical Hamming Distance [50] is the number of sites in the alignment that contain different characters, which can be considered as a nucleotide pair with two different nucleotides. The hamming distance of multiple sequence alignment is the sum of the Hamming Distances of all pairs of sequences contained in the multiple sequence alignment. The normalized Hamming Distance (NHD) is the hamming distance normalized by the length of the sequence.

2.1.1 Sequence evolution

The evolution process is a series of changes in biomolecular sequences through multiple types of mutations. Substitution, also known as point mutation, is one type of sequence mutation. Substitution is the event that a single base changes into another base. Figure 2.1 shows an example of the substitution process of a single character. This example includes two substitution events along the phylogenetic tree. Note that, we can observe the sequences at leaf nodes, but not the sequences at internal nodes and root nodes.

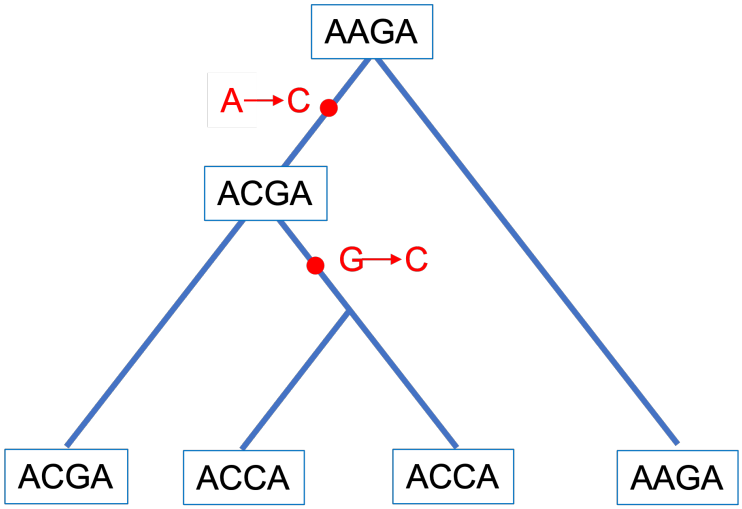
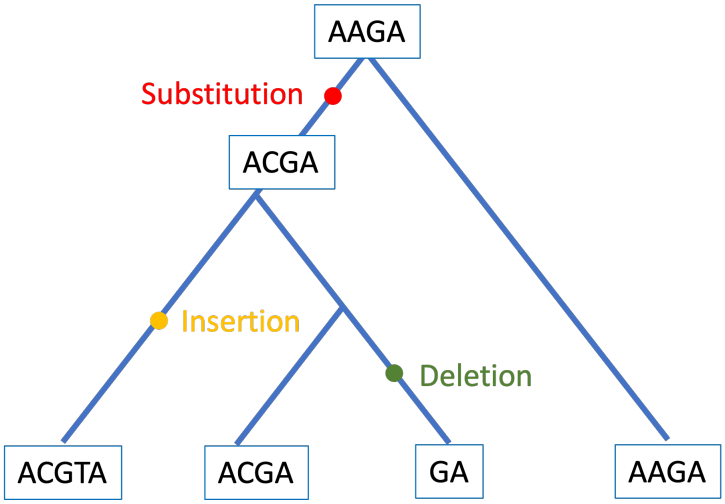


Figure 2.1: Illustration of character evolution. Characters evolve on a tree. The branching structure of the tree guides the mutation process. The red dots on the branches represent the substitution events.

Another important sequence evolution process is insertion/deletion. They can make more complicated changes to the genetic sequences. Insertions and deletions, also known as indels, are the events that occur when genetic materials are inserted or deleted during the evolutionary process. Insertion and deletion events can change the length of the sequences. The characters that used to align in the same column will no longer align with each other. Compared with the evolutionary process with only substitution events, the existence of indels makes it more challenge to identify the homologous characters and their evolutionary relationship.

An example is shown in Figure 2.2. With the existence of insertion and deletion events, the sequences at leaf nodes have different sequence lengths. The characters at the same position on the sequences are no longer homologous. We will discuss the MSA algorithms in Section 2.1.2. These methods reconstruct the homology relationships of the genetic sequence and produce an MSA, a matrix in which each column contains only homologous characters.

Dashes in the alignment represent gaps created by the historical insertion and deletion events. An example is shown in Figure 2.2c. The first two dashes in the second sequence are caused by a single deletion event, labeled as a green dot on the tree in Figure 2.2a. Dashes in the fourth column are caused by an insertion event, labeled as a yellow dot.



(a) A tree of the sequence evolution.

S1	ACGTA
S2	ACGA
S3	GA
S4	AAGA

(b) Unaligned sequences.

S1	ACGTA
S2	ACG-A
S3	--G-A
S4	AAG-A

(c) Multiple Sequence Alignment (MSA).

Figure 2.2: An example of Multiple Sequence Alignment (MSA). (a) A tree shows the evolutionary history of sequences. Evolutionary events make changes to the content of the observable sequences at leaf nodes. (b) Sequences at the leaf nodes can be observed. The unaligned sequences form a character matrix. (c) Multiple sequence alignments can be used to build a character matrix, each column consists of characters that evolve from the same ancestral character.

Evolutionary models describe the substitution events on a single site of the biomolecular sequence. Most evolutionary models assume that the sites evolve down a tree where nucleotide substitutions under a Markov process and all sites evolve independently and identically (i.i.d). A rooted tree T with edges annotated with substitution rate matrices describes the evolution of a single site down the tree. Besides the model tree, T, the model also requires a substitution rate matrix. The substitution rate matrix M describes the probability of the parent state evolving into a child state along an edge. The rate of base i changing to base j is represented by element M_{ij} in the substitution rate matrix M. For DNA sequences, the substitution rate matrix M is a 4×4 matrix. a, b, c, d, e, f denote the relative rates that one base changes to another. The frequencies of the bases A, T, C, G are represented by $\pi_A, \pi_T, \pi_C, \pi_G$ respectively. The diagonal elements of M are chosen to ensure that the sum of elements in the corresponding row equals zero. If for all pairs of bases $i, j, M_{ij} = M_{ji}$, the model is considered time-reversible. The most widely used sequence

evolution models in phylogenetic analyses are time-reversible. The generalized time-reversible model (GTR) [135] has a substitution rate matrix as follows.

$$M = \begin{pmatrix} -(a\pi_C + b\pi_G + c\pi_T) & a\pi_C & b\pi_G & c\pi_T \\ a\pi_C & -(a\pi_C + d\pi_G + e\pi_T) & d\pi_G & e\pi_T \\ b\pi_G & d\pi_G & -(d\pi_G + d\pi_G + f\pi_T) & f\pi_T \\ c\pi_T & e\pi_T & f\pi_T & -(c\pi_T + e\pi_T + f\pi_T) \end{pmatrix}$$

The Jukes-Cantor model [63] is the simplest substitution model, which is a subset of the GTR model. The Jukes-Cantor model assumes that all base frequencies are equal, $\pi_A = \pi_T = \pi_C = \pi_G = \frac{1}{4}$, and all substitution rates are equal, $a = b = c = d = e = f = 1$. The substitution rate matrix of the Jukes-Cantor model shows as follows.

$$M = \begin{pmatrix} -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} \end{pmatrix}$$

To incorporate the rate heterogeneity across sites into the sequence evolution models. We can multiply the substitution rate matrix with a rate drawn from Γ distribution. The combination of the substitution model with the Γ model produces rate variation across sites [155]. For example, GTR+Γ is the GTR model combined with the Γ model for rate variation.

2.1.2 Multiple Sequence Alignment Estimation

In computational biology, the estimation of multiple sequence alignments is one of the most fundamental problems. Multiple sequence alignment estimation is the initial step of many applications, such as predicting the structure and function of proteins and phylogenetic reconstruction.

The following is a definition of a multiple sequence alignment problem. The input is a set of unaligned sequences S. Add gaps in proper positions to align homologous characters, which

are derived from a common ancestral character, into the same column. The output is a multiple sequence alignment A that represents the true evolutionary history of S.

However, true evolutionary history is usually not available. The distance between the estimated alignment and the true alignment is not able to be measured. Thus, the multiple sequence alignment problem was simplified to an optimization problem that minimizes the differences between the sequences in the alignment. One measurement of the sequence differences is the sum-of-pairs (SP) criterion. The SP score is the number of mismatched nucleotide pairs in the MSA. The mismatched pairs include pairs of two different non-gap characters and pairs of one non-gap character with one gap character.

MSA estimation methods are generalized from the pairwise alignment. The estimation of MSA that minimizes the SP score of other similar metrics has been demonstrated to be an NP-hard problem [10, 144]. Many heuristic methods were designed for the estimation of multiple sequence alignment to find a sub-optimal solution instead, such as progressive alignment methods [38].

Progressive alignment methods first build a guide tree based on the distance of sequences. The pairwise sequence alignment is performed at the parent of two leaf nodes, then recursively performed at internal nodes. The final MSA result will be created at the root, as shown in Figure 2.3. Clustal W [73], MUSCLE [29], MAFFT [64] and T-COFFEE [107] are a few examples of progressive alignment methods.

Though progressive alignment has been widely used, there are some disadvantages associated with this type of MSA estimation algorithm. First, the choice of the initial guide tree directly influences the MSA result and the downstream analysis, such as phylogenetic inference. The inferred phylogenetic trees are biased towards the initial guide tree by using such an MSA approach [152]. Furthermore, distantly related sequences tend to be over aligned by progressive alignment [85]. The insertion and deletion events are not evenly penalized in progressive alignment.

Many standard metrics are designed for the quantitative measurement of the alignment methods' performance. The performance measurement usually compares the reference alignment with the estimated alignment. The SP-score is calculated by the percentage of all pairwise homologies in

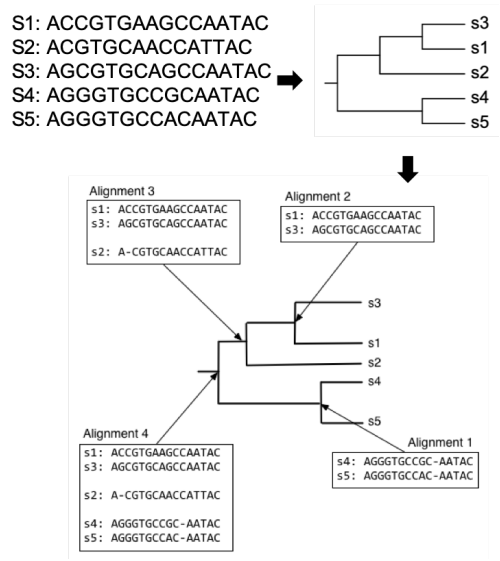


Figure 2.3: Progressive alignment algorithm. The progressive alignment algorithms take a set of sequences as input and build a distance matrix for input sequences. Then a guide tree is estimated based on the distance matrix and pairwise alignment is conducted at each internal node on the tree from leaves to the root. This figure is reproduced from http://readiab.org/.

the reference alignment recovered in the estimated alignment. The modeler score is the percentage of all pairwise homologies in the estimated alignment that are found in the reference tree. The pairs score averages the SP-score and the modeler score to penalize false positive and false negative homologies equally. TC score, which is the number of columns that are recovered entirely correctly in the estimated alignment. The most commonly used measurement for the MSA estimation problem is the SP-score. In the following studies, we used software called FastSP [160] to measure the alignment accuracy.

2.2 Phylogenies

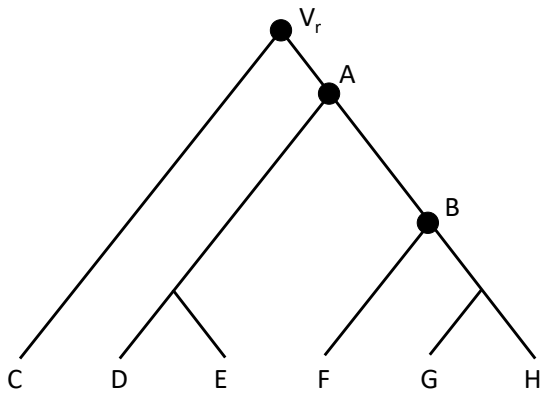
A phylogeny is a graphical representation that represents the evolutionary history of a group of entities, such as genes and species, which are usually referred to as taxa. There are mainly two types of phylogenies: phylogenetic trees and phylogenetic networks. In the following content, we will discuss about these two types of phylogenies in detail.

2.2.1 Phylogenetic Trees

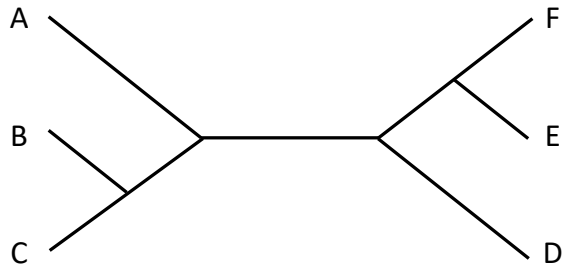
One of the most simple and widely used representations is the phylogenetic tree, a directed acyclic graph. A tree T consists of a set of vertices V and edges E. The vertices V indicate a particular taxon, and the edges E indicate evolutionary history. An internal node is a node that has a degree greater than one. The taxon at an internal node represents the ancestral taxon of the descendent taxa. A leaf node or terminal node has a degree of one, which represents an extant taxon. An edge or branch represents the evolutionary relationship between two taxa it connects. The length of the edge in a phylogenetic tree is called branch length, a non-negative number representing the quantitative measurement of the evolutionary changes or evolutionary time between the two related taxa. Trees without branch lengths attached are referred to as topologies.

A rooted tree has a root node $v_r \in V$. The root node is the most recent common ancestor of all taxa in this tree. This designation of root vertex allows us to associate ancestor-descendant relationships to the vertices. For example, if a node B is located on the path from node A to root, node B is considered an ancestor of node A, as illustrated in Figure 2.4a for examples. It usually requires additional knowledge about the taxa included in the tree or molecular clock assumption to root a phylogenetic tree. There are many different rooting methods. Some popular ones include rooting a tree utilizing an outgroup taxon, and rooting a tree by estimating the time of speciation. One method utilizes an outgroup taxon, which has a far more evolutionary relationship than all other taxa under consideration. The other method roots a phylogenetic tree based on the estimated time between speciation. This method can only be used if we assume that the evolution rate of molecular data, which we use to reconstruct a phylogeny, is constant. However, this assumption is often violated in real datasets. Unrooted trees have no such ancestor-descendant relationship. A phylogenetic tree is considered a binary tree if all vertices have a degree of three. A phylogenetic tree is called a polytomy if at least one internal node has a degree greater than three. Polytomies represent unresolved evolutionary relationships.

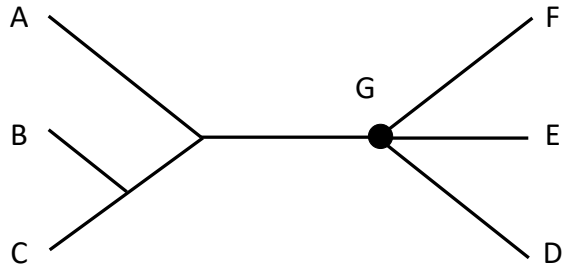
Figure 2.4a shows an example of a rooted binary phylogenetic tree over six taxa, where the direction of evolution is the direction from the root node to the leave nodes. The evolutionary



(a) A rooted phylogenetic tree.



(b) A unrooted phylogenetic tree.



(c) A polytomy tree.

Figure 2.4: An example of phylogenetic tree. (a) A rooted phylogenetic tree. V_r is the root node. Taxon B is an ancestor of taxon A. (b) The unrooted version of the same tree by removing the root node V_r. (c) An polytomy tree. Node G has a degree of four, which represents a polytomy node.

relationship among six taxa can be inferred from the tree. Figure 2.4b shows an unrooted version of the phylogenetic tree of Figure 2.4a by removing the root node V_r. Figure 2.4c shows an example of a polytomy tree.

A clade is a rooted subtree in a phylogenetic tree T, defined by a node v in T that roots the clade. Taxa included in a clade have a closer evolutionary relationship with each other than the rest of taxa in the same phylogenetic tree.

Bipartition is a similar conception as the clade in unrooted trees. A bipartition is defined by edges. By removing an edge e, the unrooted phylogenetic tree T is split into two subtrees T_1 and T_2, and the set of taxa is also split into two subsets. The leaf bipartition denotes $\{l_1|l_2\}$, where $l_1 \cup l_2$ is the complete set of leaves of T.

A phylogeny does not necessarily need to be tree-like. In certain scenarios, more complex representations such as phylogenetic networks fit better than the tree structure.

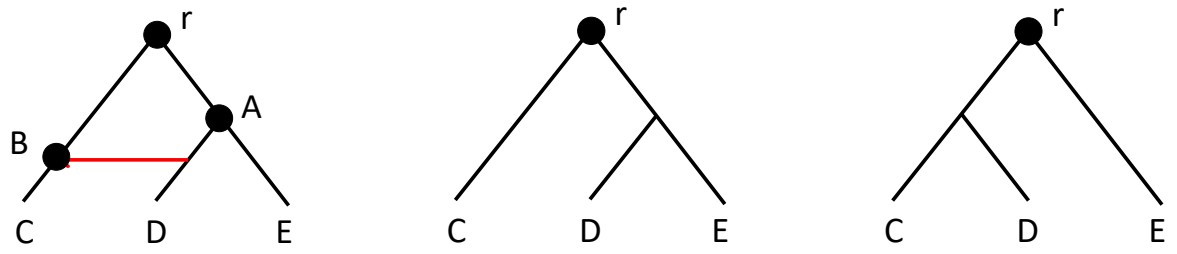
2.2.2 Phylogenetic Network

Though the phylogenetic tree is the most widely used representation of the evolutionary history of a group of species, the true evolutionary history is not always tree-like. Certain evolutionary processes, such as horizontal gene transfer (HGT), hybridization and recombination, involve reticulate gene flow between two sibling taxa, which is better represented by a phylogenetic network. By adding reticulation edges, we can better model the horizontal flow of genetic material.

A phylogenetic network N is a rooted, directed, acyclic graph (DAG) defined on a set of taxa S. Vertices V of N is defined as $V = \{r\} \cup V_L \cup V_T \cup V_N$, where

- $indeg(r) = 0$ (r is the root of N);
- $\forall v \in V_L, indeg(v) = 1$ and $outdeg(v) = 0$ (V_L are the external tree nodes or leaves of N);
- $\forall v \in V_T, indeg(v) = 1$ and $outdeg(v) \geq 2$ (V_T are the tree nodes of N);
- $\forall v \in V_N, indeg(v) = 2$ and $outdeg(v) \geq 1$ (V_N are the reticulation nodes of N);

$E \subseteq V \times V$ are the edges of network N. There are two types of edges, reticulation edges who point to reticulation nodes, and tree edges who point to tree nodes. The leaves of the network are bijectively labeled by the elements of S A phylogenetic network induces a set of trees, called



(a) A phylogenetic network N. (b) A induced tree $T1$. (c) A induced tree $T2$.

Figure 2.5: An example phylogenetic network. (a). A phylogenetic network N, r is the root node, A is a tree node, B is a reticulation node, and C, D, E are leaf nodes. The edge in red is a reticulation edge. The tree T_1 and T_2 in (b) and (c) are the inducted trees of N.

induced trees. The induced trees are obtained by keeping only one reticulation edge for each reticulation node in the network.

Figure 2.5 shows an example of a phylogenetic network N on taxa set $S = \{C, D, E\}$. In the network N, r is the root node with an indegree of 0. A is an example of a tree node with an indegree of 1 and an outdegree of 2. B is an example of the reticulation node with an indegree of 2 and an outdegree of 1. C, D, E are leaf nodes with an indegree of 1 and an outdegree of 2. The edge in red is a reticulation edge that points to a reticulation node. Before the reticulation event, all taxa evolved under a tree structure. Following the reticulation event, two lineages merge to form a new lineage, with the genetic material of the new lineage inherited from either ancestral lineage, which can be represented by two induced trees, T_1 and T_2, which are part of the network N.

2.3 Gene Trees and Species Trees

Phylogenetic trees can be used to represent evolutionary history for different types of entities. Species and genes are two closely related entities that can be modeled by phylogenetic trees.

In a species phylogeny, each leaf node represents the entire population of a species. Each internal node represents a speciation event, where the population of one species is split into subsequent species through multiple mechanisms [19], such as allopatric speciation. In an allopatric speciation event, a population is split into two populations due to geographical isolation. Each population then

evolves independently until it becomes a new species. The succession of such speciation events can be captured by a tree structure called the species tree. The genetic material of extant species can help us identify ancient speciation events.

Gene phylogeny shows the evolution path of particular genes, which are short regions of the genome across all involved species. Gene trees are not necessarily the same as the species tree or other gene trees. For example, humans and chimpanzees have a closer evolutionary relationship than humans with gorillas in some parts of the genome than in other parts [28]. The inconsistency between gene trees and the differences between gene trees and the species tree is so-called gene tree discordance or incongruence. Many evolutionary events, such as hybridization, horizontal gene transfer, recombination, and gene duplication and loss, can influence genome evolution and cause discordance among the evolutionary histories of genes and species [89, 23]. Some of those evolutionary events can cause genes to have different phylogenies compared with the species tree, but they do not contradict the tree-like structure of the species phylogeny. The other evolutionary events may result in complex evolutionary histories, which is hard for tree structure at the species level to depict.

The species tree representation is constructed based on the assumption that one species evolved from only one ancestral species. Such an assumption is accurate for many cases of vertical evolution. However, those evolutionary events involved in reticulation evolution may break the tree structure. For example, hybridization events where a new species evolves as a result of hybridization between two species [111]; horizontal gene transfer (HGT) events where genetic material can be obtained from the environment or other sibling species [44, 11]. Under such scenarios, gene phylogenies can still be represented by a tree structure, however, species phylogenies can not be a tree structure. Species networks with reticulation edges are better form of species phylogeny with such gene tree discordance. In such cases, gene phylogenies are still trees, but the species phylogeny is best modeled as a network [101, 61].

2.3.1 Coalescence

Before we introduce the concept of coalescence, let's first introduce some related concepts.

Recombination is an evolutionary process in which individuals of current generations have recombined the genomes of multiple individuals in previous generations. In this case, the ancestor of the genetic materials may vary. When we move along on a particular chromosome with recombination, the ancestors, where the genetic material is inherited, may switch for different regions on the chromosome. As the accumulation of the recombination events, the chromosome is divided into multiple regions, where sites in the same region share the same evolutionary history, but the evolutionary process may vary from one region to another.

A coalescent gene, also known as a c-gene, is a continuous region on a genome where all sites share the same evolutionary history and where there are no breakpoints caused by recombination events [42, 27, 58]. Note that, the term "gene" usually refers to stretches inside the genome that can be translated into proteins and perform certain functions, which is different from the definition of coalescent gene. The coalescent gene is considered as the most basic unit of the genome for phylogenetic analyses, where only a single evolutionary history is included.

The accumulation of recombinations results in the formation of multiple coalescent genes. The evolutionary history of one coalescent gene is represented by one tree. Different coalescent genes may have different evolutionary histories. This phenomenon can be mathematically described by the coalescent process [66]. The coalescent process models the gene variances of a genealogy. This process is easier to understand on the population level. It starts with two gene variants, which are called alleles, of the present time, and then continuously traces back their parent alleles in previous generations until we reach a point where the two alleles share the same ancestor. This point is where the two alleles coalesce.

The coalescent model was first proposed by Kingman in 1982 [66]. This model is widely used to estimate parameters such as the recombination rate, population size, and migration rates. Kingman's coalescent model assumes constant population size, random mating, a large enough population, and no overlap generation. Under such assumptions, the number of generations, which

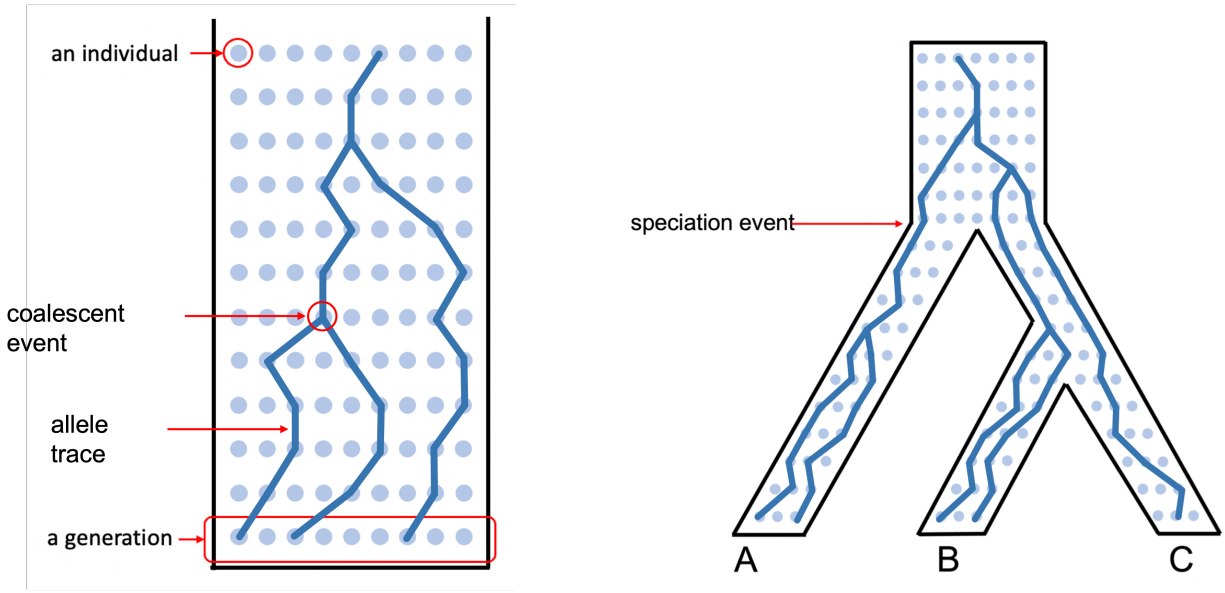
is the time that two random alleles coalesce, is exponentially distributed [66]. Let T_i be the waiting time for any two alleles from i sampled alleles be coalescent, and N_e be the population size, the waiting time T is calculated by following equations, when $i = 2$:

$$T_2(N_e) \sim \exp(t; \lambda = \frac{1}{N_e}) = \frac{1}{N_e} e^{-\frac{t}{N_e}}$$

More broadly:

$$T_i(N_e) \sim \exp(t; \lambda = \frac{\binom{i}{2}}{N_e}) = \frac{\binom{i}{2}}{N_e} e^{-t\frac{\binom{i}{2}}{N_e}}$$

The coalescent history can be represented by An example is shown in Figure 2.6.



(a) The coalescent process.

(b) The multi-species coalescent.

Figure 2.6: Illustration of coalescence and multi-species coalescence. (a) An illustration of the coalescent process. Each row represents a generation. Generations have no overlap with each other. The new generation is generated by randomly sampling from the previous generation. Each dot represents a gene copy. The blue line represents the inheritance relationship, where one gene is copied from an ancestor gene. Parents are selected at random. When populations are separated, and gene copies only come from samples from the same population. A coalescence event is a point when two lineages merge into a common ancestor. The coalescent process forms a lineage tree.

2.3.2 Multi-Species Coalescent (MSC) Model

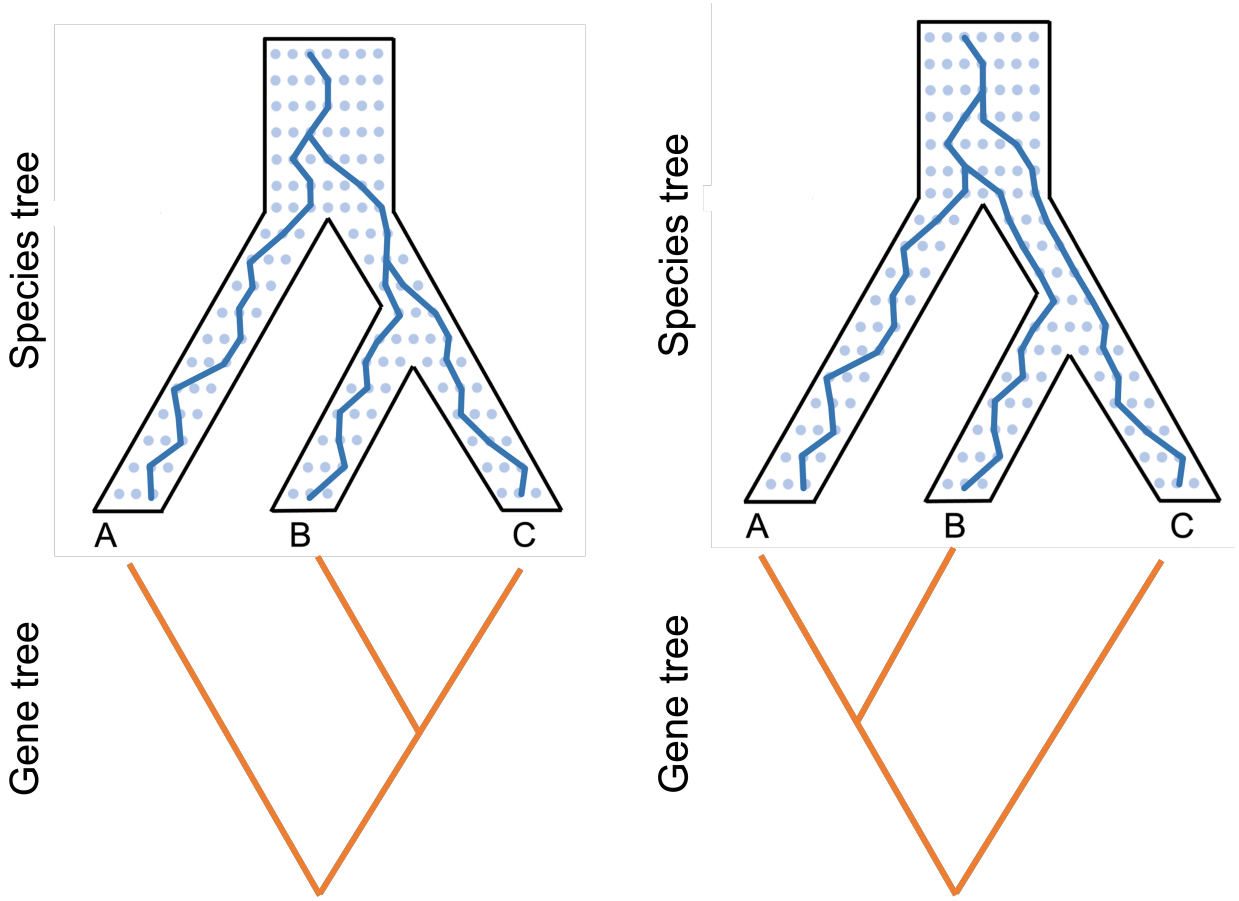
In the previous section, we discussed the coalescent model under a single population. For phylogenetic analyses involving multiple species and multiple populations, the general framework can be

extended to the Multi-Species Coalescent (MSC) model [114]. The MSC model tree is a species tree with branch length represented by coalescent units. Each leaf node of the model tree represents a population of species with fixed population size. Each branch represents one instance of the Kingman coalescent process. Random mating happens between individuals within the same population. At the internal nodes, where speciation events happen, the lineages that have not coalesced yet in the child populations are moved to the parent population. The coalescent process continues in the parent population. An example of the multi-species coalescent process is shown in Figure 2.6b.

The MSA model traces alleles back in time utilizing the following procedure. For three leaf species, A, B, and C and their parent populations, as shown in Figure 2.6b. From each leaf species, we sampled k different individuals. At the terminal branch leading to species A, we start with k_a individuals at the bottom and trace back the Kingman coalescence for t_a generations, where t_a is the length of the branch. During this time, some alleles coalesce, and some do not. At the start of the branch, r_a denotes the remaining alleles that have not coalesced yet. Assume $r_a \leq k_a$, then $k_a - r_a$ coalescent events happened on this branch. A similar process also happens in species B and C. Repeat this process on all branches until all the alleles coalesce into the root branch. The MSC model assumes that the coalescent histories in different branches of the species tree are independent. The coalescent history represents a gene tree that evolves inside the species tree. The randomness of the coalescent process may result in different gene trees. Some gene trees may have different topologies compared with the species tree. We will discuss the discordance between the gene tree and species tree in the following section.

Incomplete Lineage Sorting (ILS) describes the discordance between gene trees and the species tree. The multispecies coalescent process may result in various gene trees, as the example shown in Figure 2.7

When tracing two lineages from sibling populations back in the multispecies coalescent process, it is possible that two lineages do not coalesce before reaching the nearest ancestral population. If these two lineages do not coalesce, they go further back in time to a deeper ancestral population.



(a) A cordant gene tree.

(b) A discordance gene tree.

Figure 2.7: Illustration of deep coalescence. (a) Multi-species coalescence results in a gene tree inside a species tree. In this example, the gene tree is concordant with the species tree in terms of topology. During the process of coalescence, genes may separate before the species separate, which is called deep coalescence. The most recent common ancestor (MRCA) of gene copies sampled from species B and C is older than the speciation event of B and C. (b) Deep coalescence with incomplete fixation of gene lineages in species lineages may cause discordance between the gene trees and the species tree. In this example, the gene copies sampled from species A and B coalescent first, which is different from the order speciation events.

This scenario is called deep coalescence. In the deeper ancestral population, other lineages from other species are also present. Under the random mating assumption, the gene lineages from other species may coalesce with one of the sibling lineages before they coalesce with each other. In this situation, gene trees become discordant with the species tree, and this scenario is called Incomplete Lineage Sorting (ILS).

An example of ILS is shown in Figure 2.7b. In the ancestral population of species B and C, the gene lineages from these two species do not coalesce. Both gene lineages go back to the deeper ancestral population of species A, B, and C. In that deep ancestral population, the gene lineages from species A and B coalesce first. Then it coalesces with the linage from C. This ILS results in a gene tree where A and B are sibling species, which is different from the species tree, where B and C are siblings.

Each species tree has a unique distribution of gene trees under the MSC model [24], and it can be defined by a unique distribution of the true gene trees [24, 2]. The probability of observing a particular gene tree topology can be calculated by random sampling of a set of gene trees from this distribution. It is possible to infer the true species tree by sampling a sufficient number of gene trees, despite the discordance between the gene trees and the species tree. Yet this is not an easy problem to solve [22]. The most likely gene tree may be inconsistent with the species tree under certain conditions. We will talk about this problem in detail in the following sections.

2.4 Phylogenetic Reconstruction

The sequencing data has been used for phylogeny reconstruction for decades [56, 77, 157]. Many different methods and models are designed for phylogenetic inference using sequencing data. In the following content, we discuss the standard pipeline of phylogenetic reconstruction in detail.

2.4.1 Phylogenetic Tree Reconstruction of Single Gene

Modern phylogenetic analyses usually take molecular sequence data as input. The first step of the phylogenetic analysis is to collect data from organisms of interest and sequence the samples to get

genome data. High throughput sequencing technologies can read the whole genome or transcriptome and produce short reads that can be later assembled into longer sequences by computational methods [124]. The most frequently used biomolecular sequence data in phylogenetic analysis are DNA and RNA sequences. Note that the RNA sequences constitute only a small portion of the whole genome since the RNA sequences only contain the genetic materials of the coding genes.

The pipeline of phylogenetic tree inference of a single gene mainly consists of two steps: multiple sequence alignment and phylogenetic tree inference. An illustration is shown in Figure 2.8.

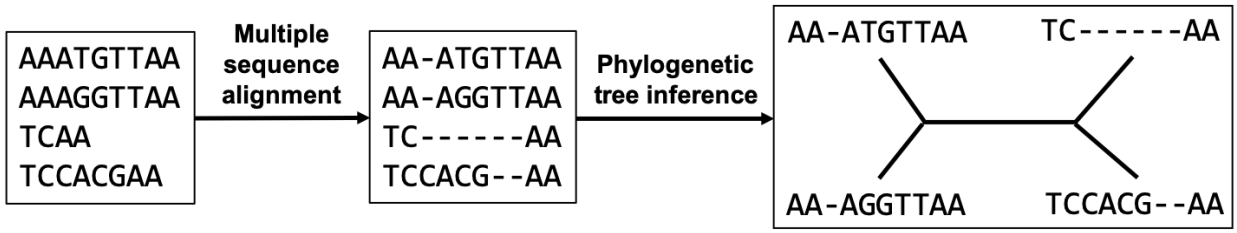


Figure 2.8: An illustration of the phylogenetic tree pipeline for a single gene. First, homologous sequences are selected and aligned. Then the phylogenetic tree is inferred based on the alignment result. The figure is adapted from [33].

The phylogenetic reconstruction pipeline takes a set of unaligned sequences as input. First, the sequences are aligned into MSA. Then the MSA estimated in the previous step is used as input for the phylogenetic tree inference. Like many other data analysis pipelines, one problem with this two-phase pipeline is that the quality of alignment estimated in the first step impacts the downstream phylogenetic tree inference [15, 109, 85, 143, 78]. The basic concepts and estimation methods of MSA are introduced in Section 2.1.

There are mainly four types of phylogenetic tree reconstruction methods, distance-based methods, Bayesian methods, Maximum Parsimony (MP) methods, and Maximum Likelihood (ML) methods. These tree inference methods usually reconstruct the phylogenetic tree based on the historical substitution events. The other more complicated evolutionary events are ignored, such as insertions and deletions. Gaps in the input MSA are treated as missing values. Though the standard version of these methods ignores other evolutionary events, it is possible to extend these methods

to more complex evolutionary events.

One important conception of a statistical inference method is statistically consistent. A method is considered statistically consistent when the inference value converges to the true value as the amount of input data increases. Many statistically consistent methods have been developed for the phylogenetic reconstruction. For example,*BEAST [44], MP-EST [124], and ASTRAL [151, 108] are some of the most widely used phylogenetic tree inference methods that are statistically consistent.

In the following content, we introduce four types of phylogenetic tree inference methods in detail.

Distance-based methods infer the evolutionary relationship based on the distance between input taxa. First, we construct a distance matrix that contains pairwise distances between all possible taxa pairs. Then the distance-based methods utilize the distance matrix to reconstruct a phylogenetic tree.

Bayesian methods calculate the posterior probability distribution of trees utilizing the prior probability of a particular tree and the likelihood of the input data. Given the prior probability, the likelihood of the data, and the correctness of the likelihood model, the posterior probability of a particular tree is the probability of this tree being the true tree. Bayesian methods usually search the tree space by utilizing the Markov Chain Monte Carlo (MCMC) algorithm.

Maximum Parsimony (MP) methods infer a phylogenetic tree by searching for the tree that best explains the observed sequencing data with the minimum number of substitution events.

The search for the best tree with the lowest score in the entire tree space has been shown to be NP-hard [20]. For datasets with a large number of taxa, heuristic search algorithms are used for the tree search [134, 9, 41, 43]. Heuristic methods utilize the hill-climbing algorithm to approach the best solution progressively, but there is no guarantee of the optimal solution.

Maximum Likelihood (ML) methods reconstruct a phylogenetic tree using the following criteria. Given a sequence evolution model and a set of sequencing data, assume that the sequencing data is generated under the evolution model and search for a tree with the maximum likelihood of producing the observed sequencing data.

Finding the ML tree is also proved to be NP-hard [117] as the MP tree. Therefore, many heuristic algorithms have been developed. Instead of finding the ML tree, the heuristic algorithms search for an approximation of the ML tree [37, 98, 112].

