

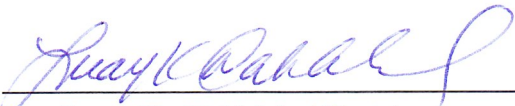
RICE UNIVERSITY

**Models and Methods for Evolutionary Histories  
Involving Hybridization and Incomplete Lineage  
Sorting**

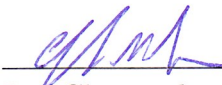
by  
**Yun Yu**

A THESIS SUBMITTED  
IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE  
**Doctor of Philosophy**

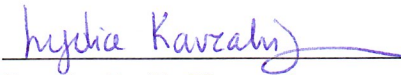
APPROVED, THESIS COMMITTEE:



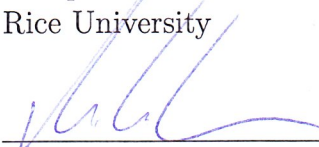
Dr. Luay K. Nakhleh (Chair),  
Associate Professor,  
Computer Science, Rice University



Dr. Christopher Jermaine,  
Associate Professor,  
Computer Science, Rice University



Dr. Lydia E. Kavvaki,  
Noah Harding Professor,  
Computer Science and Bioengineering,  
Rice University



Dr. Michael H. Kohn,  
Associate Professor,  
Ecology and Evolutionary Biology, Rice  
University

Houston, Texas

April, 2014

# Abstract

## Models and Methods for Evolutionary Histories Involving Hybridization and Incomplete Lineage Sorting

by

Yun Yu

Hybridization plays an important evolutionary role in several groups of organisms. A phylogenetic approach to detecting hybridization entails sequencing multiple loci across the genomes of a group of species of interest, reconstructing their gene trees, and exploiting their differences as signal of hybridization. However, methods that follow this approach mostly ignore population effects, such as incomplete lineage sorting (ILS). Given that hybridization occurs between closely related organisms, ILS may very well be at play and, hence, must be accounted for in the analysis framework. Methods that account for both hybridization and ILS currently exist for only very limited cases. The contributions of my work are two-fold:

- I devised the first parsimony criterion for the inference of phylogenetic networks (topologies alone) in the presence of ILS, along with new algorithms for the inference.
- I devised the first likelihood criterion for the inference of phylogenetic networks (topologies, branch lengths, and inheritance probabilities) in the presence of ILS, along with new algorithms for the inference.

I have implemented all the algorithms in our open-source, publicly available PhyloNet software package, and studied their performance in extensive simulation studies. Both the parsimony and likelihood approaches show very good performance in terms of identifying the location of hybridization events, as well as estimating the proportions of genes inherited through hybridization. Also, the parsimony approach shows good performance in terms of efficiency on handling large data sets in the experiments. For the likelihood approach, I used information criteria and cross-validation to account for the model selection issue, and used parametric bootstrap to evaluate the confidence of the inferred species phylogenies. Furthermore, I analyzed two biological data sets (a data sets of yeast genomes and another of house mouse genomes) and found support for hybridization in both.

My work will allow, for the first time, systematic phylogenomic analyses of data sets where hybridization is suspected. Thus, biologists will be able now to revisit existing analyses and conduct new ones with richer evolutionary models and inference methods. Further, the computational techniques presented here can be extended to other reticulate evolutionary events, such as horizontal gene transfer, which are believed to be ubiquitous in bacteria.

# Acknowledgments

This thesis would not have been possible without the help and support of many people, only some of whom it is possible to mention here.

First and foremost, I would like to thank my advisor, Prof. Luay Nakhleh. He is always very patient and encouraging and has excellent advice and unsurpassed knowledge of Phylogenetics. Prof. Nakhleh has been my inspiration whenever I have met with difficulties.

I would like to thank my committee members Prof. Christopher Jermaine, Prof. Lydia Kavraki and Prof. Michael Kohn for their time and patience.

I would like to thank Matt Barnett, Jianrong Dong, Kevin Liu and Nikola Ristic for collaborating some of my work. I really appreciate their insight and helpfulness.

I also want to thank all other group members, Xian Fan, Cuong Than, Justin Park, Troy Ruths, Dingqiao Wen, Natalie Yudin, Hamin Zafar, Wanding Zhou and Angela Zhu for their friendship, encouragement, and insights.

Last but not least, I would like to thank my husband, Chang Li, my son, Kevin Li, and my parents, Wende Yu and Xianping Zhou. They have been very supportive of my work, and I could not have finished this thesis without them.

# Contents

Abstract . . . . .	ii
Acknowledgments . . . . .	iv
List of Figures . . . . .	xviii
List of Tables . . . . .	xxiii
<b>1 Introduction</b>	<b>1</b>
1.1 Contributions of this thesis . . . . .	2
<b>2 Background</b>	<b>4</b>
2.1 Incomplete lineage sorting . . . . .	4
2.2 The species/gene tree problem under ILS . . . . .	6
2.3 Incomplete lineage sorting and hybridization . . . . .	8
2.4 Phylogenetic networks . . . . .	10
2.5 Coalescent histories on phylogenetic networks . . . . .	12
2.5.1 Using only topologies of gene trees . . . . .	12
2.5.2 Using both topologies and branch lengths of gene trees . . . . .	14
<b>3 Parsimonious inference of phylogenetic networks</b>	<b>16</b>

3.1	Computing the minimum number of extra lineages given a phylogenetic network and a gene tree . . . . .	17
3.1.1	An algorithm based on MUL trees . . . . .	19
3.1.2	An algorithm based on weighted ancestral configurations . . . . .	30
3.1.3	Estimating inheritance probabilities . . . . .	37
3.2	Handling gene tree uncertainty . . . . .	38
3.3	Inferring a phylogenetic network . . . . .	40
3.3.1	Neighborhood of a phylogenetic network . . . . .	40
3.3.2	Search strategies . . . . .	42
3.4	Performance . . . . .	47
3.4.1	Simulation study . . . . .	47
3.4.1.1	Evaluating the inference of phylogenetic networks . . . . .	47
3.4.1.2	More loci or more alleles? . . . . .	52
3.4.1.3	Evaluating running time on large data sets . . . . .	54
3.4.1.4	Efficiency of the algorithms . . . . .	57
3.4.1.4.1	MUL-tree based method vs. AC based method . . . . .	57
3.4.1.4.2	Factors affecting the efficiency of the AC based method . . . . .	60
3.4.2	Reanalysis of a yeast ( <i>Saccharomyces</i> ) data set . . . . .	66
3.4.3	The model selection problem . . . . .	68
<b>4</b>	<b>Probabilistic inference of phylogenetic networks</b>	<b>73</b>
4.1	Computing the probability of a gene tree given a phylogenetic network . . . . .	74

4.1.1	Using only topologies of the gene trees . . . . .	74
4.1.1.1	An algorithm based on MUL trees . . . . .	76
4.1.1.2	An algorithm based on weighted ancestral configurations . . . . .	78
4.1.2	Using both topologies and branch lengths of the gene trees . . . . .	83
4.2	Handling gene tree uncertainty . . . . .	86
4.3	Inferring a phylogenetic network . . . . .	89
4.3.1	Optimizing branch lengths and inheritance probabilities of a phylogenetic network . . . . .	90
4.3.1.1	Using only topologies of gene trees . . . . .	90
4.3.1.2	Using both topologies and branch lengths of gene trees . . . . .	92
4.4	Performance . . . . .	94
4.4.1	Simulation study . . . . .	94
4.4.1.1	Identifiability of hybridization using gene tree topologies . . . . .	95
4.4.1.1.1	Accuracy of inference . . . . .	96
4.4.1.1.2	Identifiability . . . . .	99
4.4.1.2	Efficiency of the algorithms . . . . .	110
4.4.1.2.1	MUL-tree based method vs. AC based method . . . . .	110
4.4.1.2.2	Factors affecting the efficiency of the AC based method . . . . .	112
4.4.2	Reanalysis of a yeast ( <i>Saccharomyces</i> ) data set . . . . .	114
4.4.3	Analysis of a house mouse ( <i>Mus musculus</i> ) data set . . . . .	118
4.5	Parametric Bootstrap . . . . .	123

<b>5 Usage of PhyloNet to infer phylogenetic networks</b>	<b>126</b>
<b>6 Conclusions and future work</b>	<b>128</b>
6.1 Future work . . . . .	129



# List of Figures

2.1	Gene/species tree incongruence due to ILS. Given species tree $ST$ , with constant population size throughout and time $t$ in coalescent units (number of generations divided by the population size) between the two divergence events, each of the three gene tree topologies $g_1$ , $g_2$ , and $g_3$ may be observed, with probabilities $1 - (2/3)e^{-t}$ , $(1/3)e^{-t}$ , and $(1/3)e^{-t}$ , respectively. Blue lines within the branches of the species tree is the coalescent history of gene tree $g_2$ . . . . .	5
2.2	Illustration of scenario where incomplete lineage sorting may blur the signal of hybridization. (A) A phylogenetic network and two gene trees (one in blue and the other in red) where incomplete lineage sorting is not involved. (B) A phylogenetic network and two gene trees (one in blue and the other in red) where incomplete lineage sorting is involved and blur the signal of hybridization. . . . .	9

- 2.3 A phylogenetic network  $N$ , and its associated branch lengths and inheritance probabilities. The network has 9 nodes (solid circles), which include the root  $r$ , one network-node,  $h$ , 4 leaves (bijectively labeled by the set  $\mathcal{X} = \{A, B, C, D\}$ ), and 3 internal tree-nodes. Shown also are the branch lengths (red) and inheritance probabilities (blue). . . . . 11
- 2.4 Illustrations of coalescent histories of a gene tree within the branches of a phylogenetic network. The top part is the given phylogenetic network  $N$  and gene tree  $g$  and the bottom part is ten possible coalescent histories. For visual clarity, gene tree nodes are mapped to branches in the phylogenetic network; under our mathematical definition of a coalescent history (see text), drawing a set of gene tree nodes inside a branch  $e = (u, v)$  in the phylogenetic network corresponds to mapping all those nodes to node  $v$  in the phylogenetic network. . . . . 13
- 2.5 A phylogenetic network  $N_{\lambda, \gamma}$ , a gene tree  $g_{\lambda'}$ , and the two possible coalescent histories with respect to coalescence times of  $g_{\lambda'}$  within the branches of  $N_{\lambda, \gamma}$ . One allele is sampled from taxa  $A$  and  $C$ , and two alleles from taxon  $B$ . As shown in the figure,  $\tau_1, \tau_2$  and  $\tau_3$  are the heights of the three internal nodes of  $g_{\lambda'}$ , and  $\eta_1, \eta_2, \eta_3$  and  $\eta_4$  are the heights of four internal nodes of  $N_{\lambda, \gamma}$ . . . . . 15

3.1	Illustration of the conversion from a phylogenetic network to a MUL tree, as well as all allele mappings associated with the case in which single alleles a, b, c and d were sampled from each of the four species A, B, C and D, respectively. . . . .	20
3.2	The MUL tree, branch lengths (red), and inheritance probabilities (blue), that correspond to the phylogenetic network of Fig. 2.3, as generated by Algorithm 1. In the MUL tree, each branch has an inheritance probability; values not shown here equal 1. . . . .	22
3.3	Two phylogenetic networks $N_1$ and $N_2$ , along with their corresponding MUL trees $T_1$ and $T_2$ , respectively. $T_1$ and $T_2$ share the same topology but differ in inheritance probabilities. . . . .	23
3.4	The MUL tree from Fig. 3.2 with its branches numbered. . . . .	26
3.5	Illustration of the dependence of the sets of branches in the MUL tree that correspond to single branches in the phylogenetic network. Given gene tree, MUL tree and allele mapping $f_2$ in Fig. 3.1, the optimal coalescent histories of the gene tree within the branches of the MUL tree (Left) and its corresponding coalescent history in the original phylogenetic network (Right). . . . .	29

- 3.6 The ancestral configurations that result during the computations given phylogenetic network and gene tree  $((a, d), (b, c))$  in Fig. 2.4 under the parsimony approach. Configurations in blue represent configurations generated for nodes and configurations in red represent configurations generated for branches. Curly braces and commas are removed from the ACs for compactness (e.g.,  $ady$  is the set  $\{a, d, y\}$ ). The two identical weighted ACs at the root of the network match the two optimal coalescent histories,  $h_1$  and  $h_2$ , in Table 3.1. . . . . 32
- 3.7 Illustration of estimating inheritance probabilities under MDC criterion. Given phylogenetic network in Fig. 3.1, (Left) the optimal coalescent history of gene tree  $((a, b), (c, d))$ , and (Right) the two equally optimal coalescent histories of gene tree  $((b, c), (a, d))$ . . . . . 38
- 3.8 Phylogenetic networks depicting different hybridization/divergence/extinction scenarios. The  $\alpha$  and  $\beta$  parameters denote the proportions (or, probabilities) of alleles that are inherited from the “left” parents of the reticulation nodes ( $1-\alpha$  and  $1-\beta$  denote the proportions of the alleles that are inherited from the “right” parents of the nodes). . . . . 48
- 3.9 Accuracy of the inferred phylogenetic networks and inheritance probabilities. The three columns from left to right correspond to Scenarios **I**, **II**, and **III** in Fig. 3.8, respectively. One allele per gene per species is sampled. . . 50

- 3.10 The effect of the number of alleles. Accuracy of the phylogenetic networks and inheritance probabilities estimated from gene trees simulated under Scenario **IV**, with true inheritance probabilities  $\alpha = \beta = 0.3$ , where the number of alleles sampled per species also varies. Top and bottom rows correspond to time settings 1 and 2, respectively. . . . . 53
- 3.11 Running time of phylogenetic network inference. The three columns from left to right correspond to data sets with 10, 20 and 40 taxa, respectively. The six rows from bottom to top correspond to data sets with 0, 1, 2, 3, 4 and 5 reticulation nodes, respectively. In each sub-figure, the x-axis is the number of gene trees sampled and the y-axis is the running time in seconds. 56
- 3.12 Accuracy of inferred phylogenetic networks. The three columns from left to right correspond to data sets with 10, 20 and 40 taxa, respectively. . . . . 57
- 3.13 The running times (ln of number seconds) of the MUL-tree based ( $t(MUL)$ ), and AC-based ( $t(AC)$ ) methods for computing parsimonious reconciliations, as well as the speedup  $\log_{10}(t(MUL)/t(AC))$ . . . . . 59
- 3.14 Synthetic data with controlled placements of the reticulation nodes. (Top) A species tree  $ST$ . (Middle)  $N_1$ ,  $N_2$  and  $N_3$  are three phylogenetic networks constructed by adding one reticulation edge to  $ST$  at three different locations. (Bottom) Two gene trees  $g_1$ , which is contained in all three networks, and  $g_2$ , whose coalescent events have to happen above the root of all three networks. . . . . 61

- 3.15 The effects of dependency of reticulation nodes in the species network and different gene tree topologies on the running time of the AC-based algorithms. (Left) A species tree  $ST$ . (Middle)  $N_1$  and  $N_2$  are two species networks constructed by adding seven reticulation edges to  $ST$  at different locations. (Right) two gene trees  $g_1$ , which is a contained tree of both  $N_1$  and  $N_2$ , and  $g_2$  whose coalescent events have to happen above the root of both species networks. . . . . 63
- 3.16 Analysis of the yeast data set, where gene trees are reconstructed using MP. Optimal species phylogenies, along with inheritance probabilities, inferred from gene trees reconstructed by maximum parsimony for the yeast data set of [RWKC03a]. (A) The optimal species tree (network with 0 reticulation nodes). (B) The optimal species network containing one reticulation node. (C) The optimal species network containing two reticulation nodes. For each species phylogeny, the total number of extra lineages (XL) is computed using Eq. (3.12) and reported. . . . . 67
- 3.17 Analysis of the yeast data set, where gene trees are reconstructed using Bayesian inference. Optimal species phylogenies, along with inheritance probabilities, inferred from gene trees reconstructed by MrBayes for the yeast data set of [RWKC03a]. (A) The optimal species tree (network with 0 reticulation nodes). (B) The optimal species network containing one reticulation node. (C) The optimal species network containing two reticulation nodes. For each species phylogeny, the total number of extra lineages (XL) is computed using Eq. 3.11 and reported. . . . . 68