There are some widely used ML tree methods, such as PhyML [49], FastTree [112] and RAxML [130].

2.4.2 Branch support

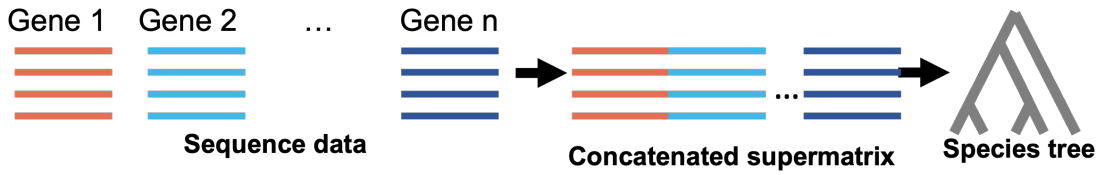
It is not an easy task to infer phylogenetic trees. Phylogenetic trees inferred by any phylogenetic tree inference methods are expected to contain errors. However, we do not have access to the true evolutionary history except in experiment settings [132]. Besides phylogenetic tree estimation, it is also crucial to have a quantitative measurement for the confidence of the inferred tree and individual branches in the inferred tree.

The posterior probability distribution of trees, which is by Bayesian methods, can be used as tree support. For the other methods, the support of the inferred tree is usually calculated by the bootstrap method [55, 36].

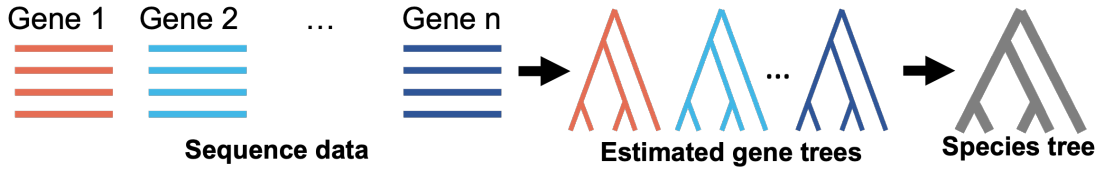
The bootstrap support estimation method first samples a sufficiently large number of replicate datasets at random. Then trees are inferred for each bootstrap replicate, respectively. A sample of the possible universe of the data that we could have observed is provided by the bootstrap replicates. For each branch in the estimated tree, the frequency that this branch appears in the bootstrap replicates is used as the branch support.

2.4.3 Phylogenetic Tree Reconstruction of Multiple Genes

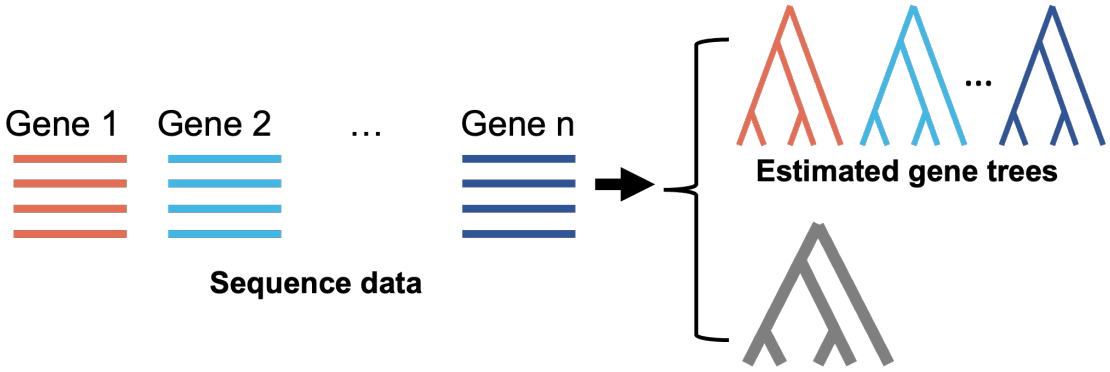
In the previous section, we introduced the standard phylogenetic reconstruction pipeline for a single gene. The limitations of such phylogenetic analysis are rather obvious. First, a single gene usually contains a few hundred to thousand base pairs. Limited sites restrict the phylogenetic signals carried by one gene. Another problem is that the gene tree topologies often disagree with the species tree due to evolutionary events, such as hybridization and horizontal gene transfer. As



(a) Pipeline of concatenation methods.



(b) Pipeline of summary-based methods.



(c) Pipeline of co-estimation methods.

Figure 2.9: An illustration of the phylogenetic tree pipeline for multiple genes. (a) Concatenation methods first concatenate the alignments of genes into a supermatrix. Then the species tree is inferred based on this super matrix. (b) Summary-based methods infer gene trees from the gene alignments first. Then the species tree is inferred from all the gene trees. (c) Co-estimation methods estimate gene trees and species trees simultaneously in a single statistical inference.

discussed in Section 2.3, the discordance between gene trees and the species tree is very common. Even though we have a perfect computational method that can infer a completely correct gene tree from given sequencing data, it is possible that the species tree has a different topology. Thus, it is critical to comprehensively analyze multiple genes and integrate the overall gene tree distribution when inferring the species tree. Phylogenetic reconstruction with multiple genes can potentially produce a more accurate species tree due to increased input data and better estimation of the gene tree distribution. There are mainly three types of multi-gene phylogenetic reconstruction pipelines: concatenation, summary-based, and co-estimation methods. An illustration is shown in Figure 2.9.

In the following content, we introduce these three types of multi-gene phylogenetic tree inference pipelines in detail. Concatenation methods are the most basic multi-gene phylogenetic reconstruction pipelines, where all the input sequencing data is simply concatenated into one supermatrix, and the phylogenetic tree is inferred based on this supermatrix. This method takes into account the statistical power provided by the entire dataset. With the gene tree discordance, the phylogenetic tree inferred from a single gene of a small set of concatenated genes has a higher probability of disagreeing with the true species tree. In contrast, the concatenated analysis of a sufficient number of genes ignores the conflicting signals shown in different genes and produces a fully resolved species tree with maximum support [120]. For the concatenated analysis, we assume that ILS is the only source of discordance between the true gene trees and the species tree.

Previous simulation studies showed that the concatenation method might give wrong species trees with high support under this assumption [31, 72, 83, 68]. One reason that may cause the wrong species tree is that the most frequent gene tree can have a different topology than the species tree.

The summary-based methods usually take two steps in the phylogenetic tree inference. First, gene trees are inferred independently. Then the gene trees are summarized to infer the species tree. The idea of the summary-based phylogenetic tree reconstruction method comes from the fact that the species tree can be uniquely defined by the probability distribution of gene trees under the MSC model, which we introduced in detail in Section 2.3.2. We can calculate the gene tree distribution by the sequencing data of a sufficiently large number of genes, then estimate the species tree using the gene tree distribution under the MSC model.

One big challenge for the summary-based method is that the accuracy of the inferred gene trees has a great impact on the downstream species tree inference. Hign estimation errors in the gene trees, which are very common for many phylogenomic datasets, can result in reduced accuracy of the inferred species tree [96, 97].

There are many summary-based methods that have been developed to infer species trees by summarizing the gene trees. One of the early approaches uses maximum parsimony criteria on

minimizing deep coalescence (MDC) [89, 90, 137, 162] to infer the species tree.

The main drawback of the summary-based method is that the gene trees are inferred independently, which limits the available data for each gene tree inference. This problem can be solved by inferring gene trees and the species tree at the same time. Such a method is called the co-estimation method. Since the gene trees are not entirely independent from each other, co-estimation of all gene trees and species tree is the best way to retain the dependence among genes during the phylogenetic reconstruction process.

Co-estimation methods infer both gene trees and species tree in one statistical inference, which ensures sufficient data for the inference of each gene tree and retains the dependence among genes. Previous studies have shown that the gene trees inferred by the co-estimation methods have higher accuracy than those inferred by the independent estimation [154, 6]. There are two widely used co-estimation methods, BEST [82] and *BEAST [54]. Both methods conduct Bayesian inference via the MCMC algorithm to simultaneously infer the probability distributions of all gene trees.

2.4.4 Phylogenetic Network Reconstruction

In previous sections, we introduced the phylogenetic reconstruction of trees from sequencing data. The MSC model plays a critical role in modeling the gene tree discordance caused by ILS in phylogenetic tree inference. As we introduced in the previous section, the reticulate evolutionary relationship between closely related species requires phylogenetic reconstruction methods that consider both ILS and reticulation events. The phylogenetic network has been proposed to represent such a complicated evolutionary history [60, 61, 101, 21].

Similar to the phylogenetic tree, a phylogenetic network topology is also represented by a rooted, directed, acyclic graph. The major difference between the phylogenetic tree and the network is that the phylogenetic network allows reticulation edges that model the horizontal gene flow. Reticulation edges represent gene flow between two sibling species or populations that exist in the same period of time. The phylogenetic network describes the evolutionary history of species, and gene trees grow within the species network.

The MSC model has been used in the phylogenetic tree inference from sequencing data of multiple genes. It has been extended to the phylogenetic network and is used for network inference from multi-gene sequencing data [161, 158].

Many computational approaches are developed for the phylogenetic network reconstruction from multiple genes, for example, the maximum parsimony method, the maximum likelihood method, and the Bayesian inference method. We introduce these methods in detail in the following content.

The Maximum Parsimony (MP) method searches for the best tree under the minimizing deep coalescence (MDC) criterion [89]. The MDC criterion was first proposed for phylogenetic tree reconstruction and then extended to the reconstruction of the phylogenetic network [90, 137]. The MP method only considers the gene tree topologies. There are mainly two problems with the MP method and the MDC criterion. One problem is that we cannot estimate parameters other than the network topology under the MDC criterion. Another problem is that the MP method with the MDC criterion is not statistically consistent, especially when short branches exist.

The Maximum Likelihood (ML) method was proposed to solve these two problems [159] with the multispecies network coalescent model [158], which models the stochastic process of gene trees growth in the branches of a phylogenetic network. The ML method searches for the optimal phylogenetic network that maximizes the probability of observing given gene trees. Hill-climbing heuristics can be applied to ML inference to improve search efficiency. The ML method can take into account both gene tree topologies and branch lengths.

Other than the computational complexity, the ML method is easy to overfit on the input data. The ML method tends to give a higher likelihood score to a more complicated network model [4]. For this reason, the ML method may generate a sub-optimal solution. This may not be a problem for the phylogenetic tree reconstruction, but it is a big obstacle for the network inference. Different from the inference of the phylogenetic tree, the phylogenetic network inference needs to determine the number of reticulation events. However, ML inference prefers more complicated networks. Networks with more reticulation edges will have a higher likelihood score. This makes the ML

method not statistically consistent. Adding more reticulation edges to the true network increases the likelihood. Thus, the number of reticulation events needs to be carefully chosen for the ML inference.

The Bayesian inference method can reduce the model complexity by regularizing the prior distribution. The bayesian method infers the posterior distribution of the network given a set of rooted gene tree topologies. The reversible-jump Markov chain Monte Carlo (RJMCMC) is often used for the Bayesian inference of the phylogenetic network.

2.4.5 Evaluation Metrics

We evaluate multiple methods through experiments in this dissertation. Both simulated and empirical datasets are used in the experiments. In simulated experiments, we generate synthetic data under sequence evolution models with various procedures, where the simulation process is under control, and the ground truth is known. We apply multiple methods to estimate the MSA and reconstruct the phylogeny. Since we have access to the ground truth, it is easy to measure the errors in the phylogenetic reconstruction process. In empirical experiments, we use empirical datasets, where the ground truth is often not available. Therefore, we evaluate the method performance using hand-curated reference alignments and trees or prior knowledge from the previous studies.

Many metrics are developed to measure the similarity of two phylogenies. Here, we will introduce some widely used metrics that measure the similarity between two given phylogenies. Note that not all the measurements of tree similarity are symmetric. In the following content, we compare a reference phylogeny to an estimated phylogeny of the same set of taxa.

2.4.5.1 Comparison of Phylogenetic Trees

Phylogenetic trees could be determined by a set of bipartitions. A bipartition is a unique split of leaves generated by deleting one internal edge in one unrooted tree. By deleting the edge, the leave nodes are split into two sets. Given an unrooted tree T, each branch defines a bipartition of taxa.

The bipartitions of T is $C(T) = \pi(e) : e \in E(T)$, where $\pi(e)$ is the bipartition on the leaf set of T produced by removing the internal edge e [147].

The False Negative (FN) rate is calculated by the proportion of bipartitions that are shown in the reference tree but not in the estimated tree. This metric is also known as the missing branch rate.

The False Positive (FP) rate is the opposite of the FN rate, which is calculated by the proportion of bipartitions that appear in the estimated tree but not in the reference tree.

The Robinson-Foulds (RF) distance [116] is the total number of bipartitions that are different between the reference tree and the estimated tree, which includes both false-positive bipartitions and false-negative bipartitions. The Robinson-Foulds (RF) distance[116] of two unrooted phylogenetic trees T_1 and T_2 is defined as

$$d(T_1, T_2) = |C(T_1) \backslash C(T_2)| + |C(T_2) \backslash C(T_1)|$$

In the following sections, we use the normalized RF rate, which is the proportion of bipartitions that are different between the two trees. The normalized RF rate is the mean of FN and FP rates. The definition of the normalized RF rate shows as following.

$$\tilde{d}(T_1, T_2) = \frac{1}{2} \times \left(\frac{|C(T_1) \backslash C(T_2)|}{C(T_1)} + \frac{|C(T_2) \backslash C(T_1)|}{C(T_2)}\right)$$

The normalized RF rate ranges from 0 to 1. When two trees are identical, the distance equals 0. FN rate, FP rate, and normalized RF rate are equal when the reference tree and the estimated tree are bifurcating. RF distance is one of the most widely used metrics for tree similarity measurement. However, when the reference tree is not bifurcating, it is inappropriate to use RN distance to compare the reference tree and the estimated tree.

2.4.5.2 Comparison of Phylogenetic Networks

Before we introduce the distance measurement between two phylogenetic networks, let's first introduce some important concepts. For a phylogenetic network N, its reduced network N' is

obtained by applying the reduction procedure. The reduction procedure is described as follows. Given a phylogenetic network N, use a single node h to replace each maximal subtree t, which does not include any network nodes. The replace node h is treated as a symbolic leaf that represents the subtree t. The equivalence mapping between two phylogenetic networks N_1 and N_2 is defined that, for node u in N_1 and node v in N_2, two nodes are considered as equivalent if both nodes are leaf nodes and share the same label, or both nodes have k children and u_i is equivalent to v_i for $1 \leq i \leq k$.

The distance, which we refer to as reduction-based distance, measures the distance between two phylogenetic networks based on their topologies. For two reduced phylogenetic networks N_1 and N_2 can be calculated by the following equation.

$$d(N_1, N_2) = \frac{1}{2}\left(\sum_{v \in U(N_1)} max\{0, \kappa_{N_1}(v) - \kappa_{N_2}(v')\} + \sum_{v \in U(N_2)} max\{0, \kappa_{N_2}(u) - \kappa_{N_1}(u')\}\right)$$

v' is a node in N_2 that is equivalent to v in N_1, similar for u', where u' is a node in N_1 that is equivalent to u in N_2. $\kappa_{N_1}(v)$ refers to the number of nodes equivalent to v in network N_1. $\kappa_{N_2}(v')$, $\kappa_{N_2}(u)$, and $\kappa_{N_1}(u')$ are defined similarly. [100]

The reduction-based distance is equivalent to the number of rooted sub-networks that appear in one network but not in the other. The reduced distance is very sensitive to small perturbations like the Robinson-Foulds distance.

CHAPTER 3

SERES: THE SEQUENTIAL RESAMPLING AND ITS APPLICATION ON MULTIPLE SEQUENCE ALIGNMENT SUPPORT ESTIMATION

3.1 Introduction

Resampling is the process of drawing samples from the original set of observations. The resampling methods are widely used in computational biology and bioinformatics for statistical support estimation, especially those non-parametric approaches, such as the standard bootstrap [32] and jackknife [140]. Support estimation utilizing resampling techniques usually includes three steps: drawing multiple resampled replicates from the original observation; performing inference or analysis on each resampled replicate; and comparing results among replicates. The standard bootstrap method, which we refer to as the bootstrap method in the following section, is a widely used resampling method for statistical support estimation. This method independently draws objects with replacements from a population.

The bootstrap approach does not require a particular model for the support estimation. However, it assumes that the observations are independent and identically distributed (i.i.d). This assumption does not always hold for biomolecular sequences. Many evolution events produce intra-sequence dependence and functional dependence in biomolecular sequences that are inconsistent with this assumption, such as recombination and hybridization.

To solve the dilemma of i.i.d assumption on the biomolecular sequences sampling, Landan and Graur proposed the Heads-or-Tails (HoT) algorithm [70] for the support estimation of multiple sequence alignment (MSA). The main idea of the HoT algorithm is that the statistical inference or analysis of MSAs should not be affected by the direction of the input alignment. That means the analysis result should be the same for the original alignment (head direction) and the reversed alignment (tail direction).

The two resampled replicates of the HoT algorithm are the original alignment and the reversed

alignment. However, for support estimation, hundreds of resampled replicates are needed. Some follow-up studies proposed new support estimation algorithms that combined the idea of the HoT with the perturbation of the parameters of the progressive MSA algorithms [71, 110, 126]. These new algorithms show advanced performance in the support estimation of MSAs compared with other state-of-the-art methods [65, 107].

In this study, we proposed a new non-parametric resampling approach called SERES, which is short for "SEquential RESampling"[145] that maintains the key property needed for non-parametric resampling, which is the neighbor preservation property. The neighbor preservation property means any pair of bases that are neighbors in the original sequence will always be neighbors in the resampled replicates. SERES utilized an improved format of the HoT algorithm, consecutive random walk, which combined the HoT algorithm with the traditional bootstrap algorithm for the resampling of the biomolecular sequences, so that we can resample many non-parametric replicates that reserve the dependence within a sequence. To test the performance of the SERES resampling method, we applied SERES to the problem of MSA support estimation. The SERES-based support estimation performs comparably or better than the state-of-the-art methods GUIDANCE[110] and GUIDANCE2[126].

3.2 Methods

The SERES resampling method combines the bootstrap method with the HoT algorithm. This method preserves the dependence within the input sequences during the resampling process. The neighboring sites are still neighbors in resampled replicates. The SERES method is capable of producing sufficient resampled replicates for support estimation. SERES can take either aligned or unaligned sequences as input for resampling purposes. Let us first introduce the SERES random walk on the aligned sequences.

3.2.1 SERES walks on aligned sequences

The SERES random walk is performed on a set of aligned sequences. The input alignment consists of MSA sites, which is a column of aligned nucleotides. The random walk starts at a randomly chosen site. The starting point is chosen uniformly at random from the input alignment. The initial moving direction is also chosen uniformly at random. Then the random walk moves on the input alignment in the initial direction. During each step of the random walk, the current site is sampled to the resampled replicate, and the moving direction could reverse at random with the reverse probability γ. The direction certainly changes at the start and end of the input alignment. The random walk ends when the resampled replicate reaches the length of the input alignment.

The SERES random walk could potentially introduce bias to the resampled replicates due to the different reversal probabilities of the input alignment. The start and end sites of the input alignment have a reversal probability of 1. But the other sites have a reverse probability of γ. However, for practical choices of walk length and reversal probability γ sampling bias is expected to be minimal. The detailed pseudocode for a non-parametric SERES walk on a fixed MSA is shown in the Algorithm 3.1.

Algorithm 3.1: SERES walk on aligned sequences

```
1:  procedure SERESWalkOnAlignedSequences(A, γ, numReplicates)
                          ▷ Input: MSA A, walk reversal probability γ, number of SERES replicates numReplicates
                                                                          ▷ Output: list of SERES replicates
2:      replicates = <>
3:      for i = 1 to numReplicates do
4:          direction = (rand() > 0.5) ? +1 : −1      ▷ Uniformly at random (UAR) choose direction (right vs. left)
5:          i = ⌊ length(A) * rand() ⌋ + 1                                  ▷ UAR draw from [1, length(A)]
                                                  ▷ rand() returns floating point number sampled UAR from [0, 1)
6:          replicate = <>
7:          while length(replicate) < length(A) do
8:              replicate .= A_i                          ▷ read A_i, which is the ith character in alignment A
                                                                ▷ Alignment characters A_i are one-indexed
9:              i+= direction
10:             if (i ≤ 0) or (i > length(A)) or (rand() < γ) then
                                                                                ▷ Reflection of random walk
11:                 direction *= -1
12:                 if (i ≤ 0) or (i > length(A)) then
13:                     i+= direction * 2                 ▷ Always reflect at start/end of alignment A
14:         replicates .= replicate
15:     return(replicates)
```

3.2.2 SERES walks on unaligned sequences

In previous section, we described the SERES random walk on the aligned sequences. For the aligned sequences, all sequences have the same length. There is no need to worry about the inconsistency of the random walk on the sequences. However, for the SERES resampling of unaligned sequences, we needs to consider the synchronization among sequences.

We use a set of anchors to ensure synchronization. Anchors are highly conservative short sequence regions. Sub-sequences in one anchor have high sequence similarity to each other. To estimate anchors for the input unaligned sequences, the best practice is to utilize multiple MSA estimation methods, and then select the most conservative short regions that appears in multiple estimated MSA results. In practical, we found that the highly similar regions in a single estimated alignment is sufficient to produce reasonable anchors.

The anchors are defined using following steps. First, estimate a guide MSA for the input sequences. Then we use Average Normalized Hamming Distance (ANHD) to measure the similarity of all possible sequence segments of given anchor length. The indels in the estimated alignments are considered as missing data. Finally, regions with the highest sequence similarity, represented by the lowest ANHD, are selected as anchors. Indices of the unaligned sequences corresponding to the start and end of each anchor are used as barriers. The start and end of the input sequences also serve as barriers. The random walks perform between barriers similar to the random walk on the alignment sequences. The change of direction only happens at the barriers with reverse probability γ. The direction certainly changes at the first and last barriers, which are the start and end of the input sequence. The random walk stops when the longest resampled sequence has reached to the length of the longest input sequence.

The application of SERES random walk on the unaligned sequences requires parametric MSA estimation for anchor selection. Therefore, the overall process is considered as semi-parametric.

An illustrated example shows in Figure 3.1 on a 5-taxon unaligned sequence dataset. First, we estimate an alignment for the input unaligned sequences. Then five anchors were selected based on the sequence similarity. The anchors' boundaries and the start and end of the input sequences are

used as barriers. Finally, the SERES random walk performs on the input sequences with barriers. The red arrows indicates the random walk paths. The dash line showed the reversal events. The random walk reverses at each encountered barrier with probability γ and reverses with certainty at the start and end barriers of the input sequence. During the random walk, the sub-sequences between previous barrier and current barrier are sampled to the replicate. The resampling procedure ends when the resampled sequences reached the length criteria.

The pseudocode for SERES resampling of a set of unaligned sequences S is shown in Algorithms 3.2 through Algorithm 3.4.

Algorithm 3.2: SERES resampling of unaligned sequences

```
1:  procedure SERESWalkOnUnalignedSequences(S, γ, numReplicates)
                ▷ Input: set of unaligned sequences S, walk reversal probability γ, number of SERES replicates numReplicates
                                                                                ▷ Output: list of SERES replicates
2:      replicates = <>
3:      barriers = <>
4:      A_init = ObtainGuideAlignments(S)                                       ▷ See Algorithm 3.3
5:      Ψ = GetAnchorsFromGuideAlignments(S, A_init)                            ▷ See Algorithm 3.3
6:      AddTrivialBarriers(barriers)
7:      for all (a⃗, b⃗) Ψ do
8:          barriers .= a⃗ . b⃗
9:      for i = 1 to numReplicates do
10:         replicates .= SERESWalkOnUnalignedSequences(S, γ, i, barriers)
11:     return(replicates)

12: static variable maxReplicateLengthFactor        ▷ Maximum replicate length is factor of longest unaligned sequence length
13: procedure SERESWalkOnUnalignedSequences(S, γ, replicateNum, barriers)
14:     direction = (rand() > 0.5) ? +1 : −1                                    ▷ UAR choose direction (left vs. right)
15:     i = ⌊ length(barriers) * rand() ⌋ + 1
16:     replicate = <>
17:     while maxLength(replicate) < maxLength(S) * maxReplicateLengthFactor do   ▷ maxLength(S) is length of longest
    unaligned sequence in S
18:         if ((i == 1) and (direction == −1)) or ((i == length(barriers)) and (direction == +1)) then   ▷ reflect at first or last
    barrier
19:             direction *= −1
20:         AsynchronousReadBetweenAdjacentBarriers(S, barriers, i, direction, replicate)   ▷ read result passed by reference to
    mutable object replicate
21:         i += direction
22:         if rand() < γ then                                                  ▷ change walk direction with probability γ
23:             direction *= −1
24:     return(replicate)

25: procedure AsynchronousReadBetweenAdjacentBarriers(S, barriers, i, direction, replicate)
26:     j = i + direction
27:     for z = i to n do
28:         replicate[z] .= (direction > 0) ? substr(S[z], barriers[i], barriers[j]) : reverse(substr(S[z], barriers[j] + 1, barriers[i] +
    1)
                                ▷ substr(x, i, j) returns substring in index interval [i, j) if i < j or empty string if i ≥ j
29:     return                                          ▷ read result passed by reference to mutable object replicate
```

Algorithm 3.3: Obtain anchors

static variable M ▷ MSA methods $M =< M1, M2, \ldots >$
procedure ObtainGuideAlignments(S)
    alignments = <>
    for all (m) M **do**
        alignments .= $m(S)$
    return(alignments)

procedure GetAnchorsFromGuideAlignments(S, A_{init})
    α = <>
    β = <>
    canonicalAlignment = $A_{\text{init}}[1]$ ▷ anchors are indexed based on a fixed alignment in A_{init} (WLOG chosen to be the first alignment in A_{init})
    C_{strict} = GetStrictConsensusColumns(A_{init}) ▷ GetStrictConsensusColumns() returns column indices into first alignment in canonicalAlignment
    $\vec{\alpha}_{\text{strict}}$ = MergeAdjacentColumns(A_{init}, C_{strict}) ▷ merges adjacent columns
        ▷ returns array of ordered pairs $(\vec{x}, \vec{y})$ where start indices $\vec{x}$ and end indices $\vec{y}$ are indexed based on canonicalAlignment
    SortAnchors($\vec{\alpha}_{\text{strict}}$, canonicalAlignment)
    for z = 1 to length($\vec{\alpha}_{\text{strict}}$) **do**
        for i = 1 to n **do**
            $(\vec{x}, \vec{y}) = \vec{\alpha}_{\text{strict}}[z]$
            if substr(canonicalAlignment[i], $\vec{x}[i]$, $\vec{y}[i]$) contains only indels **then**
                $\alpha[i][z]$ = LookupUnalignedSequenceIndex(
GetLastNonIndelIndexInPrefix(canonicalAlignment[i], $x[i]$))
                $\beta[i][z] = \alpha[i][z]$
            else
                $\alpha[i][z]$ = LookupUnalignedSequenceIndex(
GetFirstNonIndelIndexInRange(canonicalAlignment[i], $x[i]$, $y[i]$ + 1))
                $\beta[i][z]$ = LookupUnalignedSequenceIndex(
GetLastNonIndelIndexInRange(canonicalAlignment[i], $x[i]$, $y[i]$ + 1))
    return(α, β)

procedure SortAnchors($\vec{\alpha}$, canonicalAlignment)
    ▷ $\vec{\alpha}$ is an array of ordered pairs $(\vec{x}, \vec{y})$ where start indices $\vec{x}$ and end indices $\vec{y}$ are indexed based on canonicalAlignment
    sort (ComputeModifiedHammingDistance(u, canonicalAlignment) <=> ComputeModifiedHammingDistance(v, canonicalAlignment)) $\vec{\alpha}$
        ▷ perl sort syntax
        ▷ See Algorithm 3.4

Algorithm 3.4: Modified Hamming distance calculation

procedure ComputeModifiedHammingDistance(u, A)
    dist = 0
    $(\vec{x}, \vec{y}) = u$
    for i = 1 to n **do**
        for $j = i + 1$ to n **do**
            dist += ComputeModifiedHammingDistancePair(substr($A[i]$,$\vec{x}[i]$,$\vec{y}[i]$),
substr($A[j]$,$\vec{x}[j]$,$\vec{y}[j]$))
    return(dist / $\binom{n}{2}$)

procedure ComputeModifiedHammingDistancePair(x, y)
    alignedLength = length(x) ▷ aligned sequences x and y have same length
    matches = 0
    for i = 1 to alignedLength **do**
        if ($x[i]$!= INDEL) **and** ($y[i]$!= INDEL) **and** (x[i] == y[i]) **then**
            ▷ homologies involving indels are penalized as mismatch
            matches++
    return(matches / alignedLength)

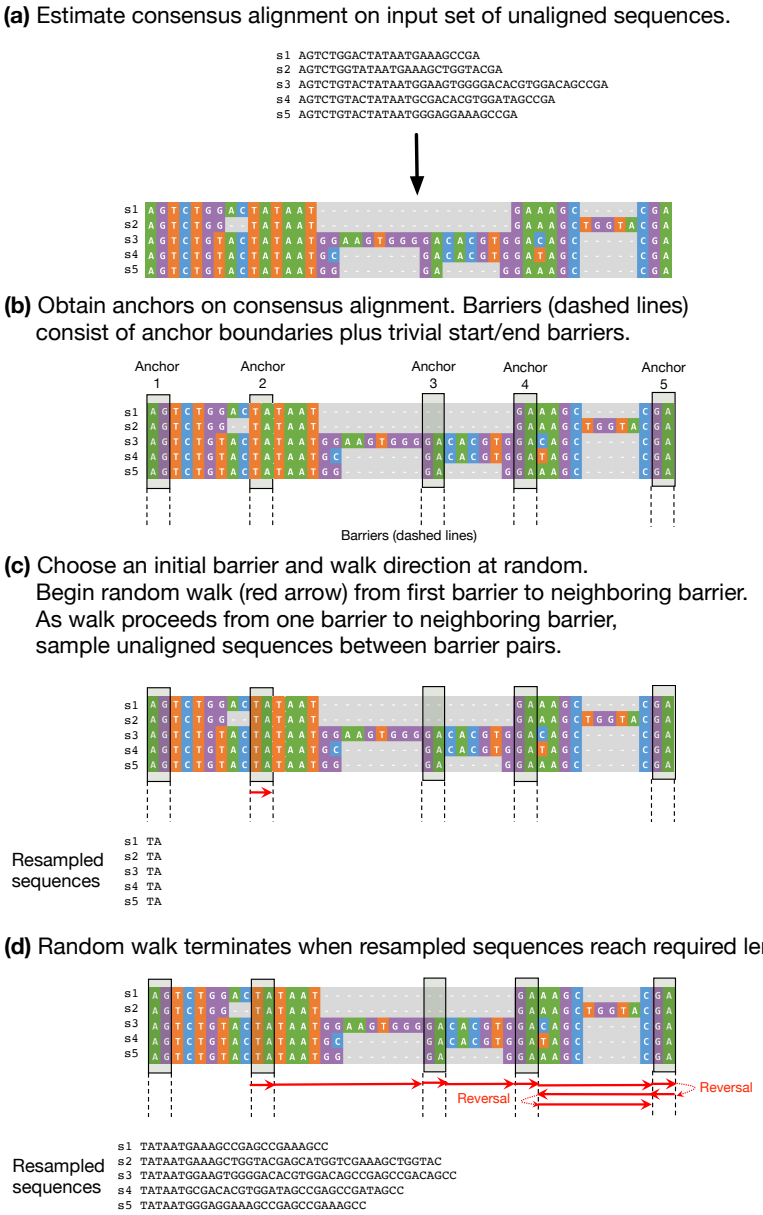


Figure 3.1: An example of SERES resampling random walk on unaligned sequences. First, we estimate an alignment for the input unaligned sequences. Then, a set of anchors with the highest similarity is estimated using the estimated alignment. The anchors' boundaries and the start and end of the input sequences are used as barriers. Finally, the SERES random walk performs on the input sequences with barriers. The random walk starts at a randomly selected barrier and moves to a randomly selected direction to the next barrier. It reverses at each encountered barrier with probability γ and reverses with certainty at the start and end barriers of the input sequence. During the random walk, the sub-sequences between the previous barrier and the current barrier are sampled for the replicate. The resampling procedure ends when the resampled sequences reach the length criteria.

3.2.3 Performance study

In this study, we applied the SERES resampling approach to the MSA support estimation problem and measured the performance of the SERES-based support estimation. The MSA support estimation is used to quantify the confidence of the estimated MSA result. For an estimated alignment, support values are inferred from multiple resampled replicates for each nucleotide-nucleotide homology in the estimated alignment. The support value indicates the confidence of the aligned residue pair.

Many computational methods are designed to solve this problem, such as PSAR[65], TCS[18], HoT[70], GUIDANCE[110] and GUIDANCE2[126]. Among those existing methods, GUIDANCE, which we refer to as GUIDANCE1 in the following section, and GUIDANCE2 are two state-of-the-art methods for MSA support estimation. Both algorithms utilize the uncertainty of the guide tree of the passive MSA methods. By using different guide trees for the passive MSA alignment, multiple alternative MSAs are produced. Then those alternative MSAs are used to calculate the support values for the residue pairs in the input MSA. In addition, GUIDANCE2 also generates alternative MSAs by varying the gap penalty score and co-optimal MSA solution.

Both GUIDANCE1 and GUIDANCE2 use the standard bootstrap to generate alternative guide trees. Our study adopted SERES to generate alternative guide trees for the passive MSA alignment. First, we applied the SERES random walk on the input unaligned sequences to generate 100 resampled replicates. The reversal probability $\gamma = 0.5$. Then, we realigned the sampled sequences using MAFFT with default settings. We also investigated MSA estimation using ClustalW [88], and FSA[12].

Each SERES replicate utilized a total of $\lfloor \frac{k}{20} \rfloor$ anchors with anchor size of 5 bp and a minimum distance between neighboring anchors of 25 bp, where k is the length of the input alignment A. The anchor parameters were selected according to our additional experiments on the impact of the anchor parameter settings, where we found that the SERES-based MSA support estimation is robust to the selection of anchor numbers and anchor size. More details is included in Section 3.3.1.

The re-estimated alignments are used as alternative MSAs for the downstream steps of GUID-

ANCE1 and GUIDANCE2.

To further explore the impact of the parameter choices, we conducted additional SERES-based support estimation experiments with varied parameter settings. Each set of experiments manipulated one parameter setting. The parameters include the number of anchors, anchor length, or the method used to estimate the input MSA. Other than the selected manipulate parameter, all the parameters used default settings for SERES-based support estimation. The number of anchors was selected from the set $\{3, 5, 20, 50, 100\}$. Anchor length in bp was chosen from the set $\{3, 5, 10, 30, 50\}$. Three different methods were used for estimating an input MSA: ClustalW [88], MAFFT [64], and FSA [12].

3.2.4 Simulated Data

We simulated the datasets under 10 model conditions to comprehensively evaluate the performance of the SERES-based MSA support estimation with different levels of complexity of the evolutionary processes. Model conditions for 10 taxa and 50 taxa are named from A to E, representing increases in sequence diversity. Parameters of model conditions are shown in Table 3.1.

We used r8s version 1.7[123] to sample a random model tree under a birth-death process. INDELible version 1.03[39] takes the model tree topology and simulates nucleotide sequences and true alignment according to the guide trees under the General Time-Reversible (GTR) substitution model. We used the simulation parameters from Liu et al. 2012 study[80]. The parameters of the GTR substitution model[118] and the indel model comes from Liu et al. 2012 study[80]. The length of the simulated sequences was 1000bp. This simulation process was repeated twenty times independently to generate twenty replicates for each model condition. All the results were the average of twenty replicates. The summary statistics of the simulated dataset are shown in Table 3.1. Additional summary statistics of estimated alignment produced by ClustalW and FSA shows in Table 3.2.

To explore the impact of gap length distribution, our study also included 10-taxon model conditions which utilized the long gap length distribution from the study of Liu et al. [80] instead

Model condition	Number of taxa	Tree height	Insertion/deletion probability	NHD	Gappiness	True align length	Est align length	SP-FN	SP-FP
10.A	10	0.4	0.13	0.297	0.474	1965	1552	0.294	0.341
10.B	10	0.7	0.1	0.394	0.512	2165	1564	0.483	0.533
10.C	10	1	0.06	0.514	0.526	2163	1554	0.657	0.684
10.D	10	1.6	0.031	0.599	0.486	1874	1508	0.747	0.753
10.E	10	4.3	0.013	0.693	0.465	1849	1613	0.945	0.943
50.A	50	0.45	0.06	0.281	0.516	2044	1786	0.086	0.088
50.B	50	0.7	0.03	0.398	0.475	1936	1714	0.106	0.103
50.C	50	1	0.02	0.514	0.498	2048	1703	0.245	0.230
50.D	50	1.8	0.012	0.594	0.471	1945	1712	0.455	0.419
50.E	50	4.3	0.004	0.688	0.459	1890	2319	0.963	0.948

Table 3.1: Simulated datasets: parameter values and summary statistics. The simulation model condition parameters consist of the number of taxa, model tree height, and insertion/deletion probability. Each model condition corresponds to a distinct set of model parameter values. The following table columns list average summary statistics for each model condition ($n = 20$). "NHD" is the average normalized Hamming distance of a pair of aligned sequences in the true alignment. "Gappiness" is the percentage of true alignment cells which consists of indels. "True align length" is the length of the true alignment. "Est align length" is the length of the MAFFT-estimated alignment [64] which was provided as input to the support estimation methods. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the MAFFT-estimated alignment and vice versa, respectively.

of the medium gap length distribution that was used in the other datasets of our simulation study.

Parameter values and summary statistics for the long-gap-length model conditions are shown in Table 3.5.

Model condition	ClustalW		
	Est align length	SP-FN	SP-FP
10.A	1208.5	0.497	0.556
10.B	1186.2	0.624	0.684
10.C	1144.8	0.711	0.754
10.D	1105.7	0.756	0.786
10.E	1060.1	0.896	0.906

Model condition	FSA		
	Est align length	SP-FN	SP-FP
10.A	2289.3	0.334	0.124
10.B	3418.5	0.585	0.164
10.C	4506.6	0.729	0.211
10.D	5000.9	0.800	0.223
10.E	6657.1	0.907	0.531

Table 3.2: Medium-gap-length model conditions: estimated alignment statistics. The MSA support estimation problem requires an input MSA. Our study included ClustalW [88] and FSA [12] alignments to explore the impact of input alignment quality on downstream support estimation. The following table columns list average statistics for estimated alignments on each model condition ($n = 20$). "Est align length" is the estimated alignment length. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the estimated alignment and vice versa, respectively.

3.2.5 Empirical data

We used the empirical benchmarks from the Comparative RNA Web (CRW) site database [14] to further test the performance of the SERES-based MSA support estimation. The CRW dataset contains alignments of ribosomal RNA sequences, which covers a wide range of dataset size and sequence divergence. The reference alignments included in the CRW dataset are produced by the combination of automatic alignment software and intensive manual correction with information of the secondary structure. The quality of the reference alignments include in the CRW are very high. This benchmark dataset has been widely used in the evaluation and comparison of MSA approaches. We selected 11 datasets from the CRW dataset with at most 250 sequences, including primary 16S rRNA, primary 23S rRNA, primary intron and seed alignments. Preprocess of the empirical datasets filtered sequences with greater or equal to 99% missing data. The summary statistics of the empirical datasets are shown in Table 3.3.