3.18 Network complexity and the number of extra lineages. The decrease in the number of extra lineages in the inferred phylogenetic network as a function of the increase in number of hybridization events inferred. The results were obtained from data pertaining to Scenario **III** in Section 3.4.1.1 under two different settings of the inheritance probabilities and two different settings of the branch lengths. . . . . 69

3.19 Distribution of the number of extra lineages in the neighborhood of an optimal species network. (Left) The distribution of the number of extra lineages of all networks formed from the species tree in Fig. 3.16A by adding a single reticulation edge in all possible ways. (Right) The distribution of the number of extra lineages of all networks formed from the species network in Fig. 3.16B by adding a single reticulation edge in all possible ways. All results are based on the gene trees reconstructed using maximum parsimony, and binned in ranges of size 10 (left) and size 5 (right). . . . . 71

4.1 The ancestral configurations that result during the computations given phylogenetic network and gene tree  $((a, d), (b, c))$  in Fig. 2.4 under the probabilistic approach. Configurations in blue represent configurations generated for nodes and configurations in red represent configurations generated for branches. Curly braces and commas are removed from the ACs for compactness (e.g.,  $ady$  is the set  $\{a, d, y\}$ ). The branch length of branch  $i$  ( $i = 1, \dots, 5$ ) is represented by  $t_i$ . . . . . 81

4.2	Phylogenetic networks depicting different hybridization/divergence/extinction scenarios. The $\alpha$ and $\beta$ parameters denote the proportions (or, probabilities) of alleles that are inherited from the “left” parents of the network-nodes ( $1-\alpha$ and $1-\beta$ denote the proportions of the alleles that are inherited from the “right” parents of the nodes). . . . .	97
4.3	Estimates of $\alpha$ and $\beta$ on Scenario <b>I</b> . Rows from top to bottom correspond to true $(\alpha, \beta)$ values of $(0.0, 0.5)$ , $(0.3, 0.3)$ , and $(0.5, 1.0)$ , respectively. . .	98
4.4	Estimates of $\alpha$ and $\beta$ on Scenario <b>IV</b> . Rows from top to bottom correspond to true $(\alpha, \beta)$ values of $(0.0, 0.5)$ , $(0.3, 0.3)$ , and $(0.5, 0.5)$ , respectively. . .	99
4.5	Estimates of $\alpha$ , $\beta$ , $t_2$ , $t_3$ , and $t_4$ on Scenario <b>V</b> . Rows from top to bottom correspond to true $(\alpha, \beta)$ values of $(0.0, 0.5)$ , $(0.3, 0.3)$ , and $(0.5, 0.5)$ , respectively. All plots correspond to true values of $t_1 = t_2 = t_3 = t_4 = 1.0$ .	100
4.6	Estimates of $\alpha$ , $t_2$ , and $t_3$ on Scenario <b>VI</b> . Rows from top to bottom correspond to true $\alpha$ values of 0.0, 0.3, and 0.5, respectively. All plots correspond to true values of $t_2 = t_3 = 1.0$ . . . . .	101
4.7	Estimates of $\alpha$ and $\beta$ on Scenario <b>II</b> . Rows from top to bottom correspond to true $(\alpha, \beta)$ values of $(0.0, 0.5)$ , $(0.3, 0.3)$ , and $(0.5, 1.0)$ , respectively. . .	102
4.8	The probabilities of the 9 different gene tree topologies (when a single allele is sampled from each of two species A and C, and two alleles are sampled from species B) on the two phylogenetic networks obtained by parameterizing the values of $\alpha$ , $\beta$ , $t_2$ and $t_3$ differently for Scenario <b>II</b> ; see text. Left to right, top to bottom: $t_1 = 0.25, 0.5, 1.0, 2.0, 4.0$ , and $8.0$ , respectively. . . . .	105



4.9	(Left) A phylogenetic network with one of the parents of the hybrids being extinct. (Right) A phylogenetic tree with divergence time $t$ between the two speciation events. . . . .	107
4.10	Values of $t(\alpha, t_2, t_3)$ based on Equation (4.20); from left to right: $\alpha = 0.1$ , 0.5, and 0.9, respectively. . . . .	107
4.11	Estimates of $\alpha$ on Scenario <b>III</b> . (Left) $\alpha = 0.0$ ; (right) $\alpha = 0.3$ . . . . .	108
4.12	Values of $t(\alpha, t_2, t_3)$ based on Equation (4.27); from left to right: $\alpha = 0.1$ , 0.5, and 0.9, respectively. . . . .	110
4.13	The running time (ln of number of seconds) of the AC based algorithm for computing the probability of gene tree topologies given a species network on the data sets described in Section 3.4.1.4.1. The columns from left to right correspond to data sets containing species networks with 1, 4 and 8 reticulation nodes, respectively. . . . .	111
4.14	Various hypotheses for the evolutionary history of a yeast data set. (A) The species tree for the five species <i>Sbay</i> , <i>Skud</i> , <i>Smik</i> , <i>Scer</i> , and <i>Spar</i> , as proposed in [RWKC03b], and inferred using a Bayesian approach [ELP07a] and a parsimony approach [TN09]. (B) A slightly suboptimal tree for the five species, as identified in [ELP07a, TN09]. (C)—(E) The three phylogenetic networks that reconcile both trees in (A) and (B), and which we reported as equally optimal evolutionary histories under a parsimony criterion in [YTDN11]. (F) A phylogenetic network that postulates <i>Smik</i> and <i>Skud</i> as two sister taxa whose divergence followed a hybridization event. . . . .	115

- 4.15 The inferred phylogenetic networks of the *M. musculus* dataset. The rows from top to bottom contain top 5 phylogenetic networks with 0, 1, 2 and 3 reticulation nodes, respectively. In each row, networks are listed from left to right with an decreasing value of log likelihood shown under each of them. . . . . 120
- 4.16 The optimal phylogenetic network inferred on the house mouse (*Mus musculus*) data set. A single individual was sampled from each of five populations: *M.m. domesticus* from France (DF), *M.m domesticus* from Germany (DG), *M.m. musculus* from the Czech Republic (MZ), *M.m. musculus* from Kazakhstan (MK), and *M.m. musculus* from China (MC). The analysis found multiple, almost equally optimal, phylogenetic networks with two reticulation events. These multiple networks all agreed on the recipient populations, but disagreed on the donor populations. One hybridization (the top dashed horizontal arrow) involves the MRCA of DF and DG as a recipient population, yet seems to have involved MK, MC, or their MRCA as the donor population. The second hybridization (the bottom dashed horizontal arrow) involves MZ as a recipient population, yet seems to have involved DF, DG, or their MRCA as the donor population. Branch lengths in coalescent units (on the tree branches) and inheritance probabilities (on the horizontal edges) are shown. . . . . 122
- 4.17 Illustration of parametric bootstrap to assess the significance of the edges in the inferred phylogenetic network. . . . . 125

# List of Tables

- 3.1 The number of extra lineages of all coalescent histories in Fig. 2.4. For every coalescent history  $h$ , columns from 2 to 7 list number of extra lineages on every branch given  $h$ . Branch 6 is the branch incident into the root of the species network  $N$ . A dash means no gene lineages enter that branch. Therefore, the total number of extra lineages of a coalescent history is the summation taken over all branches of the species network. The highlighted coalescent histories are the optimal ones under parsimony which have the minimum number of total extra lineages. . . . . 18
- 3.2 The coalescent histories of the gene tree topology and the corresponding MUL tree of phylogenetic network in Fig. 2.4. Allele mappings in first column are from Fig. 3.1. In the second column,  $x$ ,  $y$ , and  $z$  are the internal nodes of the gene tree, and each number corresponds to the branch in the MUL tree (see Fig. 3.4) to which the internal nodes of the gene tree is mapped. The last column shows the 1-1 correspondence between the coalescent history of the gene tree given the MUL tree and the coalescent history of the gene tree given the phylogenetic network in Fig. 2.4. . . . . 27

- 3.3 The results of running the AC based algorithm for computing the minimum number of extra lineages given gene trees and species networks in Fig. 3.14.  $|\mathcal{AC}_h|$  is the number of configurations at the reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node that contains the largest  $AC_v$  in post-order of traversal is labeled by  $m$  in Fig. 3.14. Furthermore, the last column is the number of allele mappings if using the MUL-tree based method. . . . . 62
- 3.4 The results of running the AC based algorithm for computing the minimum number of extra lineages given gene trees and species networks in Fig. 3.15.  $|\mathcal{AC}_h|$  is the number of configurations at the highest reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node that contains the largest  $AC_v$  set in post-order of traversal is labeled by  $m$  in Fig. 3.15. Furthermore, the last column is the number of allele mappings if using the MUL-tree based method. . . . . 65
- 3.5 Log likelihoods and values of three information criteria computed for the three species phylogeny candidates in Fig. 3.16. . . . . 72

- 4.1 The probabilities of all coalescent histories in Fig. 2.4. For every coalescent history  $h$ , columns from 2 to 7 list the probability of having  $h$  on every branch of the species network  $N$ , where  $t_i$  is the branch length of branch  $i$ . Branch 6 corresponds to the branch incident into the root of the species network  $N$ . A dash means no gene lineages enter that branch. Therefore, the total probability of a coalescent history is the product taken over all branches of the species network. In Fig. 2.4, coalescent events  $y$  and  $z$  can only happen above the root of  $N$ . For every coalescent history, the highlight cell shows where coalescent event  $x$  happens. . . . . 75
- 4.2 The probabilities of all coalescent histories with respect to coalescence times in Fig. 2.5. For every  $ht$ , columns from 2 to 7 list the probability of having  $ht$  on every branch of the species network  $N_{\lambda,\gamma}$ . Branch 6 corresponds to the branch incident into the root of the species network . A dash means no gene lineages enter that branch. Therefore, the total probability of a coalescent history with respect to coalescence times is the product taken over all branches of the species network. In Fig. 2.5, coalescent events  $y$  and  $z$  can only happen above the root of  $N_{\lambda,\gamma}$ . For every  $ht$ , the highlight cell shows where coalescent event  $x$  happens. . . . . 84

- 4.3 The results of running the AC based algorithm for computing the probability of gene tree topologies given gene trees and species networks in Fig. 3.14.  $|\mathcal{AC}_h|$  is the number of configurations at the reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node  $v$  in post-order of traversal that contains the largest  $AC_v$  set is labeled by  $m$  in Fig. 3.14. Furthermore, the last column is the number of valid allele mappings if using the MUL-tree based method. . . . . 113
- 4.4 The results of running the AC based algorithm for computing the probability of gene tree topologies given gene trees and species networks in Fig. 3.15.  $|\mathcal{AC}_h|$  is the number of configurations at the highest reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node  $v$  in post-order of traversal that contains the largest  $AC_v$  set is labeled by  $m$  in Fig. 3.15. Furthermore, the last column is the number of valid allele mappings if using the MUL-tree based method. . . . . 114
- 4.5 Parameter values estimated for the six phylogenies in Fig. 4.14, as well as the values of three information criteria, using gene tree topologies inferred by a Bayesian analysis (using MrBayes). . . . . 116
- 4.6 Parameter values estimated for the six phylogenies in Fig. 4.14, as well as the values of three information criteria, using gene tree topologies inferred by maximum parsimony (using PAUP\*). . . . . 117

- 4.7 The results of information criteria and cross validation of the optimal inferred species networks of the *M. musculus* dataset.  $N(k)$  refers to the optimal inferred species network with  $k$  reticulation nodes. . . . . 121
- 5.1 The usage of command *inferNetwork\_ML* in PhyloNet. The first two parameters are mandatory and all others are optional. . . . . 127

# Chapter 1

## Introduction

Phylogenetic trees have long been a mainstay of biology, providing an interpretative model of the evolution of molecules and characters and a backdrop against which comparative genomics and phonemics are conducted. Nevertheless, some evolutionary events, most notably horizontal gene transfer (HGT) in prokaryotes and hybridization in eukaryotes, necessitate going beyond trees [BvIJ<sup>+</sup>13]. These events result in *reticulate* evolutionary histories, best modeled by *phylogenetic networks*, which account for both vertical and non-vertical evolutionary events [Nak10]. Reticulation events result in genomic regions with local genealogies that are incongruent with the speciation pattern. Several methods and heuristics utilize this incongruence as a signal for inferring reticulation events and reconstructing phylogenetic networks from local genealogies. These methods, which are surveyed in [HRS10, Nak10, Nak13], assume that reticulation events are the sole cause of all incongruence among the gene trees and seek phylogenetic networks to explain all the incongruence.

However, in addition to hybridization, the incongruence among gene trees may be partly



caused by incomplete lineage sorting (ILS), or deep coalescence events [Mad97]. Recent studies have documented large extents of incomplete lineage sorting in groups of organisms across the Tree of Life [SWCL05, PIME06, TSIN08, KWK08, CHW<sup>+</sup>09, WAD<sup>+</sup>09, HDH<sup>+</sup>11, TKS<sup>+</sup>12]. Therefore ignoring the presence of incomplete lineage sorting could result in an over- or under-estimation of the amount of hybridization events and/or wrong inference of the location of these events. Indeed, several recent studies have shown that detecting hybridization in practice can be complicated by the presence incomplete lineage sorting (ILS) [GKB<sup>+</sup>10, EM12, SLM<sup>+</sup>12a, The12, MR12]. A wide array of methods have been developed for species tree inference from gene tree topologies when all incongruence is assumed to be due to incomplete lineage sorting; see [DR09, LYK<sup>+</sup>09, RY08] for recent surveys of such methods.

Recently, a set of methods were devised to analyze data where reticulation and ILS might both be simultaneously at play [HOLM06, MK09, Kub09, JML09, YTDN11, JSO12]. However, these methods are all applicable to simple scenarios of species evolution and mostly assume a known hypothesis about the topology of the phylogenetic network. And therefore, general methods for reconstructing such evolutionary histories are still missing.

## 1.1 Contributions of this thesis

The contributions of my work are two-fold:

- I devised the first parsimony criterion for the inference of phylogenetic networks (topologies alone) in the presence of ILS, along with new algorithms for the inference.

- I devised the first likelihood criterion for the inference of phylogenetic networks (topologies, branch lengths, and inheritance probabilities) in the presence of ILS, along with new algorithms for the inference.

I have implemented all the algorithms in our open-source, publicly available PhyloNet software package [TRN08], and studied their performance in extensive simulation studies. Both the parsimony and likelihood approaches show very good performance in terms of identifying the location of hybridization events, as well as estimating the proportions of genes that underwent hybridization. Also, the parsimony approach shows good performance in terms of efficiency on handling large data sets in the experiments. Further, I analyzed two biological data sets (a data set of yeast genomes and another of house mouse genomes) and found support for hybridization in both.

My work will allow, for the first time, systematic phylogenomic analyses of data sets where hybridization is suspected. Thus, biologists will be able now to revisit existing analyses and conduct new ones with richer evolutionary models and inference methods. Further, the computational techniques presented here can be extended to other reticulate evolutionary events, such as horizontal gene transfer, which are believed to be ubiquitous in bacteria.