Dataset	Number of taxa	NHD	Gappiness	Ref align length	Est align length	SP-FN	SP-FP
IGIA	110	0.606	0.915	10368	6675	0.734	0.784
IGIB	202	0.579	0.910	10633	7379	0.825	0.864
IGIC2	32	0.533	0.700	4243	3514	0.689	0.715
IGID	21	0.719	0.782	5061	3023	0.874	0.904
IGIE	249	0.451	0.838	2751	2775	0.393	0.376
IGIIA	174	0.668	0.814	6406	7005	0.816	0.800
PA23	142	0.293	0.267	3991	3552	0.078	0.077
PE23	117	0.300	0.612	9436	10083	0.202	0.213
PM23	102	0.361	0.797	10999	8803	0.262	0.288
SA16	132	0.212	0.205	1866	1673	0.031	0.028
SA23	144	0.304	0.460	4048	3678	0.077	0.081

Table 3.3: Empirical dataset summary statistics. The empirical study made use of reference alignments ("Ref align") from the CRW database [14]. The column description is identical to Table 3.1.

3.2.6 Performance Measure

We evaluate the performance of the MSA support estimation approaches by the receiver operating characteristic (ROC) curves, precision-recall (PR) curves, and area under ROC curves (ROC-AUC) and PR curves (PR-AUC). We focus on this application because the multiple sequence alignment problem is considered to be a classical problem in computational biology and bioinformatics and

MSAs are used as inputs for a variety of important computational problems throughout computational biology and bioinformatics, such as phylogenetics analysis, proteomics, comparative genomics, etc. It is well known that MSA quality has a major impact on downstream analysis [71, 79, 80].

The MSA support estimation produces support value for each residue pair in estimated alignment. Both the ROC curve and the PR curve are produced with thresholds ranging from 0 to 1. For each threshold, by comparing the support value with the threshold and whether the homology appears in the true alignment, we divided the homologies in the estimated alignment into four parts. True positive (TP) represents the number of residue pairs with support values greater than or equal threshold and appear in the true alignment. False positive (FP) represents the number of residue pairs with support value greater or equal to the threshold but not in the true alignment. True negative (TN) represents the number of residue pairs with support values less than the threshold and not true alignment. False negative (FN) represents the number of residue pairs with support value less than the threshold but appear in the true alignment. The ROC curve, PR curve, ROC-AUC, and PR-AUC were calculated by the scikit-learn Python library[136].

3.3 Result

3.3.1 Simulation study

For all the model conditions, the SERES-based support estimation method showed better PR-AUC and ROC-AUC performances than the GUIDANCE1 and GUIDANCE2, two state-of-the-art methods. The results of the simulation study are shown in Table 3.4. All the performance improvements achieved by the SERES-based support estimation were statistically significant. The p-values were calculated by the corrected pairwise t-test, or DeLong test [25], respectively, over all replicate datasets on each model condition. Sequence divergence showed a strong impact on the performance improvement achieved by the SERES-based method, although the SERES-based support estimation method consistently outperformed the GUIDANCE1 and GUIDANCE2 using the standard bootstrap resampling approach on all model conditions despite different dataset sizes

and sequence divergence. The performance improvement increased as the sequence divergence increased. For the 10-taxon and 50-taxon model conditions, the SERES-based method achieved at most 3% improvement on the datasets with the least sequence divergence. The SERES-based method improved the PR-AUC performance by 28% on the most divergent dataset in the simulation study. Datasets with the highest sequence divergence are the most challenging datasets.

All methods showed degraded PR-AUC as the sequence divergence increased, which is consistent with the previous GUIDANCE2 study [126]. While the SERES-based method's PR-AUC performance degraded slower than the original GUIDANCE1 and GUIDANCE2 using the bootstrap resampling approach. The performance improvement achieved by the SERES+GUIDANCE1 method over GUIDANCE1 was generally greater and statistically more significant than the comparison between the SERES+GUIDANCE2 method and GUIDANCE2. On all model conditions, the GUIDANCE2 produced better PR-AUC and ROC-AUC performance than the GUIDANCE1.

Another observation was that the PR-AUC differences were generally larger than the ROC-AUC differences, especially on model conditions with higher sequence divergence.

The performance comparisons on the long-gap-length model conditions were mostly the same as the medium-gap-length model conditions. The PR-AUC and ROC-AUC performances of the two methods are shown in Table 3.5. Similar to previous findings, SERES+GUIDANCE2 consistently produced significant improvements on both PR-AUC and ROC-AUC in comparison to GUIDANCE2. The statistical significance was calculated by the corrected pairwise t-test or DeLong test [25] across all 20 replicates for each model condition, respectively. Furthermore, as sequence divergence increased, the PR-AUC improvement that SERES+GUIDANCE2 produced relative to GUIDANCE2 tended to improve. When comparing medium-gap-length with corresponding long-gap-length model condition pairs, for example, 10.A and 10.long.A, the PR-AUC improvement of SERES+GUIDANCE2 over GUIDANCE2 was similar between the two types of gap length distribution. The differences were less than 1%. A similar finding was observed for ROC-AUC measurements. An exception is the comparison between 10.D and 10.long.D model conditions. For the 10.long.D model condition, the PR-AUC performance improvement achieved

Model condition	PR-AUC(%) GUIDANCE1	PR-AUC(%) SERES+GUIDANCE1	Pairwise t-test corrected q-value	ROC-AUC(%) GUIDANCE1	ROC-AUC(%) SERES+GUIDANCE1	DeLong et al. test corrected q-value
10.A	88.74	91.17	5.4×10^{-7}	80.22	85.57	$< 10^{-10}$
10.B	82.21	86.26	1.5×10^{-6}	84.83	88.66	$< 10^{-10}$
10.C	76.23	83.49	1.9×10^{-4}	86.98	91.23	$< 10^{-10}$
10.D	74.65	85.81	1.9×10^{-4}	88.55	93.72	$< 10^{-10}$
10.E	42.61	59.20	3.1×10^{-4}	82.24	87.40	$< 10^{-10}$
50.A	98.22	98.92	5.3×10^{-10}	83.09	90.64	$< 10^{-10}$
50.B	97.84	98.69	2.8×10^{-9}	82.85	90.39	$< 10^{-10}$
50.C	95.08	96.80	5.6×10^{-8}	85.54	90.64	$< 10^{-10}$
50.D	90.79	95.75	5.3×10^{-6}	88.89	94.56	$< 10^{-10}$
50.E	62.47	79.14	8.0×10^{-10}	91.02	93.23	$< 10^{-10}$

Model condition	PR-AUC(%) GUIDANCE2	PR-AUC(%) SERES+GUIDANCE2	Pairwise t-test corrected q-value	ROC-AUC(%) GUIDANCE2	ROC-AUC(%) SERES+GUIDANCE2	DeLong et al. test corrected q-value
10.A	92.55	93.33	7.4×10^{-6}	87.17	88.34	$< 10^{-10}$
10.B	88.08	89.31	8.4×10^{-4}	89.45	90.56	$< 10^{-10}$
10.C	84.28	86.86	3.1×10^{-4}	91.36	92.88	$< 10^{-10}$
10.D	86.03	88.75	1.9×10^{-4}	93.34	94.69	$< 10^{-10}$
10.E	51.17	62.30	1.3×10^{-3}	86.00	88.28	$< 10^{-10}$
50.A	98.98	99.14	5.3×10^{-6}	91.17	92.50	$< 10^{-10}$
50.B	98.79	98.96	1.5×10^{-6}	91.24	92.44	$< 10^{-10}$
50.C	96.86	97.45	3.2×10^{-7}	90.81	92.31	$< 10^{-10}$
50.D	94.04	96.23	1.5×10^{-5}	92.67	95.09	$< 10^{-10}$
50.E	72.61	81.47	1.5×10^{-8}	92.94	94.22	$< 10^{-10}$

Table 3.4: Support estimation method performance on simulated datasets. Results are shown for simulated datasets. The top rows show AUC comparisons of GUIDANCE1 ("GUIDANCE1") vs. SERES combined with parametric techniques from GUIDANCE1 ("SERES+GUIDANCE1"), Results AUC comparisons of GUIDANCE2 ("GUIDANCE2") vs. SERES combined with parametric techniques from GUIDANCE2 ("SERES+GUIDANCE2"); the best AUC is shown in bold. Corrected q-values are reported ($n = 20$) and all were significant ($\alpha = 0.05$).

by the SERES-based method was larger than that seen in the 10.D model condition.

We also conducted additional experiments to study the impact of the MSA estimation method choices. The experiments were conducted by SERES+GUIDANCE2 with alternative MSA methods for estimating the input MSA. A direct performance comparison is shown in Table 3.6. The three MSA methods used in our study returned input alignments with different levels of quality. Compared to the other ClustalW and MAFFT, input MSA estimated by FSA had a lower average SP-FP and the best or close to best average SP-FN measurement. Detailed summary statistics are shown in Table 3.2. Downstream support estimation PR-AUC reflected input alignment quality. In the previous experiments, we found that the PR-AUC performance decreased as sequence divergence

Model condition	PR-AUC(%)		
	GUIDANCE2	SERES+ GUIDANCE2	Pairwise t-test corrected q-value
Model condition	ROC-AUC(%)		
	GUIDANCE2	SERES+ GUIDANCE2	DeLong et al. test corrected q-value
10.long.A	89.99	90.99	$< 10^{-10}$
10.long.B	91.84	93.02	$< 10^{-10}$
10.long.C	93.14	94.59	$< 10^{-10}$
10.long.D	93.89	96.13	$< 10^{-10}$
10.long.E	92.62	94.38	$< 10^{-10}$

Table 3.5: Support estimation method performance on long-gap-length model conditions. The performance of GUIDANCE2 and SERES+GUIDANCE2 is compared across model conditions 10.long.A through 10.long.E (named in order of generally increasing sequence divergence). Aggregate PR-AUC and ROC-AUC are reported across all replicate datasets in a model condition ($n = 20$), and the best AUC for each model condition is shown in bold. Statistical significance of PR-AUC or ROC-AUC differences was assessed using a one-tailed pairwise t-test or DeLong test [25] test, respectively, and multiple test correction was performed using the method of [8]. Corrected q-values are reported ($n = 20$) and all were significant ($\alpha = 0.05$).

increased. When using FSA-estimated alignments as input, the PR-AUC performance reduction was smaller than that of MAFFT and ClustalW. The PR-AUC and ROC-AUC performance improvements obtained by SERES+GUIDANCE2 over GUIDANCE2 were robust to the quality of input alignment or choice of MSA estimation method. The SERES+GUIDANCE2 outperformed GUIDANCE2 on both PR-AUC and ROC-AUC no matter annotating more accurate input alignments, such as FSA-estimated alignments, or less accurate input alignments, such as alignments estimated by MAFFT or ClustalW.

We explored different anchor parameter settings. The method performances using differing choices for anchor length and numbers of anchors are shown in Figures 3.2 and Figure 3.3, respectively. For different anchor lengths used for the SERES resampling, the SERES+GUIDANCE2 produced roughly similar PR-AUC and ROC-AUC performance. The average ROC-AUC difference for different choices of anchor length was less than 0.01 for all model conditions. The largest PR-AUC performance difference was 0.058 on the 10.E model condition. This PR-AUC difference is considered very small compared to the PR-AUC improvement obtained by SERES+GUIDANCE2 over GUIDANCE2, which was 0.28 on the 10.E model condition. A similar outcome was seen in experiments involving different choices for the number of anchors, except in the most divergent

Model condition	PR-AUC(%)					
	ClustalW			FSA		
	GUIDANCE2	SERES+GUIDANCE2	Pairwise t-test corrected q-value	GUIDANCE2	SERES+GUIDANCE2	Pairwise t-test corrected q-value
10.A	95.37	95.78	2.8×10^{-3}	96.36	96.55	8.6×10^{-3}
10.B	92.30	92.95	8.2×10^{-4}	95.40	95.87	4.9×10^{-3}
10.C	89.36	91.23	1.7×10^{-4}	95.32	96.06	2.7×10^{-3}
10.D	88.53	90.45	8.8×10^{-5}	96.21	96.87	2.1×10^{-3}
10.E	73.96	76.50	8.2×10^{-4}	90.23	92.51	8.6×10^{-3}

Model condition	ROC-AUC(%)					
	ClustalW			FSA		
	GUIDANCE2	SERES+GUIDANCE2	DeLong et al. test corrected q-value	GUIDANCE2	SERES+GUIDANCE2	DeLong et al. test corrected q-value
10.A	96.99	97.23	$< 10^{-10}$	80.85	81.61	$< 10^{-10}$
10.B	96.64	96.94	$< 10^{-10}$	81.31	82.89	$< 10^{-10}$
10.C	96.27	96.88	$< 10^{-10}$	84.48	86.56	$< 10^{-10}$
10.D	95.78	96.65	$< 10^{-10}$	88.63	90.37	$< 10^{-10}$
10.E	89.84	90.80	$< 10^{-10}$	89.10	90.83	$< 10^{-10}$

Table 3.6: SERES+GUIDANCE2 performance using alternative methods for estimating an input MSA. Input MSAs in these experiments were estimated using either ClustalW [88] or FSA [12]. (MAFFT was used to estimate input MSAs throughout the rest of our study.) Results are shown for model conditions 10.A through 10.E (named in order of generally increasing sequence divergence). Otherwise, table layout and description are identical to Table 3.5.

10.E model condition, where an intermediate number of anchors yielded the best PR-AUC.

We compared the runtime of the GUIDANCE1 and GUIDANCE2 with or without the SERES resampling and re-estimation, the SERES-based method required slightly more runtime on all model conditions, usually a few minutes for the entire run. On average, all methods in the simulation study completed analysis of each replicate dataset in less than half an hour and with less than 1 GiB of main memory usage.

The average runtime of SERES+GUIDANCE1 was longer than GUIDANCE1 alone by 1 minute and 5 minutes on the 10-taxon and 50-taxon model conditions, respectively. The average runtime of SERES+GUIDANCE2 was longer than GUIDANCE2 alone by at most 1.4 minutes and 6.5 minutes, respectively. The runtime shows in Figure 3.4.

For the average memory usage on 10-taxon and 50-taxon model conditions, SERES+ GUIDANCE1 used 0.016 GiB to 0.610 GiB more than GUIDANCE1 alone. A similar outcome was observed when comparing SERES+GUIDANCE2 and GUIDANCE2, where SERES+GUIDANCE2 used 0.034 GiB and 0.871 GiB more memories than GUIDANCE2, respectively. The memory

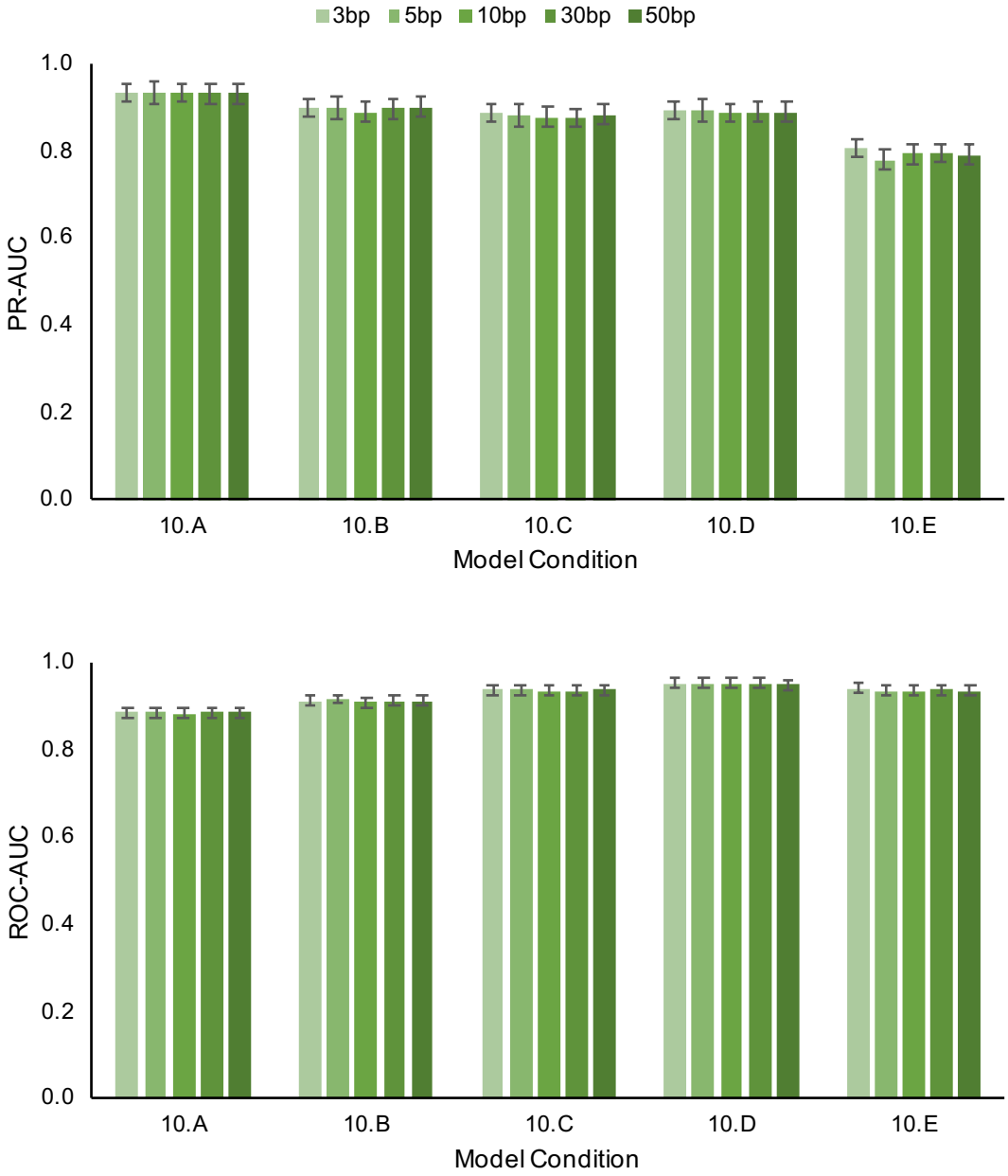


Figure 3.2: SERES+GUIDANCE2 performance using different choices for anchor length. Results are shown for five 10-taxon medium-gap-length model conditions (named 10.A through 10.E in order of generally increasing sequence divergence). We evaluated the performance of SERES+GUIDANCE2 where anchor length in bp was either 3, 5, 10, 30, or 50. We calculated each method's precision-recall (PR) and receiver operating characteristic (ROC) curves. Performance is evaluated based upon aggregate area under curve (AUC) across all replicates for a model condition ($n = 20$).

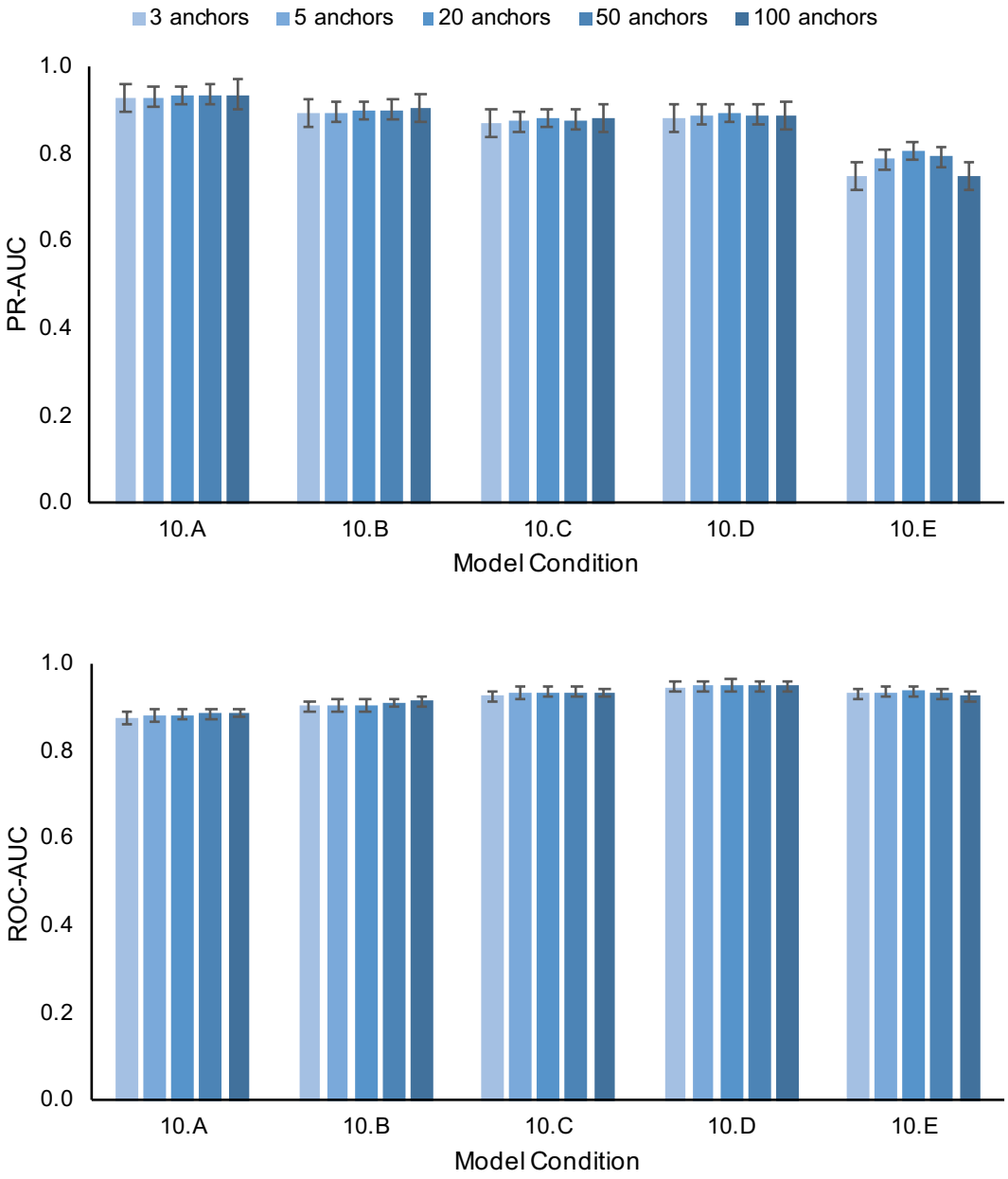


Figure 3.3: SERES+GUIDANCE2 performance using different choices for the number of anchors. We evaluated the performance of SERES+GUIDANCE2 where the number of anchors used was either 3, 5, 20, 50, or 100. Otherwise, figure layout and description are identical to Figure 3.2.

usage is shown in Figure 3.5.

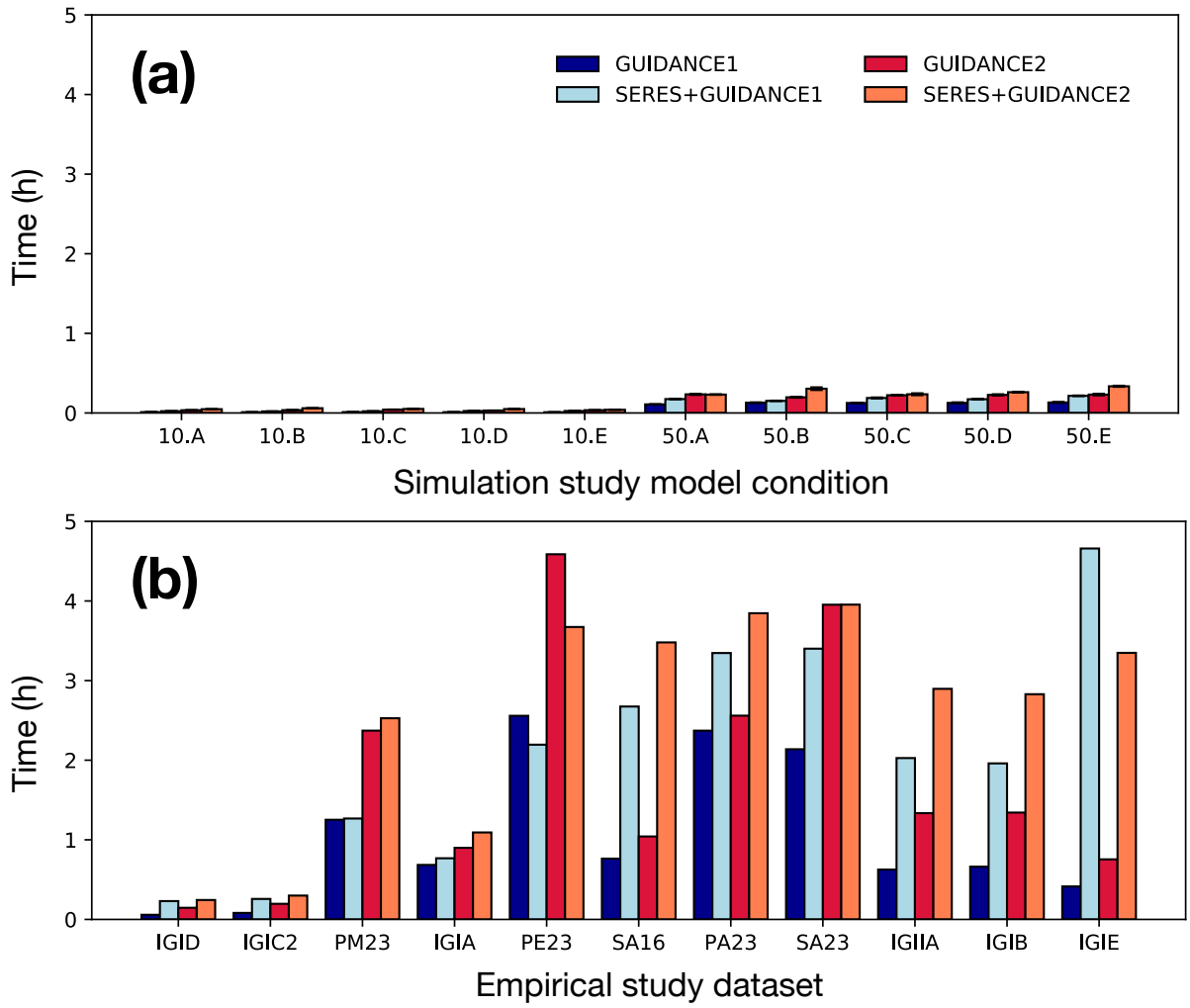


Figure 3.4: Runtime comparison of methods under study. (a) For each method, average runtime (h) across all replicates in each simulation study model condition is reported ($n = 20$); standard error bars are also shown. The 10-taxon model conditions 10.A through 10.E are shown in order from left to right, followed by the 50-taxon model conditions similarly. (b) Method runtimes are shown for each empirical study dataset. Datasets are arranged from left to right in order of increasing dataset size as measured by number of taxa.

3.3.2 Empirical study

The SERES-based support estimation methods produced better performance on all empirical datasets except for the IGIC2 dataset, where the GUIDANCE2 performed better than the SERES+GUIDANCE2 method by 1.17% on PR-AUC and 2.12% on ROC-AUC.

Consistent with the simulation study, the SERES-based method achieved larger PR-AUC performance improvements on datasets with higher sequence divergence. The PR-AUC improvements

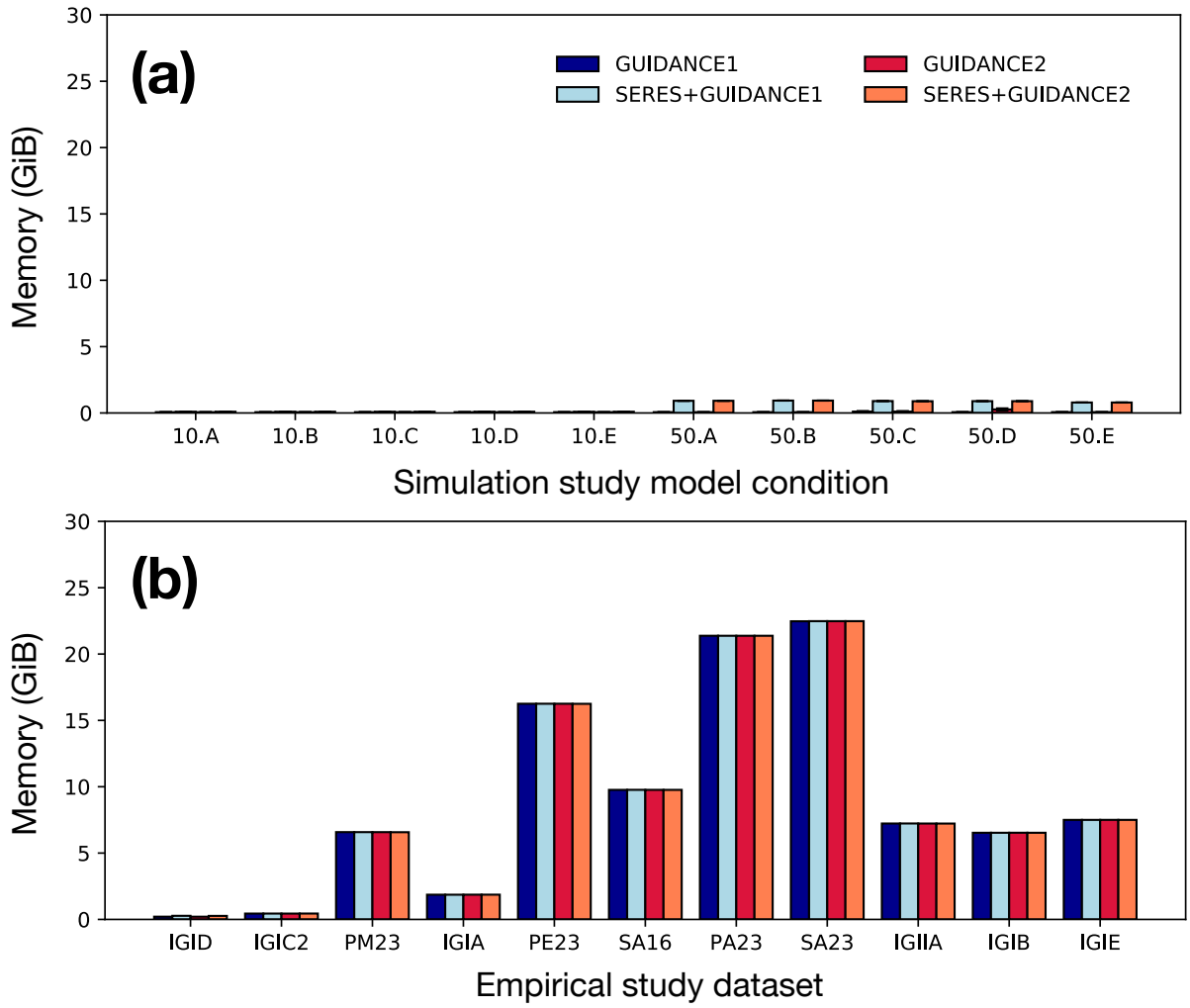


Figure 3.5: Memory usage comparison of methods under study. Memory usage is shown in GiB. Otherwise, figure layout and description are identical to Supplementary Figure 3.4.

were less than 1% on seed and primary non-intronic datasets, which were datasets with less ANHD and gappiness, refer to Table 3.3. For the intronic datasets, datasets with higher sequence divergence, the PR-AUC improvements were as much as 13.87%.

Performance improvements of the SERES+GUIDANCE1 method over GUIDANCE1 were relatively greater than that of the comparison between the SERES+GUIDANCE2 method and GUIDANCE2, which is also consistent with the simulation study.

Finally, GUIDANCE2 consistently generated better performance than GUIDANCE1 on both PR-AUC or ROC-AUC.

The runtime comparison between SERES+GUIDANCE2 and GUIDANCE2 showed larger

Model condition	PR-AUC(%)		ROC-AUC(%)	
	GUID-ANCE1	SERES+ GUID-ANCE1	GUID-ANCE1	SERES+ GUID-ANCE1
IGIA	62.67	69.28	89.50	91.62
IGIB	73.60	87.47	94.49	97.39
IGIC2	72.67	75.36	82.25	83.87
IGID	63.74	76.30	95.10	96.73
IGIE	93.56	95.42	90.08	93.30
IGIIA	73.03	83.06	86.49	96.45
PA23	98.54	99.41	82.59	93.63
PE23	98.44	99.27	94.75	97.41
PM23	97.53	98.48	94.20	96.44
SA16	99.72	99.86	91.07	95.57
SA23	98.35	99.24	81.76	92.18

Model condition	PR-AUC(%)		ROC-AUC(%)	
	GUID-ANCE2	SERES+ GUID-ANCE2	GUID-ANCE2	SERES+ GUID-ANCE2
IGIA	67.4	68.49	91.38	91.94
IGIB	80.66	86.72	96.47	97.38
IGIC2	74.44	73.27	84.63	82.51
IGID	75.15	78.38	96.44	97.09
IGIE	94.6	95.44	91.84	93.49
IGIIA	78.16	85.09	94.50	96.82
PA23	99.24	99.53	91.48	94.88
PE23	99.07	99.34	96.72	97.63
PM23	98.68	98.85	96.93	97.28
SA16	99.88	99.91	96.22	97.22
SA23	99.04	99.33	89.93	93.18

Table 3.7: Empirical study results. Results are shown for empirical datasets. For each dataset and pairwise method comparison. Table layout, and table description are otherwise identical to Table 3.4.

differences in the empirical datasets compared to the simulation study. SERES+GUIDANCE2 used at most 2.6 hours on the largest empirical datasets, which have 100-200 taxa. The variance of the runtime difference between the two methods was also larger than that of the simulation study. GUIDANCE2's main memory usage was not consistently better than SERES+GUIDANCE2 on the empirical datasets. These two methods had comparable memory usage across the empirical datasets. The maximum difference between these two methods was 0.06 GiB. Similar runtime and memory usage comparisons were observed for SERES+GUIDANCE1 and GUIDANCE1, with the former having a maximum overhead relative to the latter of 4.2 hours and 0.07 GiB.

3.4 Discussion

In both the simulation and the empirical studies, the utilization of the SERES resampling and re-estimation promoted the MSA support estimation performance. According to the experiment results, the performance improvement greatly benefited from the SERES resampling approach, which can produce many distinct replicates and reserve the intra-sequence dependence.

Under all model conditions, the support estimation produced by GUIDANCE1 and GUIDANCE2 with the SERES resampling and re-estimation showed significant improvements in the PR-AUC and ROC-AUC compared to the pipelines without the SERES resampling process. The main difference for the comparison is the resampling approach, SERES, versus the standard bootstrap. The results of the simulation study and the empirical study indicate that the SERES resampling approach outperformed the standard bootstrap resampling method in the application of the MSA support estimation. The SERSE resampling approach was designed to preserve the intra-sequence dependence caused by the insertion and deletion processes and relax the assumption made by the bootstrap resampling method that all the sites are independent and identically distributed.

The experiment results showed that the SERES+GUIDANCE1 method achieved greater PR-AUC and ROC-AUC performance improvements over the GUIDANCE1 method than the comparison between the SERES+GUIDANCE2 and GUIDANCE2. One reason is that in terms of AUC performance, the GUIDANCE2 outperformed the GUIDANCE1. We used the SERES resampling approach together with the support estimation framework of GUIDANCE1 or GUIDANCE2. GUIDANCE2 already yielded very high PR-AUC and ROC-AUC on the simulation datasets and empirical datasets, and it is hard for the SERES resampling approach to make large performance improvements upon that.

The experiments on the empirical datasets generated similar results as the simulation study. The non-intronic datasets have low ANHD and gappiness, which means they have low sequence divergence. The intronic datasets have higher sequence divergence. Therefore, the methods' performances on the non-intronic datasets were similar to low divergence model conditions, and performances on the intronic datasets were similar to high divergence model conditions. The

SERES-based support estimation methods consistently showed better performance than either GUIDANCE1 or GUIDANCE2 alone for all the empirical datasets. The SERES-based methods achieved larger performance improvements on the PR-AUC and ROC-AUC as the dataset became more divergent and challenging to align. For small datasets with low sequence divergence, the SERES resampling approach did not greatly improve the performance as on those challenging datasets. One example is the IGIC2 datasets, where the SERES+GUIDANCE2 did not outperform GUIDANCE2. IGIC2 was a small dataset, which is about an order of magnitude smaller than other datasets. This dataset also has a lower ANHD and gappiness than the other intronic datasets. Another small intronic dataset, IGID, has higher ANHD and gappiness compared to the IGIC2 dataset. When compared to the GUIDANCE2, the SERES-based support estimation yielded a relatively small performance improvement of about 3.2%. Higher sequence divergence usually means more challenges for statistical inferences and more inference errors. While the performance of the SERES-based methods degraded much slower than the corresponding non-SERES methods as the sequence divergence increased. The largest performance improvement was seen in the most divergent model conditions and empirical datasets.

The combination of GUIDANCE1 and GUIDANCE2 with SERES-based resampling and re-estimation usually leads to an increase in computational runtime in our study. The computational runtime increased by a few minutes for the 10-taxon and 50-taxon simulated datasets and a few hours for the larger empirical datasets with around 100-200 taxa. In the simulation study, the SERES-based methods also required more memory than GUIDANCE1 and GUIDANCE2 alone. The gap between the SERES-based methods and GUIDANCE1 and GUIDANCE2 decreased on the larger empirical datasets with a few hundred taxa. Compared to GUIDANCE1 and GUIDANCE2, the SERES-based methods include an additional MSA re-estimation step, occurring after SERES random walk resampling. This additional step is likely the reason for the increased computational cost.

Finally, we noticed that the combination of two different types of methods resulted in better performance than either of these methods alone. This finding indicated that the resampling techniques

are orthogonal to parametric alternatives, which is consistent with previous studies [110, 126].

3.5 Conclusions

In this study, we introduced a new non-parametric/semi-parametric resampling approach, SERES, for the resampling of biomolecular sequence data. The SERES resampling approach can produce many distinct resampling replicates while reserving the sequential dependence during the resampling process. We applied the SERES resampling approach to a classical problem in computational biology and bioinformatics, the MSA support estimation problem. We tested the SERES-based support estimation on both simulated and empirical datasets. The SERES-based method performed comparably or better than the state-of-the-art approaches.

For future research, there are several directions. Create a purely non-parametric resampling approach based on the SERES method. The SERES algorithm in this study requires the estimation of anchors to ensure the synchronization of the random walk on the unaligned input sequences. Anchor estimation is a semi-parametric procedure that makes use of progressive multiple sequence alignment algorithms. Therefore, the SERES resampling approach on the unaligned sequences is also considered a semi-parametric approach. Non-parametric resampling procedure could be obtained by replacing the anchor estimation in the SERES resampling process.

The SERES resampling approach could potentially be extended to perform the MSA estimation since alternative homologies were produced during the SERES resampling and re-estimation process. Many other problems can also make use of the SERES resampling approach, such as protein structure prediction, read mapping, and assembly. Non-parametric resampling for support estimation is widely used throughout science and engineering, and SERES resampling can also be applied in research areas outside of computational biology and bioinformatics.

CHAPTER 4

APPLICATION OF SERES RESAMPLING APPROACH TO ALIGNED SEQUENCES: PHYLOGENETIC HMM INFERENCE AND LEARNING

4.1 Introduction

Besides estimation of the confidence intervals, the statistical resampling methods are also used to produce perturbations for statistical inference and learning to improve the accuracy [13]. In this study, we applied the SERES resampling algorithm on the aligned sequences for another classical problem in computational biology and bioinformatics, recombination-aware local genealogical inference.

We selected the recombination breakpoint detection problem mainly because the recombination event is one of those evolutionary processes that creates sequence dependence. The sites involved in the recombination share the same evolutionary history. The purpose of the recombination breakpoints detection is to find the breakpoints where two neighboring regions have different local genealogy topologies. Preservation and identification of intra-sequence dependence play crucial roles in the recombination breakpoint detection problem. As introduced in the previous chapter, the SERES resampling algorithm preserves the sequence dependence within the resampled replicates. Thus, the SERES resampling approach has the ability to retain intra-sequence dependence caused by the historical recombination events during its resampling process, which makes it a perfect fit for this problem.

Traditionally, the phylogenetic tree is reconstructed under the assumption that all input MSA sites share the same evolutionary history. However, due to the existence of recombination, this assumption does not fit all conditions. Recombination is the process by which genetic materials are transferred between different organisms. This process is an essential source of genetic diversity; for example, disease-causing bacteria acquire resistance to antibiotics. This process also leads to a mixture of genetic materials, where the local genealogies of the affected region are changed and

are inconsistent with the evolutionary history of the other regions.

The Hidden Markov model was first adopted to describe evolutionary history by the study of Hein [51], where the hidden states were used to represent the topologies of evolutionary history, and the recombination event was interpreted as a transition between different states. McGuire extended the ideas of Hein's study to detect the local topologies changes caused by the recombination event [92, 93]. The HMM model was combined into a likelihood framework where the maximum likelihood method was used as part of Bayesian inference and the Markov chain was used to assign a prior probability to the sequence of topologies along a sequence alignment.

There are many HMM-based methods for local genealogical inference [59, 150, 91, 81]. In this study, We mainly focus on the recHMM [150] for the following reasons. RecHMM identifies local genealogy by applying heuristic searches on the space of all possible partitions. This algorithm is powerful in detecting local genealogy changes, especially when the regions that involve in recombination are long and the dataset size is large. It reveals the most likely recombination breakpoint locations with high accuracy and fewer requirements on parameter settings such as window schemes. Another reason is that recHMM takes aligned sequences as input to annotating mosaic genome structures, making it possible to combine the SERES resampling approach.

The recHMM algorithm utilizes a statistical model that combines a finite-sites substitution model and a phylo-HMM to capture intra-sequence dependence due to recombination. An EM-based approach is used to learn recHMM model parameters. [150] also applied a structural EM heuristic [40] to automatically learn the set of local gene trees represented by the recHMM's states. In the simulation study, we compared the method performance of the recHMM method and the combined method of the recHMM with the SERES resampling approach on local genealogies inference. We also evaluated the combined method performance on an empirical HIV genome sequence dataset.

4.2 Methods

4.2.1 Standalone recHMM analysis

We first ran the recHMM alone to get a baseline performance on the local genealogy inference in the simulation study.

Users need to assign a state size ϕ for the HMM model of recHMM. In our simulation study, we ran recHMM with the default setting, $\phi = 3$. For recHMM, gene trees with either different topologies or branch lengths are considered different trees. In the structural EM used by [150], HMM states are distinguished by both gene tree topologies and branch lengths. In Westesson's study, recHMM employed ψ independent optimization trials to avoid local optima and selected the best trial under the maximum likelihood criterion. In our simulation study, we applied the same strategy as in the original study of recHMM, where recHMM ran with $\psi = 100$ independent optimization trials, and selected the best trail. We used the same posterior decoding algorithm as used in the original study to perform statistical inference of local phylogenies [113]. Let G be the set of all possible unrooted tree topologies on n taxa. The input alignment A contains n sequences, each sequence comes from one taxon. The length of the input alignment is k, which means there are k sites in A. The input alignment is assumed to contain recombined regions, where the corresponding local genealogies are different from the other regions due to the historical recombinations [52]. For each site a_i, where $1 \leq i \leq k$, in the input alignment A, recHMM outputs the conditional probability for the gene tree $g \in G$ correspond to each hidden state conditional on all sites in A and the fitted HMM model.

4.2.2 The SERES+recHMM pipeline

To test the SERES resampling approach on the local genealogy inference problem, we combined recHMM with the SERES random walk.

First, we ran SERES resampling on the input alignment A with a default reversal probability $\gamma = 0.005$. We also conducted additional experiments with alternative reversal probability $\gamma \in$

$\{0, 0.01, 0.1\}$. We used the SERES random walk to generate ten resampled replicates for each dataset in our study.

Then, we ran recHMM on each resampled replicate. To get a fair comparison, we restricted the number of independent learning trials ψ used in the SERES-based pipeline. We ran recHMM on each SERES replicate with $\psi = 10$. So for each dataset, the total number of independent learning trials used in the SERES-based pipeline was equal to the number of independent learning trials used by the standalone recHMM method. The recHMM generated posterior probabilities for each site over ϕ gene tree topologies.

For each site, the posterior probability distributions were aggregated across all the replicates in which the site appeared. The aggregated distribution was then normalized to obtain a valid probability distribution.

4.2.3 Simulated datasets

We used ms [57] to simulate the gene trees under the coalescent-with-recombination model with either 4, 5, or 6 taxa. The recombination rate of the simulation is $\rho \in \{0.5, 1.0, 2.0\}$ and the total sequence length is 1 Kb per replicate. For each gene tree, we used Seq-Gen [48] to simulate the sequences under the Jukes-Cantor substitution model [63]. The substitution rate is set to be $\theta \in \{0.5, 1.0, 2.0\}$. The simulation procedures described above were repeated 30 times independently. We got 30 replicate datasets for each model condition. We simulated datasets that covers a wide range of recombination rates and mutation rates over 4, 5, or 6 taxa. The number of taxa are chosen to test the scalability of the SERES resampling algorithm on the HMM inference. The number of unique unrooted tree topologies for 4 taxa is 3. There are 15 unique unrooted tree topologies for 5 taxa, and this number increases to 105 for 6 taxa.

Summary statistics for the simulated datasets are shown in Table 4.1. We used the Robinson-Foulds distance [116] to measure the topological accuracy of inferred gene trees compared with the ground truth. The Robinson-Foulds distance is the proportion of bipartitions that appear in an inferred gene tree but not in the true gene tree or vice versa.

Number of taxa	Recombination rate ρ	Mutation rate θ	# gene trees Avg	# gene trees SE	ANHD Avg	ANHD SE
4	0.5	0.5	3.4	1.4	0.359	0.080
4	0.5	1	3.0	1.2	0.498	0.101
4	0.5	2	3.0	1.2	0.620	0.086
4	1	0.5	4.1	1.7	0.344	0.102
4	1	1	4.1	1.7	0.495	0.105
4	1	2	4.1	1.7	0.625	0.079
4	2	0.5	5.6	2.4	0.321	0.091
4	2	1	5.6	2.4	0.462	0.103
4	2	2	5.6	2.4	0.581	0.094
5	0.5	0.5	3.100	1.165	0.331	0.093
5	0.5	1	3.200	1.137	0.464	0.083
5	0.5	2	3.200	1.137	0.585	0.068
5	1	0.5	4.700	1.917	0.357	0.085
5	1	1	4.700	1.917	0.492	0.063
5	1	2	4.700	1.917	0.608	0.044
5	2	0.5	6.367	2.834	0.331	0.078
5	2	1	6.367	2.834	0.467	0.083
5	2	2	6.367	2.834	0.587	0.062
6	0.5	0.5	3.167	1.293	0.336	0.088
6	0.5	1	3.300	1.159	0.439	0.077
6	0.5	2	3.300	1.159	0.554	0.072
6	1	0.5	4.667	2.134	0.304	0.082
6	1	1	4.667	2.134	0.435	0.079
6	1	2	4.667	2.134	0.554	0.064
6	2	0.5	6.100	2.587	0.312	0.079
6	2	1	6.100	2.587	0.448	0.085
6	2	2	6.100	2.587	0.565	0.068

Table 4.1: Simulated dataset statistics. The number of true gene trees and average normalized Hamming distance ("ANHD") are reported for simulated datasets from the simulation study; average ("Avg") and standard error ("SE") are shown for all experimental replicates from each model condition ($n = 30$).

4.2.4 Empirical datasets

We also re-analyzed an HIV dataset from the study of [150]. The dataset consisted of Indian samples that were originally studied by [84]. The dataset was sub-sampled to include four sequences, including the putatively recombinant sequence 95IN21301.

4.3 Results

4.3.1 Simulation study

We calculated the correlation between the inferred posterior probability of a gene tree topology g and the topological accuracy of g to measure the performance of the standalone method and the SERES-based method. Table 4.2 shows the correlation results of the 4 taxa model conditions. For all 4-taxon model conditions, the posterior probabilities produced by the SERES-based method

are consistently better correlated with topological accuracy than the probabilities produced by the standalone recHMM. The absolute correlation improvement was at least 0.203, where the recombination rate $\rho = 2$ and the mutation rate $\theta = 1$. The largest correlation improvement was 0.305, where the recombination rate $\rho = 1$ and the mutation rate $\theta = 0.5$. This correlation improvement achieved by the SERES-based method is robust for all model conditions with various mutation rates and recombination rates.

Then we compared the distributions of the posterior probability inferred by the standalone recHMM and the SERES+recHMM method for the 4-taxon model conditions. Figure 4.1 shows the distribution of the posterior probabilities inferred by recHMM alone. Figure 4.2 shows the distribution of the posterior probabilities inferred by the SERES+recHMM method. In Figure 4.1 and Figure 4.2, "true class" refers to the true gene tree topologies used as guide trees for the sequence simulation, and all the other gene tree topologies are labeled as "false class".

Ideally, the posterior probability of the true gene tree topologies should be 100%, and the posterior probability of the other gene tree topologies should be 0%. The right-most blue bar is the highest in Figure 4.2 for all the model conditions, which means the majority of the posterior probabilities inferred by the SERES+recHMM for the true gene tree topologies were higher than 90%. However, in Figure 4.1, the right-most bar and the left-most bar are almost identical in height. This means almost half of the posterior probabilities inferred by the standalone recHMM are less than 10%.

An opposite trend was observed for the false class. The orange bar representing the highest posterior probability inferences, which are 90% posterior probability or greater, was the second-highest in Figure 4.1. In contrast, SERES+recHMM consistently returned fewer inferences in the top decile of the posterior probability range. The SERES+recHMM-inferred posterior distribution for the false class of per-site inferences was consistently shifted leftward compared to standalone recHMM.

We saw a similar performance on the 5-taxon model conditions. In Table 4.3, the SERES+ recHMM inference had a stronger correlation with topological accuracy compared with the

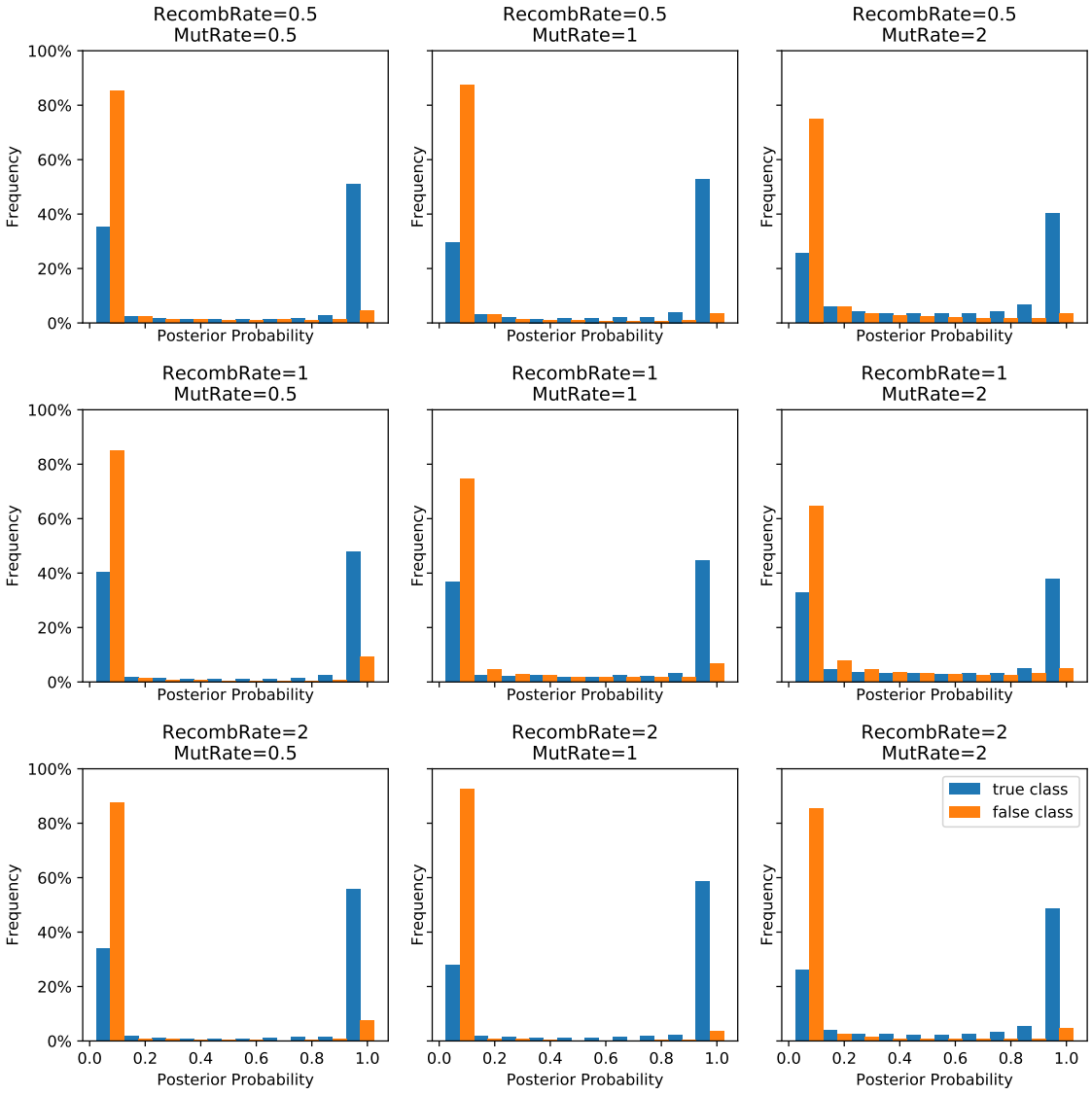


Figure 4.1: The posterior probability distribution inferred by the standalone recHMM method on 4-taxon model conditions. For each site, we split the local gene tree topologies into true class, which contains the true gene tree topologies for the site, and false class', which contains all other gene tree topologies. For each class and each replicate dataset in a model condition, the inferred posterior probabilities for gene trees at any site were binned into deciles; the resulting histogram was normalized over all replicates in a model condition (n = 30). The normalized histograms for the true and false classes are shown in blue and orange, respectively.

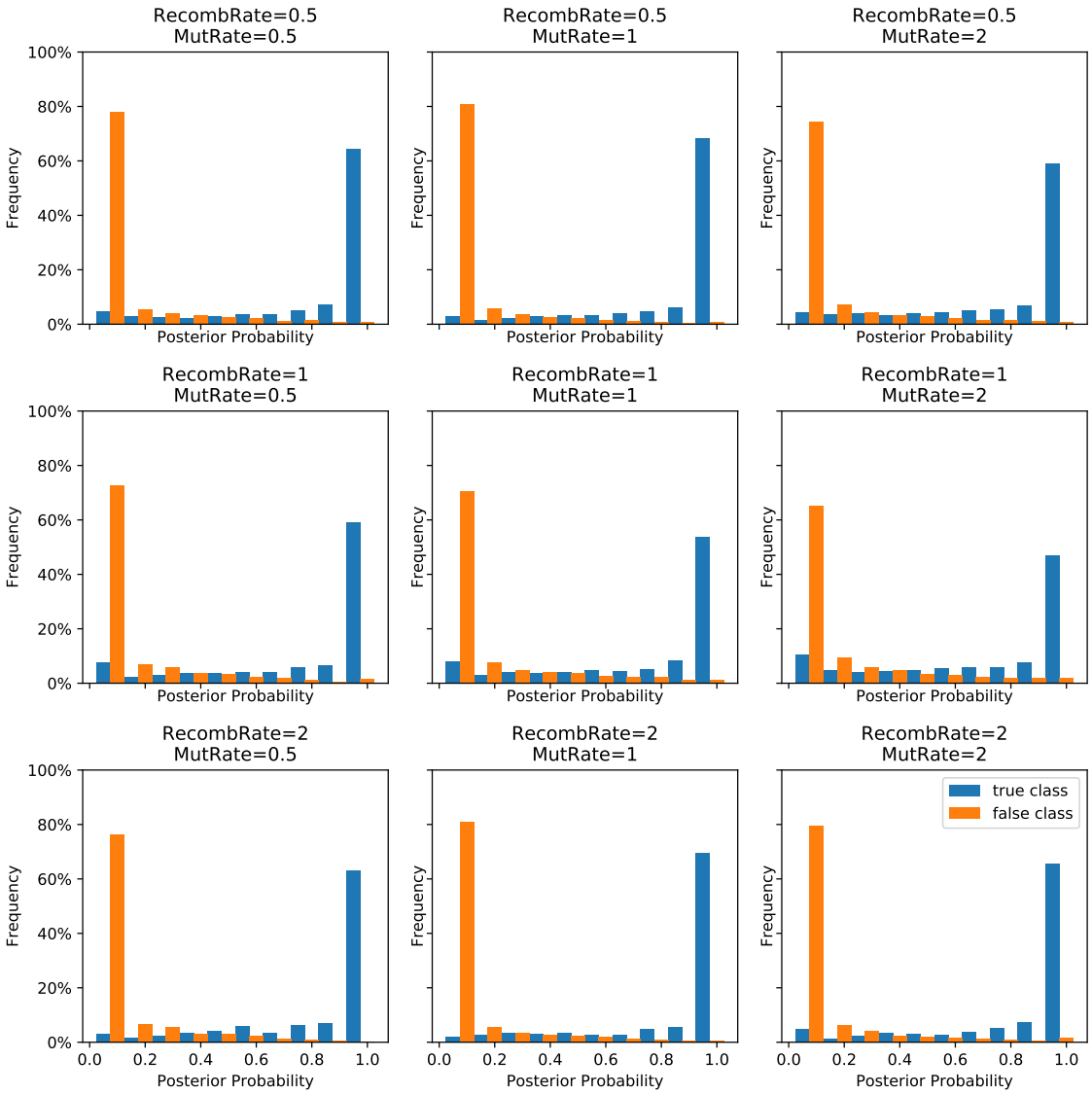


Figure 4.2: Histogram of posterior probabilities inferred by SERES+recHMM method on 4-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1.

Number of taxa	Recombination rate ρ	Mutation rate θ	recHMM correlation	SERES+recHMM correlation
4	0.5	0.5	-0.547	-0.830
4	0.5	1	-0.622	-0.866
4	0.5	2	-0.554	-0.799
4	1	0.5	-0.470	-0.775
4	1	1	-0.460	-0.742
4	1	2	-0.427	-0.677
4	2	0.5	-0.560	-0.855
4	2	1	-0.664	-0.867
4	2	2	-0.609	-0.837

Table 4.2: On 4-taxon model conditions, the posterior probabilities inferred by SERES+recHMM were better correlated with topological accuracy compared with the standalone recHMM. For each method, we calculated the Pearson correlation between the inferred posterior probability of a gene tree g and the topological distance between g and the true gene tree of a site. Average correlation for a method is calculated across all replicates in a model condition ($n = 30$).

recHMM inference for all model conditions.

The SERES+recHMM improved the correlation coefficients by at least 0.066 when the recombination rate $\rho = 1$ and the mutation rate $\theta = 2$. The largest correlation improvement was 0.155, where the recombination rate $\rho = 2$ and the mutation rate $\theta = 0.5$.

Number of taxa	Recombination rate ρ	Mutation rate θ	recHMM correlation	SERES+recHMM correlation
5	0.5	0.5	-0.571	-0.692
5	0.5	1	-0.526	-0.676
5	0.5	2	-0.525	-0.651
5	1	0.5	-0.597	-0.675
5	1	1	-0.569	-0.678
5	1	2	-0.618	-0.684
5	2	0.5	-0.506	-0.661
5	2	1	-0.543	-0.665
5	2	2	-0.56	-0.648

Table 4.3: On 5-taxon model conditions, posterior probabilities inferred by the SERES+recHMM had stronger correlation with topological accuracy compared with the posterior probabilities inferred by the standalone recHMM. Otherwise, table layout and description are identical to Table 4.2.

Similar to the 4-taxon and 5-taxon dataset comparisons, the SERES+recHMM's inference was more strongly correlated with topological accuracy across all 6-taxon model conditions when compared to standalone recHMM (Table 4.4). However, compared with the 4-taxon and 5-taxon datasets, the correlation coefficients for both methods were generally weaker. Moreover, the absolute improvement in the correlation achieved by SERES+recHMM was more negligible as

well.

The comparison of the posterior probability was the same for the false class. For the false class of inferences, the posterior probability distributions inferred by SERES+recHMM were more strongly shifted leftward than recHMM, despite the fact that posterior probabilities of the false class of inferences were more than double of the inferences in the 4-taxon and 5-taxon experiments.

However, a different outcome was observed for the true class of per-site inferences. For the true class, the posterior probability distributions inferred by the SERES+recHMM were more diffuse than recHMM rather than a rightward shift. One reason for this observation is that when the number of sequences involved in the HMM inference and learning increased, the computational complexity increased dramatically.

Number of taxa	Recombination rate ρ	Mutation rate θ	recHMM correlation	SERES+recHMM correlation
6	0.5	0.5	-0.3312	-0.494
6	0.5	1	-0.251	-0.457
6	0.5	2	-0.360	-0.472
6	1	0.5	-0.376	-0.486
6	1	1	-0.469	-0.473
6	1	2	-0.507	-0.535
6	2	0.5	-0.382	-0.506
6	2	1	-0.504	-0.515
6	2	2	-0.455	-0.554

Table 4.4: On 6-taxon model conditions, posterior probabilities inferred using SERES+recHMM were more highly correlated with topological accuracy compared to standalone recHMM. Otherwise, table layout and description are identical to Table 4.2.

We also conducted additional experiments to evaluate the impact of key method parameters. The results are shown in Table 4.5. By comparing the inference accuracy of recHMM versus SERES+recHMM with different SERES reversal probability γ, we found that the performance improvement achieved by SERES+recHMM over the standalone recHMM was robust to the choice of reversal probability γ, as long as the chosen value was not too high. Reasonable choices are equivalent to reversal breakpoints separated by an average of at least 100 bp of sequence length. The results are consistent with the original motivation for sequence-aware resampling and re-estimation. [145].

We tested the impact of the HMM state space size ϕ too. The results are shown in Table 4.6.

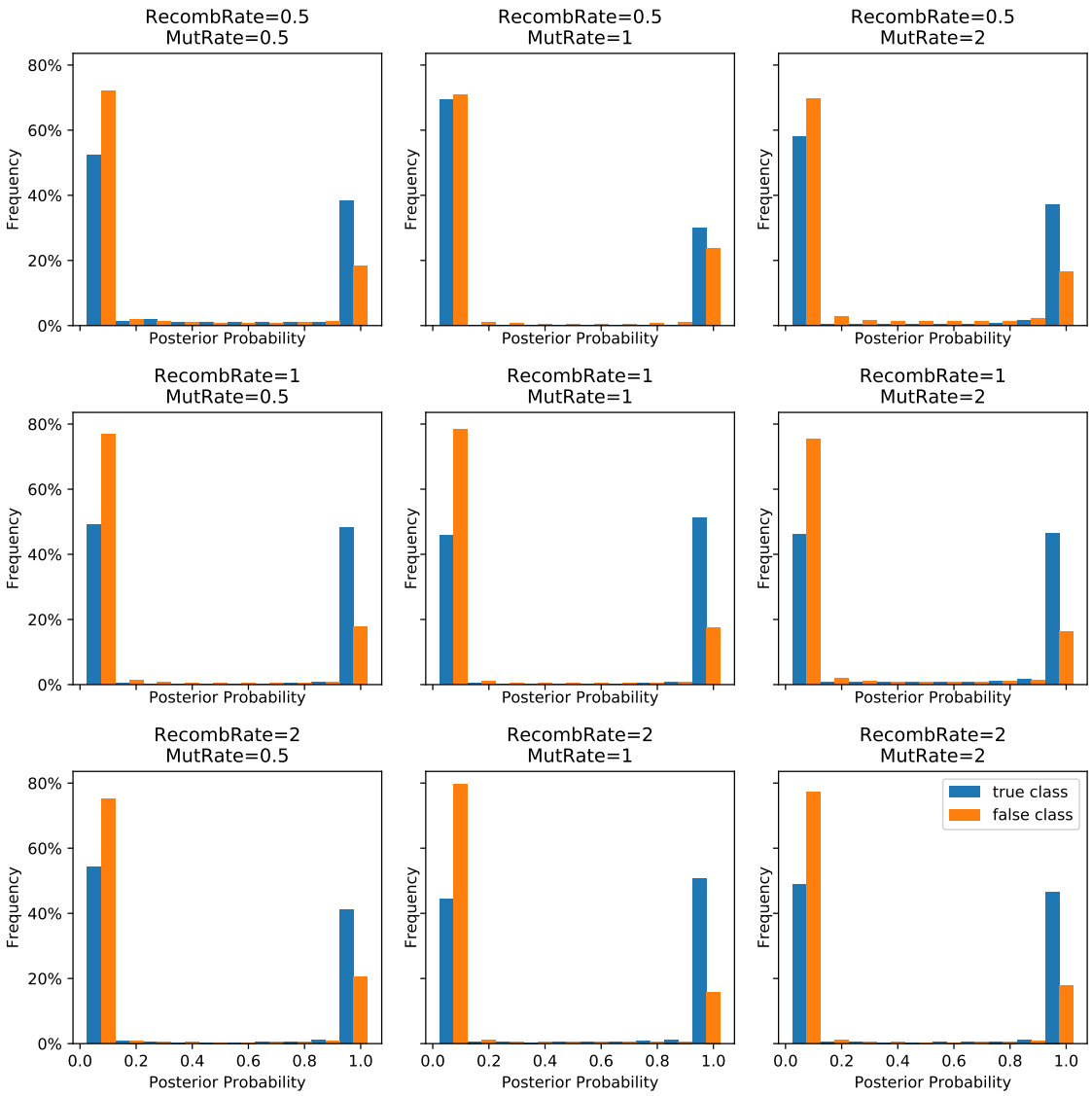


Figure 4.3: Distribution of posterior probabilities inferred by standalone recHMM method on 6-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1.

There are approximately 3 to 6 distinct true gene tree topologies in each simulated sequence. For the recHMM, using an HMM state space size ϕ larger than the number of local gene tree topologies, the inferred results were easily overfitting. This finding is consistent with the original study of recHMM [150].

The average runtime for the two methods was roughly comparable, and neither method consistently ran faster than the other one. We observed low memory usage (less than 100 MiB) for both methods throughout our study.

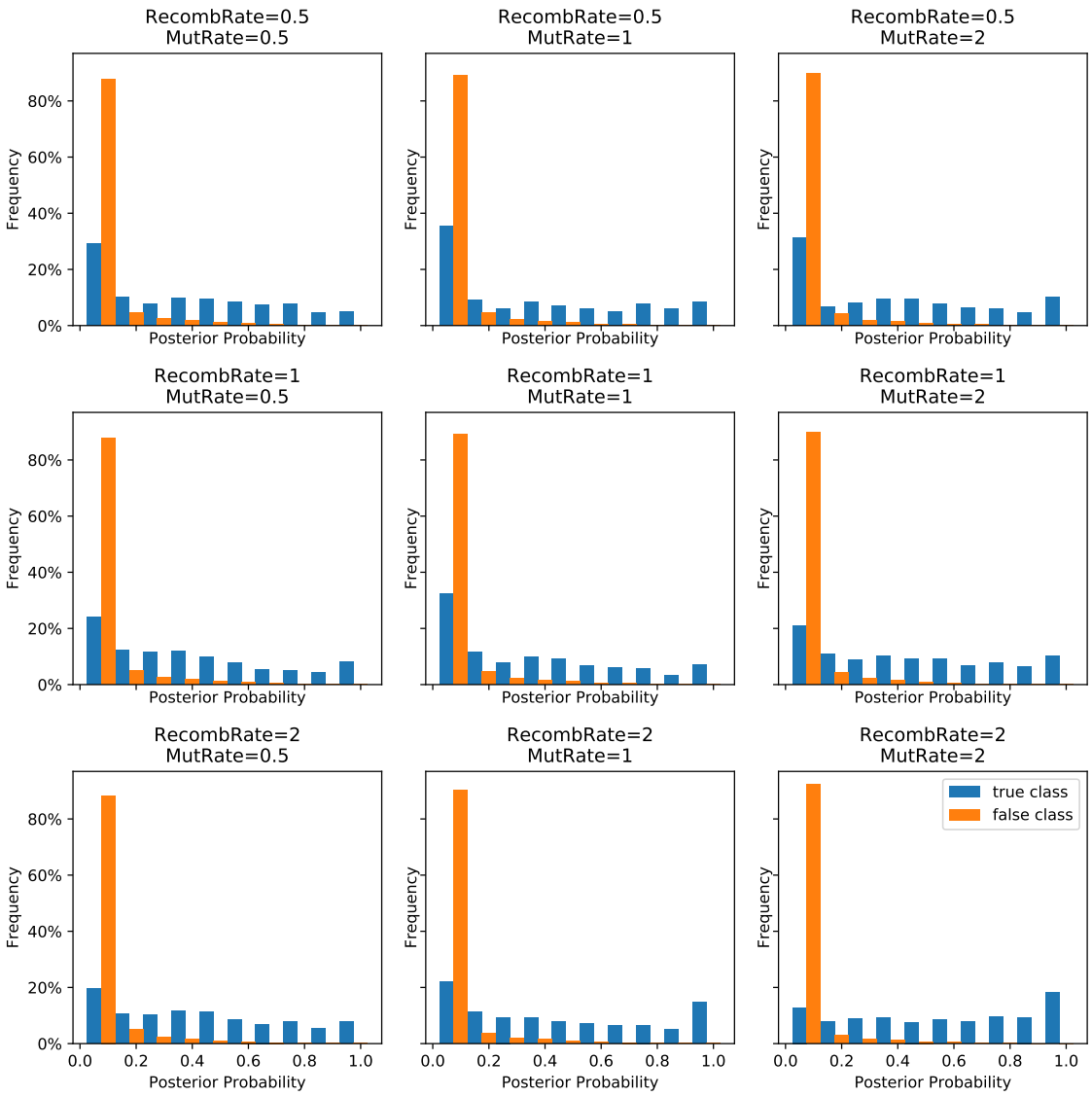


Figure 4.4: Distribution of posterior probabilities inferred by SERES+recHMM method on 6-taxon model conditions. Figure layout and description are otherwise identical to Figure 4.1.

4.3.2 Empirical study

Figure 4.5 shows the posterior probability distribution obtained by the SERES+recHMM method. The breakpoints refer to positions where the blue topology and the orange topology are switched. The SERES+recHMM results recovered five breakpoints that had been described in both Lole's study and Westesson's study [84, 150], which are located at 6402 bp, 6969 bp, 7073 bp, 9431 bp, and 9585 bp on the input alignment. The posterior probability distribution inferred by the

Number of taxa	Recombination rate ρ	Mutation rate θ	recHMM No γ	SERES+recHMM $\gamma = 0$	$\gamma = 0.005$	$\gamma = 0.01$	$\gamma = 0.1$
4	0.5	0.5	-0.614	-0.851	-0.842	-0.854	-0.679
4	0.5	1	-0.670	-0.868	-0.869	-0.847	-0.697
4	0.5	2	-0.651	-0.875	-0.840	-0.876	-0.700
4	1	0.5	-0.554	-0.873	-0.867	-0.888	-0.681
4	1	1	-0.506	-0.803	-0.798	-0.782	-0.575
4	1	2	-0.539	-0.760	-0.725	-0.748	-0.551
4	2	0.5	-0.651	-0.818	-0.844	-0.833	-0.664
4	2	1	-0.756	-0.841	-0.865	-0.845	-0.663
4	2	2	-0.667	-0.851	-0.867	-0.838	-0.676
5	0.5	0.5	-0.571	-0.732	-0.692	-0.673	-0.605
5	0.5	1	-0.526	-0.699	-0.676	-0.679	-0.601
5	0.5	2	-0.525	-0.689	-0.651	-0.662	-0.577
5	1	0.5	-0.597	-0.690	-0.675	-0.654	-0.591
5	1	1	-0.569	-0.705	-0.678	-0.669	-0.601
5	1	2	-0.618	-0.693	-0.684	-0.671	-0.593
5	2	0.5	-0.506	-0.669	-0.661	-0.629	-0.595
5	2	1	-0.547	-0.677	-0.665	-0.638	-0.572
5	2	2	-0.560	-0.675	-0.648	-0.653	-0.573
6	0.5	0.5	-0.489	-0.597	-0.573	-0.549	-0.465
6	0.5	1	-0.343	-0.591	-0.533	-0.533	-0.446
6	0.5	2	-0.374	-0.565	-0.537	-0.557	-0.44
6	1	0.5	-0.314	-0.609	-0.568	-0.556	-0.471
6	1	1	-0.375	-0.608	-0.568	-0.530	-0.453
6	1	2	-0.331	-0.607	-0.589	-0.548	-0.466
6	2	0.5	-0.511	-0.582	-0.558	-0.547	-0.477
6	2	1	-0.498	-0.606	-0.564	-0.556	-0.475
6	2	2	-0.528	-0.610	-0.584	-0.566	-0.509

Table 4.5: The comparison among different reversal probabilities γ on 4-, 5- and 6-taxon model conditions. The methods utilize models with $\phi = 3$ to infer a posterior probability distribution over gene tree topologies. For each method's inference, we calculated the Pearson correlation between the inferred posterior probability for a gene tree g and the topological distance between g and the true evolutionary history of a site (i.e., the true local gene tree). The averages are reported across all n replicates in a model condition ($n = 30$).

SERES+recHMM method clearly showed inference uncertainty in the first few hundred bp of the input alignment.

The study of Westesson [150] also reported two additional breakpoints at 4328 bp and 4401 bp that were not described in the study of Lole [84]. The standalone recHMM found local topology switches in the region of 4000 bp to 4200 bp. However, the SERES-based method results in more uncertainty in the corresponding regions.

Inconsistencies exist between the posterior probability distribution inferred by the standalone recHMM and the SERES+recHMM. Some local topology changes found by the standalone recHMM were not supported by the SERES+recHMM analysis, for example, the region from 6000 to 6500 bp.

Number of States ϕ	Recombination rate ρ	Mutation rate θ	recHMM correlation	SERES+recHMM correlation
3	0.5	0.5	-0.571	-0.692
3	0.5	1	-0.526	-0.676
3	0.5	2	-0.525	-0.651
3	1	0.5	-0.597	-0.675
3	1	1	-0.569	-0.678
3	1	2	-0.618	-0.684
3	2	0.5	-0.506	-0.661
3	2	1	-0.543	-0.665
3	2	2	-0.56	-0.648
10	0.5	0.5	-0.546	-0.681
10	0.5	1	-0.563	-0.671
10	0.5	2	-0.555	-0.66
10	1	0.5	-0.563	-0.677
10	1	1	-0.54	-0.674
10	1	2	-0.534	-0.661
10	2	0.5	-0.433	-0.639
10	2	1	-0.464	-0.651
10	2	2	-0.533	-0.651
15	0.5	0.5	-0.446	-0.685
15	0.5	1	-0.458	-0.685
15	0.5	2	-0.487	-0.655
15	1	0.5	-0.501	-0.679
15	1	1	-0.472	-0.671
15	1	2	-0.501	-0.657
15	2	0.5	-0.388	-0.648
15	2	1	-0.426	-0.652
15	2	2	-0.456	-0.645

Table 4.6: The comparison among different number of states ϕ on 5-taxon model conditions. The methods utilize models with $\gamma = 0.005$ to infer a posterior probability distribution over gene tree topologies. For each method's inference, we calculated the Pearson correlation between the inferred posterior probability for a gene tree g and the topological distance between g and the true evolutionary history of a site (i.e., the true local gene tree). The averages are reported across all n replicates in a model condition ($n = 30$).

Finally, for the gene tree topology, the posterior probability inferred by SERES+recHMM was generally lower than the standalone recHMM. One example is shown in the region located between 5000 and 8000 bp. The SERES+recHMM inferred almost zero probability for the green topology within this region, whereas the recHMM inferred a highly variable probability.

The SERES-based recHMM detected local genealogy changes in the HIV dataset, which is consistent with the previous studies [84, 150]. These pieces of evidence suggested that the sequence 95IN21301 is recombinant.

# of seq	Rec rate ρ	Mut rate θ	Runtime (h) recHMM	Runtime (h) SERES+ recHMM	Memory (GiB) recHMM	Memory (GiB) SERES+ recHMM
4	0.5	0.5	1.788	1.731	0.055	0.055
4	0.5	1	1.951	1.838	0.055	0.055
4	0.5	2	1.852	1.944	0.055	0.055
4	1	0.5	1.958	1.868	0.055	0.055
4	1	1	1.935	1.674	0.055	0.055
4	1	2	1.919	1.672	0.055	0.055
4	2	0.5	1.712	1.878	0.055	0.055
4	2	1	1.676	1.887	0.055	0.055
4	2	2	2.199	1.852	0.055	0.055
5	0.5	0.5	2.907	3.001	0.056	0.056
5	0.5	1	3.168	3.030	0.056	0.056
5	0.5	2	3.973	3.139	0.056	0.056
5	1	0.5	3.417	3.468	0.056	0.056
5	1	1	3.164	3.153	0.056	0.056
5	1	2	3.258	2.968	0.056	0.056
5	2	0.5	3.651	2.971	0.056	0.056
5	2	1	3.849	3.253	0.056	0.056
5	2	2	3.109	3.100	0.056	0.056
6	0.5	0.5	5.542	4.959	0.058	0.058
6	0.5	1	4.969	4.748	0.058	0.058
6	0.5	2	4.715	5.353	0.058	0.058
6	1	0.5	5.652	5.423	0.058	0.058
6	1	1	5.471	5.529	0.058	0.058
6	1	2	4.398	5.314	0.058	0.058
6	2	0.5	4.819	6.177	0.059	0.058
6	2	1	4.271	5.136	0.058	0.058
6	2	2	4.221	5.056	0.058	0.058

Table 4.7: The runtime and memory usage information for standalone recHMM and SERES+recHMM methods on simulation study model conditions. Model conditions were parameterized by the number of sequences, recombination rate ρ, and mutation rate θ. Both methods utilize models with $\phi = 3$ and $\gamma = 0.005$ to infer a posterior probability distribution over gene tree topologies. Average runtime in hours and peak memory usage in GiB are reported across all replicates in a model condition ($n = 30$).