# Chapter 2

## Background

### 2.1 Incomplete lineage sorting

Incomplete lineage sorting is best understood under the *coalescent model* [DR06, DS05a, Hud83, Nei86, Nei87, Ros02, Taj83, Tak89]. The coalescent model views gene lineages moving backward in time, eventually coalescing down to one lineage. The term *coalescence* refers to the process in which, looking backward in time, two gene lineages merge at a common ancestor. Under the coalescent model, the evolution of a gene is viewed “backward” in time; that is, from the leaves toward the root. Therefore, we refer to lineages “entering” a directed branch  $b = (u, v)$  as those that, when looking backward in time, come directly from under node  $v$ . Similarly, we refer to lineages “exiting” a branch  $b = (u, v)$  as those that, when looking backward in time, come directly from under node  $u$ . In each time interval between species divergences, lineages entering the interval from a more recent time period might or might not coalesce—an event whose probability is determined largely by the population size and branch lengths. ILS, or *deep coalescence*, refers to the

case in which two lineages fail to coalesce before their speciation events. It is more likely to happen for a larger population or a shorter branch length. For example, in Fig. 2.1, the reconciliation (a way gene tree is reconciled within the branches of a species phylogeny) shown in blue lines within the branches of the species tree  $ST$  indicates that lineage from  $A$  and lineage from  $B$  did not coalesce on branch  $(r, w)$ , and instead, both of them went further. After entering the branch incident with the root  $r$ , lineage from  $B$  coalesce with  $C$  first and then coalesce with  $A$ . The resulting gene tree is  $g_2$  on the left, which disagree with the species tree  $ST$ . In this case, we say incomplete lineage sorting occurred on branch  $(r, w)$  of  $ST$ .

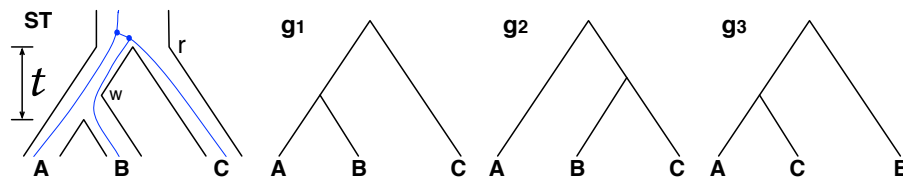


Figure 2.1: Gene/species tree incongruence due to ILS. Given species tree  $ST$ , with constant population size throughout and time  $t$  in coalescent units (number of generations divided by the population size) between the two divergence events, each of the three gene tree topologies  $g_1$ ,  $g_2$ , and  $g_3$  may be observed, with probabilities  $1 - (2/3)e^{-t}$ ,  $(1/3)e^{-t}$ , and  $(1/3)e^{-t}$ , respectively. Blue lines within the branches of the species tree is the coalescent history of gene tree  $g_2$ .

## 2.2 The species/gene tree problem under ILS

The phenomenon of species tree/gene tree incongruence arises in phylogenomic studies [Mad97], and incomplete lineage sorting is one of the factor that may cause this incongruence. A wide array of methods have been developed for species tree inference from gene tree topologies when all incongruence is assumed to be due to incomplete lineage sorting; see [DR09, LYK<sup>+</sup>09, RY08] for recent surveys of such methods.

In Fig. 2.1, the blue lines within the branches of  $ST$  is called a *coalescent history*, which describes how a gene evolves within the branches of a phylogenetic tree. Let  $V(t)$  denote the set of nodes in a tree  $t$ , and let  $t_u$  denote the subtree of tree  $t$  that is rooted at node  $u$ . Given gene tree  $g$  and species tree  $T$ , a *coalescent history* is a function  $h : V(g) \rightarrow V(T)$  such that the following conditions hold:

- if  $w$  is a leaf in  $g$ , then  $h(w)$  is the leaf in  $T$  with the same label (in the case of multiple alleles,  $h(w)$  is the leaf in  $T$  with the label of the species from which the allele labeling leaf  $w$  in  $g$  is sampled); and,
- if  $w$  is a node in  $g_v$ , then  $h(w)$  is a node in  $T_{h(v)}$ .

Given a species tree  $T$  and a gene tree  $g$ ,  $H_T(g)$  denotes the set of all coalescent histories; mathematical properties and algorithms for computing  $H_T(g)$  have been given [Ros07, TRIN07].

Under the coalescent model, a gene tree can be viewed as a random variable conditional on a species tree. For the species tree  $((A, B), C)$ , with time  $t$  between species divergences, Fig. 2.1 shows the three possible outcomes for the gene tree topology random variable,

along with their probabilities. In fact, Degnan and Salter [DS05b] gave the mass probability function of a gene tree topology  $g$  for a given species tree  $\psi$  with branch lengths  $\lambda$  as

$$P_{\psi_\lambda}(G = g) = \sum_{h \in H_\psi(g)} \prod_b p(b, h), \quad (2.1)$$

where  $p(b, h)$  is the probability within a given branch  $b$  of the species tree  $\psi$  that the coalescence events specified by the coalescent history  $h$  occur. In this equation, the summation is taken over all coalescent histories of the gene tree topology, given the species tree and its branch lengths, and the product is taken over all branches of the species tree. Later, Wu [Wu12] proposed an algorithm for faster computation of  $P_{\psi_\lambda}(G = g)$  without explicitly enumerating coalescent histories. Given a collection of gene trees  $\mathcal{G}$ , the inference of species tree becomes finding the optimal species tree  $\psi_{\lambda^*}^*$  such that

$$\psi_{\lambda^*}^* = \operatorname{argmax}_{\psi_\lambda} P(\mathcal{G} | \psi_\lambda) = \operatorname{argmax}_{\psi_\lambda} \prod_{g \in \mathcal{G}} P(g | \psi_\lambda). \quad (2.2)$$

On the other hand, a parsimony approach was proposed for the same goal using minimizing deep coalescence (MDC) as criterion. Given a coalescent history  $h$ , the *number of extra lineages* arising from  $h$  on a branch  $b = (u, v)$  in a species tree  $\psi$  is the number of gene tree lineages exiting branch  $b$  from below node  $u$  toward the root, minus one. So for the species tree and coalescent history shown in Fig. 2.1, the number of extra lineages on branch  $(r, w)$  is 1. Then  $XL(\psi, g)$ , the minimum number of extra lineages required to reconcile a gene tree  $g$  within the branches of a species tree  $\psi$ , can be calculated as

$$XL(\psi, g) = \min_{h \in H_\psi(g)} XL(\psi, h). \quad (2.3)$$

where  $XL(\psi, h)$  is the number of extra lineages arising from coalescent history  $h$  on the entire species tree  $\psi$  which is the sum of the extra lineages over all branches of  $\psi$  given  $h$ .

Finally, the inference of species tree given a collection of gene trees  $\mathcal{G}$  becomes finding the optimal species tree  $\psi^*$  such that

$$\psi^* = \operatorname{argmin}_{\psi} XL(\psi, \mathcal{G}) = \operatorname{argmin}_{\psi} \sum_{g \in \mathcal{G}} XL(\psi, g). \quad (2.4)$$

Efficient algorithms for finding optimal  $\psi^*$  have been developed for cases when the gene tree is rooted or unrooted, binary or non-binary, and on single and multiple alleles [TN09, YWN11a, YWN11b].

## 2.3 Incomplete lineage sorting and hybridization

Hybridization plays an important evolutionary role in several groups of organisms. A phylogenetic approach to detect hybridization entails sequencing multiple loci across the genomes of a group of species of interest, reconstructing their gene trees, and taking their differences as indicators of hybridization. For example, in Fig. 2.2A, there are two gene trees growing within the branches of the phylogenetic network. At the reticulation node, the ancestral alleles  $B$  in these two gene trees are inherited from different parents, which results in two different gene tree topologies  $((A, B), C)$  and  $(A, (B, C))$ . So the incongruence among gene trees can be signal of hybridization when incomplete lineage sorting is not involved.

As hybridization occur between closely related species, incomplete lineage sorting occurs in similar scenario. Fig. 2.2B gave an example of what may happen if hybridization and incomplete lineage sorting are both taken into the picture. The ancestral alleles  $B$  in red gene tree and blue gene tree are inherited from different parents at the reticulation node, which is the same as what is shown in Fig. 2.2A. However, after that, incomplete lineage

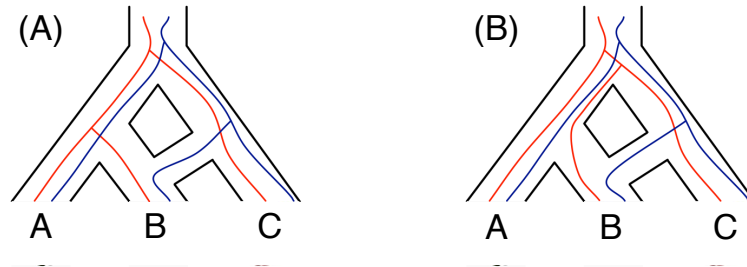


Figure 2.2: Illustration of scenario where incomplete lineage sorting may blur the signal of hybridization. (A) A phylogenetic network and two gene trees (one in blue and the other in red) where incomplete lineage sorting is not involved. (B) A phylogenetic network and two gene trees (one in blue and the other in red) where incomplete lineage sorting is involved and blur the signal of hybridization.

sorting occurred in the red gene tree, and as a result, the red gene tree and blue gene tree have the same topology. In this case, incomplete lineage sorting actually blurs the signal of hybridization. And therefore, methods ignoring the presence of incomplete lineage sorting could result in an over- or under-estimation of the amount of hybridization events and/or wrong inference of the location of these events. It is important to account for incomplete lineage sorting and hybridization simultaneously.

Recently, attempts have been made for this task [HOLM06, MK09, Kub09, JML09, YTDN11, JSO12]. However, they all focused on very limited special cases where the phylogenetic network topology is known and contains one or two hybridization events, and a single allele sampled per species.



## 2.4 Phylogenetic networks

The term *phylogenetic network* has grown to become an umbrella term that encompasses any non-treelike model [HRS10]; therefore, it is important to explicitly describe the phylogenetic network model used. Since I am concerned with hybridization and deep coalescence, I use the evolutionary, or hybridization, phylogenetic network model given in [Nak10], which I now briefly review.

**Definition 1** A phylogenetic  $\mathcal{X}$ -network, or  $\mathcal{X}$ -network for short,  $N$  is an ordered pair  $(G, \ell)$ , where  $G = (V, E)$  is a directed, acyclic graph (DAG) with  $V = \{r\} \cup V_L \cup V_T \cup V_N$ , where

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

$E \subseteq V \times V$  are the network's edges, including reticulation edges whose heads are reticulation nodes, and tree edges whose heads are tree nodes., and  $\ell : V_L \rightarrow \mathcal{X}$  is the leaf-labeling function, which is a bijection from  $V_L$  to  $\mathcal{X}$ .

I use  $V(N)$  and  $E(N)$  to denote the set of nodes and edges of phylogenetic network  $N$  respectively. Fig. 2.3 shows an example of a phylogenetic network based on Definition 1.

In addition to the topology of a phylogenetic network  $N$ , I associate with each branch  $b = (u, v)$  in the network a branch length, denoted by  $\lambda_b$  (equivalently,  $\lambda_{(u,v)}$ ), which reflects the time in coalescent units between the two endpoints of the branch. To describe all branch lengths of a phylogenetic network, a vector  $\lambda$  with one entry per branch is provided. In addition, for each reticulation node  $h$ , with two parent edges  $b_1 = (u, h)$  and  $b_2 = (v, h)$ , I associate inheritance probabilities  $\gamma_{b_1}$  (equivalently,  $\gamma_{(u,h)}$ ) and  $\gamma_{b_2}$  (equivalently,  $\gamma_{(v,h)}$ ), such that  $\gamma_{b_1}, \gamma_{b_2} \in [0, 1]$  and  $\gamma_{b_1} + \gamma_{b_2} = 1$ . The parameter  $\gamma_{(x,h)}$  is taken to denote the proportion of alleles in the population  $h$  that are inherited from population  $x$ . To describe all hybridization probabilities associated with a phylogenetic network, a vector  $\gamma$  with one entry per reticulation edge is provided.

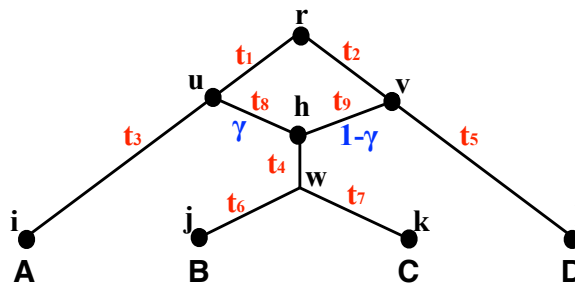


Figure 2.3: A phylogenetic network  $N$ , and its associated branch lengths and inheritance probabilities. The network has 9 nodes (solid circles), which include the root  $r$ , one network-node,  $h$ , 4 leaves (bijectively labeled by the set  $\mathcal{X} = \{A, B, C, D\}$ ), and 3 internal tree-nodes. Shown also are the branch lengths (red) and inheritance probabilities (blue).

Note that a phylogenetic tree is a phylogenetic network with  $V_N = \emptyset$ . For a phylogenetic tree  $T$ ,  $\gamma_b = 1$  for all  $b \in E(T)$ . Hence, we omit the inheritance probabilities  $\gamma$  and

use  $T_\lambda$  to denote a phylogenetic tree  $T$  with branch lengths  $\lambda$ .

## 2.5 Coalescent histories on phylogenetic networks

The notion of *coalescent histories* is central to my work. In this section, I will introduce the definition of coalescent history given a gene tree and a phylogenetic network [YDN12, YDLN14]. Here, I distinguish between cases where whether the branch lengths of the gene tree and phylogenetic network are taken into account or not.

### 2.5.1 Using only topologies of gene trees

In Section 2.2, the definition of coalescent history on species tree is introduced. A similar notion of coalescent histories can be defined on phylogenetic networks. Let  $N$  be a phylogenetic network and  $u$  be a node in  $V(N)$ . I denote by  $N_u$  the set of nodes in  $N$  that are under node  $u$  (that is, the set of nodes that are reachable from the root of  $N$  via at least one path that goes through node  $u$ ). I can now define a coalescent history of a gene tree  $g$  and a species (phylogenetic) network  $N$  as a function  $h : V(g) \rightarrow V(N)$  such that the following conditions hold:

- if  $w$  is a leaf in  $g$ , then  $h(w)$  is the leaf in  $N$  with the same label (the same as above in the case of multiple alleles); and,
- if  $w$  is a node in  $g_v$ , then  $h(w)$  is a node in  $N_{h(v)}$ .

Given a phylogenetic network  $N$  and a gene tree  $g$ , I denote by  $H_N(g)$  the set of all coalescent histories. See Fig. 2.4 for an illustration.