4.4 Discussion

To evaluate to what extent the posterior probability inferred by each method reflects the topological accuracy, we compared the method performances of the standalone recHMM method and the combined SERES+recHMM method. The correlation between the inferred per-site posterior probability for a gene tree topology g and the topological accuracy of g as measured by the Robinson-Foulds distance between g and the true gene tree topology for a site was used to assess method performance. Across all the simulation model conditions, the posterior probabilities inferred by the SERES+recHMM method had a consistently better correlation with topological accuracy compared to standalone recHMM. The improved performance obtained by combining recHMM inference with

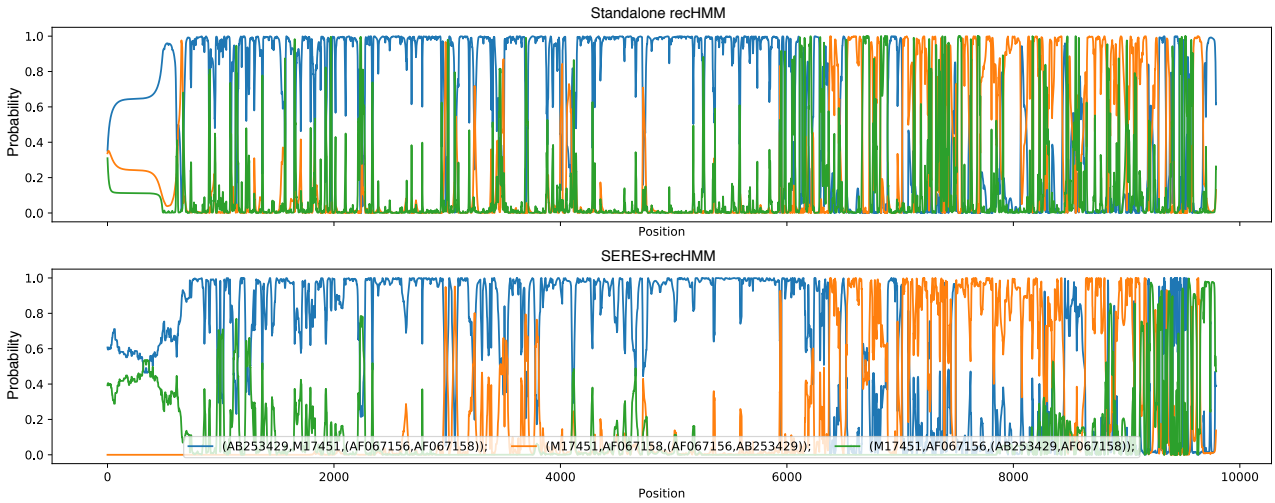


Figure 4.5: Posterior probability distribution of local gene tree topologies inferred by standalone recHMM versus SERES+recHMM method on Indian HIV-1 dataset. We re-analyzed a subset of the Indian HIV-1 genome dataset that was published by [84]; [150] re-analyzed the original dataset using recHMM. Our re-analysis compared local gene tree probabilities computed using standalone recHMM posterior decoding (top panel) versus SERES+recHMM posterior decoding (bottom panel). The plots show posterior decoding probabilities (y-axis) versus genome coordinate (x-axis). Local gene tree probabilities are colored based on the three possible unrooted topologies for the four-taxon dataset (shown in either blue, orange, or green).

SERES resampling and re-estimation was robust to a wide range of mutation and recombination rates.

We also compared the inferred posterior probability distributions produced by these two methods. We found that the standalone recHMM method produced a similar posterior probability distribution for both true classes and false classes. The SERES+recHMM method produced more distinguishable posterior probability distributions for the true classes and false classes.

We attribute these findings to two factors. First, the application of the SERES resampling and re-estimation appears to be conducive to the improved inference of true gene tree topologies. The SERES resampling approach has the ability to retain the sequence dependence in the input alignment to the resampled replicates. And, the intra-sequence dependence among sites provides additional information on the historical evolutionary events, especially those that caused the dependence. The local genealogy inference greatly benefited from the SERES resampling and re-estimation process. Second, the SERES resampling algorithm reveals uncertainties in the inference of local

genealogy. Incorrect inferences for the gene tree topologies were less repeatable. The posterior probability distribution of the false class was leftward shifted consistently for all model conditions of the simulated datasets, which indicates that the SERES resampling and re-estimation process produced a consistently low posterior probability for those incorrect local genealogy topologies. Even for the larger dataset with 6 taxa, where solution space was an order of magnitude larger than that of the 4-taxon and 5-taxon datasets, the SERES+recHMM method produced consistently low posterior probabilities for those incorrect gene tree topologies.

Although the SERES+recHMM produced low posterior probabilities for the false class over all model conditions, it returned diffused posterior probability distribution for the true class when the dataset size was large, which indicates that uncertainties included in the inference of true gene tree topology were also reflected in the posterior probabilities when the dataset size increased. This result may be caused by the increased computational complexity of HMM learning optimization as the number of input sequences increases. It is likely that conservatively limiting SERES-based re-estimation to 10 learning iterations is insufficient for the larger model conditions in our study. More intensive learning optimization may yield improved re-estimation and a greater performance benefit from augmenting recHMM with SERES.

Additional experiments that we performed to evaluate how the choice of the parameters impacted the method performance indicated that the performance advantage returned by SERES+recHMM over standalone recHMM was robust to the choice of reversal probability γ. The results are consistent with the original motivation for sequence-aware resampling and re-estimation. We noted the correspondence between an rth order Markov process and a SERES random walk with a reversal probability γ. For $\gamma = 0.5$, a first-order Markov process suffices; for $\gamma < 0.5$, higher-order Markovian processes are needed to capture sequential dependence. Essentially, smaller γ values mean that longer-distance sequential dependence is retained. Our results suggest that there is a certain threshold. When passing that threshold, longer-distance sequential dependence is critical to the performance of resampling and re-estimation for sequence-based inference problems. Experiments with alternative settings for the HMM state space size showed SERES+recHMM's

performance was relatively robust to overfitting, as compared to standalone recHMM analysis.

4.5 Conclusions

This study introduced the application of SERES random walks on aligned sequences and showed SERES as a data perturbation technique to improve statistical inference and learning. The simulation experiments showed that the combination of the The SERES resampling approach with the recHMM achieved great improvement in the local genealogy inferences. The empirical study on the HIV genome sequence dataset confirmed the breakpoints detected in the previous studies [84, 150]. The SERES resampling approach achieved great success on recombination detection and local genealogical inference problems. SERES resampling and re-estimation may prove to be similarly beneficial in ancestral recombination inference problems other than local genealogical inference, such as recombination rate estimation [133], recombination hotspot or coldspot detection [99, 5], etc.

CHAPTER 5

PHYLOGENETIC SUPPORT ESTIMATION WITH THE RANDOM WALK RESAMPLING APPROACH

5.1 Introduction

In modern phylogenetic studies, the true evolutionary history is usually not available for analysis. Phylogenies are generally inferred by statistical methods. Therefore, it is critical to evaluate the reproducibility of the inferred phylogenies. In 1985, Felsenstein proposed a standard bootstrap approach to estimate the confidence intervals for an input phylogeny [36]. This bootstrap method takes a set of sequence alignmentss as input. Then this method generates bootstrap replicates by sampling sites from the input alignment at random with replacements. Alignment sites are considered independently and identically distributed (i.i.d). Phylogenies are inferred for all bootstrap replicates. Given an annotation phylogeny, the bootstrap support of the internal edges of the annotation phylogeny is calculated. The support value is the proportion of the re-estimated phylogenies that contain the same internal edge. Bootstrap support estimation has become the de facto standard for assessing reproducibility in phylogenetics and phylogenomics analysis. This method has been used in almost all modern phylogenetic studies to show the confidence of the involved phylogenies. This makes Felsenstein's ground-breaking work one of the most cited works in history. Felsenstein's 1985 paper has become the 41st most cited in all of science, according to the survey of [141], which has been cited over 44,000 times.

Many alternative computational algorithms can be used for the purposes of phylogenetic support estimation. For example, non-parametric resampling approaches such as the jackknife [140]; the MSA confidence measurement approaches such as GUIDANCE1 [71, 110], GUIDANCE2 [126], PSAR [65], T-COFFEE [107], and Divvier [1]); parametric MSA resampling or filtering methods applied to the problem of phylogenetic support estimation (e.g., TCS [18], Gblocks [17], and Trimal [16], and Bayesian posterior probability methods [156]). However, these alternative methods are

not widely used due to limitations such as parametric models and specific application scenarios.

In this study, we proposed a new non-parametric resampling approach, RAWR, which is short for "RAndom Walk Resampling." The RAWR resampling method is a simplified version of the SERES resampling approach described in previous sections. The SERES algorithm is a semi-parametric algorithm for the application of unaligned sequences that requires anchor estimation for synchronization purposes. This increases the method complexity by adding extra parameters such as anchor length, count, and sequence similarity measurement. The RAWR resampling approach does not require anchor estimation. This new algorithm is designed for the phylogenetic support estimation problem by resampling and re-estimation using unaligned sequences. The RAWR-based phylogenetic support estimation shows comparable or typically better performance than the traditional bootstrap support estimation approach.

5.2 Methods

The computational problem of the phylogenetic support estimation is described below. The inputs contain an MSA A and a phylogenetic tree T. The MSA A is estimated using MSA method f, and the phylogenetic tree $T = (V, E)$ is estimated using the phylogenetic inference method g on the MSA A. The output is a set of confidence values ranging from 0 to 1 for all bipartitions split by internal edges in $e \in E$.

5.2.1 RAWR-based Phylogenetic Support Estimation

The RAWR-based phylogenetic support estimation takes an estimated MSA A and a phylogenetic tree $T = (V, E)$. The problem output consists of confidence interval estimates for each bipartitions defined by non-leaf edge $e \in E$.

The estimated MSA A is inferred from the unaligned sequences S by the MSA method f. First, the RAWR resampling approach generates sequence replicates from the input MSA A.

The RAWR resampling approach also utilizes the random walk to conduct the resampling process, similar to the SERES method. The random walk starts with a randomly selected start point

and a randomly selected direction. The random walk moves along the input alignment in the initial direction. All sites located on the trace of the random walk are sampled to the resampled replicate.

The RAWR algorithm does not require estimation of the anchors to ensure synchronization during the random walk, where the random walk only changes its direction at the barriers, which are sites located at the start and end of anchors, with reverse probability γ. The RAWR resampling approach simplifies this setting and uses all sites of the input estimated MSA A as barriers. The random walk certainly changes direction when encountering the first and last sites. The reverse probability is set to γ elsewhere. The random walk procedure ends once the sampled replicate reaches the desired length. The details of this resampling procedure are described in Algorithm 5.1.

Algorithm 5.1: RAWR phylogenetic support estimation

```
 1: procedure RAWRSupport(A, T, f(), g(), γ, k_r)
                                  ▷ Input: MSA A, phylogenetic tree T, MSA method f(),
                  ▷ phylogenetic tree estimation method g(), reversal probability γ,
                                                  ▷ number of replicates k_r
                                        ▷ Output: phylogenetic support estimates ε
 2:     reestimates = <>
 3:     for i = 1 to k_r do
 4:         X_i = resampleRAWRReplicate(A)
 5:         reestimates .= g(f(X_i))
 6:     for all non-leaf edge e in T do
 7:         ε(e) = proportion of T_i in list reestimates
                       that display bipartition corresponding to e
 8:     return(ε)

 9: procedure resampleRAWRReplicate(A,γ)
10:     Y = <>
11:     select i ∈ [1, |A|] and walkDirection uniformly at random
12:     while !converged(Y,A) do
13:         Y .= A[i]                                  ▷ add ith column of A to Y
14:         if reversal(γ) || (i == 1 && walkDirection is left)
15:            || (i == |A| && walkDirection is right) then
16:             reverse(walkDirection)
17:         i = next column index after i in walkDirection order
18:     return(unalign(Y))                             ▷ unalign(Y) drops indels from Y

19: procedure converged(Y,A)
20:     return(length(Y) ≥ length(A))
                                  ▷ Sequence-length-based convergence criterion requires
                                  ▷ number of resampled sites ≥ input MSA length
```

In this study, we set the length of the resampled replicate to be equal to the length of A. Other ending criteria are also feasible for the random walk, for example, statistical criteria based on the random walk procedure. The indels are removed from the sampled alignment to produce a set of

unaligned sequences. Each round of the random walk procedure produces a resampled replicate which consists of a set of unaligned sequences. The random walk process is repeated multiple times independently to obtain a set of resampled replicates.

The two-phase methods for phylogenetic inference are widely used in systematic studies. First, the MSA is estimated for the unaligned sequence. Then the estimated MSA is used to infer the phylogenetic tree. We used MAFFT version 7.222 [64] with default settings for the MSA re-estimation. The MAFFT [64] is one of the most accurate MSA estimation methods. We also explored the choice of MSA estimation methods, including ClustalW [88], and FSA[12]. Moreover, it is proved to result in good accuracy in the following phylogenetic inference [79, 80]. The summary statistics for the estimated MSAs are shown in Table 5.3. The phylogenetic tree was inferred from the re-estimated MSA by RAxML version 8.2.11 [130] under the GTR+Γ model with maximum likelihood criteria [118, 155, 142].

The phylogenetic support values are calculated for all internal branches of the input phylogenetic tree. The support value of the internal branch $e \in E$ is the fraction of the re-estimated trees that contain the same internal branch e. In this study, the default setting of γ was 1×10^{-1}. We also explored other reverse probability, $\gamma \in \{1 \times 10^{-3}, 1 \times 10^{-2}, 2 \times 10^{-2}, 5 \times 10^{-2}, 1 \times 10^{-1}, 2 \times 10^{-1}, 3 \times 10^{-1}\}$. For each experiment, RAWR was conducted to generate 100 resampled replicates for each input dataset.

An illustrated example shows in Figures 5.1, where the RAWR random walk is applied on a 8-taxon dataset. Then an alignment is generated based on the RAWR replicate and a tree is inferred from the estimated alignment.

5.2.2 Bootstrap Phylogenetic Support Estimation.

We performed the bootstrap analysis using the RAxML version 8.2.11. The bootstrap was implemented in the RAxML software. The bootstrap support estimation first generates 100 bootstrap replicates from the input MSA. Then the phylogenetic tree was re-estimated for the bootstrap replicate. The support values of the internal branches are calculated using the same process as the

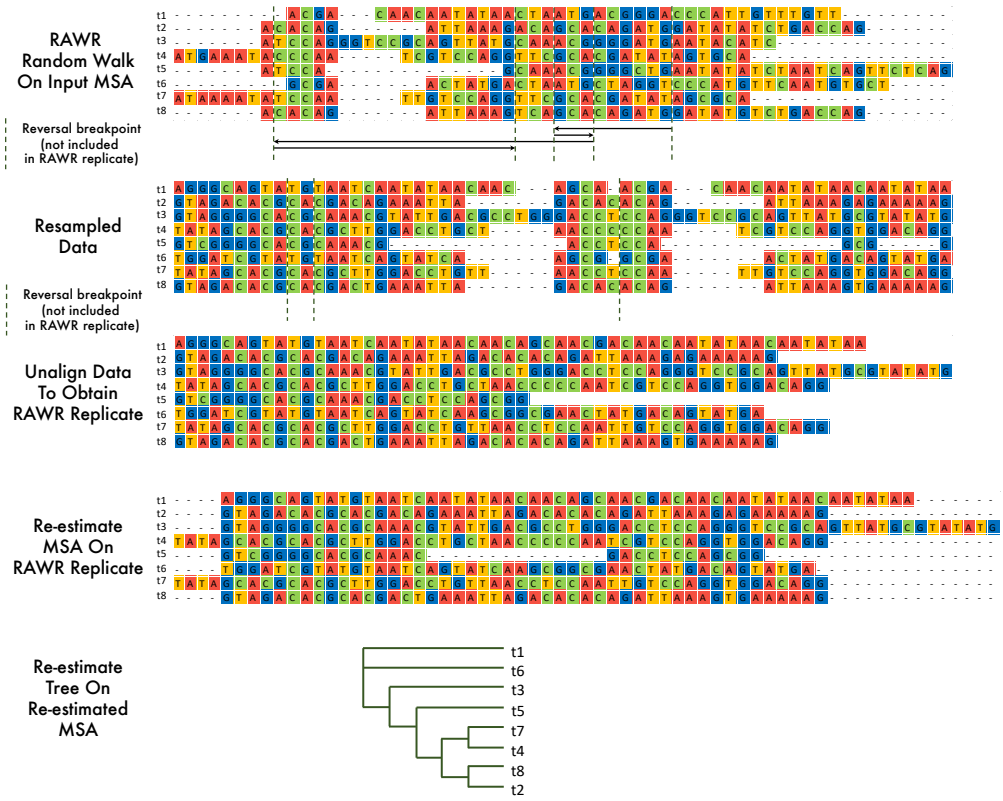


Figure 5.1: An illustrated example of RAWR resampling and re-estimation. The first step of the RAWR-based phylogenetic support estimation is sequence resampling. A random walk is performed on the input MSA. MSA sites are resampled during the random walk. The indels are removed from the resampled sites to produce a resampled replicate. Then, MSA is re-estimated from the resampled sequence. Finally, a phylogenetic tree is re-estimated using the re-estimated MSA as input. The dashed lines in the first and second subplots show the reversal breakpoints.

RAWR-based phylogenetic support estimation process.

5.2.3 Additional Performance Study

We explored different parameter settings, such as the choice of the MSA estimation method, for the RAWR-based support estimation method. We explored alternative MSA methods for the estimation and re-estimation of MSAs. We used ClustalW in the performance study to explore the impact of alignment quality on downstream phylogenetic inference and support estimation. We selected ClustalW because it is one of the most widely used MSA methods in computational biology and bioinformatics. For our experiments, we used ClustalW version 2.1 with default settings. The detailed summary statistics of the ClustalW-estimated MSAs are shown in Table 5.1.

Model condition	ClustalW alignment			RAxML(ClustalW) nRF
	length	SP-FN	SP-FP	
10.A	1216	0.670	0.747	0.207
10.B	1236	0.726	0.809	0.236
10.C	1184	0.835	0.889	0.379
10.D	1171	0.828	0.874	0.500
10.E	1163	0.901	0.926	0.650

Table 5.1: Summary statistics for ClustalW-estimated alignments and RAxML(ClustalW) trees on 10-taxon model conditions. Table layout and description are otherwise identical to Table 5.3.

To better evaluate the performance of the RAWR-based support estimation method, we applied the RAWR resampling approach to model conditions with different parameter settings of the insertion/deletion model. We conducted a simulation experiment using additional model conditions from the original study of the SERES resampling algorithm [145] where we utilized the long gap length distribution from the study of Liu et al. [80], rather than the medium gap length distribution used elsewhere in our simulation study. The simulation and experimental procedures were exactly the same across all model conditions in our simulation study. The model parameters and summary statistics of model conditions using the long gap length distribution are listed in Table 5.2.

Model condition	Model tree height	Insertion deletion rate	True alignment			MAFFT alignment		
			ANHD	Gapiness	Length	Length	SP-FN	SP-FP
10.long.A	0.4	0.13	0.276	0.440	1804.8	1433.7	0.272	0.315
10.long.B	0.7	0.1	0.363	0.481	1926.7	1447.8	0.381	0.426
10.long.C	1	0.06	0.455	0.456	1853.5	1413.3	0.510	0.537
10.long.D	1.6	0.031	0.542	0.432	1754.1	1403.1	0.725	0.729
10.long.E	4.3	0.013	0.660	0.445	1811.0	1560.1	0.899	0.897

Table 5.2: Long-gap-length model conditions: parameter values and summary statistics. Our simulation study included additional 10-taxon model conditions that utilized the long gap length distribution from the study of 2012 study of Liu et al. [80]. The model parameters consisted of model tree height and insertion/deletion probability, and each model condition corresponds to a distinct set of model parameter values. The long-gap-length model conditions are named 10.long.A through 10.long.E in order of generally increasing sequence divergence. The following table columns list average summary statistics for each model condition (n = 20). "NHD" is the average normalized Hamming distance of a pair of aligned sequences in the true alignment. "Gappiness" is the percentage of true alignment cells which consists of indels. "True align length" is the length of the true alignment. "Est align length" is the length of the MAFFT-estimated alignment [64] which was provided as input to the support estimation methods. "SP-FN" and "SP-FP" are the proportion of homologies that appear in the true alignment but not in the MAFFT-estimated alignment and vice versa, respectively. The table and caption are reproduced from the original study of the SERES resampling algorithm [145].

We mainly compared the method performance of the RAWR-based support estimation method and the bootstrap method. There are many other well-designed methods for the problem of phylogenetic tree support estimation. Alternatives include other non-parametric resampling methods such as the jackknife [140] and parametric resampling [147, 37], such as MSA-specific confidence measures and other alignment-oblivious phylogenetic support estimation methods. Alternatives have also been proposed for the last step of the phylogenetic support calculation. For example, the transfer bootstrap expectation (TBE) method [74], which pairs bootstrap resampling of MSAs and phylogenetic tree re-estimation with an alternative support calculation. We conducted experiments using one of the MSA-specific confidence measures, GUIDANCE2, and parametric resampling to perform the support estimation of phylogenetic trees. We also compared the RAWR-based support estimation performance with the alRT, one of the alignment-oblivious phylogenetic support estimation methods. Finally, we compared the method performance of the RAWR+TBE and bootstrap+TBE support estimations.

The GUIDANCE2 [126] is one of these methods that was originally developed for the estimation of confidence intervals of multiple sequence alignments. GUIDANCE2 combines the HoT algorithm with altering parameters used in progressive MSA estimation methods, such as guide tree and gap penalties, to generate parametric resampling techniques. In our experiment, we ran GUIDANCE2 with default settings. For each dataset, GUIDANCE2 was used to resample 100 replicates, and re-estimation was performed on each resampled replicate using an identical procedure as in the RAWR and bootstrap analyses. We calculated the phylogenetic tree support using the RAxML software with the same command as used in other simulation experiments in our study. Although the HoT algorithm was initially designed for MSAs, and GUIDANCE2 was originally developed for MSA confidence interval placement, it is natural to consider the impact of MSA quality on downstream phylogenetic inference. As we demonstrate in the performance study, a new application of these parametric and semi-parametric techniques beyond their originally intended use can bring value.

We used the aLRT [3] implementation in PhyML version 3.0 [49] to run two aLRT analy-

ses. First, We ran a parametric aLRT support analysis under the general time reversible (GTR) nucleotide substitution model [135], where a phylogenetic tree topology was estimated alongside GTR substitution model parameters and branch lengths. The second analysis is that, we ran a fixed-topology analysis, where the fixed topology consisted of the annotation topology that was estimated by RAxML. The fixed-topology analysis included estimation of branch lengths and GTR substitution rates/frequencies using PhyML.

TBE [74] was run using the same annotation MSA, annotation tree, and re-estimated trees as in the rest of our performance study. Re-estimated trees were generated by either bootstrap resampling and re-estimation or RAWR resampling and re-estimation. The equivalent inputs enable a comparison across TBE and other methods in our study. The TBE analyses were run using Booster v. 0.1.2.

Since the RAWR random walk is continuous for each replicate, the random walk can be concentrated in a narrow region. In this scenario, sites outside the narrow region covered by the random walk were ignored in the statistical inference involving RAWR resampling. We proposed and evaluated an alternative random-walk-based phylogenetic support estimation procedure to help the RAWR random walk better explore the input alignment. The alternative procedure replaced random reversals in the RAWR resampling procedure with random teleportation. The details of this new support estimation algorithm shows in Algorithm 5.2. For this reason, we refer to the alternative method as "RAWR+teleport."

Algorithm 5.2: RAWR+teleport resampling procedure

```
1:  procedure resampleWithRAWR+Teleport(A,γ)
2:      Y = <>
3:      select i ∈ [1, |A|] and walkDirection uniformly at random
4:      while !converged(Y,A) do
5:          Y .= A[i]                                   ▷ add ith column of A to Y
6:          if teleport(γ) then                         ▷ Biased coin flip with bias γ
7:              select i ∈ [1, |A|] and walkDirection uniformly at random
8:          else
9:              if (i == 1 && walkDirection is left) || (i == |A| && walkDirection is right) then
10:                 reverse(walkDirection)
11:             i = next column index after i in walkDirection order
12:     return(unalign(Y))                              ▷ unalign(Y) drops indels from Y
```

5.2.4 Simulated datasets

We utilized the same model conditions and the simulation datasets used in the previous studies [145, 80] for the simulation dataset. The simulated datasets cover a wide range of dataset sizes and evolutionary divergence.

The simulation process of the benchmark datasets is described as follows. For the 10-taxon and 50-taxon datasets, the INDELible version 1.03 [39] was used to sample non-ultrametric trees. The branch lengths were sampled randomly within the range of 0 and 1. For the 100-taxon datasets, r8s version 1.7 [123] was used to sample random birth-death model trees. The sampled ultrametric model trees were deviated using the procedure described in the study by Roshan et al. [102]. The deviation factor $c = 2.0$. All model trees were rescaled to the specified tree height h described in the corresponding model conditions. The nucleotide sequences were simulated along the model trees under the general time-reversible (GTR) model of substitution and the insertion/deletion model [39]. Details about the GTR model and insertion/deletion models are described in Chapter 2. The base frequency and substitution rate parameters were obtained from the 2012 study by Liu et al. [80]. The root sequence length was set to be 1kb.

INDELible version 1.03 was used to simulate nucleotide sequences for the 10-taxon and 50-taxon model conditions. ROSE was used to simulate nucleotide sequences for the 100-taxon model conditions with the indel model described in Liu's 2012 study [80] with a medium gap length distribution. The above simulation process was repeated 20 times independently to produce multiple replicate datasets. The model parameters and summary statistics of the simulated datasets are shown in Table 5.3.

5.2.5 Empirical datasets

We used the empirical benchmark dataset from the Comparative RNA Website (CRW) database (www.rna.icmb.utexas.edu) [14]. The CRW rRNA datasets [14] contain sequence alignments that were generated based on structural information and intensive manual correction. This benchmark dataset has been widely used in the evaluation and comparison of MSA approaches. We estimated

Model condition	Number of taxa	Model tree height	Insertion deletion rate	True alignment			MAFFT alignment			RAxML nRF
				ANHD	Gapiness	Length	Length	SP-FN	SP-FP	
10.A	10	0.47	0.13	0.380	0.591	2466	1543	0.566	0.629	0.186
10.B	10	0.7	0.1	0.479	0.618	2691	1602	0.687	0.750	0.243
10.C	10	1.2	0.06	0.591	0.645	2832	1588	0.811	0.850	0.443
10.D	10	2	0.031	0.642	0.591	2490	1583	0.815	0.841	0.464
10.E	10	4.4	0.013	0.696	0.578	2390	1623	0.904	0.913	0.664
50.A	50	0.45	0.06	0.415	0.667	3070	2053	0.340	0.336	0.084
50.B	50	0.73	0.03	0.513	0.603	2525	1834	0.451	0.431	0.146
50.C	50	1.2	0.02	0.598	0.620	2646	1950	0.731	0.704	0.322
50.D	50	2	0.012	0.667	0.629	2720	2171	0.902	0.881	0.517
50.E	50	4.3	0.005	0.715	0.591	2474	2385	0.974	0.965	0.755
100.A	100	4	1×10^{-5}	0.454	0.331	1682	1533	0.054	0.046	0.075
100.B	100	7	1×10^{-5}	0.540	0.439	2263	1861	0.209	0.176	0.119
100.C	100	15	5×10^{-5}	0.646	0.571	2317	2418	0.680	0.603	0.470
100.D	100	25	2×10^{-5}	0.683	0.634	1837	2799	0.899	0.853	0.607
100.E	100	20	4×10^{-5}	0.672	0.614	2487	2701	0.848	0.796	0.661

Table 5.3: Model condition parameters and summary statistics of the simulation datasets. Model condition parameters consisted of the number of taxa, tree height, and insertion/deletion probability. The model conditions are named from A to E to represent increasing evolutionary divergence. The average summary statistics are reported for the true alignments, and the MAFFT-estimated alignments over n replicate datasets ($n = 20$). "ANHD" is the average normalized Hamming distance of a pair of aligned sequences in an MSA, "Gappiness" is the proportion of an MSA matrix that consists of indels, "length" is the number of MSA columns, and "SP-FN" and "SP-FP" are the proportions of residue pairs that appear in the true alignment but not in the estimated alignment or vice versa, respectively. The average normalized Robinson-Foulds distance ("nRF") between the model tree and the RAxML(MAFFT)-inferred tree is also reported over n replicate datasets ($n = 20$).

MLE trees from the reference alignment. The RAWR-based phylogenetic support estimation and the bootstrap phylogenetic support estimation were conducted on the empirical benchmark dataset using the same command as in the simulation study. Simulation studies and empirical benchmarking were designed to focus on non-coding DNA sequence evolution. Therefore, we selected the intronic rRNA datasets with a range of evolutionary divergence and dataset size to perform the empirical experiments for consistency purposes. Sequences with more than 99% missing data were filtered from the analysis. The summary statistics of the empirical datasets are shown in Table 5.4.

5.2.6 Performance measurement.

We used the precision-recall (PR) curve and the area under the PR curve (PR-AUC) to evaluate the performance of both phylogenetic support estimation methods. For the following reasons, we

Dataset	Number of taxa	Reference alignment			MAFFT alignment			RAxML nRF
		ANHD	Gapiness	Length	Length	SP-FN	SP-FP	
IGIA	110	0.606	0.915	10368	6065	0.732	0.780	0.645
IGIB	202	0.579	0.910	10633	7070	0.825	0.863	0.678
IGIC2	32	0.533	0.700	4243	3530	0.691	0.716	0.517
IGID	21	0.719	0.782	5061	3063	0.874	0.905	0.778
IGIE	249	0.451	0.838	2751	2847	0.406	0.389	0.585
IGIIA	174	0.668	0.814	6406	6945	0.817	0.800	0.450

Table 5.4: Summary statistics of empirical datasets. Summary statistic calculations and descriptions identical to Table 5.3.

selected the PR curve and its area, PR-AUC, to measure the method performance rather than the Receiver Operating Characteristic (ROC) curve and the area beneath it, ROC-AUC. Although the ROC curve is one of the most commonly used statistics for evaluating the performance of binary-response statistical inferences, it can be misleading when the proportions of bipartitions that appear in or not in the true phylogeny are very imbalanced [121]. The precision is sensitive to the imbalance of the dataset. This makes the PR curve a more accurate measurement of the imbalanced dataset. The PR curve is calculated on thresholds ranges from 0 to 1. A confusion matrix is calculated for each threshold based on the model prediction and the ground truth. The support values estimated by the phylogenetic support estimation methods serve as the model predictions. The phylogenetic support estimation methods generate support values for bipartitions, which is a unique split of the leaf set defined by the internal branches of a given annotation phylogeny. In both simulation studies and empirical study, the annotation phylogenies are inferred from the simulated sequences.

The labels indicating whether a bipartition appears in the model tree were generated by the comparison of the inferred tree and the model/reference tree. The bipartitions that appear in the model tree/reference tree are represented by 1, and the other branches are represented by 0.

True positives (TP) consist of bipartitions of the estimated tree that have support values greater than or equal to a given threshold and appear in the reference tree. False positives (FP) consist of bipartitions of the estimated tree that have support value greater than or equal to a given threshold but do not appear in the reference tree. False negatives (FN) consist of bipartitions of the estimated tree that have support less than a given threshold but appear in the reference tree. True negatives (TN) consist of bipartitions of the estimated tree that have support less than a given threshold and

do not appear in the reference tree. The PR curve plots the recall versus the precision of each threshold. The recall is calculated by $\frac{|TP|}{|TP|+|FN|}$. And the precision is calculated by $\frac{|TP|}{|TP|+|FP|}$. We used custom scripts, and the scikit-learn Python library [136] to calculate the curves and AUC quantities.

We also compared the runtime and peak memory usage for these two phylogenetic support estimation methods. All experiments were conducted on computing facilities in the Michigan State University High-Performance Computing Center. We used compute nodes in the intel16-k80 cluster, each with a 2.4 GHz 14-core Intel Xeon E5-2680v4 processor.

5.3 Results

5.3.1 Simulation Study

5.3.1.1 Performance comparison of RAWR versus bootstrap.

The PR-AUC results of both methods on the simulation datasets are shown in Table 5.5. The RAWR-based approach produces comparable or better PR-AUC results than the bootstrap method on all simulation datasets. We conducted pairwise t-tests with Benjamini Hochberg correction [8] with $n = 20$ and $\alpha = 0.05$. The results show that the improvements achieved by the RAWR-based approach were statistically significant for all model conditions except for two 100-taxon model conditions with the lowest sequence divergence. The performance of the two phylogenetic support estimation approaches was comparable on these two 100-taxon model conditions.

The PR-AUC improvement achieved by the RAWR-based approach increases as the sequence divergence grows. The PR-AUC improvement grew from 0.045 to 0.246 for the 10.A model condition to 10.E model condition, 0.009 to 0.334 for the 50.A model condition to 50.E model condition, and -0.004 to 0.291 for the 100.A to 100.E model condition. This finding indicates that the RAWR support estimates show better performance on more challenging datasets.

The largest PR-AUC improvements of the RAWR-based approaches over the bootstrap approach were 0.334 and 0.160, respectively. The average PR-AUC improvements were 0.136 and 0.039,

respectively. One possible reason is that the RAWR-based support estimation approaches conduct re-estimation for both MSAs and phylogenetic trees. While the bootstrap support estimation only includes the re-estimation of phylogenetic trees. The bootstrap resampling process breaks the intra-sequence dependence, and the bootstrap replicates lose the sequence homology and cannot produce meaningful alignments.

Model condition	PR-AUC			Corrected q-value
	Bootstrap	RAWR-reduced	RAWR	
10.A	0.951	0.989	0.996	8.2×10^{-3}
10.B	0.920	0.978	0.990	4.2×10^{-3}
10.C	0.784	0.927	0.977	4.2×10^{-3}
10.D	0.822	0.950	0.968	4.2×10^{-3}
10.E	0.679	0.976	0.925	1.5×10^{-4}
50.A	0.988	0.993	0.997	4.3×10^{-3}
50.B	0.970	0.990	0.994	5.4×10^{-4}
50.C	0.900	0.980	0.989	4.9×10^{-6}
50.D	0.798	0.981	0.988	$< 10^{-10}$
50.E	0.663	0.990	0.997	$< 10^{-10}$
100.A	0.997	0.990	0.993	$< 10^{-10}$
100.B	0.990	0.986	0.991	$< 10^{-10}$
100.C	0.828	0.971	0.982	7.2×10^{-9}
100.D	0.735	0.973	0.983	$< 10^{-10}$
100.E	0.695	0.975	0.986	$< 10^{-10}$

Table 5.5: PR-AUC performances on the simulation datasets. The PR-AUC are aggregated over n replicate datasets for a model condition ($n = 20$). Statistical significance of PR-AUC differences between RAWR and bootstrap were evaluated using a one-tailed pairwise t-test and a multiple test correction was performed using the method of [8]. Corrected q-values are reported ($n = 20$).

The runtime and peak memory usage for all methods are shown in Figure 5.2. Compared to the bootstrap method, the RAWR-based support estimation methods require one extra step: re-estimation of the resampled replicates. This additional step greatly increased the runtime of the RAWR-based approaches. The runtimes of all the methods were relatively short for the 10-taxon model conditions, which usually takes less than an hour. As the taxa number and the sequence divergence increased, the runtime increased by an order of magnitude. For the 50-taxon model conditions, the most time-consuming method, the GUIDANCE2 method, took over 10 hours for all five model conditions. For the 100-taxon model condition with the highest sequence divergence, the support estimation procedure costs half a day to multiple days. The runtime increase was mainly caused by the increase in the computational complexity of MSA estimation and phylogenetic tree

inference.

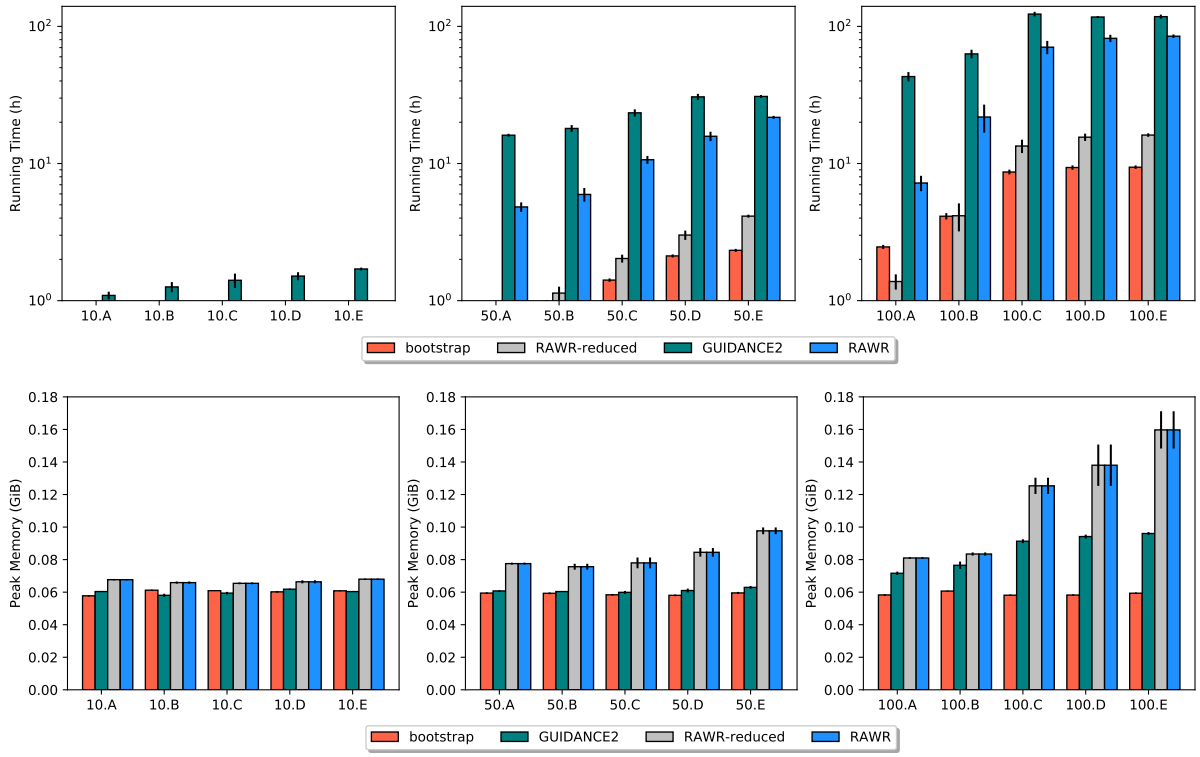


Figure 5.2: Runtime and memory usage of the phylogenetic support estimation methods on simulation datasets. The top row includes average runtime usage for each model condition in the simulated study. The y-axis shows runtime in hours and is in log-scale. The left, middle right subplots represent 10-taxon, 50-taxon, and 100-taxon model conditions respectively. The bottom row includes average memory usage for each model condition in the simulated study. The y-axis shows runtime peak memory usage in GiB. The left, middle right subplots represent 10-taxon, 50-taxon, and 100-taxon model conditions respectively. The average runtime or peak memory usage were calculated across all replicate datasets in the model condition (n = 20).