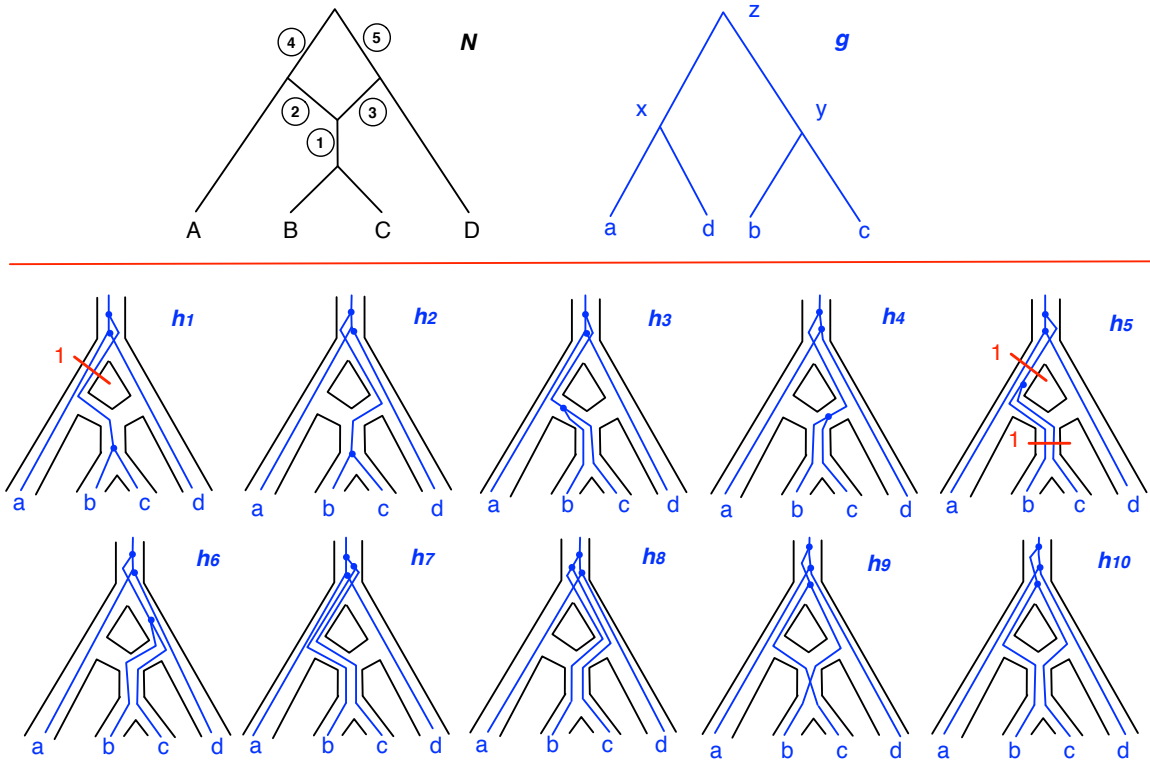


Figure 2.4: Illustrations of coalescent histories of a gene tree within the branches of a phylogenetic network. The top part is the given phylogenetic network  $N$  and gene tree  $g$  and the bottom part is ten possible coalescent histories. For visual clarity, gene tree nodes are mapped to branches in the phylogenetic network; under our mathematical definition of a coalescent history (see text), drawing a set of gene tree nodes inside a branch  $e = (u, v)$  in the phylogenetic network corresponds to mapping all those nodes to node  $v$  in the phylogenetic network.

The algorithm given in [TRIN07] for computing the set  $H_T(g)$  does not apply to the case when the species phylogeny is a network; that is, for computing  $H_N(g)$ . Further, a phylogenetic network is parameterized with inheritance probabilities  $\gamma$  that must be associated properly with the coalescent histories to obtain the gene tree probability.

### 2.5.2 Using both topologies and branch lengths of gene trees

Given a gene tree  $g$  and a species tree  $ST$ , if both the topology and branch lengths of the gene tree are taken into account, then there is only one way of reconciling  $g$  within the branches of  $ST$ . However, when the species phylogeny is a network  $N$ , there might be more than one reconciliation due to different paths gene lineages can take at reticulation nodes of  $N$  when tracing them backwards in time.

First, I use  $\tau_{\psi_{\lambda}}(v)$  to denote the height of node  $v$  in phylogeny  $\psi$  with branch lengths  $\lambda$ . Then given a gene tree  $g_{\lambda'}$  and a phylogenetic network  $N_{\lambda,\gamma}$ , a coalescent history *with respect to coalescence times* can be defined as a function  $ht : V(g_{\lambda'}) \rightarrow E(N_{\lambda,\gamma})$ , such that the following condition holds: for  $h \in H_N(g)$ , if  $h(v) = (x, y)$  and  $\tau_{N_{\lambda}}(x) > \tau_{g_{\lambda'}}(v) \geq \tau_{N_{\lambda}}(y)$ , then  $ht(v) = (x, y)$ . And  $\tau_{g_{\lambda'}}(v)$  tells us exactly on which point of branch  $(x, y)$  coalescent event  $v$  happens. Furthermore, I denote the set of coalescent histories with respect to coalescence times for gene tree  $g_{\lambda'}$  and phylogenetic network  $N_{\lambda,\gamma}$  by  $H_{N_{\lambda,\gamma}}(g_{\lambda'})$ . Clearly,  $H_{N_{\lambda,\gamma}}(g_{\lambda'}) \subseteq H_N(g)$ , but  $H_{N_{\lambda,\gamma}}(g_{\lambda'})$  itself changes with both  $\lambda$  and  $\lambda'$ .

To better illustrate it, an example is shown in Fig. 2.5, where the same phylogenetic network and gene tree are used as the ones in Fig. 2.4, but with branch lengths. We can see that there are only two coalescent histories with respect to coalescence times,  $ht_1$  and  $ht_2$ , resulting from different paths  $b_1$  and  $b_2$  took at the reticulation node. And their corresponding coalescent histories in Fig. 2.4 are  $h_5$  and  $h_6$ , respectively. It is important to note that some  $\lambda$  and  $\lambda'$  may result in  $H_{N_{\lambda,\gamma}}(g_{\lambda'}) = \emptyset$ , which means  $g_{\lambda'}$  cannot be reconciled within the branches of  $N_{\lambda,\gamma}$  with respect to their coalescence times.

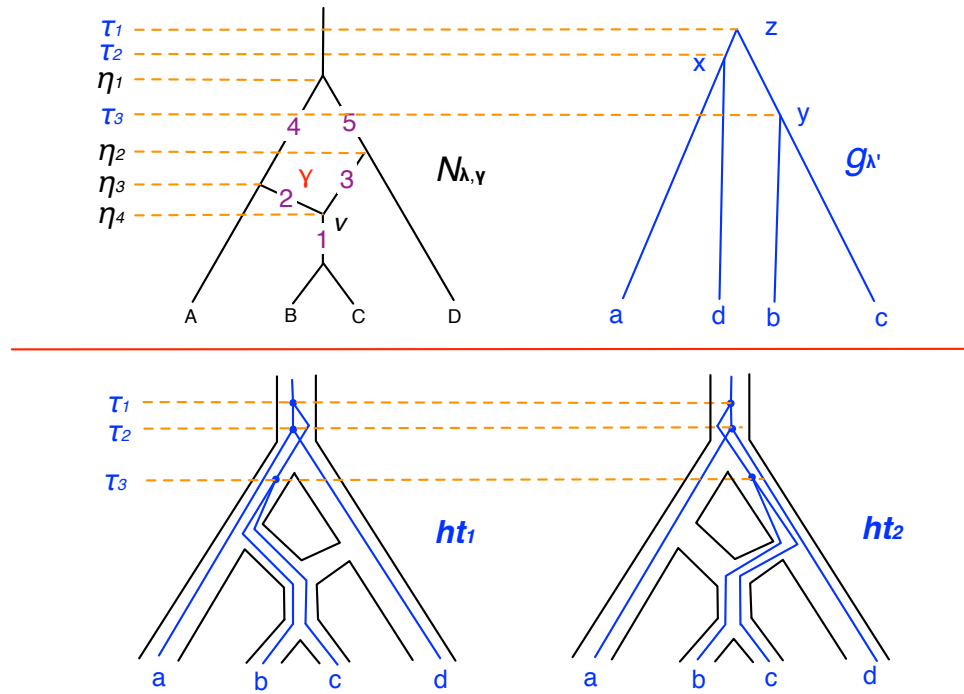


Figure 2.5: A phylogenetic network  $N_{\lambda, \gamma}$ , a gene tree  $g_{\lambda'}$ , and the two possible coalescent histories with respect to coalescence times of  $g_{\lambda'}$  within the branches of  $N_{\lambda, \gamma}$ . One allele is sampled from taxa  $A$  and  $C$ , and two alleles from taxon  $B$ . As shown in the figure,  $\tau_1$ ,  $\tau_2$  and  $\tau_3$  are the heights of the three internal nodes of  $g_{\lambda'}$ , and  $\eta_1$ ,  $\eta_2$ ,  $\eta_3$  and  $\eta_4$  are the heights of four internal nodes of  $N_{\lambda, \gamma}$ .

## Chapter 3

# Parsimonious inference of phylogenetic networks

In this chapter, I propose methods for inferring phylogenetic network from a collection of gene trees under a parsimony criterion MDC (minimizing deep coalescence). More specifically, given a phylogenetic network  $N$  and a collection of gene trees  $\mathcal{G}$ , the goal is to infer the optimal phylogenetic network  $N^*$  such that

$$N^* = \operatorname{argmin}_N XL(N, \mathcal{G}) \quad (3.1)$$

where  $XL(N, \mathcal{G})$  is the minimum total number of extra lineages required to reconcile all gene trees  $\mathcal{G}$  within the branches of  $N$ , which equals

$$XL(N, \mathcal{G}) = \sum_{g \in \mathcal{G}} XL(N, g) \quad (3.2)$$

where  $XL(N, g)$  denotes the minimum number of extra lineages required to reconcile  $g$  within the branches of  $N$ . Note that for this maximum parsimony method  $N$  represents the topology of the network only.

### 3.1 Computing the minimum number of extra lineages given a phylogenetic network and a gene tree

Given a coalescent history  $h$  and a phylogenetic network  $N$ , the number of extra lineages arising from  $h$  on the entire network  $N$ , denoted by  $XL(N, h)$ , is the sum of the extra lineages over all branches of  $N$  (excluding branches that have zero lineages exiting them).

Table 3.1 lists the number of extra lineages of all coalescent histories in Fig. 2.4.

Using coalescent histories, the minimum number of extra lineages required to reconcile gene tree  $g$  within the branches of  $N$ , denoted by  $XL(N, g)$ , can then be calculated by

$$XL(N, g) = \min_{h \in H_N(g)} XL(N, h) \quad (3.3)$$

Obviously, under MDC (minimizing deep coalescence) criterion, the optimal coalescent history refers to the one that results in the fewest number of extra lineages [Mad97, TN09], and thus,

$$XL(N, g) = \sum_{e \in E(N)} [k_e(g) - 1] \quad (3.4)$$

where  $k_e(g)$  is the number of extra lineages on edge  $e$  of  $N$  in the optimal coalescent history of gene tree  $g$ . When the species phylogeny  $N$  is a tree, efficient algorithms have been developed to compute term  $k_e(g)$  in Eq. 3.4 for cases when the gene tree is rooted or unrooted, binary or non-binary, and on single and multiple alleles [TN09, YWN11a, YWN11b].

In this section, I propose two methods for computing  $XL(N, g)$  the minimum number of extra lineages that is required to reconcile a given gene tree within the branches of a given phylogenetic network. One of the two methods is based on the concept of multil-



Table 3.1: The number of extra lineages of all coalescent histories in Fig. 2.4. For every coalescent history  $h$ , columns from 2 to 7 list number of extra lineages on every branch given  $h$ . Branch 6 is the branch incident into the root of the species network  $N$ . A dash means no gene lineages enter that branch. Therefore, the total number of extra lineages of a coalescent history is the summation taken over all branches of the species network. The highlighted coalescent histories are the optimal ones under parsimony which have the minimum number of total extra lineages.

	$XL(N, h)$ on each branch						Total
	1	2	3	4	5	6	
$h_1$	0	0	–	1	–	0	1
$h_2$	0	–	0	–	1	0	1
$h_3$	1	0	–	1	–	0	2
$h_4$	1	–	0	–	1	0	2
$h_5$	1	1	–	1	–	0	3
$h_6$	1	–	1	–	1	0	3
$h_7$	1	1	–	2	–	0	4
$h_8$	1	–	1	–	2	0	4
$h_9$	1	0	0	1	1	0	3
$h_{10}$	1	0	0	1	1	0	3

abeled (MUL) tree [YBN13], and the other is based on the concept of weighted ancestral configurations [YRN13].

### 3.1.1 An algorithm based on MUL trees

Central to this algorithm is converting the phylogenetic network to a multilabeled tree, or MUL tree [HOLM06]. A MUL tree is not a true phylogenetic tree, since its leaves are not uniquely labeled by a taxa set. However, I show in this work that the MUL tree representation of a phylogenetic network allows us to extend the calculation of the minimum number of extra lineages of gene tree on a phylogenetic tree in a straightforward manner to cases where hybridization may be involved.

Given a phylogenetic network  $N$  and a gene tree  $\mathcal{G}$ , the approach for computing the minimum number of extra lineages to reconcile gene tree  $\mathcal{G}$  within the branches of network  $N$  has three steps. First,  $N$  is converted into a MUL tree  $T$ ; second, the alleles at the tips of  $\mathcal{G}$  are mapped in every valid way to the tips of  $T$ ; and, finally, the minimum number of extra lineages of  $N$  and  $\mathcal{G}$  is computed as the minimum, over all allele mappings, of number of extra lineages of  $\mathcal{G}$  given  $T$  (see Fig. 3.1).

#### Step 1: Converting the phylogenetic network to MUL tree

Central to our formulation/algorithm for computing the probability of a gene tree given a phylogenetic network is converting the phylogenetic network to a multilabeled tree, or MUL tree [HOLM06]. A MUL tree is not a true phylogenetic tree, since its leaves are not uniquely labeled by a taxa set. However, we show in this work that the MUL tree representation of a phylogenetic network allows us to extend coalescent-based calculations

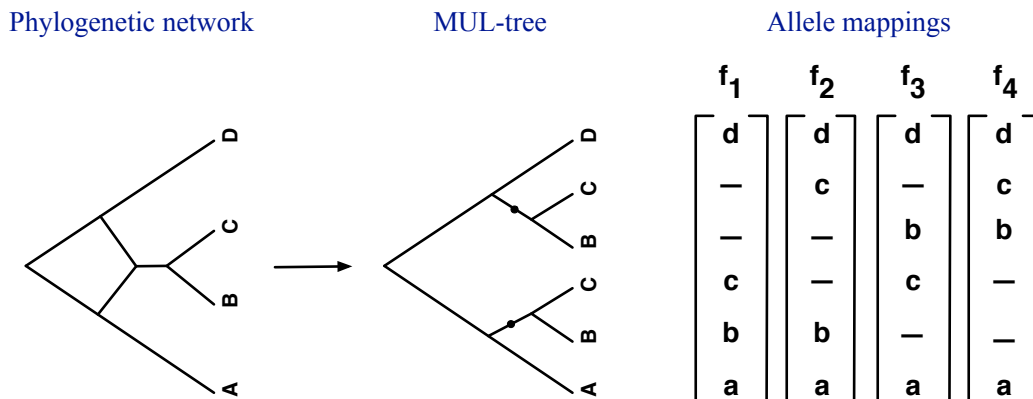


Figure 3.1: Illustration of the conversion from a phylogenetic network to a MUL tree, as well as all allele mappings associated with the case in which single alleles a, b, c and d were sampled from each of the four species A, B, C and D, respectively.

of gene tree probabilities in a straightforward manner to cases where hybridization may be involved.

It is straightforward to convert a phylogenetic network into its corresponding MUL tree. The main idea is to process the phylogenetic network in a bottom-up fashion, traversing its nodes from the leaves towards the root. Every time a network-node  $h$  is encountered, a copy of the tree rooted at  $h$  is created, and each of  $h$ 's two parents points to exactly one of these two copies. As the traversal operates in a bottom-up fashion, it is guaranteed that when a network-node is encountered, there are no network-nodes remaining “under” it (they would have been processed already). In addition to the topology, the conversion maps the branch lengths and inheritance probabilities to the appropriate branches as well. Finally, as a single edge in a phylogenetic network  $N$  may give rise to multiple edges in the MUL tree  $T$ , in order to keep track of which branches in the MUL tree originated from the same branch in the phylogenetic network, we build during the conversion a mapping  $w$

from the set of the MUL tree branches to the set of the phylogenetic network branches, such that  $\phi(e') = e$  if branch  $e'$  in the MUL tree corresponds to branch  $e$  in the phylogenetic network. The usage of this  $\phi$  mapping will become clearer below. Upon completion of this step of converting the phylogenetic network  $N$ , its branch lengths  $\lambda$  and inheritance probabilities  $\gamma$ , the result is a MUL tree  $T$  along with its branch lengths  $\lambda'$ , inheritance probabilities  $\gamma'$ , and the branch mapping  $\phi : E(T) \rightarrow E(N)$ . The full description of the procedure is given formally in Algorithm 1 (**NetworkToMULTree**).

---

**Algorithm 1: NetworkToMULTree.**

---

**Input:** Phylogenetic  $\mathcal{X}$ -network  $N$ ; branch lengths  $\lambda$ ; inheritance probabilities  $\gamma$ .

**Output:** MUL tree  $T$ ; branch lengths  $\lambda'$ ; inheritance probabilities  $\gamma'$ ; edge mapping

$$\phi : E(T) \rightarrow E(N).$$

$T \leftarrow N$  and set  $\phi(e') = e$  where  $e' \in E(T)$  is a copy of  $e \in E(N)$ ;

$\lambda' \leftarrow \lambda$ ;

**foreach**  $b \in E(T)$  **do**

|  $\gamma'_b \leftarrow 1$ ;

**while** *traversing the nodes of  $T$  bottom-up* **do**

| **if** *node  $h$  has two parents,  $u$  and  $v$*  **then**

| Create a copy of  $T_h$  whose root is new node  $h'$  and set  $\phi(e') = \phi(e)$  where

|  $e' \in E(T_{h'})$  is a copy of  $e \in E(T_h)$ ;

| Add a new edge  $(v, h')$  to  $T$  and  $\gamma'_{(v,h')} \leftarrow \gamma_{(v,h)}$ ;

| Delete edge  $(v, h)$  from  $T$ , as well as  $\gamma'_{(v,h)}$ ,  $\lambda'_{(v,h)}$  and  $\phi_{(v,h)}$ ;

**return**  $T$ ;

---

The MUL tree  $T$  that corresponds to the phylogenetic network  $N$  of Fig. 2.3 is given in Fig. 3.2. In this example, traversing the phylogenetic network from the leaves towards the root, the reticulation node  $h$  is encountered who has parents  $u$  and  $v$ , a copy of the subtree rooted at  $h$  is created, and then one of the two copies of  $h$  is attached as a child of  $u$ , and the other is attached as a child of  $v$ , resulting in the MUL tree shown in Fig. 3.2 along with its branch lengths and inheritance probabilities. The corresponding branch mapping  $\phi : E(T) \rightarrow E(N)$  is listed below:

- $\phi((u, i)) = (u, i), \phi((v, l)) = (v, l), \phi((u, h)) = (u, h), \phi((v, h')) = (v, h), \phi((r, u)) = (r, u)$ , and  $\phi((r, v)) = (r, v)$ .
- $\phi((h, w)) = \phi((h', w')) = (h, w)$ .
- $\phi((w, j)) = \phi((w', j')) = (w, j)$ .
- $\phi((w, k)) = \phi((w', k')) = (w, k)$ .

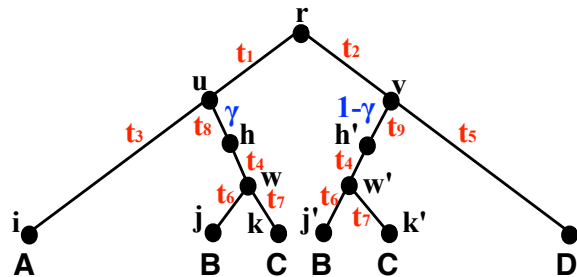


Figure 3.2: The MUL tree, branch lengths (red), and inheritance probabilities (blue), that correspond to the phylogenetic network of Fig. 2.3, as generated by Algorithm 1. In the MUL tree, each branch has an inheritance probability; values not shown here equal 1.

It is important to note that it is possible that two different phylogenetic network topologies give rise to the same MUL tree topology, and under certain settings of branch lengths and inheritance probabilities, the networks may also give rise to identical MUL tree topologies and branch parameters (which, by definition, would result in non-identifiability of the topology and/or parameter values). However, if the parameter values differ between the two networks, they may still be identifiable, even though the two networks give rise to the same MUL tree topology. This issue is illustrated in Fig. 3.3.

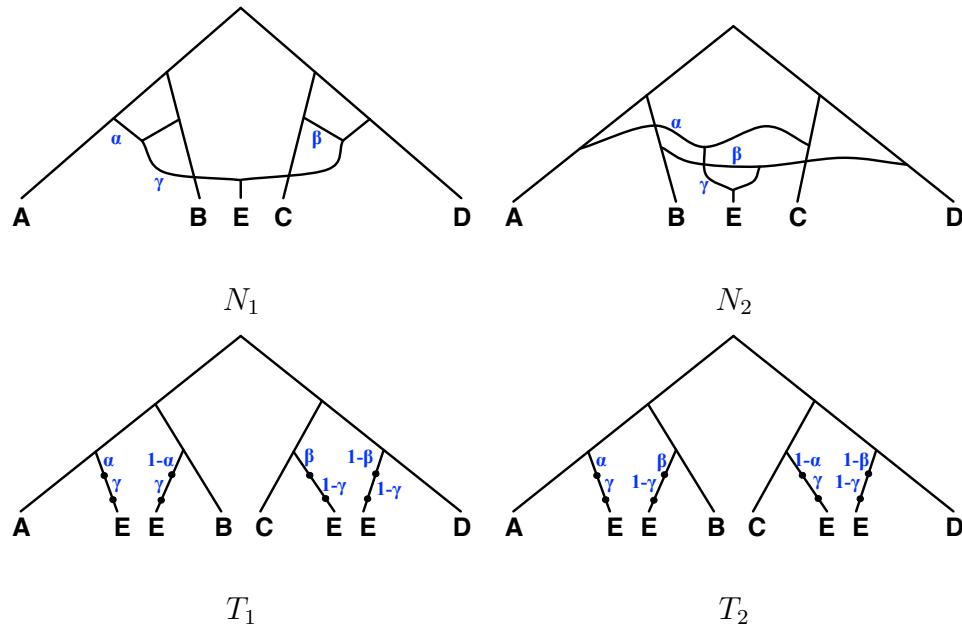


Figure 3.3: Two phylogenetic networks  $N_1$  and  $N_2$ , along with their corresponding MUL trees  $T_1$  and  $T_2$ , respectively.  $T_1$  and  $T_2$  share the same topology but differ in inheritance probabilities.

The phylogenetic network  $N_1$  involves a hybridization between A and B, a hybridization between C and D, and a hybridization of the two hybrids. MUL tree  $T_1$  is obtained from  $N_1$ . The phylogenetic network  $N_2$  involves a hybridization between A and C, a hy-

bridization between B and D, and a hybridization between the two hybrids. MUL tree  $T_2$  is obtained from  $N_2$ . As shown in the figure,  $T_1$  and  $T_2$  share the same topology, but differ in inheritance probabilities. Further, different lengths of the branches of the two networks would result in different branch lengths of the MUL trees produced from each of the networks.

## Step 2: Mapping the alleles to the leaves of the MUL tree

In computing the minimum number of extra lineages required to reconcile a gene tree within the branches of a phylogenetic network, all the alleles sampled from species  $x$  are mapped to the single leaf labeled  $x$  in the species phylogeny. However, unless the phylogenetic network  $N$  does not have any reticulation nodes, the resulting MUL tree  $T$  contains leaf sets that are labeled by the same species  $x$ . For example, in Fig. 3.1, the MUL tree has two leaves labeled  $B$  and two leaves labeled  $C$ . In this case, it is important to map the alleles systematically to the leaves of the MUL tree so as to cover *exactly* all the coalescence patterns that would arise had the alleles been mapped to the phylogenetic network.

I denote by  $c_x$  the set of leaf nodes in  $T$  that are labeled by species  $x$ . For example,  $c_B$  for the MUL tree in Fig. 3.1 is the set of the two leaves labeled by  $B$ . Now, consider a locus  $l$ . I denote by  $A_x$  (for  $x \in \mathcal{X}$ ) the set of alleles sampled from species  $x$  for locus  $l$ , and by  $a_x$  the size of this set (i.e.,  $a_x = |A_x|$ ). In the example of Fig. 3.1, one allele  $b$  was sampled from species  $B$ ; hence,  $A_B = \{b\}$  and  $a_B = 1$ . An *allele mapping* is a function  $f : (\cup_{x \in \mathcal{X}} A_x) \rightarrow (\cup_{x \in \mathcal{X}} c_x)$  such that if  $f(a) = d$ , and  $d \in c_x$ , then  $a \in A_x$ . In other words,  $f$  maps an allele from species  $x$  to a leaf in the MUL tree labeled by  $x$ . Let  $\mathcal{F}_{T,g}$  denote the set of all such allele mappings  $f$  given MUL tree  $T$  and gene tree  $\mathcal{G}$ ; in Fig. 3.1,

$$\mathcal{F}_{T,g} = \{f_1, f_2, f_3, f_4\}.$$

### Step 3: Computing the minimum number of extra lineages of a gene tree on the MUL tree

Once the MUL tree  $T$  and the set of all allele mappings  $\mathcal{F}_{T,g}$  are obtained, the minimum number of extra lineages of gene tree  $\mathcal{G}$  within MUL tree  $T$  can be computed as the minimum number of extra lineages of  $\mathcal{G}$  within  $T$  over all possible allele mappings. Let  $XL(T, g)$  be the minimum number of extra lineages of  $\mathcal{G}$  within  $T$ , and  $XL(T, g, f)$  be the minimum number of extra lineages of  $\mathcal{G}$  within  $T$  under allele mapping  $f$ . Then  $XL(T, g)$  can be computed as

$$XL(T, g) = \min_{f \in \mathcal{F}_{T,g}} XL(T, g, f). \quad (3.5)$$

Before introducing how to complete the computation in Eq. 3.5, it is important to first understand what is coalescent history given a gene tree and a MUL tree. Let  $T$  be a MUL tree,  $\mathcal{G}$  be a gene tree, and  $f$  be an allele mapping. Then, a *coalescent history* is a function  $h : V(g) \rightarrow V(T)$  such that the following conditions hold:

- if  $w$  is a leaf in  $\mathcal{G}$ , then  $h(w) = f(a)$  where  $a$  is the allele that labels leaf  $w$ ; and,
- if  $w$  is a node in  $g_v$ , then  $h(w)$  is a node in  $T_{h(v)}$ .

I denote by  $H_{T,f}(g)$  the set of all coalescent histories of gene tree  $\mathcal{G}$  within the branches of MUL tree  $T$  given the allele mapping  $f$ .

Table 3.2 lists all the coalescent histories of the gene tree and the corresponding MUL tree of the phylogenetic network in Fig. 2.4. Each row in the table gives the branches of the MUL tree (see Fig. 3.4 for numbers of branches of the MUL tree) on which the coalescent



events, represented by the gene tree internal nodes  $x$ ,  $y$  and  $z$ , occur. For each coalescent history within the branches of the MUL tree, the corresponding coalescent history within the branches of the original phylogenetic network in Fig. 2.4 is given in the last column. Note that there is a 1-1 correspondence, which implies that the allele mappings of the MUL tree do cover exactly all the coalescence patterns that would arise had the alleles been mapped to the phylogenetic network.

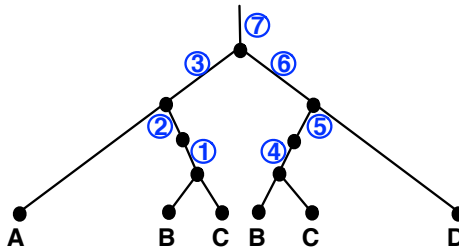


Figure 3.4: The MUL tree from Fig. 3.2 with its branches numbered.

As shown in Eq. 3.4,  $XL(N, g)$  is equal to the minimum number of extra lineages over all coalescent histories of gene tree  $\mathcal{G}$  within the branches of phylogenetic network  $N$ . This also applies to  $XL(T, g, f)$ . I denote by  $H_{T,f}(g)$  the set of all coalescent histories of  $\mathcal{G}$  within the branches of  $T$  under mapping  $f$ , and by  $XL(T, h)$  the number of extra lineages of a certain coalescent history  $h \in H_{T,f}(g)$ . Then  $XL(T, g, f)$  in Eq. 3.5 can be calculated as

$$XL(T, g, f) = \min_{h \in H_{T,f}(g)} XL(T, h). \quad (3.6)$$

Converting a phylogenetic network into a MUL tree enables me to avoid dealing with network topologies and make use of the existing techniques that have been developed for trees. More specifically, techniques for fast computing Eq. 3.4 without explicitly enumerating all coalescent histories when  $N$  is a tree [TN09, YWN11a, YWN11b] can be applied to

Table 3.2: The coalescent histories of the gene tree topology and the corresponding MUL tree of phylogenetic network in Fig. 2.4. Allele mappings in first column are from Fig. 3.1. In the second column,  $x$ ,  $y$ , and  $z$  are the internal nodes of the gene tree, and each number corresponds to the branch in the MUL tree (see Fig. 3.4) to which the internal nodes of the gene tree is mapped. The last column shows the 1-1 correspondence between the coalescent history of the gene tree given the MUL tree and the coalescent history of the gene tree given the phylogenetic network in Fig. 2.4.

Allele mapping	$x$	$y$	$z$	Coal. hist. in Fig. 2.4
$f_1$	7	1	7	$h_1$
	7	2	7	$h_3$
	7	3	7	$h_5$
	7	7	7	$h_7$
$f_2$	7	7	7	$h_{10}$
$f_3$	7	7	7	$h_9$
$f_4$	7	4	7	$h_2$
	7	5	7	$h_4$
	7	6	7	$h_6$
	7	7	7	$h_8$

$XL(T, g, f)$ , too. However, MUL tree is not a regular species tree, special attention needs to be paid to sets of branches in the MUL tree that correspond to single branches in the phylogenetic network, since coalescence events within these branches are not independent. Let me illustrate this issue using MUL tree  $T$  and allele mapping  $f_2$  in Fig. 3.1. Under this mapping, the optimal coalescent history is shown on the left in Fig. 3.5. Tracing allele from  $B$  and allele from  $C$  independently implicitly indicates that tracing the evolution of these two alleles in the phylogenetic network, no coalescence event should occur on the branch incident into leaf  $B$  in the network. And the number extra lineages is 0 for both edge  $(h_1, w_1)$  and  $(h_2, w_2)$ . However, if going back to the original phylogenetic network, the corresponding coalescent history on the right in Fig. 3.5 shows clearly that there is 1 extra lineage on edge  $(h, w)$  of the network. In fact, edge  $(h_1, w_1)$  and  $(h_2, w_2)$  in MUL tree are both copies of edge  $(h, w)$  in the network, or  $\phi(h_1, w_1) = \phi(h_2, w_2) = (h, w)$ , which implies that branches in the MUL tree that originally come from the same branch in the phylogenetic network should be handled together. More specifically, let  $e'$  be an edge in  $N$ . Given the mapping  $\phi$  from the branches of  $T$  to the branches of  $N$ , the pre-image (or, inverse image)  $\phi^{-1}(e')$  is the set of all branches in  $T$  that map to  $e'$  under  $\phi$ . That is,  $\phi^{-1}(e') = \{e \in E(T) : \phi(e) = e'\}$ , where  $E(T)$  is the set of  $T$ 's branches. to account for this issue. Then, Eq. 3.4 is modified for MUL tree as follows

$$XL(T, g, f) = \sum_{e' \in E(N)} \left[ \sum_{e \in \phi^{-1}(e')} k_e(g, f) - 1 \right]. \quad (3.7)$$

where  $k_e(g, f)$  is number of lineages on branch  $e$  in the optimal coalescent history of  $\mathcal{G}$  under allele mapping  $f$ . Techniques to compute  $k_e(g, f)$  completely follow the existing

methods [TN09, YWN11a, YWN11b].