For the peak memory usage, the bootstrap method used the least amount of memory compared to the other methods, which is similar to the quantitative comparison of the runtime. The differences in the peak memory usage among all the support estimation methods were relatively small. The peak memory usage of the support estimation methods was usually about a few hundred MiB. However, according to the previous studies [80, 95] and the computational difficulty in this study, we expect that as the dataset size increases, the memory limitations will quickly become a significant bottleneck.

Overall, the simulation study experiments indicate that the RAWR-based support estimation ap-

proach improves performance compared with the widely-used bootstrap support estimation method. This improvement requires additional time and memory usage compared to a standard bootstrap analysis.

5.3.1.2 RAWR support estimation using reduced resampling replication.

In the simulation study, we included a reduced RAWR-based support estimation method, which we refer to as the "RAWR-reduced" method in the following. For the reduced RAWR-based method, we used an order of magnitude fewer resampled replicates compared with the regular RAWR-based method analysis and the bootstrap method. The standard RAWR-based method or the bootstrap method requires 100 resampled replicates, and the reduced RAWR-based approach conducts the support estimation with 10 resampled replicates.

Though the resampled replicates were reduced by an order of magnitude, the RAWR-reduced method produced comparable or better performance than the bootstrap method. Like the standard RAWR method, the RAWR-reduced method achieved greater PR-AUC improvements under model conditions with a large dataset size or higher sequence divergence. In comparing the PR-AUC of the RAWR-reduced method to the regular RAWR method, there was an average improvement of 0.007 across all model conditions, which is shown in Table 5.5.

The RAWR-reduced method used slightly more runtime than the bootstrap method on all model conditions and less runtime than the regular RAWR method. The RAWR-reduced method used similar peak memory usage compared to the regular RAWR method. The RAWR-reduced method uses much less runtime and comparable peak memory by reducing the resampled replicates, but the performance is comparable or even better than the regular RAWR method.

5.3.1.3 Results of Additional Performance Study

As in other performance studies of MSA and phylogenetic tree estimation from unaligned sequence inputs [79, 80], we found that MAFFT generally produced more accurate alignments than ClustalW on the 10-taxon model conditions, although this accuracy improvement did not translate directly to

Model condition	PR-AUC	
	MAFFT	ClustalW
10.A	0.996	0.997
10.B	0.990	0.991
10.C	0.977	0.988
10.D	0.968	0.992
10.E	0.925	0.964

Table 5.6: RAWR support estimation using alternative estimation/re-estimation methods. We compared RAWR support estimation using two different estimation/re-estimation methods: either MAFFT and RAxML(MAFFT) or ClustalW and RAxML(ClustalW). For each of the two methods, aggregate PR-AUC is shown across all replicate datasets of each model condition ($n = 20$).

more accurate downstream phylogenetic inference. The details of the alignment summary statistics refer to Table 5.3 and Table 5.1.

Despite the alignment accuracy, RAWR returned comparable PR-AUC regardless of which of the two MSA methods were used on the 10.A and 10.B model conditions. On the more divergent 10.C through 10.E model conditions, when using ClustalW for estimation and re-estimation of the MSA, RAWR improved the PR-AUC performance by 0.011, 0.024, and 0.039 comparing to that used MAFFT. The results are shown in Table 5.6. Our finding suggests that RAWR support estimation is robust to the quality of the annotation MSA. Furthermore, this result suggests that neighbor-preserving random walks may yield better support estimates where computational problems are more challenging, and estimation uncertainty is greater.

For the experiments where we used alternative choices for reversal probability γ for the RAWR support estimation, on each 10-taxon model condition except for the 10.C model condition, RAWR returned similar PR-AUC as the reversal probability γ was increased from 0.001 up until a critical threshold. PR-AUC then dropped as γ increased past the threshold (Table 5.7). The exact threshold varied somewhat across model conditions. More generally, we observed a range of RAWR γ settings that returned the highest PR-AUC, where the range typically crossed one to two orders of magnitude.

For the experiments on long-gap-length model conditions, similar performance outcomes were observed compared to the medium-gap-length simulations in the rest of our simulation study. The RAWR method produced a comparable or better PR-AUC than bootstrap. The improvement

Model condition	Reversal probability γ						
	1×10^{-3}	1×10^{-2}	2×10^{-2}	5×10^{-2}	1×10^{-1}	2×10^{-1}	3×10^{-1}
10.A	0.997	0.998	0.998	0.996	0.994	0.986	0.980
10.B	0.994	0.990	0.991	0.990	0.987	0.985	0.977
10.C	0.942	0.941	0.950	0.977	0.968	0.957	0.901
10.D	0.977	0.982	0.978	0.95	0.944	0.935	0.934
10.E	0.969	0.978	0.971	0.983	0.929	0.923	0.922

Table 5.7: Simulation study: RAWR support estimation using different choices for reversal probability γ. Aggregate PR-AUC is reported across all replicate datasets of each 10-taxon model condition ($n = 20$).

Model condition	PR-AUC	
	Bootstrap	RAWR
10.long.A	0.997	0.997
10.long.B	0.992	0.994
10.long.C	0.904	0.937
10.long.D	0.829	0.949
10.long.E	0.552	0.788

Table 5.8: PR-AUC comparison of bootstrap and RAWR methods on 10-taxon long-gap-length model conditions. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. Each method's PR-AUC is reported as an aggregate across all replicate datasets for a model condition ($n = 20$).

achieved by the RAWR method was as much as 0.236. The RAWR-based method achieved a larger PR-AUC advantage on model conditions with greater evolutionary divergence. Both methods returned slightly lower PR-AUC for the long-gap-length model conditions compared to the medium-gap-length model conditions. But the PR-AUC improvements obtained by the RAWR-based support estimation method over the bootstrap method were largely unaffected by the gap length distribution used for simulation. This suggests that RAWR's performance is robust to increasing dependence between neighboring sites due to longer insertions and deletion events.

For the PR-AUC comparisons of aLRT and RAWR-based methods on simulated datasets, the RAWR-based support estimation method consistently produced a better PR-AUC than the aLRT methods over all model conditions in the simulation study. The results are shown in Tables 5.9. Furthermore, as the sequence divergence increases, the PR-AUC improvement of the RAWR support estimation method tended to increase. The original aLRT method was proposed to address the high computational cost of the standard phylogenetic bootstrap method, which requires re-estimation of multiple bootstrap replicates. The computational efficiency of the LRT method

Model condition	PR-AUC		
	aLRT with free topology	aLRT with fixed topology	RAWR
10.A	0.952	0.939	0.996
10.B	0.884	0.884	0.990
10.C	0.722	0.745	0.977
10.D	0.757	0.784	0.968
10.E	0.631	0.621	0.925
50.A	0.979	0.980	0.997
50.B	0.960	0.961	0.994
50.C	0.870	0.876	0.989
50.D	0.711	0.710	0.988
50.E	0.548	0.556	0.997
100.A	0.986	0.987	0.993
100.B	0.976	0.969	0.991
100.C	0.775	0.773	0.982
100.D	0.663	0.670	0.983
100.E	0.592	0.593	0.986

Table 5.9: PR-AUC comparison of aLRT and RAWR methods for phylogenetic support estimation. We used PhyML [49] to run two types of aLRT analyses: (1) support estimation or a free tree topology that was also estimated as part of the analysis, and (2) support estimation for a RAxML-inferred tree topology. The latter methodology for obtaining an annotation tree is identical to the approach used in all other experiments in our study, and its PR-AUC performance is therefore directly comparable to other simulation study experiments. Table layout and description are otherwise identical to Table 5.8.

is obtained using statistical approximations that represent a potential tradeoff in terms of type I/II error. Our findings support these observations, as the PR-AUC achieved by both the aLRT methods also underperformed the traditional bootstrap support estimation. We note one critical difference between our study and the study of Anisimova and Gascuel [3]. Our study provided estimated annotation MSAs as input to phylogenetic support estimation methods, whereas the study of Anisimova and Gascuel utilized the true alignments. The relative performance comparisons of aLRT and RAWR can be attributed in part to the major impact of MSA quality on downstream phylogenetic and phylogenetic support estimation.

In the PR-AUC comparisons of the TBE and RAWR methods on simulated datasets, the RAWR method consistently outperformed the original TBE method [74] over all model conditions in our simulation study. One exception was the least divergent 100-taxon model condition, where both methods returned comparable PR-AUC. The PR-AUC comparisons of the TBE and RAWR methods are shown in Table 5.10.

TBE support calculation is downstream of input data resampling. The original TBE support

estimation method utilized standard bootstrap resampling, which we refer to as TBE+bootstrap. As noted in the original study of TBE [74], the orthogonality of these two problems allows other resampling techniques to be easily replaced. Thus, we included a third method that joined TBE support calculation with RAWR resampling, which we refer to as TBE+RAWR. TBE+RAWR returned comparable or improved PR-AUC compared to TBE+bootstrap, and the largest improvements were obtained on the most divergent model conditions.

However, neither of the TBE methods outperformed standard RAWR support estimation, which uses a traditional binary test for the presence or absence of bipartition to calculate phylogenetic support. We also did not observe PR-AUC comparisons that suggested a type I/II error advantage for the original TBE method over traditional phylogenetic bootstrap support estimation. Our findings differ from the original study of the TBE method [74], which we attribute to the following factors. As noted above, a major difference between the two studies is MSA quality: the former utilizes estimated MSAs, and the latter utilizes true MSAs. Furthermore, Lemoine et al. [74] noted that, by definition of the bipartition transfer distance, TBE support is always greater than or equal to traditional bootstrap support for a given set of inputs. Based on our findings, we conjecture that an optimistic support measure is beneficial for addressing type II errors but could be counterproductive for type I errors.

Among other important assumptions, for example, treating indels as missing data or an additional state, theoretical guarantees about TBE and phylogenetic bootstrap support implicitly assume that input sequences are aligned without error. However, incorrect sequence homology and other misalignments will require a different set of theoretical and applied considerations. Our experiments suggest that sequence-aware resampling and re-estimation have an important role to play in phylogenetic support estimation.

We also compared the performance of RAWR versus GUIDANCE2, a state-of-the-art purpose-built fully parametric method for placing confidence intervals on estimated multiple sequence alignments. The application of an MSA confidence assessment method like GUIDANCE2 to the downstream task of phylogenetic support estimation differs from its original intended purpose.

Model condition	PR-AUC		
	TBE with bootstrap resampling	TBE with RAWR resampling	RAWR
10.A	0.943	0.982	0.996
10.B	0.913	0.959	0.990
10.C	0.773	0.894	0.977
10.D	0.823	0.924	0.968
10.E	0.670	0.869	0.925
50.A	0.986	0.983	0.997
50.B	0.965	0.967	0.994
50.C	0.888	0.943	0.989
50.D	0.785	0.950	0.988
50.E	0.655	0.968	0.997
100.A	0.995	0.984	0.993
100.B	0.985	0.962	0.991
100.C	0.806	0.922	0.982
100.D	0.722	0.934	0.983
100.E	0.680	0.930	0.986

Table 5.10: PR-AUC comparison of TBE with bootstrap resampling, TBE with RAWR resampling, and RAWR. TBE was used to estimate phylogenetic support using two different resampling approaches: either (1) standard bootstrap resampling, which corresponds to the method originally proposed and studied by Lemoine et al. [74], or (2) RAWR resampling. The former is denoted "TBE with bootstrap resampling", and the latter is denoted "TBE with RAWR resampling". For comparison purposes, RAWR resampling and re-estimation was also run as a third method (denoted "RAWR"), and we used the same methodology as elsewhere in our study (i.e., using a standard branch presence/absence calculation to assess phylogenetic support).

However, we note that GUIDANCE2 incorporates standard bootstrap resampling as a first step, and subsequent steps focus on guide tree re-estimation and other re-estimation tasks as part of progressive MSA re-estimation. For this reason, GUIDANCE2 can be seen as an adaptation of the standard bootstrap to MSA and tree re-estimation.

The performance comparison between RAWR and GUIDANCE2 was qualitatively similar to that of RAWR and bootstrap. RAWR returned comparable or better PR-AUC compared to GUIDANCE2 on the simulated datasets, and RAWR's PR-AUC advantage over GUIDANCE2 tended to grow as model conditions grew larger and more divergent. The results show in Table 5.11). GUIDANCE2 was the slowest method overall due to the complexity of its special-purpose MSA re-estimation approach, and both GUIDANCE2 and RAWR required more main memory compared to bootstrap.

Finally, we note that GUIDANCE2 is purpose-built for MSA re-estimation, whereas bootstrap and RAWR are general-purpose non-parametric resampling methods, both resample an MSA

Model condition	PR-AUC	
	GUIDANCE2	RAWR
10.A	0.989	0.996
10.B	0.983	0.990
10.C	0.921	0.977
10.D	0.939	0.968
10.E	0.997	0.997
50.B	0.994	0.994
50.C	0.975	0.989
50.D	0.942	0.988
50.E	0.837	0.997
100.A	0.988	0.993
100.B	0.993	0.991
100.C	0.939	0.982
100.D	0.894	0.983
100.E	0.881	0.986

Table 5.11: PR-AUC comparison of GUIDANCE2 and RAWR phylogenetic support estimation methods. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively. We report each method's aggregate PR-AUC across all replicate datasets for a model condition ($n = 20$).

Model Condition	PR-AUC
10.A	0.998
10.B	0.99
10.C	0.978
10.D	0.966
10.E	0.964

Table 5.12: PR-AUC performance of RAWR+teleport on 10-taxon model conditions.

without utilizing an explicit parametric model. Despite this, RAWR was able to match or exceed GUIDANCE2's PR-AUC performance.

For the simulation study experiments with an alternative random walk resampling procedure, the PR-AUC values returned by RAWR+teleport on the 10-taxon model conditions are shown in Table 5.12. In terms of PR-AUC, RAWR+teleport had comparable performance to the standard RAWR method. For the model conditions in our experiment, downstream re-estimation and support calculations may be relatively tolerant of discontinuities introduced by teleportation, at least relative to related desynchronization injected by random reversals in a standard RAWR resampled replicate. Further experimentation will help to clarify the tradeoffs between the different resampling methods.

5.3.2 Empirical Study

5.3.2.1 Performance comparison of RAWR versus bootstrap.

We evaluated the RAWR-based methods versus the bootstrap methods based on their PR-AUC on the empirical datasets, shown in Table 5.13. The RAWR method returned a similarly better PR-AUC compared to the bootstrap method, which is consistent with the results of the simulation study. The RAWR-based method outperformed the bootstrap method on all empirical datasets, and the average improvement was 0.105. This result indicates that the RAWR-based support estimation method can handle large datasets containing hundreds of sequences and still produce accurate support estimations than the traditional bootstrap method.

There are some differences between the empirical study and the simulation study that are worth noting. First, the reference trees do not equal the true evolutionary history. The reference trees used in the empirical study were inferred from the reference alignment by the MLE method. Although the reference alignments were aligned by an automatic method and manual correction, and the alignments are very accurate, there is no guarantee that the reference alignments are true alignments. While in the simulation study, the model trees and the true MSAs are the ground truth. The model trees were used to guide the simulation of the sequence evolution. Secondly, the number of empirical datasets is different from the number of simulation datasets. This is caused by the large amount of effort required to curate reference alignments for empirical datasets. Finally, the intronic rRNA datasets have similar summary statistics compared to the simulated datasets. However, the intronic rRNA datasets are not exactly the same as the simulated datasets. The selected empirical datasets involve secondary structure evolution, strong selective pressures, and other evolutionary and biophysical constraints that are not considered in the simulation process in this study.

5.3.2.2 Results of Additional Performance Study

The PR-AUC comparisons of the aLRT and RAWR methods on the empirical benchmarking datasets were consistent with the simulation datasets. RAWR consistently returned PR-AUC improvements

Model condition	PR-AUC	
	Bootstrap	RAWR
IGIA	0.725	0.804
IGIB	0.629	0.695
IGIC2	0.778	0.957
IGID	0.670	0.884
IGIE	0.772	0.808
IGIIA	0.830	0.884

Table 5.13: PR-AUC performances on the emprircal datasets. PR-AUC comparison of bootstrap and RAWR methods for phylogenetic support estimation. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively.

relative to both aLRT methods across all of the empirical benchmarking datasets. The results are shown in Table 5.14.

Dataset	PR-AUC		
	aLRT with free topology	aLRT with fixed topology	RAWR
IGIA	0.6696	0.7094	0.7845
IGIB	0.4904	0.5515	0.8332
IGIC2	0.6368	0.7242	0.8808
IGID	0.7998	0.7299	0.8524
IGIE	0.6832	0.6864	0.8206
IGIIA	0.7774	0.8036	0.9053

Table 5.14: Empirical study: PR-AUC comparison of aLRT and RAWR methods on CRW benchmarking datasets. Table layout and description are otherwise identical to Table 5.9.

The PR-AUC comparisons of the TBE and RAWR methods on the empirical benchmarking datasets were also consistent with the simulated datasets. The RAWR-based support estimation method consistently outperformed the original TBE method [74] across all empirical benchmarks. The results are shown in Table 5.15).

Dataset	PR-AUC		
	TBE with bootstrap resampling	TBE with RAWR resampling	RAWR
IGIA	0.7100	0.7195	0.7845
IGIB	0.6194	0.7232	0.8332
IGIC2	0.7508	0.8559	0.8808
IGID	0.5915	0.8044	0.8524
IGIE	0.7235	0.7622	0.8206
IGIIA	0.8252	0.8524	0.9053

Table 5.15: PR-AUC comparison of TBE and RAWR methods on CRW benchmarking datasets. Table layout and description are otherwise identical to Table 5.9.

The performance comparison between RAWR and GUIDANCE2 was similar on the empirical datasets compared to the simulated datasets. The RAWR-based support estimation method produced

comparable or better PR-AUC compared to GUIDANCE2. The results are shown in Table 5.16, RAWR outperformed GUIDANCE2 on all empirical benchmarks except for IGIB. The average absolute difference between the two methods' PR-AUC values was 0.055. This discrepancy may be attributed to the relative difficulty that the IGIB dataset presents: the worst PR-AUC values in our entire study were observed on this dataset. One primary factor for this outcome is the high gappiness of the reference alignments for the IGIB and IGIA datasets, for example, the fraction of the reference alignment that consists of indels, as compared to every other dataset in our study. RAWR resampling of datasets with high gappiness may require additional safeguards to mitigate desynchronization.

	PR-AUC	
Dataset	GUIDANCE2	RAWR
IGIA	0.705	0.804
IGIB	0.737	0.695
IGIC2	0.874	0.957
IGID	0.740	0.884
IGIE	0.777	0.808
IGIIA	0.870	0.884

Table 5.16: PR-AUC comparison of GUIDANCE2 and RAWR methods for phylogenetic support estimation. MAFFT and RAxML(MAFFT) were used to perform MSA and tree estimation/re-estimation, respectively.

5.4 Discussion

The results of the simulation study and the empirical study indicate that the neighbor preservation property plays a critical role in the phylogenetic support estimation. For the phylogenetic support estimation problem, the regular RAWR-based method and the RAWR-reduced method outperformed the bootstrap method on almost all the simulation model conditions and the empirical datasets. The difference between the RAWR-based methods and the bootstrap method is that the RAWR resampling approach produces the resampled replicates that retain the intra-sequence dependence followed by the alignment re-estimation, while bootstrap replicates do not. Meaningful sequence homology is retained in the RAWR replicates, which makes it possible to re-estimate alignments from the RAWR replicates. Therefore, the phylogenetic trees are inferred from the re-estimated alignments following the same two-phase pipeline and serve as a better data perturbation

source for the phylogenetic support estimation.

Compared with the SERES resampling process, the RAWR resampling process is more similar to the bootstrap method. There is no need for anchor estimation for synchronization purposes, and the parameters involved in the anchor estimation steps are unnecessary. The random walk procedure improved the HoT algorithm by producing many replicates with reverse orientation. More replicates usually result in better performance when it comes to support estimation.

However, there are problems associated with the resampled replicates. First, synchronization may be broken during the resampling process by frequent reversal of the direction. If true alignments were accessible, all sites would be correctly aligned. In this case, breakpoints caused by the direction change do not break the synchronization of the sequence homology. However, true alignments are usually not available for analysis. Using estimated alignment, which contains mismatches near the reversal breakpoints, the synchronization is interrupted during the resampling process. This may impact the downstream re-estimation. A relatively low reversal probability limits the number of unique resampled replicates. However, high reversal probability results in more reversal breakpoints, which may break the synchronization due to the inaccuracy of the estimated alignments. Our experiments show that the choice of reversal probability has little impact on the downstream support estimates if we use a relatively low reversal probability. But the impact of the choice of reversal probability is still worth noting.

Another problem with the RAWR resampling method is that a high reversal probability will reduce the sequential dependence of the RAWR resampled replicates. To the extreme, the bootstrap replicates do not take into account the sequential dependence. The RAWR resampling with $\gamma = 0$ is equivalent to mirrored inputs but with a random start point, and reverse only at the start or end of the input alignments. The RAWR resampling with $\gamma = 0.5$ is a first-order Markovian process, as discussed in [145]. The RAWR resampling with $0 < \gamma < 0.5$, where γ is relatively small, the resampled replicates contain more sequential dependence. According to our experiments on the choice of reversal probability γ, smaller reversal probabilities are likely to be more practical for most applications.

The resampled site distribution depends on the choice of reversal probability γ. For an extreme example, the RAWR resampling with $\gamma = 1$ is equivalent to sampling a single site many times and leads to a significant data loss. Even if we use a relatively small reversal probability, an individual RAWR replicate can still be biased around the initial start position. Also, the RAWR random walk certainly changes direction at the start and end of the input alignment, which can also introduce RAWR resampling bias.

One solution to avoid the sampling bias is to sample a sufficiently large set of RAWR replicates with a relatively small reversal probability and aggregate the statistical inference. The RAWR start positions are chosen uniformly at random. We explored the sampling frequency of the input alignment sites for the RAWR resampling and found that the results are consistent with the above hypothesis. Therefore, the RAWR support estimation is less affected by the resampling bias since many resampled RAWR replicates are re-estimated and aggregated into a single support estimation.

The bootstrap support estimation usually treats the input alignment as the true alignment and does not consider the alignment errors. However, true alignments are not available in practice, and the alignment errors are commonly found in the estimated alignments. The RAWR-based support estimation benefits from the inaccuracy of the input alignment. The RAWR replicates serve as an effective data augmentation source, which provides more alternative inference results for the MSA and the tree estimation.

This study applies the RAWR resampling and re-estimation to sequence dependence due to insertion and deletion processes. Below, we highlight other factors as part of future work. Many other factors also lead to sequence dependence, which we do not take into account yet. The new random walk resampling and re-estimation techniques paved the road toward solving the i.i.d. assumptions in biomolecular sequence analysis and other topics. There is still more progress to be made.

5.5 Conclusion

In this study, we introduced a new non-parametric resampling technique, RAWR. We applied the RAWR resampling approach to a classical bioinformatics problem, phylogenetic support estimation. We conducted experiments on both simulated and empirical datasets that covered a wide range of dataset sizes and sequence divergence. The performance study showed that the RAWR-based support estimation method produced comparable or better performance than the widely used bootstrap support estimation method. At the same time, the RAWR-based support estimation methods request longer runtime and memory usage due to the RAWR resampling and re-estimation procedure. The tradeoff between accuracy and computational runtime and memory can be offset by reducing the number of resampled replicates.

For future research, application-specified resampling and re-estimation can better utilize the sequence dependence in biomolecular sequence analysis. For example, taking the structure information into account for the resampling procedure. Another direction is applying the non-parametric resampling that reserves the intra-sequence dependence to other biological analyses that require biomolecular sequences as input or account for the sequential dependence. The non-parametric resampling methods require fewer assumptions and are not restricted to a specific application. Therefore, the RAWR resampling approach can be easily applied to other problems that deal with sequencing data. One important example is the statistical inference of species trees under models of sequence evolution that take into account the sequential dependence [139, 146].

CHAPTER 6

AN APPLICATION OF RANDOM WALK RESAMPLING TO PHYLOGENOMIC ANALYSIS OF DARWIN'S FINCHES

6.1 Introduction

We proposed the RAWR resampling algorithm to simplify the SERES random walk and serve the phylogenetic support estimation problem. The RAWR-based method estimates the support of phylogenetic trees by resampling and re-estimation using unaligned DNA sequences. The RAWR-based phylogenetic support estimation shows comparable or typically better performance than the traditional bootstrap support estimation approach.

In previous experiments, we tested the RAWR resampling and re-estimation phylogenetic support estimation approach on single loci. Traditionally, phylogenetic inference and learning assume that all sites of the input alignment evolve along the same tree, and many population-level evolutionary events are ignored during the phylogenetic tree reconstruction.[34, 45]. Thus phylogenetic tree reconstruction is often conducted on a single locus or a few loci. Such approaches may work well on distantly related taxa. However, for closely related species, due to evolutionary events such as incomplete lineage sorting (ILS) and horizontal gene flow, a phylogenetic tree inferred from one or a few given loci may be inconsistent with the species tree. The inconsistent gene tree topologies may be obtained across the entire genome[89, 106, 30].

As the costs of next-generation sequencing techniques drop, sequencing a large number of genes or even whole genomes is feasible for phylogenetic inference and learning. Many computational algorithms are designed for coalescent-based species tree inference, which takes multiple gene trees as input and reconstructs the species tree with the presence of both ILS and horizontal gene flow. The whole-genome sequencing dataset is challenging in both its size and the variance of sequence divergence. In this study, we compared the RAWR-based support estimation method and the standard bootstrap method on the whole-genome sequencing dataset of Darwin's finch from

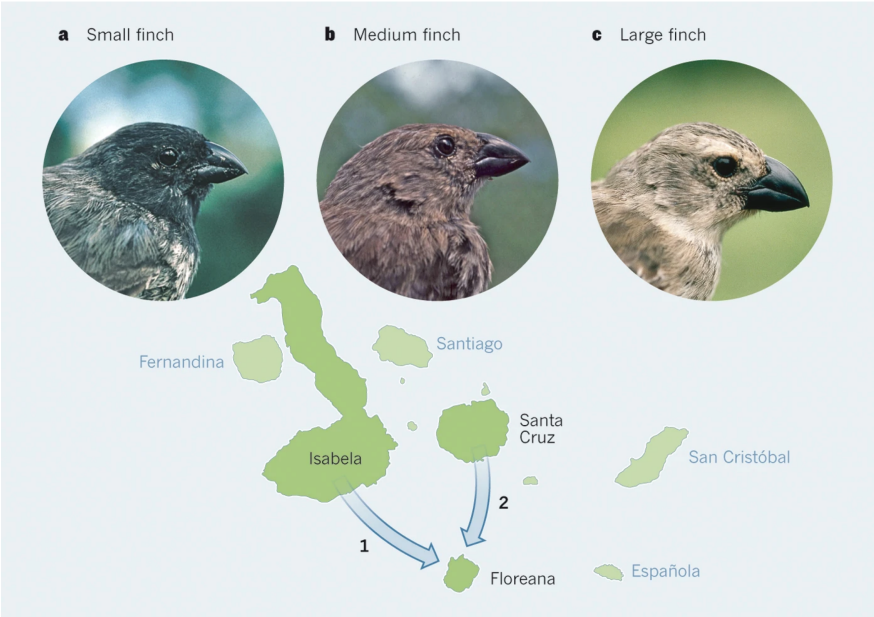


Figure 6.1: Pictures of three Darwin's finch species. (a) The small tree finch *Camarhynchus parvulus*. (b) The medium tree finch *Camarhynchus pauper*. (c) The large tree finch *Camarhynchus psittacula*. All three species live on the Galapagos island of Floreana. Arrows indicate the migration of two populations of *C. psittacula* from Isabela and Santa Cruz. Figure comes from [47].

Larmichhaney's study in 2015 [69]. We also contributed an analysis pipeline for the phylogenetic tree reconstruction and support estimation from the raw sequencing data.

6.2 Methods

6.2.1 Dataset

In this study, we reanalyzed the whole-genome sequencing data of Darwin's finches that were initially studied by Lamichhaney et al. 2015 study [69].

Lamichhaney's study conducted phylogenetic analysis on the whole-genome sequencing data of 120 individuals representing all of the Darwin's finch species and two close relatives, in total 25 species. We randomly selected one sample for each species, resulting in a dataset with 25 samples (accession numbers SRR1607296, SRR1607504, SRR1607439, SRR1607359, SRR1607385, SRR1607440, SRR1607547, SRR1607403, SRR1607458, SRR1607472, SRR1607551, SRR1607494,

SRR1607399, SRR1607462, SRR1607343, SRR1607534, SRR1607406, SRR1607485, SRR1607508, SRR1607543, SRR1607365, SRR1607420, SRR1607466, SRR1607529, and SRR1607480). We downloaded the raw Illumina HiSeq2000 paired-end short read data from the NCBI SRA database (accession number PRJNA263122 at http://www.ncbi.nlm.nih.gov/sra).

The Lamichhaney study used the reference genome of medium ground finch *Geospiza fortis* for mapping and variance calling purposes. We used the same reference genome and gene annotations in our analysis. The reference genome and annotation file were downloaded from the GigaDB database (http://gigadb.org/dataset/100040).

6.2.2 Process of the raw sequencing data

Due to limitations in software and script availability, we explored an alternative process and analysis pipeline for the phylogenetic tree reconstruction and support estimation of the whole-genome sequencing raw data. After the raw NGS read data was downloaded, the raw data went through NGS read mapping, quality filtering, variant calling, and phasing steps to obtain a multi-locus sequence dataset. The details of the raw reads processing are described as below.

First, the paired-end raw reads of each sample were mapped to the reference genome using BWA version 0.7.17 with default parameters [75]. Then the mapped results were filtered based on the mapping quality and sorted by coordinate order using SAMtools [76]. We also used SAMtools to conduct variant calling on the preprocessed data to identify variants. The variants include SNPs and short indels polymorphisms. To obtain the haplotype sequences, those bi-allelic SNPs were phased using fastPHASE version 1.4.8 [125]. The phased calls for bi-allelic SNPs were combined with the genotypic data of homozygous multi-allelic SNPs and homozygous indel polymorphisms. The heterozygous multi-allele SNPs and heterozygous indels, representing less than 1% of the input data, were treated as missing data. In total, we got 28,507 scaffold sequences for each haplotype.

The gene annotation data provided with the reference genome include coordinate information of locus corresponded to a gene, an intergenic region between annotated genes on a scaffold, or a scaffold with no annotated genes. We extracted loci sequences based on the coordinate information

of the gene annotations and the scaffold sequences produced in previous steps. Then we estimated alignment for each locus using MAFFT version 7.222 [64] with default settings. Finally, estimated alignments were filtered by the sequence length to achieve better efficiency. Very short alignments will cause errors in the following tree inference analysis, while very long alignments will cost much time for the resampling process. Alignments with the length between 30bp and 1.5Mb were used in the following analysis. After filtering, we got 34,972 alignments for each haplotype. The final multi-locus sequence dataset consisted of 13,321 loci corresponding to annotated genes, 15,275 loci corresponding to intergenic regions between annotated genes on a scaffold, and 6,376 loci corresponding to scaffolds that lacked annotated genes.

6.2.3 Concatenated MLE phylogenetic tree inference

We concatenated multiple sequence alignments of all loci to produce a concatenated genomic sequence alignment A for phylogenetic tree inference and support estimation. The i^{th} alignment a_i served as partition p_i in the concatenated alignment and the following analyses. To obtain an annotation phylogeny for the phylogenetic support estimation methods under study, we estimated a species tree using maximum likelihood estimation (MLE) on the concatenated and partitioned genomic sequence alignment A. The concatenated alignment A had a sequence length of multiple orders of magnitude larger than the other datasets that we used in our previous studies. The RAxML algorithm we used in previous studies was inefficient for such big, partitioned phylogenomic datasets. Therefore, we utilized ExaML [67] to perform MLE phylogenetic tree inference. The ExaML algorithm was designed for large-scale phylogenetic analyses by performing parallelized computation on a high-performance computing cluster. First, we inferred an initial tree by maximum parsimony optimization using RAxML version 8.2.9 [131]. The start tree was used for ExaML's local search heuristics. Then the concatenated and partitioned alignment A were filtered only to contain partitions where all four types of nucleotides. Partitions without full coverage of all types of nucleotides will cause errors during the substitution model parameters inference process. The filtered alignment was transformed into a binary file format by a dedicated parser component of

ExaML version 3.0.21. The binary alignment file contains global data information such as alignment length, data types, and partition boundaries, allowing each ExaML process to concurrently read only those parts of the alignment on which it will be computing likelihoods and greatly improving the computational efficiency. Then the ExaML version 3.0.21 was used to perform phylogenetic MLE on the concatenated and partitioned alignment A.

6.2.4 Phylogenetic support estimation using bootstrap resampling.

We used the standard bootstrap method to sample 100 bootstrap replicates from the concatenated and partitioned genomic sequence alignment A. The bootstrap resampling method was implemented in RAxML version 8.2.9. The bootstrap replicates were filtered only to contain partitions containing all four types of nucleotides. We used ExaML to transform the filtered alignment into binary format and used RAxML to infer an initial tree for each bootstrap replicate. Then, for each bootstrap replicate, an MLE tree was inferred by the ExaML using the initial tree and binary file as input. The software version and commands we used to perform phylogenetic MLE inference were the same as for annotating tree estimation. The re-estimated trees of bootstrap replicates were used to estimate phylogenetic bootstrap support for the annotation tree that we estimated using the original concatenated and partitioned alignment A.

6.2.5 Phylogenetic support estimation using RAWR resampling.

We conducted the RAWR random walk on each partitioned alignment a_i of the original alignment A with default reverse rate $\gamma = 1 \times 10^{-1}$. For each estimated alignment a_i, we used the RAWR resampling approach to obtain 100 resampled replicates $\{b_{ij}\}$ for $1 \leq j \leq 100$.

To produce a single concatenated and partitioned alignment for each resampled replicate, we concatenated estimated alignments $\{b_{ij}\}$ across all partitions p_i for one resampled replicate j, where $1 \leq j \leq 100$. As in the bootstrap analyses, the concatenated alignments were filtered to contain partitions that cover all four types of nucleotides. Phylogenetic re-estimation was conducted on each RAWR replicate alignment with the same procedure and parameter settings as the bootstrap

support estimation. The re-estimated RAWR trees were used to calculate support for the annotation tree. We used Dendroscope [62] to visualize phylogenetic support estimates on the species tree.

6.3 Results and Discussion

We produced a phylogenetic tree for Darwin's finches using the MLE phylogenetic tree inference method on the concatenated alignment of the whole-genome sequence data. The inferred phylogenetic tree, which is shown in Figure 6.3, was topologically identical to the phylogenetic tree provided by Lamichhaney's study [69], which is shown in Figure 6.2, except for one single internal branch in the tree of finches clade. The phylogenetic tree produced by the original study showed that the *C. pauper* has a closer relationship with *C. parvulus*, and then the clade of these two species coalescents with the other tree finch species. The species tree we got from our MLE analysis showed that instead of *C. parvulus*, *C. pauper* is genetically closer to the other three tree finch species, *C. psittacula*, *C. heliobates*, and *C. pallidus*. This result indicates that the analysis pipeline we used in our study produced a reasonable phylogenetic tree, which is mostly consistent with the tree provided by the original study using the whole-genome sequencing data. The change in raw sequencing data processing tools and pipelines did not impact the downstream phylogenetic tree reconstruction, unlike the MSA estimation, where the choice of MSA methods greatly influences the accuracy of downstream phylogenetic analysis.

The inconsistent tree finch clade did not mean that the phylogenetic tree inferred in this study was wrong. We found that two support estimation methods produced different estimations for this inconsistent clade. The supports produced by the RAWR-based method are shown in 6.4, and the supports produced by the bootstrap method are shown in 6.3. The RAWR-based support estimation approach produced lower support than the bootstrap method for the tree finch clade. The RAWR-based support estimation approach yielded 46% support on the parent edge for the clade of *C. psittacula*, *C. heliobates*, and *C. pallidus*, and 68% support on the parent edge of *C. psittacula*, *C. heliobates*, *C. pallidus*, and *C. pauper*. However, the support produced by the bootstrap method is 100% for the two internal branches mentioned above. The different support

estimations produced by the two methods on the inconsistent clade indicate that there are more uncertainties in the phylogenetic relationships of the tree finch species. Thus, it is more difficult to obtain a confident inference from the sequencing data for this set of finch species. Furthermore, our findings suggest that the RAWR resampling and re-estimation process aids in revealing the uncertainties of the downstream phylogenetic tree reconstruction, particularly variances introduced by MSA estimation. The bootstrap method is not sensitive to the influence of MSA quality on the downstream phylogenetic reconstruction since the bootstrap replicates lose intra-sequence dependence during the resampling process. Minor discrepancies were also noted in the non-sharp-beaked ground finch clade, at most a 7% difference. Otherwise, both methods closely agreed on estimated bipartition support in the rest of the species phylogeny, with an average support difference of 0.6%.

The RAWR-based support estimation took a longer runtime than the standard bootstrap support estimation method on the same multi-locus sequencing dataset because the RAWR-based method requires an additional step than the bootstrap method: the re-estimation of the resampled replicate sequences. This additional step significantly increased the runtime of the RAWR-based approaches. This is consistent with our previous study on the single-locus dataset, where the runtime of the RAWR-based support estimation was an order of magnitude longer than the bootstrap method. Parallel computing can easily be applied to the re-estimation of resampled replicates to solve this problem.

6.4 Conclusion

In this study, we reanalyzed the whole-genome sequencing data of Darwin's finches that were initially studied by Lamichhaney et al. 2015 study[69]. We explored the application of RAWR-based support estimation on the multi-locus sequence dataset. We reproduced the phylogenetic tree inference by concatenated MLE analysis. Though inferring phylogenetic relationships of birds is often challenging because of frequent hybridization and rapid radiations (Grant and Grant 1992; Ericson et al. 2006), the reconstructed phylogeny shares an identical topology with the tree

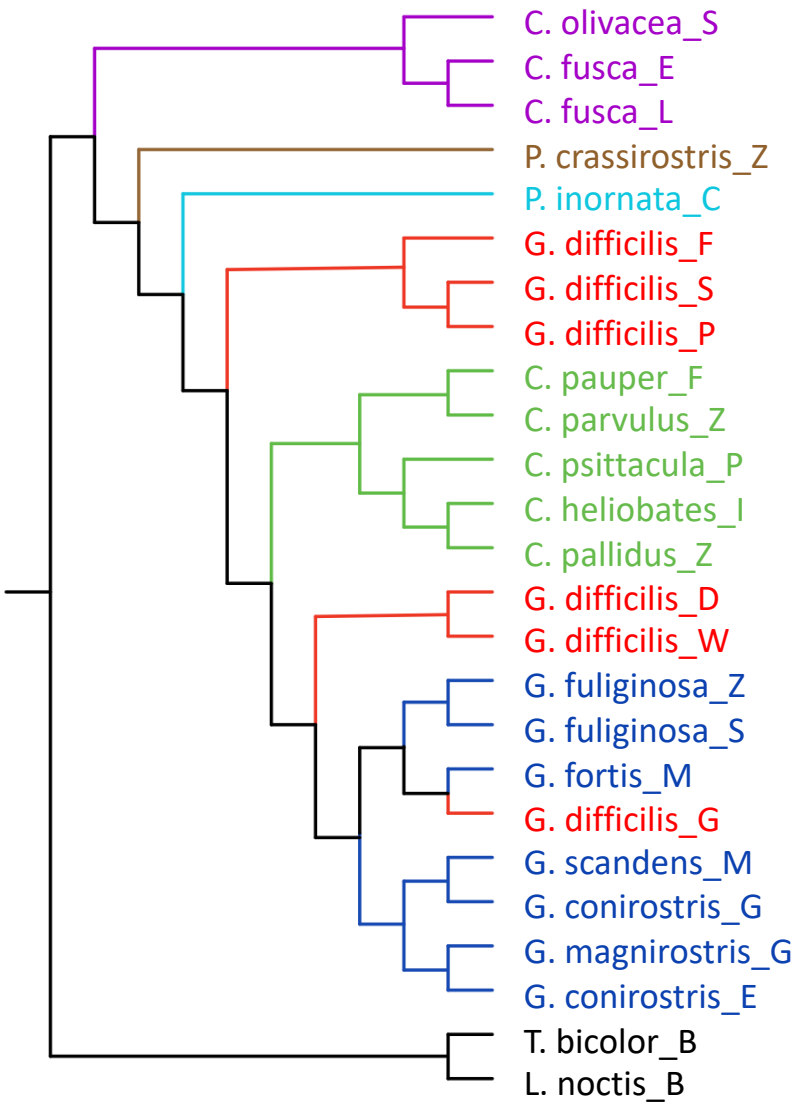


Figure 6.2: Phylogenetic tree of Darwin's finch species and two close relatives reported in Lamichhaney et al. 2015 study[69]. This phylogenetic tree was reproduced from the Figure 1 panel b in Lamichhaney's 2015 paper. The branch length were ignored and the tree was rescaled to an ultrametric tree, where all the leaves have the same distance to the root. The color of the branches and the species name represents the group that the species belongs to. We used the same color as the original study, the purple, brown, cyan, red, green, blue and black color represent the group of warbler finches, vegetarian finch, cocos finch, sharp-beaked ground finches, tree finches, all other ground finches and the outgroups.

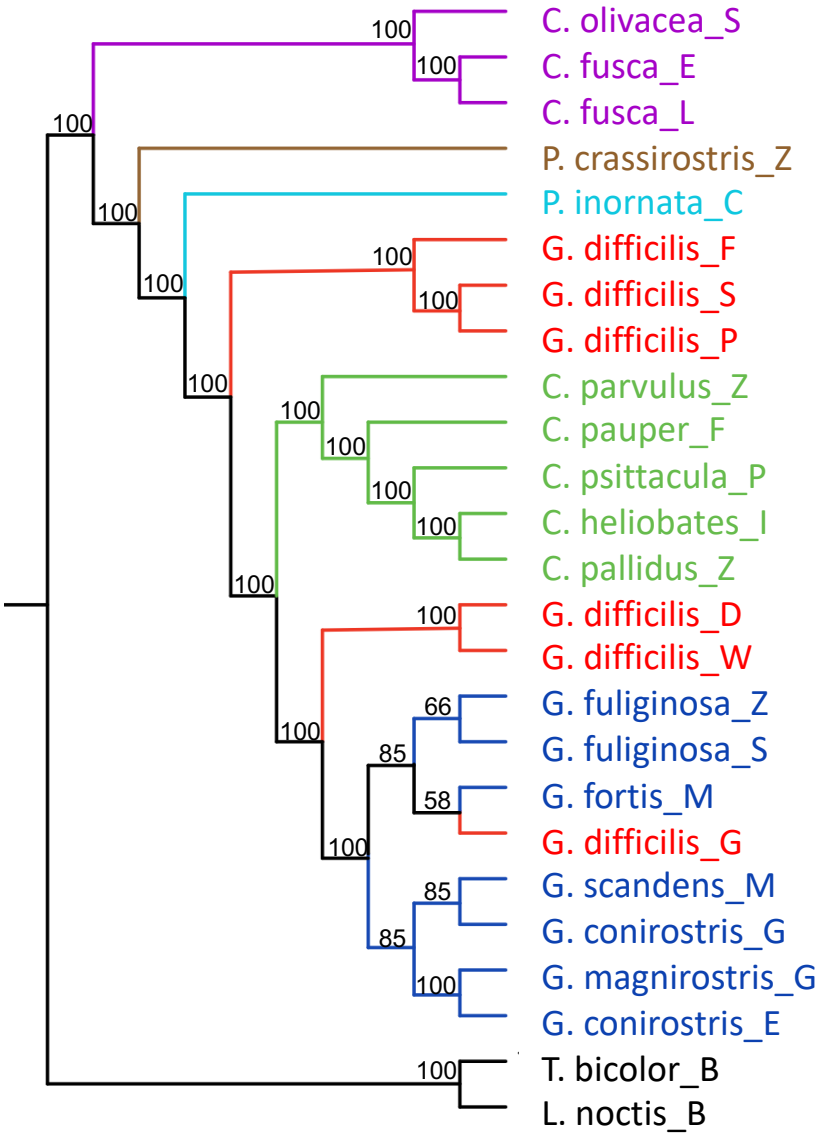


Figure 6.3: The re-estimated phylogenetic tree for Darwin's finch species and two close relatives with supports estimated by the standard bootstrap method. We re-estimated a species tree using maximum likelihood estimation (MLE) on the concatenated and partitioned genomic sequence alignment of Darwin's finch species and two close relatives. We estimated supports using the standard bootstrap method, which is implemented by RAxML version 8.2.9. The branch length were ignored and the tree was rescaled to an ultrametric tree. The color mapping is the same as Figure 6.2.

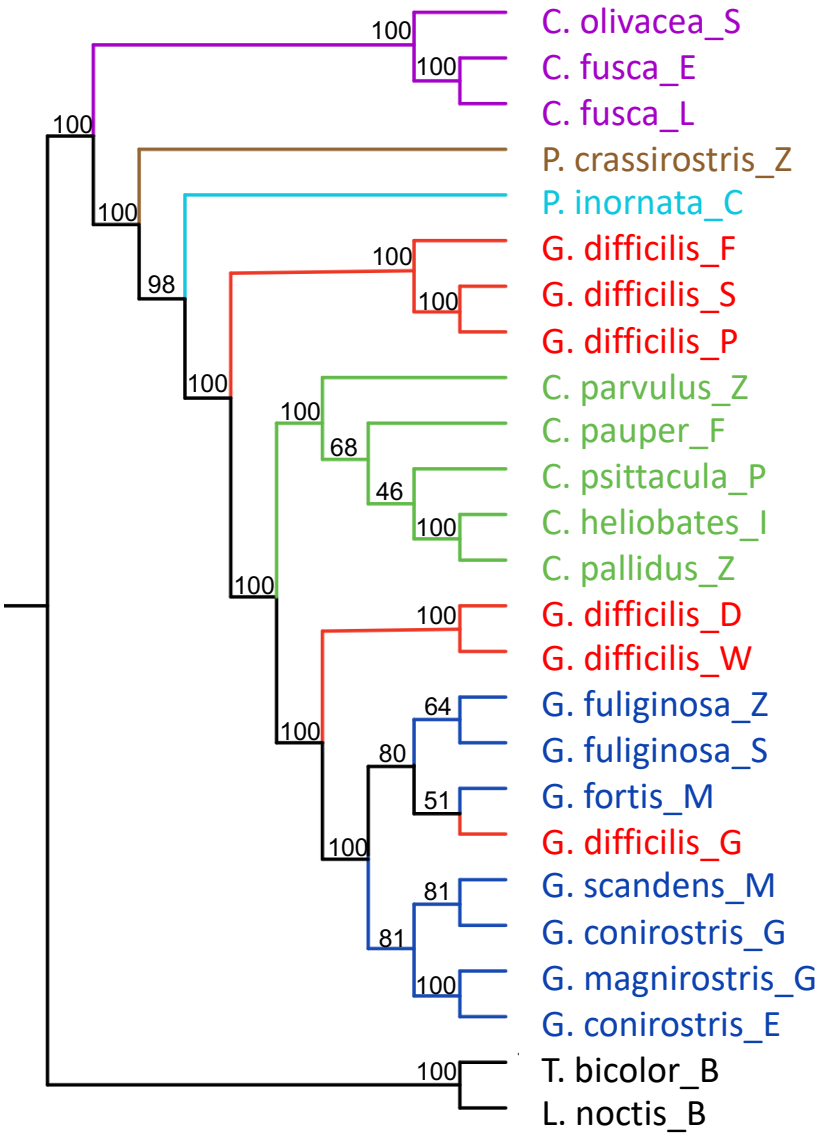


Figure 6.4: The re-estimated phylogenetic tree for Darwin's finch species and two close relatives with supports estimated by the RAWR-based support estimation method. We calculated supports for the re-estimated phylogenetic tree using the RAWR-based support estimation method. The annotated tree is the same tree as Figure 6.3. The branch length were ignored and the tree was rescaled to an ultrametric tree. The color mapping is the same as Figure 6.2.

reported in Lamichhaney's study except for the tree finches. The only inconsistent internal branch showed uncertainty about the evolutionary relationship of the *C. pauper* with the other tree finches. The RAWR-based method produced low confidence intervals for this clade, which confirmed that the phylogenetic relationships between tree finches were challenging to infer, and the upstream MSA estimation process had a certain impact on the phylogenetic reconstruction. We performed the bootstrap support estimation on Darwin's finch dataset, and the bootstrap method returned 100% supports for the clade of the tree finches. This result indicates that the bootstrap method is not sensitive enough to identify uncertain phylogenies caused by the MSA re-estimation. Our experiment on the whole-genome sequencing data of Darwin's finches confirmed that the neighbor preservation property plays a critical role in the phylogenetic support estimation.

For future research, the RAWR-based support estimation can be applied to the analysis of many other types of sequencing data. Transcriptome data has been used in diverse applications, including phylogenetic inference and learning. Though the phylogenetic analysis using transcriptome data is not well developed as the phylogenetic analysis of DNA and protein sequence, it has been proved that the phylogenetic inference using transcriptome data could effectively reproduce previous phylogenies, such as phylogeny of the genus Flaveria reconstructed by Lyu et al. [87] and the phylogenetic analyses of Orchidaceae studied by Deng et al. [26]. The RAWR resampling approach can be extended to phylogenetic studies with transcriptome data, where the phylogenetic analysis can be associated with the evolution of gene expression, post-transcriptional modifications, alternative splicing, and gene fusions.

CHAPTER 7

IMPACT OF MULTIPLE SEQUENCE ALIGNMENT ERROR ON THE SUMMARY-BASED PHYLOGENETIC NETWORK RECONSTRUCTION

7.1 Introduction

The phylogenetic tree is the traditional model for studying the evolutionary history of a set of organisms. The tree structure can effectively represent the relationship among a group of species or genes. However, phylogenetic trees are unable to depict more complex evolutionary events, such as horizontal gene transfer, hybridization, recombination, introgression, or gene duplication and loss. In these scenarios, the genetic materials of some sites are not inherited vertically from the parents but rather horizontally. Therefore, a more general structure, phylogenetic network, was proposed to better represent the evolutionary relationship with reticulation events involved [61, 101].

Various computational methods are designed to reconstruct phylogenetic networks from large-scale genomic sequence data, such as distance-based, maximum parsimony, and maximum likelihood methods. Many of these methods have a two-phase pipeline similar to that of phylogenetic tree inference. The first step is to estimation a set of gene trees from multiple sequence alignments (MSAs) of multiple loci. The second step is to reconstruct the species phylogeny by gene trees. The multiple sequence alignments from the loci are the very initial step of the phylogeny reconstruction.

Previous studies show that the error introduced during the alignment estimation greatly impacts the downstream tree inference and learning [153, 152]. However, there is little discussion about the impact of MSA error on the phylogenetic network reconstruction.

Here we conducted a performance study to investigate how the MSA error impacts the reconstruction of phylogenetic networks. We performed our study on both simulated datasets and two benchmark datasets. The results show that the errors introduced into the sequence alignments significantly impact the accuracy of the downstream phylogenetic network inference. This effect becomes more pronounced as sequence divergence or taxa size increase.

The study offers some critical insights into the quality of input MSAs and a new direction to improve the accuracy of phylogenetic network reconstruction. The computational methods should take into account the alignment error.

7.2 Methods

7.2.1 Simulated Dataset

We performed our simulation study on randomly selected model networks with one reticulation event, following the simulation procedure of Hejase and Liu's 2016 study [53]. The model networks contain either four taxa or eight taxa. We simulated the sequence replicates through three main steps: randomly sample the model network, simulate local genealogies under model networks, simulate sequence evolution for each gene tree.

To simulate the model network, we started with a randomly sampled tree using r8s version 1.8.1 [123]. We scaled all the sampled trees to 1.0 to achieve better control of the divergence of simulated sequences. Then we randomly chose a reticulation time t_M, where $t_M \in [0.01, 0.25]$, and two populations. One reticulation event was added at time t_M between these two populations with a randomly chosen direction. The reticulation event is shown as a directed edge on the tree, representing the gene flow between two populations at time t_M. Last, for rooting purposes, we added one outgroup taxon to the model network at time 1.5.

We simulated the local evolutionary history of 1000 loci using ms [57] under the multi-species coalescent model for each model network. The migration event occurs between time $t_M - 0.01$ and $t_M + 0.01$ with a migration rate of 5.0.

Then, we applied INDELible v1.03 [39] to simulate the sequence evolution of each gene tree from the previous step. We conducted the sequence simulation under the General Time-Reversible (GTR) nucleotide substitution model [118]. We used the medium gap length distribution for the insertion and deletion model. The parameters of the GTR model and the insertion and deletion models come from the study of Liu et al. [80]. To obtain replicated datasets, we repeat this simulation process independently 100 times for each model condition.

The detailed statistical summary of the simulated dataset is listed in Table 7.1. The average normalized Hamming distance (ANHD) represents the sequence divergence. The gappiness represents the percentage of gaps in the true alignments.

Model condition	Insertion/deletion rate	Model phylogeny height	ANHD	Gappiness
4.A	0.1	0.5	0.3146	0.3301
4.B	0.05	0.8	0.4212	0.2855
4.C	0.03	1.4	0.5417	0.2956
4.D	0.02	2.5	0.6388	0.3326
4.E	0.01	5	0.7053	0.3313
8.A	0.03	1	0.4468	0.2825
8.B	0.02	2	0.5844	0.3418
8.C	0.01	3	0.6454	0.2809
8.D	0.006	7	0.7153	0.3531
8.E	0.004	10	0.7270	0.3433

Table 7.1: Model parameters and summary statistics of the simulated datasets. The 4-taxon model conditions are named 4.A through 4.E in order of increasing evolutionary divergence; the 8-taxon model conditions are named 8.A through 8.E similarly. Additional model condition parameters include the insertion/deletion rate and the model phylogeny height (see Methods section for details). Average normalized Hamming distance ("ANHD") and the percentage of true MSA cells that consist of indels ("Gappiness") are reported as an average for each model condition.

7.2.2 Simulation Experiments

We conducted two sets of experiments to investigate the impact of the MSA error on phylogenetic network inference. Both sets of experiments took MSAs as input. The input alignments were either the true alignments simulated under the local genealogies or the estimated alignments inferred from simulated sequences. We used the true alignments as ground truth, which does not have alignment errors. We aligned the simulated sequences using ClustalW version 2.1, MAFFT version 7.222, and FSA version 1.15.9 [73, 64, 12], three widely used MSA methods. The estimated alignments contain errors introduced by the alignment process. By comparing the signals of gene flow or the accuracy of inferred phylogenies, we can learn the effects of alignment error on the phylogenetic network reconstruction.

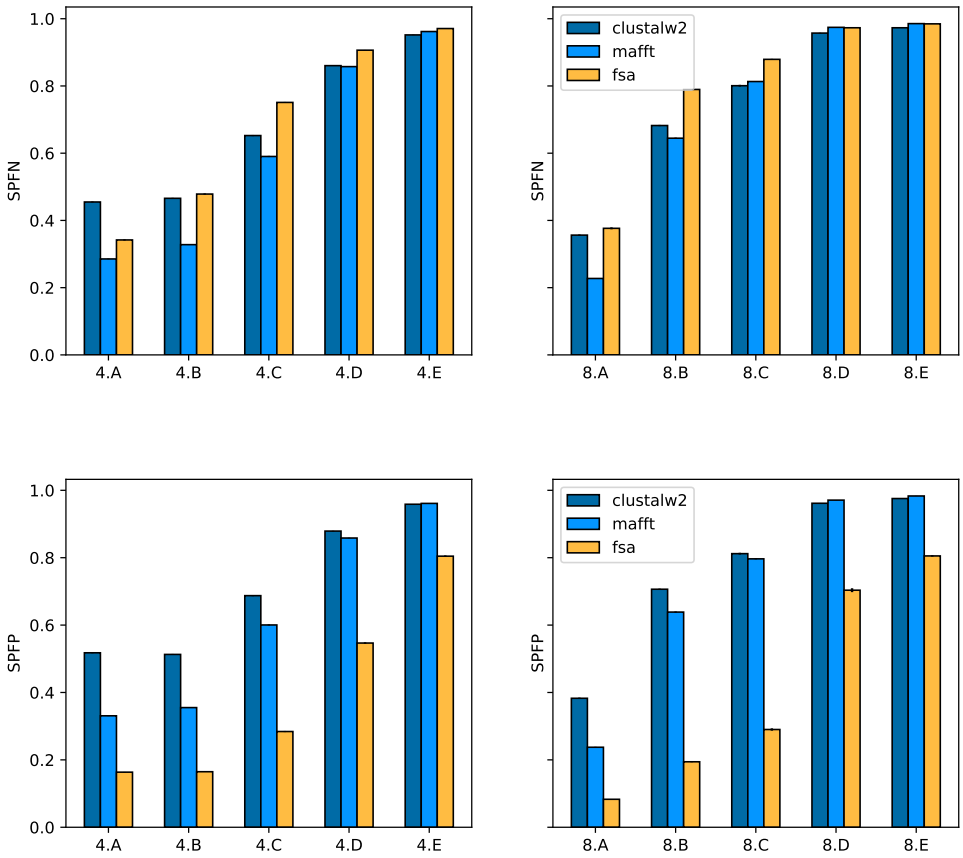


Figure 7.1: Alignment error for MSA estimation methods of the simulated datasets. The MSA methods in our study consisted of MAFFT, ClustalW, and FSA. We assessed MSA estimation error based on type I and type II error: the former was assessed based on SP-FP proportion ("SPFP"), which is the proportion of nucleotide-nucleotide homologies that appear in the estimated alignment but not the true alignment, and the latter was assessed based on SP-FN proportion ("SPFN"), which is the proportion of nucleotide-nucleotide homologies that appear in the true alignment but not the estimated alignment. Average SPFN and SPFP are shown for each MSA method on each model condition.

The first set of experiments applied D-statistics on input MSAs to detect the gene flow signals. D-statistics, or ABBA-BABA statistics, is a parsimony algorithm designed for gene flow detection of closely related species. This method takes sequence alignments as input and calculates D-value based on the numbers of ABBA and BABA sites, which represent two types of inconsistent local genealogies with the species tree. Without significant gene flow, ABBA and BABA sites appear in alignment with the same probability. Therefore, the expected D-value is 0 in this scenario. However, if the D-value is significantly different from zero, two non-sibling species are closer than their sibling species. Such D-values indicate that there is a significant difference between the number of ABBA and BABA sites and that two non-sister species are more similar to each other than expected. This is considered as a signal of gene flow.

The other set of experiments is to reconstruct phylogenetic network from the simulated sequences using summary-based algorithms. The state-of-the-art phylogenetic network reconstruction algorithms we used in this study are implemented in the PhyloNet software package [149, 138].

The phylogenetic network inference pipeline is similar to the phylogenetic tree inference pipeline, which also applies two steps to the biomolecular sequences. The first step is to estimate an MSA from the simulated sequences. and infer gene tree from the estimated MSA. The summary-based inference methods take the inferred gene trees as input and reconstruct the network from the sampled loci.

MSAs in the first pipeline stage consisted of either the true alignment or an estimated alignment that was produced using the above procedure. Gene trees in the second pipeline stage consisted of either the true gene trees or inferred gene trees that were obtained using the following procedure: on either the true MSA or an estimated MSA, we ran FastTree version 2.1.11 [112] with default settings. to perform maximum likelihood estimation of an unrooted gene tree under the GTR+Γ model of nucleotide substitution [118, 35, 105], and rooted gene trees were obtained using outgroup rooting. Since outgroups were used solely for rooting gene trees, the leaf edge to the outgroup taxon was then pruned from each rooted gene tree. Finally, for each set of rooted gene trees – either true gene trees, estimated gene trees that were obtained used MLE on true alignments, or estimated

gene trees that were obtained using MLE on estimated alignments –

PhyloNet was used to perform summary-based network inference under one of three different optimization criteria, model likelihood given gene tree topologies as input [160], model likelihood given gene tree topologies and branch lengths as input [160], or pseudo-likelihood given gene tree topologies as input [160]. We refer to the two summary-based inference methods as MLE, and MPL. All two methods were run using default settings and version 3.6.0 of the PhyloNet software package.

We measured the method performance using the topological errors of the inferred phylogenies compared with the corresponding model phylogenies. We used the Robinson-Foulds distance [116] between the inferred phylogeny and the model phylogeny to measure the topological error of the inferred gene trees. The Robinson-Foulds distance was calculated by the the proportion of bipartitions that appear in the inferred gene tree but not in the true tree or vice versa. We used the metric proposed by Nakhleh's 2019 study [100] to measure the inferred species networks' topological error. The metric, which we refer to as the reduced distance in the following section, is calculated on the set of reduced phylogenetic networks by the number of rooted subnetworks that appear in the inferred network but not the model network or vice versa.

7.2.3 Empirical Datasets and Experiments

In our empirical study, we re-analyzed the datasets from two previous studies [148, 122]. The mosquito dataset is the dataset used in the study of Wen et al. 2016[148] for adaptive introgression in mosquitoes, which was sampled from the whole genome alignment of Fontaine et al. 2015 [103]. This dataset consists of mosquito genetic sequence data of six species and one outgroup taxon. The mosquito dataset includes the following 6 species: *Anopheles gambiae*, *Anopheles coluzzii*, *Anopheles arabiensis*, *Anopheles quadriannulatus*, *Anopheles merus*, and *Anopheles melas*, which are represented by G, C, A, Q, R, and L, respectively in our analysis. *Anopheles christyi* serves as the outgroup taxon. A total of 3019 loci are included.

The second dataset used in the empirical study was obtained from the study of Salichos and

Rokas. [122]. This dataset contains genomic sequence data from 23 yeast species, 4435 loci in total.

We used the same summary-based phylogenetic inference approach to reconstruct species networks as the simulation study. We utilized three different MSA methods to estimate alignments for each locus, for the purpose of obtaining estimated alignments of different quality. The MSA methods we used are ClustalW version 2.1, MAFFT version 7.222, and FSA version 1.15.9 [73, 64, 12]. Then we used FastTree to infer unrooted gene tree for each locus based on the estimated alignments of different quality. For the mosquito-6taxa dataset, the inferred gene trees were rooted by removing the outgroup taxon. For yeast dataset, gene trees were rooted under the MDC criterion using the species tree from Neafsey's 2015 study and Salichos's 2013 study [104, 122]. Finally, we used the rooted gene trees as input to reconstruct a species network with r reticulations, where $r \in [0, 4]$. The network was inferred using the MPL approach, which is implemented in PhyloNet version 3.6.0.

7.3 Results

7.3.1 Simulation Study

7.3.1.1 D-statistics for gene flow detection

In the first set of experiments in the simulation study, we used the D-statistic analysis to detect the gene flow from estimated alignments. A score obtained by D-statistic analysis, which is also called D-value, that is significantly different from zero indicates gene flow between two taxa. A larger absolute score means a stronger gene flow signal. We conducted D-statistics on sequences derived from either model networks or model trees. For the simulation datasets generated from tree-like model phylogenies, no gene flow was included. Such simulation datasets were considered as the negative control group. Ideally, the D-value inferred from the sequences of the negative control group should be close to zero, which means no gene flow was detected. The average scores produced by the D-statistics on the true alignments were close to zero. These results indicated weak

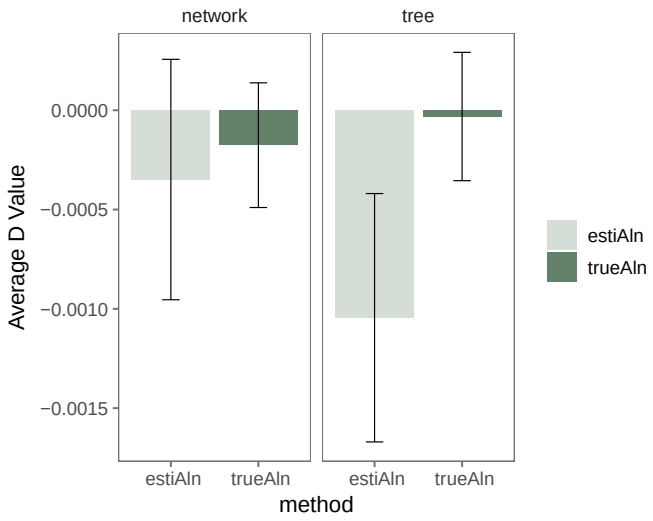


Figure 7.2: Gene flow detection using the D-statistic on simulated datasets. The left subplot shows the D-statistic score distribution on simulated datasets with model networks. The right subplot shows the D-statistic score distribution on simulated datasets with model trees. The D-statistic values were calculated using MAFFT-estimated alignments, which refer to ”estiAln’, and true alignments, which refer to ”trueAln”. Average D-statistic values are reported, and standard error bars are shown over 20 replicates.

gene flow signals were detected for the true alignments of the negative control datasets. However, the average score on the MAFFT-estimated alignments are the largest among all the groups of simulated datasets. Gene flow signals detected from the estimated alignments of the negative control datasets are even larger than the gene flow signals detected from the network datasets.

The simulation datasets derived from network-like model phylogenies contain gene flow signals, which should result in D-values away from zero. As expected, the D-values of both estimated and true alignments were larger than zero. However, the variance of the D-statistic scores tended to be larger on estimated alignments than that of the corresponding true alignments, which means the gene flow detected from the MAFFT-estimated alignments were stronger than that detected from the true alignments.

7.3.1.2 Phylogenetic network inference

We calculated the topological errors of the inferred phylogenetic networks. The topological errors were quantified by the reduction-based distance, which is also called reduced distance, between the

inferred networks and the corresponding model networks. The reduced distance results from the inferred phylogenetic networks of the simulated datasets are shown in Figure 7.3.

Generally, the species networks generated by the MLE network inference method from the true alignments and the true gene trees were the most accurate. The estimated alignments and the estimated gene trees produced the least accurate species networks. There are two exceptions, 4.A and 8.A, two model conditions with the least sequence divergence. For these two model conditions, all the species networks produced by the MLE inference method had comparable topological accuracy.

The topological error difference between species networks inferred from different data sources increased as the sequence divergence increased. For the three least divergent model conditions of 4 taxa and 8 taxa, the MLE method with the true alignments and true gene trees produced similar topological errors compared to the MLE method with true alignments and estimated gene trees. However, for the two most divergent model conditions of 4 taxa and 8 taxa, the MLE method with the true alignments and estimated gene trees returned higher topological error than that using the true gene trees. The species networks inferred by the MLE method using estimated alignments and estimated gene trees were generally less accurate than the other methods, especially for the model conditions with higher sequence divergence. Also, the choice of the MSA method had a minor impact on the topological accuracy of the downstream inferred phylogenetic networks. Furthermore, the topological errors obtained on the 8-taxon model conditions were generally higher than those obtained on the 4-taxon model conditions.

The comparisons among MPL methods with different alignments and gene trees were similar to those of the MLE methods. The MPL method generally produced the most accurate phylogenetic networks with true alignments and true gene trees produced, and the least accurate networks with estimated alignments and estimated gene trees, except for the least divergent model conditions of 4 taxa and 8 taxa, where the MPL method produced similar topological errors despite the alignment errors and gene tree errors.

The MLE method usually produces more accurate species networks than the MPL method under

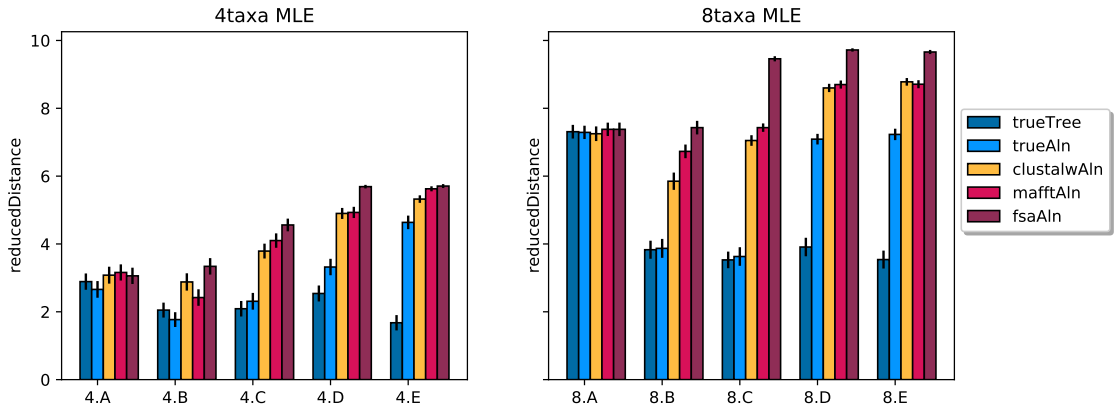


Figure 7.3: Topological errors of MLE analysis on simulation datasets. The MLE method was conducted on five different inputs: (1) true MSAs and true gene trees (”trueTree”), (2) true alignments and gene trees estimated using FastTree on the true alignments (”trueAln”), (3) ClustalW-estimated alignments and gene trees estimated using FastTree on the ClustalW-estimated MSAs (”clustalwAln”), (4) MAFFT-estimated alignments and gene trees estimated using FastTree on MAFFT-estimated alignments (”mafftAln”), or (5) FSA-estimated alignments and gene trees estimated using FastTree on FSA-estimated alignments (”fsaAln”). Topological error was measured using the reduced distance [100]. Averages and standard error bars are shown for each model condition in the simulation study ($n = 20$).

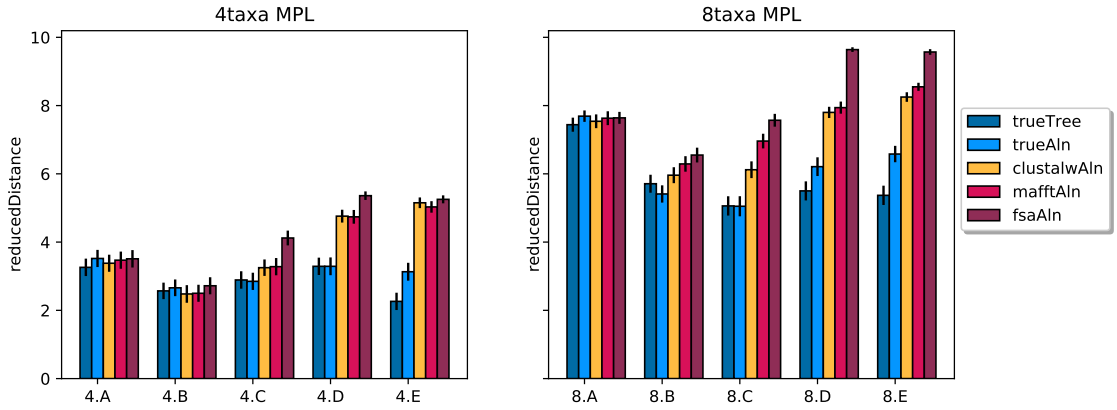


Figure 7.4: Topological errors of MPL analysis on simulation datasets. Figure description and layout are otherwise identical to Figure 7.3.

the same model condition. The MLE method utilizes the full model likelihood criterion, while the MPL method uses a pseudolikelihood criterion, which approximates the full model likelihood criterion [160, 129]. The MPL method is faster but less accurate than the MLE method.

The runtime and peak memory usage of the simulation study are shown in Figures 7.5 and Figure 7.6. The MPL method consistently used less time and memory than the MLE method on

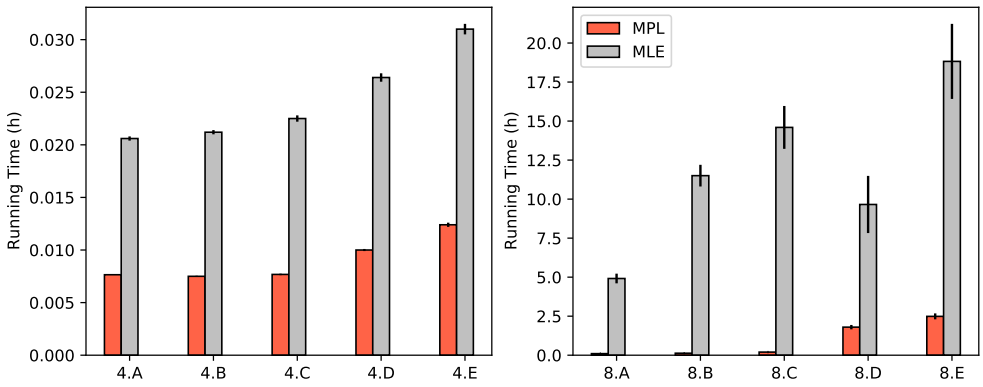


Figure 7.5: Computational runtime requirements of summary-based species network inference methods for simulation study. The runtime of the MPL and MLE methods on simulation datasets is shown in hours. Averages and standard error bars are shown for each model condition over 20 replicates in the simulation study.

all model conditions. On the 8.E model condition, which has the highest sequence divergence, the MLE method's runtime reached almost 20 hours. However, large datasets are very common for modern phylogenomic studies, where the datasets contain many dozens of genomic sequences. The runtime of the MLE method has become a major bottleneck in its application. Our finding is consistent with an earlier performance study [53]. The peak memory usage of both the MPL and MLE methods was less than 800 MiB on all model conditions. Relative differences in peak memory usage were smaller than runtime comparisons as well.

7.3.2 Empirical Study

For the mosquito dataset, we compared the topologies of estimated networks using different input MSAs, including reference and estimated alignments estimated by ClustalW, MAFFT, or FSA. The pairwise reduced distances between the inferred networks are shown in Table 7.2. We inferred phylogenetic network with a single reticulation event from each set of input alignment. The networks inferred from the ClustalW-estimated alignments and MAFFT-estimated alignments were identical in topology compared to the networks inferred from the reference alignments. However, networks inferred from the FSA-estimated alignments had different topologies compared to the networks

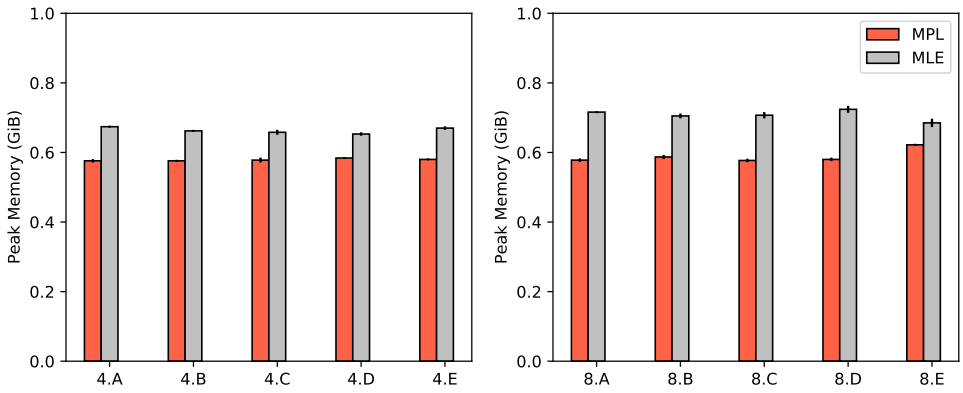


Figure 7.6: Computational memory requirements of summary-based species network inference methods of simulation study. The peak main memory usage of the MPL and MLE method on simulation datasets shows in GiB. Figure description and layout are identical to Figure 7.5.

inferred from the other alignments.

As the reticulations increased, the networks became more complex, and the topological distance between the different methods also increased. There was no identical network inferred from different alignments with two or more reticulation events, and there was no clear trend in terms of the input alignments. One important difference between the empirical study and the simulation study was that the reference alignments used in the empirical study were partially estimated by the computational approaches, which is not the true alignment.

The yeast dataset showed similar results, except the reference alignments were not available. The topological differences were lowest for the networks with a single reticulation event. As the number of reticulations increased, the hypotheses became more complex, and topological differences increased too. As with the mosquito dataset, there were no clear trends in the topological differences between the pairwise comparison of networks inferred using different input MSAs.

7.4 Discussion

The results of the simulation study and the empirical study indicate that the estimation error greatly impacts the downstream phylogenetic network reconstruction. The estimation error introduced to MSA and gene trees potentially create false positive gene flow signals. According to the

1 ret.	ClustalW	MAFFT	FSA	Reference
ClustalW	-	0	6	0
MAFFT		-	6	0
FSA		6	-	6
Reference				-

2 ret.	ClustalW	MAFFT	FSA	Reference
ClustalW	-	5	7	9
MAFFT		-	9	9
FSA			-	7
Reference				-

3 ret.	ClustalW	MAFFT	FSA	Reference
ClustalW	-	11	6	10
MAFFT		-	9	11
FSA			-	8
Reference			-	

4 ret.	ClustalW	MAFFT	FSA	Reference
ClustalW	-	13	9	15
MAFFT		-	12	11
FSA			-	13
Reference			-	

Table 7.2: Topological distance pairwise comparison of species networks inferred by MLE on different MSAs of the mosquito dataset. We obtained the reference alignment from the original study of [103]. The estimated alignments were generated by ClustalW, MAFFT, or FSA. We also compared estimation of species networks with differnt reticulations, which represents different model complexity. The MLE method was used to estimate species networks with at most 1, 2, 3, or 4 reticulations (”ret.”). Topological distances [100] of pairwise comparison between estimated networks were measured. Only upper triangular entries in the pairwise distance matrix are shown.

D-statistics study, the D-values inferred from the estimated alignments showed stronger gene flow signals than those inferred from true alignments for both tree and network datasets. The MSA estimation error enhanced the gene flow signals. The sequence alignment problem is to reconstruct the homology of unaligned sequences. However, if homologous are not correctly aligned, the evolutionary history resolved from such estimated alignments tells a different story.

As the dataset size of sequence divergence increases, the MSA estimation becomes more challenging. More estimation errors are included in the estimated alignments of large datasets with high sequence divergence. The dataset size and sequence divergence of sequences greatly impact the accuracy of the inferred species network, which is proved by the simulation experiments.

1 ret.	ClustalW	MAFFT	FSA
ClustalW	-	5	8
MAFFT		-	4
FSA			-

2 ret.	ClustalW	MAFFT	FSA
ClustalW	-	13	15
MAFFT		-	15
FSA			-

3 ret.	ClustalW	MAFFT	FSA
ClustalW	-	15	20
MAFFT		-	22
FSA			-

4 ret.	ClustalW	MAFFT	FSA
ClustalW	-	16	18
MAFFT		-	21
FSA			-

Table 7.3: Topological distance pairwise comparison of species networks inferred by MLE on different MSAs of the yeast dataset. The MSAs were estimated using ClustalW, MAFFT, or FSA. Table description and layout are identical to Table 7.2.