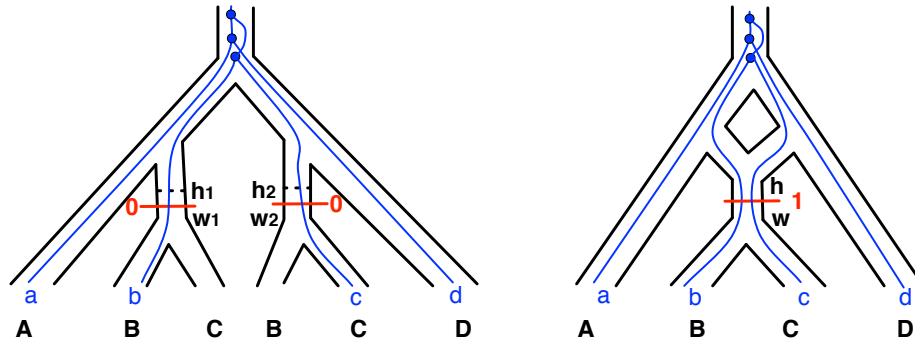


Figure 3.5: Illustration of the dependence of the sets of branches in the MUL tree that correspond to single branches in the phylogenetic network. Given gene tree, MUL tree and allele mapping  $f_2$  in Fig. 3.1, the optimal coalescent histories of the gene tree within the branches of the MUL tree (Left) and its corresponding coalescent history in the original phylogenetic network (Right).

Finally, plugging Eq. 3.7 into Eq. 3.5, the minimum number of extra lineages of a gene tree  $\mathcal{G}$  within a MUL tree  $T$  can be calculated by

$$XL(T, g) = \min_{f \in \mathcal{F}_{T, g}} \sum_{e' \in E(N)} \left[ \sum_{e \in \phi^{-1}(e')} k_e(g, f) - 1 \right]. \quad (3.8)$$

The running time of this method depends on the number of allele mappings which is exponential in a combination of the number of alleles sampled and the number of reticulation nodes. More precisely, for every leaf  $x$  in  $N$ , let  $a(x)$  be the number of alleles sampled from  $x$  and  $h(x)$  be the maximum number of reticulation nodes on all possible paths from  $x$  to the root of  $N$ , then the number of allele mappings is at most  $2^{\sum_{x \in V_L(N)} h(x)a(x)}$  where  $V_L(N)$  is the set of leaves of  $N$ . In addition, the MUL tree  $T$  converted from  $N$  can have at most  $\sum_{x \in V_L(N)} h(x)$  leaves, so  $XL(T, g, f)$  can be computed in

$O(\sum_{x \in V_L(N)} h(x))$  time [YWN11b, YWN11a]. As a result,  $XL(T, g)$  can be computed in  $O(\sum_{x \in V_L(N)} h(x) \cdot 2^{\sum_{x \in V_L(N)} h(x)a(x)})$  time. Clearly, the running time of this method does not change for different gene tree topologies.

### 3.1.2 An algorithm based on weighted ancestral configurations

#### Ancestral configurations on networks

Central to this method is the concept of weighted *ancestral configuration* (AC, or simply configuration). When its unweighted version was first introduced, it was defined on species trees for computing the probability of gene tree topologies [Wu12]. However, the concept is extended significantly here to apply to networks.

Given a species network  $N$  with  $q = |V_N|$  reticulation nodes numbered  $1, 2, \dots, q$  and a gene tree  $g$  on set  $\mathcal{Y}$  of alleles, an ancestral configuration can be associated with a node  $v$  of  $N$ , denoted by  $AC_v$ , or an edge  $e$  of  $N$ , denoted by  $AC_e$ , and is an element of the set  $2^{\mathcal{Y}} \times \mathbb{Z}^q \times \mathbb{R}$  where the first element is the set of all subsets of alleles in  $\mathcal{Y}$ , the second is the set of all vectors of integers of size  $q$ , and the third element is the set of real numbers. When the context is clear, I omit the subscript. For an AC  $(B, a, w)$ , the interpretation is as follows:

- $B \subseteq A$ : a set of lineages that exist at the point (node or edge) with which the AC is associated.
- $a[i]$ , for  $1 \leq i \leq q$ : an index for the AC split that occurred at reticulation node  $i$  and gave rise to  $B$ .

- $w$ : a weight of the AC; I discuss below how to set/use this entry.

Given two ACs,  $AC_1 = (B_1, a_1, w_1)$  and  $AC_2 = (B_2, a_2, w_2)$ , then  $AC_1$  and  $AC_2$  are considered to be *compatible* if for each  $i$ ,  $1 \leq i \leq q$ , either  $a_1[i] = a_2[i]$  or  $a_1[i] \cdot a_2[i] = 0$ ; otherwise, the two ACs are *incompatible*. Further, if  $B_1 = B_2$  and  $a_1 = a_2$ , I say that the two ACs are *identical*.

Ancestral configurations are computed in a bottom-up fashion by algorithm below. Two major operations that occur as the algorithm proceed bottom-up are:

- Splitting an AC whenever a reticulation node is encountered. Let  $(B, a, w)$  be an AC on the edge incident out of reticulation node  $k$ . Further, assume that for each reticulation node  $i$  ( $1 \leq i \leq q$ ), we have a counter  $o_i$ , that is initialized to 0 at the start of an algorithm. Splitting  $(B, a, w)$  at node  $k$  results in two ACs  $AC_1 = (B_1, a_1, w_1)$  and  $AC_2(B_2, a_2, w_2)$ , each associated with one of the two reticulation edges, such that  $B_1 \cup B_2 = B$ ,  $B_1 \cap B_2 = \emptyset$ ,  $a_1[k] = a_2[k] = o_k + 1$ , and  $o_k$  is incremented by 1. For the weights,  $w_1 = w$  and  $w_2 = 0$ .
- Merging two ACs whenever an internal tree node is encountered. Let  $(B_1, a_1, w_1)$  and  $(B_2, a_2, w_2)$  be two compatible ACs associated with the edges incident from a tree node  $u$ . Then, these two ACs are merged into one AC  $(B, a, w)$  at node  $u$  where  $B = B_1 \cup B_2$  and  $a[i] = \max\{a_1[i], a_2[i]\}$  for all  $1 \leq i \leq q$ . For the weights,  $w = w_1 + w_2$ .

For  $AC = (B, a, w)$  I denote by  $n(AC)$  the quantity  $|B|$ . I denote by  $\mathcal{AC}$  the set of ACs associated with a node or edge. When  $\mathcal{AC}$  is associated with an edge, it denotes the set of ACs that result after all coalescence events took place on the edge. Fig. 3.6 shows the

sets of all ACs constructed during the executions of the algorithms CountXL below. In this example, in the network on the right in Fig. 3.6, the first configuration listed (with the constituent set  $\{a, b_1, b_2\}$ ) on branch 4 is compatible with the first configuration listed (with the constituent set  $\{c\}$ ) on branch 5, but is incompatible with the second configuration listed (with the constituent set  $\{c, b_1, b_2\}$ ) on branch 5.

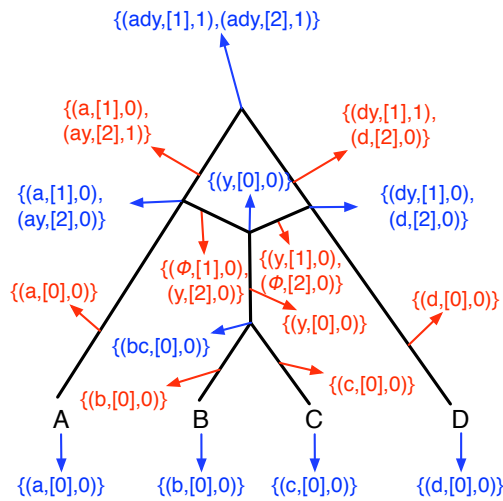


Figure 3.6: The ancestral configurations that result during the computations given phylogenetic network and gene tree  $((a, d), (b, c))$  in Fig. 2.4 under the parsimony approach. Configurations in blue represent configurations generated for nodes and configurations in red represent configurations generated for branches. Curly braces and commas are removed from the ACs for compactness (e.g.,  $ady$  is the set  $\{a, d, y\}$ ). The two identical weighted ACs at the root of the network match the two optimal coalescent histories,  $h_1$  and  $h_2$ , in Table 3.1.

Assume  $m$  and  $n$  are two gene lineages that meet at some node in a gene tree  $g$ . When reconciling  $g$  within the edges of a species network  $N$ , after the two entered the same edge of  $N$ , they might or might not have coalesced before leaving that edge, the probability of

which depends on the length (in terms of time) and width (in terms of population size) of that edge. Therefore, one configuration entering a edge of  $N$  might give rise to several different configurations leaving that edge with different probabilities. For example, suppose a gene tree  $g$  has a subtree  $((a, b)x, c)y$  (tree with root  $y$ , leaf-child  $c$  of the root, child  $x$  of the root, and two leaves  $a$  and  $b$  that are children of  $x$ ). Then if a configuration  $(\{a, b, c\}, p, w)$  entered a edge of  $N$ , it could give rise to one of three different configurations leaving that branch, including  $\{a, b, c\}$ ,  $\{x, c\}$  and  $\{y\}$ . I denote by  $Coal(B, g)$ , for a set  $B$  of lineages and gene tree  $g$ , the set of all sets of lineages that  $B$  could coalesce into with respect to the topology of  $g$ . Ancestral configurations provide a compact representation of coalescent histories, thus allowing for efficient computing: while redundant parts that appear in different coalescent histories must be computed explicitly every time they are encountered, particularly over the different allele mappings employed in the approaches of [YBN13] introduced in Chapter 3.1, using ancestral configurations ameliorates this by computing the values only once for each ancestral configuration. Further, when these computations are coupled with network space search, local perturbations to candidate networks necessitate new computations to only a small number of ancestral configurations. I now show how to use configurations to compute  $XL(N, g)$  efficiently.

### Counting the number of extra lineages under ILS and hybridization

For a configuration  $AC$ , I denote by  $xl(AC)$  the minimum number of extra lineages arising from coalescing the extant gene lineages in  $AC$  to the present gene lineages in  $AC$ . In this method, weight  $w$  in  $(B, a, w) \in \mathcal{AC}$  corresponds to  $xl(AC)$ , where  $\mathcal{AC}$  is either  $\mathcal{AC}_v$  where  $v$  is a node, or  $\mathcal{AC}_b$  where  $b$  is a edge.



**Observation 1** Let  $AC = (B, a, w)$  be a configuration entering a edge  $b$  and  $AC^+ = (B^+, a^+, w^+)$  be a configuration that  $AC$  coalesced into before leaving  $b$ . Then

$$w^+ = w + \max\{n(AC^+) - 1, 0\} \quad (3.9)$$

where  $n(AC^+)$  is the number of lineages on edge  $b$ .

I define a function called **CreateCACsForXL** which takes a gene tree  $g$ , an edge  $b = (u, v)$  of the network  $N$  and a set of ACs  $\mathcal{AC}_v$  that enter edge  $b$ , and returns a set of ACs  $\mathcal{AC}_{(u,v)}$  that exit edge  $b$ . See Alg. 2 for details. Note that although one configuration can coalesce into several different configurations along an edge, under parsimony only the one that has the minimum total number of extra lineages needs to be kept. Therefore  $|\mathcal{AC}_v| = |\mathcal{AC}_{(u,v)}|$  and there is a 1-1 correspondence between configurations in  $|\mathcal{AC}_v|$  and configurations in  $|\mathcal{AC}_{(u,v)}|$ . Note that if node  $v$  is a reticulation node,  $|\mathcal{AC}_v|$  here represents the set of ACs that about to enter branch  $(u, v)$  after splitting.

---

**Algorithm 2: CreateCACsForXL.**

---

**Input:** Gene tree  $g$ , edge  $b = (u, v)$ , set of ACs  $\mathcal{AC}_v$

**Output:** A set of ACs  $\mathcal{AC}_{(u,v)}$

**foreach**  $(B, a, w) \in \mathcal{AC}_v$  **do**

$B^+ \leftarrow \operatorname{argmin}_{B' \in \text{Coal}(B, g)} |B'|;$

Compute  $w^+$  using Eq. 3.9;

$\mathcal{AC}_{(u,v)} \leftarrow \mathcal{AC}_{(u,v)} \cup (B^+, a, w^+);$

**return**  $\mathcal{AC}_{(u,v)}$

---

For a phylogenetic network  $N$  and a gene tree  $g$ , the algorithm for computing the minimum number of extra lineages required to reconcile  $g$  within  $N$  is shown in Alg. 3. Ba-

sically, I traverse the nodes of the network in post-order. For every node  $v$  I visit, the set of ACs  $\mathcal{AC}_v$  for node  $v$  will be constructed based on its type. Recall that there are four types of nodes in a phylogenetic network, which are leaves, reticulation nodes, internal tree nodes, and the root. Finally when the root of  $N$  is reached, we are able to obtain  $XL(N, g)$ .

It is important to note that although this algorithm is much more efficient than the MUL tree based one in simulation study (see Section 3.4.1.4.1), the running time of this algorithm is still exponential for some data sets, as the complexity of the problem is open and conjectured to be NP-hard.

### **Reducing the number of configurations**

At every reticulation node  $v$  in the species network, every configuration  $AC$  in  $\mathcal{AC}_v$  is split in all  $2^{n(AC)}$  possible ways. This may result in multiple configurations which contain the same set of gene lineages but are all distinct because of different index values (the second element of an AC) in some  $\mathcal{AC}$ . Since the running time (and memory usage) of the algorithms depends on the number of configurations, it is important to reduce the number of configurations so as to speed up the computation. Here, I make use of *articulation* nodes in the graph (an articulation node is a node whose removal disconnects the phylogenetic network). Obviously, the reticulation nodes inside the sub-network rooted at an articulation node are independent of the reticulation nodes outside the sub-network. So at articulation node  $v$  I reset the index vectors in all ACs in  $\mathcal{AC}_v$  to 0's so that all configurations at  $v$  containing the same set of gene lineages become identical. More precisely, when traversing the species network, after constructing  $\mathcal{AC}_v$  for some internal tree node  $v$  as described in Alg. 3, if  $v$  is an articulation node, the index vector is reset to 0's in every AC in  $\mathcal{AC}_v$ .