Other than the dataset size and sequence divergence, the choice of the MSA method also makes an impact on the topological accuracy of the downstream inferred phylogenetic networks. Although, the FSA method is considered to produce the most accurate MSA estimation. The species network inferred from FSA-estimated alignments usually had the highest topological error. The alignments estimated by ClustalW and MAFFT resulted in comparable topological errors. The FSA method seeks for a global alignment with the minimum number of gap openings across the sequences. Such optimization criterion makes minor impacts on the phylogenetic tree inference since the tree reconstruction assumes that the substitution, insertion, and deletion events occur independently for all sites. However, evolutionary events, such as recombination, hybridization, and horizontal gene transfer, create dependence among sites. Incorrect indels are likely to contribute to false-positive gene flow signals. Generally, the sequence alignment method reconstructs homology based on historical substitution, insertion, and deletion events. More complicated evolutionary events are often ignored. Based on our experiments, more sophisticated MSA methods are needed to take into account those evolutionary processes that result in reticulate gene flow.

Similar to MSA estimation, errors introduced to estimated gene trees also impact the phylogenetic networks, especially under model conditions with high sequence divergence. When the sequence divergence is low, the gene tree errors are less impactable for the network inference, which indicate that less sequence divergence usually represents fewer evolutionary events. However, for more challenging datasets, the estimation errors included in the estimated MSA accumulate into the estimation of gene trees.

Another factor that influences the network inference is the number of reticulation hypotheses. As the number of reticulations increased, the hypotheses became more complicated, and the search space of phylogenetic network inference increased dramatically. Expanded search space creates a major obstacle for network inference and possibly results in lower inference accuracy.

7.5 Conclusion

In this study, we investigate the impact of MSA quality on the inference of the phylogenetic network. We compared networks inferred from either true alignments or estimated alignments with either true gene trees or inferred gene trees from the input alignments. The networks inferred from the true alignments and true gene trees were the most accurate, and the networks inferred from the estimated alignments and estimated gene trees were the least accurate. The estimation errors included in MSA greatly impacted the downstream phylogenetic network inference and learning. Alignment estimation becomes more difficult as dataset size and sequence divergence increase. As a consequence, the inferred networks were less accurate for larger and more divergent datasets.

Besides the MSA estimation errors, the choice of MSA estimation method and the errors included in gene trees also affect the species network. The results are consistent across all model conditions in the simulation study. The topological comparisons of the empirical datasets were also consistent with the simulation study. The phylogenetic network reconstruction is also influenced by the hypothesis of network complexity. As the network complexity increases, the search space of the phylogenetic network also increases dramatically, which makes it harder to approach the optimal solution.

Current MSA estimation methods do not take into account the evolutionary processes that produce reticulate gene flow, such as recombination and horizontal gene transfer. Thus, an advanced computational method for MSA-aware phylogenetic network reconstruction is needed for better accuracy. However, computational scalability is a big challenge for such a method. Due to the complexity of the network inference problem, the computational methods that we used in our study are already computationally intensive. For better network inferences, the network inference problem of a large dataset can be divided into small problem sets, and parallel computing can be used to improve the computational scalability.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

In this dissertation, we addressed problems caused by some over-simplified assumptions in the phylogenetic analysis. Phylogenetic reconstruction is an important and fundamental problem for many biological studies. More and more biomolecular sequencing data are available for phylogenetic analyses as the technological advances in molecular biology and genomics. Due to the complexity of the phylogenetic reconstruction problem, many over-simplified assumptions have been made in modern phylogenetic studies. We combined the resampling and re-estimation procedures with multiple phylogenetic analyses to utilize the phylogenetic information carried by indels, which are often ignored in the traditional phylogenetic reconstruction. We proposed a new sequential resampling algorithm, SERES, to address the assumption made by many widely used non-parametric resampling algorithms that all sites of an alignment have evolved independently and identically distributed (i.i.d). A biomolecular sequence resampling algorithm is proposed for the resampling of biomolecular sequence data that counts for the sequence dependence. The SERES resampling approach combines the standard bootstrap resampling algorithm with the form of a random walk, which can produce many distinct resampling replicates while reserving the sequential dependence during the resampling process. The SERES resampling approach outperformed the state-of-the-art approaches, which utilize the bootstrap method, on a classical problem in computational biology and bioinformatics, the MSA support estimation problem.

The SERES resampling approach has a wide range of applications for various types of data. We introduced the application of SERES random walks on aligned sequences and showed SERES as a data perturbation technique to improve statistical inference and learning. The combination of the SERES resampling approach with the recHMM obtained great improvement in the local genealogy inferences. This finding is confirmed by the breakpoint detection problem of the HIV genome sequence dataset [84, 150]. SERES resampling and re-estimation may be similarly beneficial in ancestral recombination inference problems other than local genealogical inference, such as

recombination rate estimation [133], recombination hotspot or coldspot detection [99, 5], etc.

Due to the synchronization needs, the SERES algorithm is a semi-parametric algorithm when applied to the unaligned sequences, which requires anchor estimation for synchronization purposes. We introduced a new non-parametric resampling technique, RAWR. The RAWR resampling approach does not require additional parameters. We applied the RAWR resampling approach to another classical bioinformatics problem, phylogenetic support estimation. The performance study showed that the RAWR-based support estimation produced better performance than the widely used bootstrap support estimation method, especially for sequences with higher divergence. This finding is consistent with the SERES resampling approach to the MSA support estimation problem.

In this dissertation, we further relaxed the tree structure assumption of phylogenies and investigated the impact of MSA uncertainties on the phylogenetic network inference. Previous research has shown that estimation errors in MSA have a great impact on downstream tree inference [153, 152]. We showed that the MSA estimation errors also affect the phylogenetic network reconstruction. The accuracy of the inferred networks decreased as the accumulation of estimation errors of alignments and gene trees. The impact of alignment errors on the topological accuracy grows as the sequence divergence increases. The topological comparison results of the empirical datasets are consistent with the simulation datasets, and they show that the network hypothesis complexity also impacts the phylogenetic network inferences.

This dissertation mainly addressed problems caused by over-simplified assumptions that are commonly used in the traditional phylogenetic analysis. We took into account the indels uncertainties and the intra-sequence dependence for the analyses of large sequencing datasets in phylogenetic studies, but many such challenges remain. We point out some directions for future work.

The SERES and RAWR resampling algorithms require few assumptions and are not restricted to a specific application. It is easy to apply the new resampling algorithms to other evolutionary analyses which require biomolecular sequences as input or accounting for the sequential dependence, such as protein structure prediction, reads mapping, and assembly. One important application is the statistical inference of species trees under models of sequence evolution that take into account the

sequential dependence. Based on our performance study, the MSA quality has a great impact on the phylogenetic network. An advanced computational method for MSA-aware phylogenetic network reconstruction can utilize the RAWR resampling approach to produce phylogenetic networks with better accuracy. However, computational scalability would be a major challenge for such a method due to the complexity of the network inference problem. For the network inferences with the sequence resampling algorithm, the original inference problem can be divided into small problem sets. In this scenario, parallel computing can be applied to solve small problem sets parallelly and improve computational scalability. Non-parametric resampling is also widely used throughout science and engineering, and SERES/RAWR resampling can also be applied in research areas outside of computational biology and bioinformatics.

In this dissertation, we discussed how does the new resampling algorithms deal with the sequence dependence caused by historical insertion and deletion events. Many other factors also lead to sequence dependence, which we do not take into account yet. The non-parametric resampling algorithm can be extended to application-specified resampling and re-estimation. For example, take the structure information into account for the resampling procedure. Such a resampling algorithm can better utilize the sequence dependence caused by particular evolutionary events in the biomolecular sequence analysis involved in those evolutionary events.

Since alternative homologous or bipartitions are produced during the resampling and re-estimation process, the sequential resampling approach could potentially be extended to perform the MSA estimation or phylogenetic inference. However, the optimization criterion and computational efficiency are two main challenges for such a method. In the study of RAWR resampling and re-estimation in phylogenetic support estimation, we observed that the re-estimation process is the bottleneck of computational runtime. Since the resampling and re-estimation processes of the resampled replicates are independent to each other, parallel computing can be applied to boost the computational efficiency of such methods.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Raja Hashim Ali, Marcin Bogusz, and Simon Whelan. Identifying clusters of high confidence homologies in multiple sequence alignments. *Molecular biology and evolution*, 36(10):2340–2351, 2019.

[2] Elizabeth S Allman, James H Degnan, and John A Rhodes. Identifying the rooted species tree from the distribution of unrooted gene trees under the coalescent. *Journal of mathematical biology*, 62(6):833–862, 2011.

[3] Maria Anisimova and Olivier Gascuel. Approximate likelihood-ratio test for branches: a fast, accurate, and powerful alternative. *Systematic biology*, 55(4):539–552, 2006.

[4] Hagai Attias et al. A variational baysian framework for graphical models. In *NIPS*, volume 12. Citeseer, 1999.

[5] Adam Auton and Gil McVean. Recombination rate estimation in the presence of hotspots. *Genome research*, 17(8):1219–1227, 2007.

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

[7] Frida Belinky, Ofir Cohen, and Dorothée Huchon. Large-scale parsimony analysis of metazoan indels in protein-coding genes. *Molecular biology and evolution*, 27(2):441–451, 2010.

[8] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[9] Maria Bonet, Mike Steel, Tandy Warnow, and Shibu Yooseph. Better methods for solving parsimony and compatibility. *Journal of Computational Biology*, 5(3):391–407, 1998.

[10] Paola Bonizzoni and Gianluca Della Vedova. The complexity of multiple sequence alignment with sp-score that is a metric. *Theoretical Computer Science*, 259(1-2):63–79, 2001.

[11] Luis Boto. Horizontal gene transfer in evolution: facts and challenges. *Proceedings of the Royal Society B: Biological Sciences*, 277(1683):819–827, 2010.

[12] Robert K. Bradley, Adam Roberts, Michael Smoot, Sudeep Juvekar, Jaeyoung Do, Colin Dewey, Ian Holmes, and Lior Pachter. Fast statistical alignment. *PLoS Computational Biology*, 5(5):e1000392, may 2009.

[13] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[14] Jamie J Cannone, Sankar Subramanian, Murray N Schnare, James R Collett, Lisa M D'Souza, Yushi Du, Brian Feng, Nan Lin, Lakshmi V Madabusi, Kirsten M Müller, et al. The comparative rna web (crw) site: an online database of comparative sequence and structure information for ribosomal, intron, and other rnas. *BMC bioinformatics*, 3(1):1–31, 2002.

[15] Brandi L Cantarel, Hilary G Morrison, and William Pearson. Exploring the relationship between sequence similarity and accurate phylogenetic trees. *Molecular biology and evolution*, 23(11):2090–2100, 2006.

[16] Salvador Capella-Gutiérrez, José M Silla-Martínez, and Toni Gabaldón. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 2009.

[17] Jose Castresana. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Molecular biology and evolution*, 17(4):540–552, 2000.

[18] Jia Ming Chang, Paolo Di Tommaso, and Cedric Notredame. TCS: A new multiple sequence alignment reliability measure to estimate alignment accuracy and improve phylogenetic tree reconstruction. *Molecular Biology and Evolution*, 31(6):1625–1637, jun 2014.

[19] Jerry A Coyne, H Allen Orr, et al. *Speciation*, volume 37. Sinauer Associates Sunderland, MA, 2004.

[20] William HE Day. Computational complexity of inferring phylogenies from dissimilarity matrices. *Bulletin of mathematical biology*, 49(4):461–467, 1987.

[21] Andrew H Debevec and James B Whitfield. Introduction to phylogenetic networks.—david a. morrison., 2013.

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

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

[24] James H Degnan and Laura A Salter. Gene tree distributions under the coalescent process. *Evolution*, 59(1):24–37, 2005.

[25] E R DeLong, D M DeLong, and D L Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, 44(3):837–45, 1988.

[26] Hua Deng, Guo-Qiang Zhang, Min Lin, Yan Wang, and Zhong-Jian Liu. Mining from transcriptomes: 315 single-copy orthologous genes concatenated for the phylogenetic analyses of orchidaceae. *Ecology and Evolution*, 5(17):3800–3807, 2015.

[27] Jeff J Doyle. Trees within trees: genes and species, molecules and morphology. *Systematic Biology*, 46(3):537–553, 1997.

[28] Ingo Ebersberger, Petra Galgoczy, Stefan Taudien, Simone Taenzer, Matthias Platzer, and Arndt Von Haeseler. Mapping human genetic ancestry. *Molecular biology and evolution*, 24(10):2266–2276, 2007.

[29] Robert C Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–7, 2004.

[30] Scott V Edwards. Is a new and general theory of molecular systematics emerging? *Evolution: International Journal of Organic Evolution*, 63(1):1–19, 2009.

[31] Scott V Edwards, Liang Liu, and Dennis K Pearl. High-resolution species trees without concatenation. *Proceedings of the National Academy of Sciences*, 104(14):5936–5941, 2007.

[32] B Efron. Bootstrap methods: another look at the jackknife. e annals of statistics, 7 (1): 1–26. *URL http://www. jstor. org/stable/2958830*, 1979.

[33] R. A. L. Elworth, H. A. Ogilvie, J. Zhu, and L. Nakhleh. Advances in Computational Methods for Phylogenetic Networks in the Presence of Hybridization. Technical report, 2018.

[34] Joseph Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, 1981.

[35] Joseph Felsenstein. Taking variation of evolutionary rates between sites into account in inferring phylogenies. *Journal of Molecular Evolution*, 53(4-5):447–455, 2001.

[36] Joseph Felsenstein. Confidence Limits on Phylogenies: An Approach Using the Bootstrap. *Evolution*, 39(4):783, jul 2006.

[37] Joseph Felsenstein and Joseph Felenstein. *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

[38] Da-Fei Feng and Russell F Doolittle. Journal of Molecular Evolution Progressive Sequence Alignment as a Prerequisite to Correct Phylogenetic Trees. *J Mol Evol*, 25(4):351–360, 1987.

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

[40] Nir Friedman, Matan Ninio, Itsik Pe'er, and Tal Pupko. A structural em algorithm for phylogenetic inference. In *Proceedings of the fifth annual international conference on Computational biology*, pages 132–140, 2001.

[41] Ganeshkumar Ganapathy, Vijaya Ramachandran, and Tandy Warnow. Better hill-climbing searches for parsimony. In *International Workshop on Algorithms in Bioinformatics*, pages 245–258. Springer, 2003.

[42] John Gatesy and Mark S Springer. Phylogenetic analysis at deep timescales: unreliable gene trees, bypassed hidden support, and the coalescence/concatalescence conundrum. *Molecular phylogenetics and evolution*, 80:231–266, 2014.

[43] Gonzalo Giribet. Tnt: tree analysis using new technology, 2005.

[44] J Peter Gogarten and Jeffrey P Townsend. Horizontal gene transfer, genome innovation and evolution. *Nature Reviews Microbiology*, 3(9):679–687, 2005.

[45] Brian Golding and Joe Felsenstein. A maximum likelihood approach to the detection of selection from a phylogeny. *Journal of molecular evolution*, 31(6):511–523, 1990.

[46] Edward M Golenberg, Michael T Clegg, Mary L Durbin, John Doebley, and Din Pow Ma. Evolution of a noncoding region of the chloroplast genome. *Molecular phylogenetics and evolution*, 2(1):52–64, 1993.

[47] Peter R Grant and B Rosemary Grant. Speciation undone. *Nature*, 507(7491):178–179, 2014.

[48] Nicholas C. Grassly, Jun Adachj, and Andrew Rambaut. PSeq-Gen: an application for the Monte Carlo simulation of protein sequence evolution along phylogenetic trees. *Bioinformatics*, 13(5):559–560, 2007.

[49] Stéphane Guindon, Jean-François Dufayard, Vincent Lefort, Maria Anisimova, Wim Hordijk, and Olivier Gascuel. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of phyml 3.0. *Systematic biology*, 59(3):307–321, 2010.

[50] Richard W Hamming. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160, 1950.

[51] Jotun Hein. A heuristic method to reconstruct the history of sequences subject to recombination. *Journal of Molecular Evolution*, 36(4):396–405, 1993.

[52] Jotun Hein, Mikkel Schierup, and Carsten Wiuf. *Gene genealogies, variation and evolution: a primer in coalescent theory*. Oxford University Press, USA, 2004.

[53] Hussein A Hejase and Kevin J Liu. A scalability study of phylogenetic network inference methods using empirical datasets and simulations involving a single reticulation. *BMC bioinformatics*, 17(1):1–12, 2016.

[54] Joseph Heled and Alexei J Drummond. Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution*, 27(3):570–580, 2009.

[55] David M Hillis and James J Bull. An empirical test of bootstrapping as a method for assessing confidence in phylogenetic analysis. *Systematic biology*, 42(2):182–192, 1993.

[56] David M Hillis, Craig Moritz, Barbara K Mable, and Richard G Olmstead. *Molecular systematics*, volume 23. Sinauer Associates Sunderland, MA, 1996.

[57] Richard R. Hudson. Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18(2):337–338, 2002.

[58] Richard R Hudson et al. Gene genealogies and the coalescent process. *Oxford surveys in evolutionary biology*, 7(1):44, 1990.

[59] Dirk Husmeier and Frank Wright. Detection of recombination in dna multiple alignments with hidden markov models. *Journal of Computational Biology*, 8(4):401–427, 2001.

[60] Daniel H Huson and David Bryant. Application of phylogenetic networks in evolutionary studies. *Molecular biology and evolution*, 23(2):254–267, 2006.

[61] Daniel H Huson, Regula Rupp, and Celine Scornavacca. *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press, 2010.

[62] Daniel H Huson and Celine Scornavacca. Dendroscope 3: an interactive tool for rooted phylogenetic trees and networks. *Systematic biology*, 61(6):1061–1067, 2012.

[63] S. JEFFERY. Evolution of Protein Molecules. *Biochemical Society Transactions*, 7(2):452–453, 2015.

[64] Kazutaka Katoh and Daron M. Standley. MAFFT multiple sequence alignment software version 7: Improvements in performance and usability. *Molecular Biology and Evolution*, 30(4):772–780, apr 2013.

[65] Jaebum Kim and Jian Ma. Psar: measuring multiple sequence alignment reliability by probabilistic sampling. *Nucleic acids research*, 39(15):6359–6368, 2011.

[66] John FC Kingman. On the genealogy of large populations. *Journal of Applied Probability*, 19(A):27–43, 1982.

[67] Alexey M Kozlov, Andre J Aberer, and Alexandros Stamatakis. Examl version 3: a tool for phylogenomic analyses on supercomputers. *Bioinformatics*, 31(15):2577–2579, 2015.

[68] Laura Salter Kubatko and James H Degnan. Inconsistency of phylogenetic estimates from concatenated data under coalescence. *Systematic biology*, 56(1):17–24, 2007.

[69] Sangeet Lamichhaney, Jonas Berglund, Markus Sällman Almén, Khurram Maqbool, Manfred Grabherr, Alvaro Martinez-Barrio, Marta Promerová, Carl-Johan Rubin, Chao Wang, Neda Zamani, et al. Evolution of darwin's finches and their beaks revealed by genome sequencing. *Nature*, 518(7539):371–375, 2015.

[70] Giddy Landan and Dan Graur. Heads or tails: A simple reliability check for multiple sequence alignments. *Molecular Biology and Evolution*, 24(6):1380–1383, mar 2007.

[71] Giddy Landan and Dan Graur. Local reliability measures from sets of co-optimal multiple sequence alignments. In *Biocomputing 2008*, pages 15–24. World Scientific, 2008.

[72] Bret R Larget, Satish K Kotha, Colin N Dewey, and Cécile Ané. Bucky: gene tree/species tree reconciliation with bayesian concordance analysis. *Bioinformatics*, 26(22):2910–2911, 2010.

[73] M. A. Larkin, G. Blackshields, N. P. Brown, R. Chenna, P. A. Mcgettigan, H. McWilliam, F. Valentin, I. M. Wallace, A. Wilm, R. Lopez, J. D. Thompson, T. J. Gibson, and D. G. Higgins. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23(21):2947–2948, nov 2007.

[74] Fréderic Lemoine, J-B Domelevo Entfellner, Eduan Wilkinson, Damien Correia, M Dávila Felipe, Tulio De Oliveira, and Olivier Gascuel. Renewing felsenstein's phylogenetic bootstrap in the era of big data. *Nature*, 556(7702):452–456, 2018.

[75] Heng Li and Richard Durbin. Fast and accurate short read alignment with burrows–wheeler transform. *bioinformatics*, 25(14):1754–1760, 2009.

[76] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.

[77] C Randal Linder and Tandy Warnow. An overview of phylogeny reconstruction. 2001.

[78] Kevin Liu, C Randal Linder, and Tandy Warnow. Multiple sequence alignment: a major challenge to large-scale phylogenetics. *PLoS currents*, 2, 2010.

[79] Kevin Liu, Sindhu Raghavan, Serita Nelesen, C Randal Linder, and Tandy Warnow. Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. *Science*, 324(5934):1561–1564, 2009.

[80] Kevin Liu, Tandy J. Warnow, Mark T. Holder, Serita M. Nelesen, Jiaye Yu, Alexandros P. Stamatakis, and C. Randal Linder. SATé-II: Very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. *Systematic Biology*, 61(1):90–106, jan 2012.

[81] Kevin J Liu, Jingxuan Dai, Kathy Truong, Ying Song, Michael H Kohn, and Luay Nakhleh. An hmm-based comparative genomic framework for detecting introgression in eukaryotes. *PLoS computational biology*, 10(6):e1003649, 2014.

[82] Liang Liu. Best: Bayesian estimation of species trees under the coalescent model. *Bioinformatics*, 24(21):2542–2543, 2008.

[83] Liang Liu, Lili Yu, and Scott V Edwards. A maximum pseudo-likelihood approach for estimating species trees under the coalescent model. *BMC evolutionary biology*, 10(1):1–18, 2010.

[84] K S Lole, R C Bollinger, R S Paranjape, D Gadkari, S S Kulkarni, N G Novak, R Ingersoll, H W Sheppard, and S C Ray. Full-length human immunodeficiency virus type 1 genomes from subtype C-infected seroconverters in India, with evidence of intersubtype recombination. *Journal of virology*, 73(1):152–60, 1999.

[85] Ari Löytynoja and Nick Goldman. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, 320(5883):1632–1635, 2008.

[86] Peng-tao Luan, Oliver A Ryder, Heidi Davis, Ya-ping Zhang, and Li Yu. Incorporating indels as phylogenetic characters: impact for interfamilial relationships within arctoidea (mammalia: Carnivora). *Molecular Phylogenetics and Evolution*, 66(3):748–756, 2013.

[87] Ming-Ju Amy Lyu, Udo Gowik, Steve Kelly, Sarah Covshoff, Julia Mallmann, Peter Westhoff, Julian M Hibberd, Matt Stata, Rowan F Sage, Haorong Lu, et al. Rna-seq based phylogeny recapitulates previous phylogeny of the genus flaveria (asteraceae) with some modifications. *BMC evolutionary biology*, 15(1):1–14, 2015.

[88] G Blackshields1 M.A. Larkin1 N.P. Brown3, R. Chenna3, P.A. McGettigan1,, F Valentin4 H. McWilliam4 I.M. Wallace1, A. Wilm1, R. Lopez4, J.D. Thompson2,, T J Gibson3 Higgins, and D G. Sequence analysis, Clustal W and Clustal X version 2.0. *Bioinformatics Applications Note*, 23(21):2947–2948, 2007.

[89] Wayne P Maddison. Gene trees in species trees. *Systematic Biology*, 46(3):523–536, 1997.

[90] Wayne P Maddison and L Lacey Knowles. Inferring phylogeny despite incomplete lineage sorting. *Systematic biology*, 55(1):21–30, 2006.

[91] Thomas Mailund, Julien Y Dutheil, Asger Hobolth, Gerton Lunter, and Mikkel H Schierup. Estimating divergence time and ancestral effective population size of bornean and sumatran orangutan subspecies using a coalescent hidden markov model. *PLoS genetics*, 7(3):e1001319, 2011.

[92] Grainne McGuire. *Statistical methods for DNA sequences: detection of recombination and distance estimation*. PhD thesis, University of Edinburgh, 1998.

[93] Gráinne McGuire, Frank Wright, and Michael J Prentice. A bayesian model for detecting past recombination events in dna multiple alignments. *Journal of Computational Biology*, 7(1-2):159–170, 2000.

[94] Michael L Metzker. Sequencing technologies—the next generation. *Nature reviews genetics*, 11(1):31–46, 2010.

[95] Siavash Mirarab, Nam Nguyen, Sheng Guo, Li-San Wang, Junhyong Kim, and Tandy Warnow. Pasta: ultra-large multiple sequence alignment for nucleotide and amino-acid sequences. *Journal of Computational Biology*, 22(5):377–386, 2015.

[96] Erin K Molloy and Tandy Warnow. To include or not to include: the impact of gene filtering on species tree estimation methods. *Systematic Biology*, 67(2):285–303, 2018.

[97] Erin K Molloy and Tandy Warnow. Statistically consistent divide-and-conquer pipelines for phylogeny estimation using njmerge. *Algorithms for Molecular Biology*, 14(1):1–17, 2019.

[98] Daniel Money and Simon Whelan. Characterizing the phylogenetic tree-search problem. *Systematic biology*, 61(2):228, 2012.

[99] Simon Myers, Leonardo Bottolo, Colin Freeman, Gil McVean, and Peter Donnelly. A fine-scale map of recombination rates and hotspots across the human genome. *Science*, 310(5746):321–324, 2005.

[100] Luay Nakhleh. A metric on the space of reduced phylogenetic networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(2):218–222, 2009.

[101] Luay Nakhleh. Evolutionary Phylogenetic Networks: Models and Issues. In *Problem Solving Handbook in Computational Biology and Bioinformatics*, pages 125–158. Springer, 2010.

[102] Luay Nakhleh, Bernard ME Moret, Usman Roshan, Katherine St. John, Jerry Sun, and Tandy Warnow. The accuracy of fast phylogenetic methods for large datasets. In *Biocomputing 2002*, pages 211–222. World Scientific, 2001.

[103] Daniel E Neafsey, Igor V Sharakhov, Xiaofang Jiang, Andrew B Hall, Evdoxia Kakani, Sara N Mitchell, Yi-chieh Wu, Hilary A Smith, Matthew W Hahn, and Nora J Besansky. Extensive introgression in a malaria vector species complex revealed by phylogenomics. *Science*, 347(6217):1–20, 2015.

[104] Daniel E. Neafsey, Robert M. Waterhouse, Mohammad R. Abai, Sergey S. Aganezov, Max A. Alekseyev, James E. Allen, James Amon, Bruno Arcà, Peter Arensburger, Gleb Artemov, Lauren A. Assour, Hamidreza Basseri, Aaron Berlin, Bruce W. Birren, Stephanie A. Blandin, Andrew I. Brockman, Thomas R. Burkot, Austin Burt, Clara S. Chan, Cedric Chauve, Joanna C. Chiu, Mikkel Christensen, Carlo Costantini, Victoria L.M. Davidson, Elena Deligianni, Tania Dottorini, Vicky Dritsou, Stacey B. Gabriel, Wamdaogo M. Guelbeogo, Andrew B. Hall, Mira V. Han, Thaung Hlaing, Daniel S.T. Hughes, Adam M. Jenkins, Xiaofang Jiang, Irwin Jungreis, Evdoxia G. Kakani, Maryam Kamali, Petri Kemppainen, Ryan C. Kennedy, Ioannis K. Kirmitzoglou, Lizette L. Koekemoer, Njoroge Laban, Nicholas Langridge, Mara K.N. Lawniczak, Manolis Lirakis, Neil F. Lobo, Ernesto Lowy, Robert M. MacCallum, Chunhong Mao, Gareth Maslen, Charles Mbogo, Jenny McCarthy, Kristin Michel, Sara N. Mitchell, Wendy Moore, Katherine A. Murphy, Anastasia N. Naumenko, Tony Nolan, Eva M. Novoa, Samantha O'Loughlin, Chioma Oringanje, Mohammad A. Oshaghi, Nazzy Pakpour, Philippos A. Papathanos, Ashley N. Peery, Michael Povelones, Anil Prakash, David P. Price, Ashok Rajaraman, Lisa J. Reimer, David C. Rinker, Antonis Rokas, Tanya L. Russell, N'Fale Sagnon, Maria V. Sharakhova, Terrance Shea, Felipe A. Simão, Frederic Simard, Michel A. Slotman, Pradya Somboon, Vladimir Stegniy, Claudio J. Struchiner, Gregg W.C. Thomas, Marta Tojo, Pantelis Topalis, José M.C. Tubio, Maria F. Unger, John Vontas, Catherine Walton, Craig S. Wilding, Judith H. Willis, Yi Chieh Wu, Guiyun Yan, Evgeny M. Zdobnov, Xiaofan Zhou, Flaminia Catteruccia, George K. Christophides, Frank H. Collins, Robert S. Cornman, Andrea Crisanti, Martin J. Donnelly, Scott J. Emrich, Michael C. Fontaine, William Gelbart, Matthew W. Hahn, Immo A. Hansen, Paul I. Howell, Fotis C. Kafatos, Manolis Kellis, Daniel Lawson, Christos Louis, Shirley Luckhart, Marc A.T. Muskavitch, José M. Ribeiro, Michael A. Riehle, Igor V. Sharakhov, Zhijian Tu, Laurence J. Zwiebel, and Nora J. Besansky. Highly evolvable malaria vectors: The genomes of 16 Anopheles mosquitoes. *Science*, 347(6217):1258522, 2015.

[105] MASATOSHI NEi, Ranajit Chakraborty, and Paul A Fuerst. Infinite allele model with varying mutation rate. *Proceedings of the National Academy of Sciences*, 73(11):4164–4168, 1976.

[106] Richard Nichols. Gene trees and species trees are not the same. *Trends in Ecology & Evolution*, 16(7):358–364, 2001.

[107] C Notredame, D G Higgins, and J Heringa. Turnbaugh et al. 16S rRNA sequences\rT-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol*, 302(1):205–217, 2000.

[108] Cédric Notredame, Desmond G. Higgins, and Jaap Heringa. T-coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302(1):205–217, sep 2000.

[109] T Heath Ogden and Michael S Rosenberg. Multiple sequence alignment accuracy and phylogenetic inference. *Systematic biology*, 55(2):314–328, 2006.

[110] Osnat Penn, Eyal Privman, Giddy Landan, Dan Graur, and Tal Pupko. An alignment confidence score capturing robustness to guide tree uncertainty. *Molecular Biology and Evolution*, 27(8):1759–1767, 2010.

[111] David Posada and Keith A Crandall. Intraspecific gene genealogies: trees grafting into networks. *Trends in ecology & evolution*, 16(1):37–45, 2001.

[112] Morgan N Price, Paramvir S Dehal, and Adam P Arkin. Fasttree 2–approximately maximum-likelihood trees for large alignments. *PloS one*, 5(3):e9490, 2010.

[113] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[114] Bruce Rannala and Ziheng Yang. Bayes estimation of species divergence times and ancestral population sizes using dna sequences from multiple loci. *Genetics*, 164(4):1645–1656, 2003.

[115] Maria C Rivera and James A Lake. Evidence that eukaryotes and eocyte prokaryotes are immediate relatives. *Science*, 257(5066):74–76, 1992.

[116] D. F. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147, feb 1981.

[117] Sebastien Roch. A short proof that phylogenetic tree reconstruction by maximum likelihood is hard. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(1):92–94, 2006.

[118] F. Rodríguez, J. L. Oliver, A. Marín, and J. R. Medina. The general stochastic model of nucleotide substitution. *Journal of Theoretical Biology*, 142(4):485–501, feb 1990.

[119] Antonis Rokas and Peter WH Holland. Rare genomic changes as a tool for phylogenetics. *Trends in ecology & evolution*, 15(11):454–459, 2000.

[120] Antonis Rokas, Barry L Williams, Nicole King, and Sean B Carroll. Genome-scale approaches to resolving incongruence in molecular phylogenies. *Nature*, 425(6960):798–804, 2003.

[121] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.

[122] Leonidas Salichos and Antonis Rokas. Inferring ancient divergences requires genes with strong phylogenetic signals. *Nature*, 497(7449):327–331, 2013.

[123] Michael J Sanderson. Sanderson - 2003 - r8s inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock.pdf. *Bioinformatics*, 19(2):301–302, 2003.

[124] Michael C Schatz, Arthur L Delcher, and Steven L Salzberg. Assembly of large genomes using second-generation sequencing. *Genome research*, 20(9):1165–1173, 2010.

[125] Paul Scheet and Matthew Stephens. A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. *The American Journal of Human Genetics*, 78(4):629–644, 2006.

[126] Itamar Sela, Haim Ashkenazy, Kazutaka Katoh, and Tal Pupko. GUIDANCE2: Accurate detection of unreliable alignment regions accounting for the uncertainty of multiple parameters. *Nucleic Acids Research*, 43(W1):W7–W14, 2015.

[127] Mark P Simmons and Helga Ochoterena. Gaps as characters in sequence-based phylogenetic analyses. *Systematic biology*, 49(2):369–381, 2000.

[128] Mark P Simmons, Helga Ochoterena, and Timothy G Carr. Incorporation, relative homoplasy, and effect of gap characters in sequence-based phylogenetic analyses. *Systematic Biology*, 50(3):454–462, 2001.

[129] Claudia Solís-Lemus and Cécile Ané. Inferring phylogenetic networks with maximum pseudolikelihood under incomplete lineage sorting. *PLoS genetics*, 12(3):e1005896, 2016.

[130] Alexandros Stamatakis. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

[131] Alexandros Stamatakis, Paul Hoover, and Jacques Rougemont. A rapid bootstrap algorithm for the RAxML web servers. *Systematic Biology*, 57(5):758–771, oct 2008.

[132] Michael Steel. Recovering a tree from the leaf colourations it generates under a markov model. *Applied Mathematics Letters*, 7(2):19–23, 1994.

[133] Michael PH Stumpf and Gilean AT McVean. Estimating recombination rates from population-genetic data. *Nature Reviews Genetics*, 4(12):959–968, 2003.

[134] David L Swofford. Paup: phylogenetic analysis using parsimony. *Mac Version 3. 1. 1.(Computer program and manual).*, 1993.

[135] Simon Tavaré. Some probabilistic and statistical problems in the analysis of dna sequences. *Lectures on Mathematics in the Life Sciences*, 17(2):57–86, 1986.

[136] Allan Thach, Elizabeth C., Thompson, Karen J., Morris. A fresh look at followership: A model for matching Followership and leadership styles. *Journal of Behavioral & Applied Management*, 14(1):1–5, 2006.

[137] Cuong Than and Luay Nakhleh. Species tree inference by minimizing deep coalescences. *PLoS Computational Biology*, 5(9):e1000501, 2009.

[138] Cuong Than, Derek Ruths, and Luay Nakhleh. Phylonet: a software package for analyzing and reconstructing reticulate evolutionary relationships. *BMC bioinformatics*, 9(1):1–16, 2008.

[139] Jeffrey L Thorne, Hirohisa Kishino, and Joseph Felsenstein. Inching toward reality: an improved likelihood model of sequence evolution. *Journal of molecular evolution*, 34(1):3–16, 1992.

[140] John Tukey. Bias and confidence in not quite large samples. *Ann. Math. Statist.*, 29:614, 1958.

[141] Richard Van Noorden, Brendan Maher, and Regina Nuzzo. The top 100 papers. *Nature News*, 514(7524):550, 2014.

[142] John Wakeley. Substitution rate variation among sites in hypervariable region 1 of human mitochondrial DNA. *Journal of Molecular Evolution*, 37(6):613–623, 1993.

[143] Li-San Wang, Jim Leebens-Mack, P Kerr Wall, Kevin Beckmann, Claude W DePamphilis, and Tandy Warnow. The impact of multiple protein sequence alignment on phylogenetic estimation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(4):1108–1119, 2009.

[144] LUSHENG WANG and TAO JIANG. On the Complexity of Multiple Sequence Alignment. *Journal of Computational Biology*, 1(4):337–348, 2009.

[145] Wei Wang, Jack Smith, Hussein A. Hejase, and Kevin J. Liu. Non-parametric and semi-parametric support estimation using SEquential RESampling random walks on biomolecular sequences. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11183 LNBI:294–308, 2018.

[146] Zhiwei Wang and Kevin J Liu. A performance study of the impact of recombination on species tree analysis. *BMC genomics*, 17(10):165–174, 2016.

[147] Tandy Warnow. Computational phylogenetics: An introduction to designing methods for phylogeny estimation. *Computational Phylogenetics: An Introduction to Designing Methods for Phylogeny Estimation*, pages 1–379, 2017.

[148] Dingqiao Wen, Yun Yu, Matthew W. Hahn, and Luay Nakhleh. Reticulate evolutionary history and extensive introgression in mosquito species revealed by phylogenetic network analysis. *Molecular Ecology*, 25(11):2361–2372, 2016.

[149] Dingqiao Wen, Yun Yu, Jiafan Zhu, and Luay Nakhleh. Inferring phylogenetic networks using phylonet. *Systematic Biology*, 67(4):735–740, 2018.

[150] Oscar Westesson and Ian Holmes. Accurate detection of recombinant breakpoints in whole-genome alignments. *PLoS Computational Biology*, 5(3):e1000318, mar 2009.

[151] Travis J Wheeler and John D Kececioglu. Multiple alignment by aligning alignments. *Bioinformatics*, 23(13):i559–i568, 2007.

[152] K.M. Wong, M.A. Suchard, and J.P. Huelsenbeck. Alignment uncertainty and genomic analysis Supporting online material. *Science*, 319(5862):473, 2008.

[153] Martin Wu, Sourav Chatterji, and Jonathan A. Eisen. Accounting for alignment uncertainty in phylogenomics. *PLoS ONE*, 7(1):e30288, 2012.

[154] Jimmy Yang and Tandy Warnow. Fast and accurate methods for phylogenomic analyses. In *BMC bioinformatics*, volume 12, pages 1–12. BioMed Central, 2011.

[155] Ziheng Yang. Maximum-likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *Molecular biology and evolution*, 10(6):1396–1401, 1993.

[156] Ziheng Yang and Bruce Rannala. Bayesian phylogenetic inference using DNA sequences: A Markov Chain Monte Carlo method. *Molecular Biology and Evolution*, 14(7):717–724, jul 1997.

[157] Ziheng Yang and Bruce Rannala. Molecular phylogenetics: principles and practice. *Nature reviews genetics*, 13(5):303–314, 2012.

[158] Yun Yu, James H Degnan, and Luay Nakhleh. The probability of a gene tree topology within a phylogenetic network with applications to hybridization detection. *PLoS genetics*, 8(4):e1002660, 2012.

[159] Yun Yu, Jianrong Dong, Kevin J Liu, and Luay Nakhleh. Maximum likelihood inference of reticulate evolutionary histories. *Proceedings of the National Academy of Sciences*, 111(46):16448–16453, 2014.

[160] Yun Yu and Luay Nakhleh. A maximum pseudo-likelihood approach for phylogenetic networks. *BMC genomics*, 16(10):1–10, 2015.

[161] Yun Yu, Cuong Than, James H Degnan, and Luay Nakhleh. Coalescent histories on phylogenetic networks and detection of hybridization despite incomplete lineage sorting. *Systematic Biology*, 60(2):138–149, 2011.

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