---

**Algorithm 3: CountXL.**


---

**Input:** Phylogenetic network  $N$  with  $q$  reticulation nodes, gene tree  $g$

**Output:**  $XL(N, g)$

**while** *traversing the nodes of  $N$  in post-order* **do**

**if** *node  $v$  is a leaf, who has parent  $u$*  **then**

$\mathcal{AC}_v \leftarrow \{(B, a, 0)\}$  where  $B$  is the set of leaves in  $g$  that are sampled from  
the species associated with  $v$  and  $a$  is a vector of  $q$  0's ;

$\mathcal{AC}_{(u,v)} \leftarrow \text{CreateCACsForXL}(g, (u, v), \mathcal{AC}_v)$ ;

**else if** *node  $v$  is a reticulation node, who has child  $w$ , and two parents  $u_1$  and  $u_2$*

**then**

$\mathcal{AC}_v \leftarrow \mathcal{AC}_{(v,w)}$ ;

**foreach**  $AC \in \mathcal{AC}_v$  **do**

        Split  $AC$  in every possible way into pairs of ACs, and for each pair, add  
        one AC to  $\mathcal{AC}_{(u_1,v)}$  and the other AC to  $\mathcal{AC}_{(u_2,v)}$ ;

**else if** *node  $v$  is an internal tree node, who has two children  $w_1$  and  $w_2$*  **then**

**foreach** *pair*  $(AC_1, AC_2)$  *of compatible ACs in*  $\mathcal{AC}_{(v,w_1)} \times \mathcal{AC}_{(v,w_2)}$  **do**

        Merge  $AC_1$  and  $AC_2$  and add the resulting AC to  $\mathcal{AC}_v$ ;

**if** *node  $v$  is an internal tree node, who has a parent  $u$*  **then**

$\mathcal{AC}_{(u,v)} \leftarrow \text{CreateCACsForXL}(g, (u, v), \mathcal{AC}_v)$ ;

**else**

**return**  $\min_{AC \in \mathcal{AC}_v} xl(AC)$ ;

---

Then  $\mathcal{AC}_v$  is updated to be  $\mathcal{AC}'_v$  as follows such that only the configuration containing the minimum weight is left:

$$\mathcal{AC}'_v = \{\operatorname{argmin}_{(B,a,w) \in \mathcal{AC}_v} w\} \quad (3.10)$$

where  $a$  is a zero vector.

### 3.1.3 Estimating inheritance probabilities

In this section, I describe how to estimate inheritance probabilities under MDC criterion [YBN13]. Given a collection  $\mathcal{G}$  of gene trees, once the optimal coalescent histories for all of them are computed within the branches of a phylogenetic network  $N$ , the inheritance probabilities associated with the reticulation nodes are estimated as follows. Let  $x$  be a reticulation node in  $N$ . Given the optimal coalescent histories computed, let  $l_x$  be the number of lineages that trace the left parent in all the coalescent histories, and let  $r_x$  be the number of lineages that trace the right parent in all the coalescent histories. Then, the probability associated with the left reticulation edge incident with  $x$  is  $l_x/(l_x + r_x)$  and the probability associated with the right reticulation edge incident with  $x$  is  $r_x/(l_x + r_x)$ .

Note that some gene tree may have multiple equally optimal coalescent histories, which implies that at some reticulation node  $x$  of  $N$  some gene lineages going left or right yields the same number of extra lineages. In this case, these gene lineages are considered to be informative when estimating inheritance probability of  $x$ , and hence they are ignored in the computation. For example, given phylogenetic network  $N$  in Fig. 3.1, for gene tree  $((a, b), (c, d))$ , according to its optimal coalescent history shown on left in Fig. 3.7, gene lineage went left and one gene lineage went right at reticulation node  $v$ , so for this gene

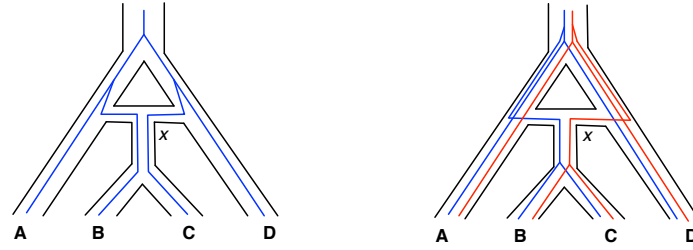


Figure 3.7: Illustration of estimating inheritance probabilities under MDC criterion. Given phylogenetic network in Fig. 3.1, (Left) the optimal coalescent history of gene tree  $((a, b), (c, d))$ , and (Right) the two equally optimal coalescent histories of gene tree  $((b, c), (a, d))$ .

tree itself, inheritance probability of both edge incident with  $x$  is 0.5. However, consider gene tree  $((a, d), (b, c))$  which has two equally optimal coalescent histories shown on right in Fig. 3.7. Clearly, at reticulation node  $x$  the ancestral allele of  $(b, c)$  went left in one optimal coalescent history (blue lines) and went right in the other (red lines), both of which yield the same number of extra lineages. In other word, both these two gene lineages are informative when estimating the probability of alleles going left (or right) at  $x$  under MDC criterion. Therefore, it will be considered as 0 gene lineages going left and 0 gene lineages going right at reticulation node  $x$  during the computation.

## 3.2 Handling gene tree uncertainty

When analyzing biological data set, gene tree topologies are estimated from sequence data, as such, there is uncertainty about them. In Bayesian inference, this uncertainty is reflected by a posterior distribution of gene tree topologies. In a parsimony analysis, several equally

optimal trees are computed. I propose here a way for incorporating this uncertainty into the framework above [YBN13].

Assume there are  $k$  loci under analysis, and for each locus  $i$ , a Bayesian analysis of the sequence alignment returns a collection of gene trees  $g_1^i, \dots, g_q^i$ , along with their associated posterior probabilities  $p_1^i, \dots, p_q^i$  ( $p_1^i + \dots + p_q^i = 1$ ). Now, let  $\mathcal{G}$  be the set of all distinct tree topologies computed on all  $k$  loci, and for each  $g \in \mathcal{G}$  let  $p_g$  be the sum of posterior probabilities associated with all gene trees computed over all loci whose topology is  $g$ . Thus,  $p_g = \sum_{i=1}^k p_g^i$  and  $\sum_{g \in \mathcal{G}} p_g = k$ . Then, Eq. 3.2 is replaced by

$$XL(N, \mathcal{G}) = \sum_{g \in \mathcal{G}} [XL(N, g) \times p_g] \quad (3.11)$$

Note that if  $p_j^i = 1$  or  $0$  for each  $i$  and  $j$ , then Eq. 3.11 is equivalent to Eq. 3.2. I additionally allow the  $p_j^i$  terms to be between  $0$  and  $1$  (and therefore  $p_g$  to be non-integer values) in order to reflect uncertainty in the estimated gene trees.

In the case where a maximum parsimony analysis is conducted to infer gene trees on the individual loci, a different treatment is necessary, since for each locus, all inferred trees are equally optimal. For locus  $i$ , let  $g$  be the strict consensus of all optimal gene tree topologies found. Then, Eq. 3.2 becomes

$$XL(N, \mathcal{G}) = \sum_{g \in \mathcal{G}} \min_{g' \in b(g)} XL(N, g') \quad (3.12)$$

where  $b(g)$  is the set of all binary refinements of gene tree topology  $g$ . Note that if  $g$  contains nodes of very high degrees, this approach is computationally infeasible if done in a brute-force fashion (explicitly considering all possible refinements). However, using the MUL-tree conversion technique, the efficient algorithms for [YWN11a, YWN11b] apply directly and achieve this computation in polynomial time, as opposed to the exponential

time of the brute-force approach. Also, AC-based algorithm is modified slightly to avoid explicitly considering all possible refinements of the gene tree,

### 3.3 Inferring a phylogenetic network

In this section, I describe that given a collection of gene trees  $\mathcal{G}$  how I search the network space to find the optimal phylogenetic network  $N^*$  such that  $N^* = \operatorname{argmin}_N XL(N, G)$ . Note that when I say the space of phylogenetic networks, only network topologies are considered. The materials in this section are from paper [YBN13, YDLN14].

#### 3.3.1 Neighborhood of a phylogenetic network

For a fixed number of taxa  $n$ , the space of phylogenetic networks consists of an infinite set of non-overlapping subspaces, each of which contains phylogenetic networks that have the same number of reticulation nodes. I denote each subspace by  $\Omega(n, k)$  where  $k$  is the number of reticulation nodes. From this definition, clearly  $\Omega(n, 0)$  is the tree space.

Given a phylogenetic network  $N \in \Omega(n, k)$ , I define four types of operations for network rearrangement as follows.

- Adding a reticulation edge ( $\delta_1$ ):
  1. Let  $(u_1, v_1)$  and  $(u_2, v_2)$  be two distinct edges in  $N$  such that  $v_2$  is not a predecessor of  $u_1$ .
  2. Delete the two edges  $(u_1, v_1)$  and  $(u_2, v_2)$ .

3. Add two new nodes  $x_1$  and  $x_2$  and five new edges  $(u_1, x_1)$ ,  $(x_1, v_1)$ ,  $(u_2, x_2)$ ,  $(x_2, v_2)$ , and  $(x_1, x_2)$  to network  $N$ .

- Removing a reticulation edge ( $\delta_2$ ):

1. Let  $(u, v)$  be an edge in  $N$  such that  $v$  is a reticulation node and  $u$  is a tree node.
2. Delete the two nodes  $u$  and  $v$  and the five edges  $(w, u)$ ,  $(u, z)$ ,  $(u, v)$ ,  $(x, v)$  and  $(v, y)$ , where  $w$  is the parent node of  $u$ ,  $z$  is the child node of  $u$  other than  $v$ ,  $x$  is the parent node of  $v$  other than  $u$ , and  $y$  is the child node of  $v$ .
3. Add two new edges  $(w, z)$  and  $(x, y)$  to network  $N$ .

- Relocating the destination of a reticulation edge ( $\delta_3$ ):

1. Let  $(u_1, v_1)$  and  $(u_2, v_2)$  be two distinct edges in  $N$  such that  $v_1$  is a reticulation node and  $v_2$  is not a predecessor of  $u_1$ .
2. Delete node  $v_1$  and the four edges  $(u_1, v_1)$ ,  $(u_2, v_2)$ ,  $(w, v_1)$ , and  $(v_1, z)$ , where  $w$  is the parent node of  $v_1$  other than  $u_1$  and  $z$  is the child node of  $v_1$ .
3. Add a new nodes  $x$  and four new edges  $(u_2, x)$ ,  $(x, v_2)$ ,  $(u_1, x)$ , and  $(w, z)$  to network  $N$ .

- Relocating the source of an edge ( $\delta_4$ ):

1. Let  $(u_1, v_1)$  and  $(u_2, v_2)$  be two distinct edges in  $N$  such that  $u_1$  is neither a reticulation node nor a predecessor of  $v_2$ .
2. Delete node  $u_1$  and the four edges  $(u_1, v_1)$ ,  $(u_2, v_2)$ ,  $(w, u_1)$ , and  $(u_1, z)$ , where  $w$  is the parent node of  $u_1$  and  $z$  is a child node of  $u_1$  other than  $v_1$ .



3. Add a new nodes  $x$  and four new edges  $(u_2, x)$ ,  $(x, v_2)$ ,  $(x, v_1)$ , and  $(w, z)$  to network  $N$ .

I denote the set of phylogenetic networks that can be obtained by applying operation  $\delta_i$  to  $N$  by  $\delta_i(N)$ , where  $1 \leq i \leq 4$ . Clearly, for a phylogenetic network  $N \in \Omega(n, k)$ , all networks in  $\delta_1(N)$  are in  $\Omega(n, k + 1)$ , all networks in  $\delta_2(N)$  are in  $\Omega(n, k - 1)$ , and all networks in  $\delta_3(N) \cup \delta_4(N)$  are in  $\Omega(n, k)$ . Finally, the neighborhood of a phylogenetic network  $N$ , denoted by  $\Delta(N)$ , can be defined based on these operations, depending on the searching strategies (see 3.3.2).

### 3.3.2 Search strategies

Due to the fact that the space of phylogenetic networks is very big, it is infeasible to enumerate and evaluate all possible phylogenetic networks during the search even when an upper bound of the number of reticulation nodes is given. Hence, I employ the hill climbing heuristic to search the network space in order to find the optimal phylogenetic network given a collection of gene trees. Hill-climbing is a commonly used mathematical optimization technique for local search. Here, I implemented simple hill climbing, as well as one of its variants steepest ascent hill climbing.

#### Simple hill climbing

Starting from some phylogenetic network  $N$ , I randomly pick a neighbor of  $N$ , say  $N'$ . If  $N'$  is a better candidate than  $N$  where  $XL(N', G) < XL(N, G)$ ,  $N$  is replaced by  $N'$ . Then the search continues. This process is repeated until the number of consecutive failure

reaches some preset value.

For simple hill climbing, the neighborhood of a phylogenetic network  $N$  is defined to be  $\bigcup_{1 \leq i \leq 4} \delta_i(N)$ , so a neighbor of a phylogenetic network  $N$  can be generated by applying one of the four types of operations of network rearrangement defined in the Section 3.3.1. Here, each of these four operations is associated with a weight. In order to propose a random neighbor of a network, the type of operation to be applied to generate the neighbor is first randomly selected according to their weights, and then the edges involved in that operation are randomly picked.

The simple hill climbing heuristic I am employing here does not guarantee to find global optimal solution. Due to random choices during the search, two different runs may take completely different paths and end up with different local optimums, even if these two runs start the search from the same starting network. So in order to avoid getting stuck at some local optimum, the whole process is performed multiple times and finally the phylogenetic network with the highest likelihood score from those runs is claimed to be the optimal solution. On the other hand, it is known that the choice of starting point of the search is very important, so I will discuss about it next. Generally speaking, the farther the starting phylogenetic network is from the global optimal one, the less likely the search is to discover the global optimal one, or the longer time it takes. So it is good to start the search from some reasonable species phylogenies, like binary resolutions of majority consensus species tree, or the optimal species tree under MDC criterion [Mad97, TN09, YWN11a, YWN11b]. However, given a collection of gene trees of  $n$  taxa, the optimal phylogenetic network in  $\Omega(n, k + 1)$  does not have to be in  $\delta_1(N_k^*)$  where  $N_k^*$  is the optimal phylogenetic network in  $\Omega(n, k)$ . So it is also important to start the search from some random points so that

more phylogenetic network space will be covered during the search, and hopefully by that I could avoid getting stuck at some local optimum. However, sometimes it may not be ideal to choose some totally random starting network due to the fact that the space of phylogenetic network is very big especially for a large number of taxa. So some random networks which are 2 or 3 operations (see Section 3.3.1) away from the one of best guess, like MDC tree, may be a good choice. To sum up, it is very important that the search will be performed from multiple starting points which are carefully chosen so that the method will have a higher chance to infer the global optimal solution.

### **Steepest ascent hill climbing**

Given a collection of gene trees  $\mathcal{G}$  with  $n$  taxa, in order to infer the optimal parsimonious phylogenetic network with at most  $m$  reticulation nodes, in addition to simple hill climbing, I proposed the steepest ascent, one of the variants of hill climbing, for searching the space of phylogenetic networks (See Alg. 4).

Basically, my strategy is to start search in space  $\Omega(n, k)$  from the starting network  $N$ , where  $k$  is the number of reticulation nodes in  $N$ , until some local optimum is reached in that space, then jump to  $\Omega(n, k + 1)$  and continue the search there. This process terminates when the local optimum in  $\Omega(n, m)$  is reached, or the locally optimal score in  $\Omega(n, i)$  is no better than that in  $\Omega(n, i + 1)$  where  $i < m$ .

More specifically, starting from some network  $N \in \Omega(n, k)$  where  $k \leq m$ , I first examine every network in  $\delta_3(N) \cup \delta_4(N)$  and find the one that results in the minimum number of extra lineages, say  $N^*$ , such that  $N^* = \operatorname{argmin}_{N' \in \delta_3(N) \cup \delta_4(N)} XL(N', G)$ . If  $XL(N^*, G) < XL(N, G)$ , which means  $N^*$  is a better network than the current one given

---

**Algorithm 4: SteepestAscentForNetworkSearch.**


---

**Input:** Gene trees  $\mathcal{G}$ , a starting phylogenetic network  $N$  and the maximum number of reticulation nodes  $m$

**Output:** A phylogenetic network  $N$

$continueSearch \leftarrow true$  ;

**while**  $continueSearch = true$  **do**

$N^* \leftarrow N$  ;

$flag \leftarrow true$  ;

**while**  $flag = true$  **do**

**foreach**  $N' \in \delta_3(N) \cup \delta_4(N)$  **do**

**if**  $XL(N', G) < XL(N^*, G)$  **then**

$N^* \leftarrow N'$  ;

**if**  $XL(N, G) > XL(N^*, G)$  **then**

$N \leftarrow N^*$  ;

**else**

$flag \leftarrow false$  ;

**if** the number of reticulation nodes in  $N$  is less than  $m$  **then**

**foreach**  $N' \in \delta_1(N)$  **do**

**if**  $XL(N', G) < XL(N^*, G)$  **then**

$N^* \leftarrow N'$  ;

**if**  $XL(N, G) > XL(N^*, G)$  **then**

$N \leftarrow N^*$  ;

**else**

$continueSearch \leftarrow false$  ;

**else**

$continueSearch \leftarrow false$  ;

**return**  $N$ ;

---

$\mathcal{G}$  under MDC criterion,  $N$  is replaced by  $N^*$  and the search is then continued in  $\Omega(n, k)$  from the new  $N$ ; otherwise, the search ends in  $\Omega(n, k)$  since the local optimum in that subspace is considered to be achieved. In the latter case, if  $k$  has reached the pre-specified upper bound of the number of reticulation nodes  $m$ , the whole search terminates and returns  $N$  as the final inferred phylogenetic network. Otherwise, we are now considering moving to space  $\Omega(n, k + 1)$ . More specifically, the current network  $N$  will be compared with network  $N^* = \operatorname{argmin}_{N' \in \delta_1(N)} XL(N', G)$ . If  $N^*$  is better,  $N$  will be replaced and the search will then continue in space  $\Omega(n, k + 1)$  from the new  $N$ ; otherwise,  $N$  is considered to be the final inferred network and the whole search terminates.

Similar to simple hill climbing, steepest ascent does not guarantee to find global optimal solution either. In my implementation, it is more deterministic than simple hill climbing, in the sense that given gene trees  $\mathcal{G}$ , a starting phylogenetic network  $N$  and the maximum number of reticulation nodes  $m$ , the method always returns the same network. This is because when examining all networks in  $\delta_i$ , the orders of those networks being enumerated are fixed. Therefore, different from simple hill climbing, for each starting point, the search only needs to be performed once. However, in order to avoid getting stuck at some local optimum, it is still important to start the search from multiple different starting points so that hopefully more phylogenetic network space will be covered during the search.

## 3.4 Performance

### 3.4.1 Simulation study

To study the performance of the criterion and the method in terms of the accuracy of the inferred phylogenetic networks, the accuracy of the inheritance probabilities they estimate and the efficiency on large dataset, I did intensive simulation studies [YBN13]. Also, I compared the efficiency between the MUL-tree based method and AC based method [YRN13].

#### 3.4.1.1 Evaluating the inference of phylogenetic networks

To evaluate the power of this parsimony approach at inferring phylogenetic networks along with inheritance probabilities, I considered four phylogenetic networks (Fig. 3.8) depicting evolutionary scenarios that present different challenges.

The phylogenetic network in Scenario **I** includes speciation after hybridization. Scenario **II** presents two independent hybridization events involving terminal taxa (leaves). Scenario **III** includes a hybrid species that further speciates, and then the two sister taxa hybridize again. Scenario **IV** includes two hybridization events the more recent of which involves a descendant and a descendant of a parent of the of the earlier hybrid. These different phylogenetic networks allow me to examine how combinations of speciation and hybridization affect the detectability of hybridization in particular, and the inference of phylogenetic networks in general. Further, I varied the inheritance probabilities associated with the hybridization events in the phylogenetic networks. For Scenario **I**, I considered  $\alpha \in \{0.0, 0.3, 0.5\}$ . For Scenario **II** and **III**, I considered  $(\alpha, \beta) \in \{(0.0, 0.5), (0.3, 0.3), (0.5, 0.5)\}$ . Since the hybridization events in Scenario **IV** are overlapping, I considered  $(\alpha, \beta) \in$

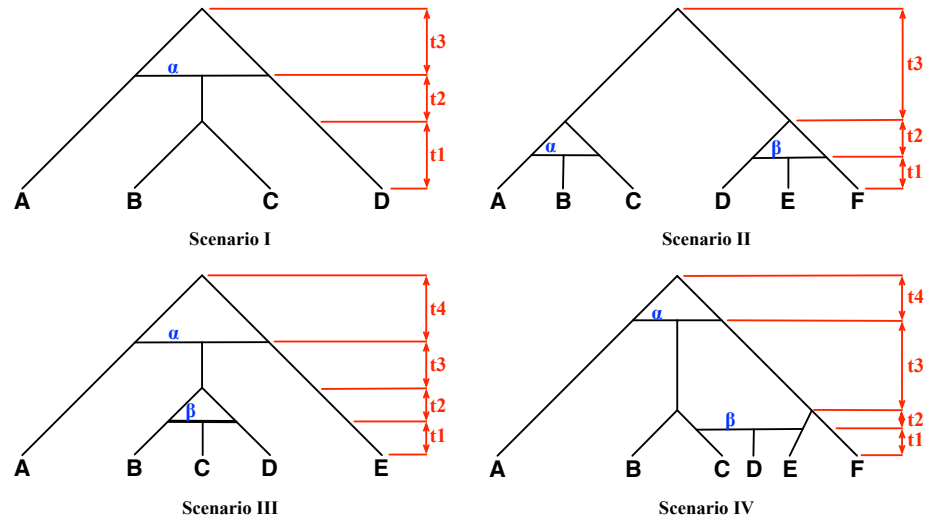


Figure 3.8: Phylogenetic networks depicting different hybridization/divergence/extinction scenarios. The  $\alpha$  and  $\beta$  parameters denote the proportions (or, probabilities) of alleles that are inherited from the “left” parents of the reticulation nodes ( $1 - \alpha$  and  $1 - \beta$  denote the proportions of the alleles that are inherited from the “right” parents of the nodes).

$\{(0.0, 0.5), (0.3, 0.3), (0.5, 0.0), (0.5, 0.5), (0.5, 1.0)\}$  in this case. The rationale for selecting the three values 0.0, 0.3, and 0.5 is that they represent no hybridization, “skewed” hybridization (different genetic contributions of the two parents to the hybrid), and perfect hybridization (equal genetic contributions of the two parents to the hybrid). Finally, to vary the extend of deep coalescence within each of the four evolutionary histories, I considered two settings for branch lengths (are measured in coalescent units): setting 1, in which  $t_1 = t_2 = t_3 = t_4 = 1.0$ , and setting 2, in which  $t_1 = t_2 = t_3 = t_4 = 2.0$ . All reticulation branches have length 0. As the extent of ILS increases as branches become shorter, I expect setting 1 to provide more challenging data for the method.

Using each combination of phylogenetic network, inheritance probabilities, and branch

length setting, I used the `ms` program [Hud02] to generate 10, 25, 50, 100, 500, 1000 and 2000 gene trees within the branches of the phylogenetic networks. To obtain statistically significant results, I generated 100 data sets per parameter setting and evaluated the performance as averaged over these 100 data sets, for each point in the parameter space. In these experiments, a single allele per species per gene was sampled.

Using the input sets of gene tree topologies, I inferred phylogenetic networks along with inheritance probabilities. In this experiments, I started from the optimal species tree under MDC by the exact method in [TN09] and searched the network space using steepest ascent described in Section 3.3.2. I assume knowledge of the true number of hybridization events and made inference with these (known) numbers of hybridization events. More specifically, for data sets corresponding to Scenario **I**, I inferred phylogenetic networks with one reticulation node, and for the other three scenarios, I inferred phylogenetic networks with two reticulation nodes. I discuss later the issues arising when I do not control for the number of hybridization events. I compared each inferred phylogenetic network against the (known) true phylogenetic network in terms of the topology and estimated inheritance probability. For comparing the topologies of two phylogenetic networks, I used the dissimilarity measure of [NWL04, TRN08] which computes the symmetric difference between the two sets of taxa clusters induced by the two networks. Results of the application of my methods to gene trees under Scenarios **I**, **II**, and **III** are given in Fig. 3.9.

In terms of the accuracy of the inferred phylogenetic network topology, we observe that as the number of gene trees used increases, the error in the estimated network decreases. For all three evolutionary scenarios, using about 50 gene trees under time setting 2 for branch lengths results in phylogenetic network inferences with 0 error. However, the per-



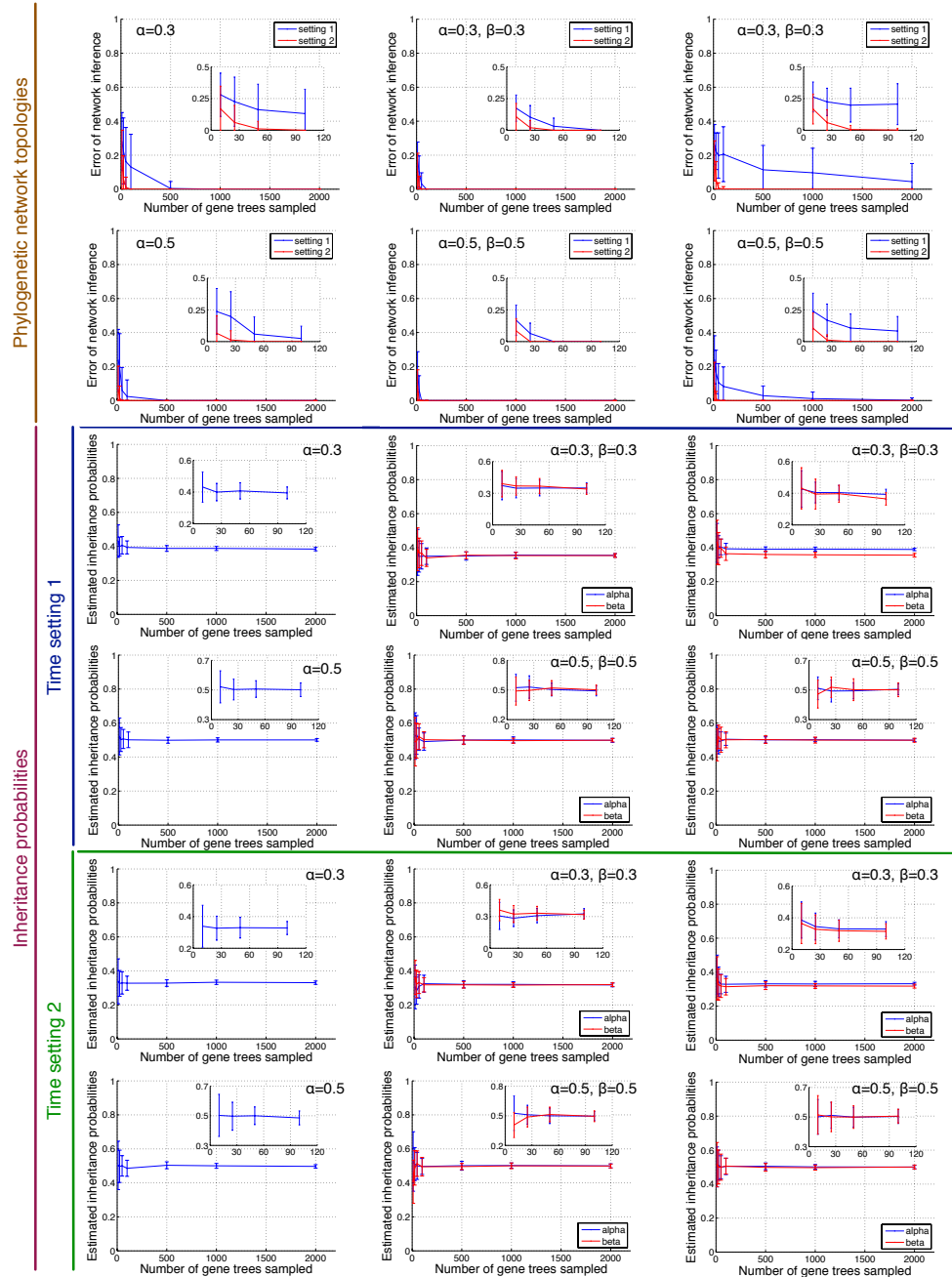


Figure 3.9: Accuracy of the inferred phylogenetic networks and inheritance probabilities.

The three columns from left to right correspond to Scenarios **I**, **II**, and **III** in Fig. 3.8, respectively. One allele per gene per species is sampled.

formance is different under time setting 1, which incorporates larger extents of incomplete lineage sorting. Here, we see that using about 50 gene trees results in correct network inference only under Scenario **II**, which is the least challenging for all scenarios considered. When we consider Scenario **I**, which adds to Scenario **II** the complexity of divergence after hybridization, we observe that the number of genes required to obtain accurate phylogenetic networks increases significantly (by an order of magnitude). For Scenario **III**, we observe that even with 2000 gene trees, the search heuristic fails to identify the true phylogenetic network. It is important to note here that we must distinguish between the performance of the optimality criterion and that of the search heuristic employed for inference. In this case, my search heuristic begins with a species tree that minimizes the number of extra lineages (or, deep coalescences) over all possible tree candidates, given the set of gene tree. Using this tree, the search proceeds in a hill descent fashion, each time exploring all neighboring topologies of the current optimal network, and continuing with the best found. An artifact of this search heuristic is that if the true network cannot be obtained from the starting tree in any possible way, then this search heuristic would not converge to the true network. Of course, this problem could be ameliorated by random restarts of the search heuristic or by exhaustively starting from all possible trees. While the former is also not guaranteed to result in convergence to the true network, the latter is prohibitive but for data sets with very small numbers of taxa, given the exponentially large size of the tree space. Nevertheless, we have inspected the cases pertaining to Scenario **III** and verified that the reason behind the lack of convergence to 0-error networks is the criterion: The number of extra lineages in the optimal network that the heuristic infers is *smaller* than that in the true network. This is not surprising, since parsimonious reconciliation and inference is known to have

consistency issues, even when ILS is the only event at play [TN09, TN10, TR11]. Finally, we observe that the performance is better for inheritance probabilities that are closer to 0.5. This is due to the fact that under these settings the contributions of the two parents to the genetic makeup of a hybrid species is more balanced, providing more phylogenetic signal for the method to infer the correct evolutionary history.

In terms of estimating the inheritance probabilities, the results show that my search heuristic makes very good estimates, regardless of the evolutionary scenario and branch length setting. Even though branch length setting 2 yields slightly more accurate estimates, which is expected, it is important to note that the method produces very good estimates even for the shorter branch lengths, where the extent of ILS is much larger. Further, it is worth emphasizing that these good estimates are obtained even with the smallest data sets (in terms of the number gene trees). This is a strength of the method.

#### **3.4.1.2 More loci or more alleles?**

Given the finite resources associated with any phylogenomic analysis, a natural question to ask is: In order to obtain more accurate inferences of phylogenetic networks and inheritance probabilities, should one sample more loci across the genomes or more alleles per locus? To explore this question, I used the above simulation procedure to generate gene trees under evolutionary Scenario **IV**, where 1, 2, 4 and 8 alleles per locus per species were sampled. The multi-allele gene trees were then used as input in the inference procedure. The results of this experiment are shown in Fig. 3.10.

Several observations are in order. First, in the case of this evolutionary scenario, the ability of the method to infer the correct topology of the phylogenetic network is not af-

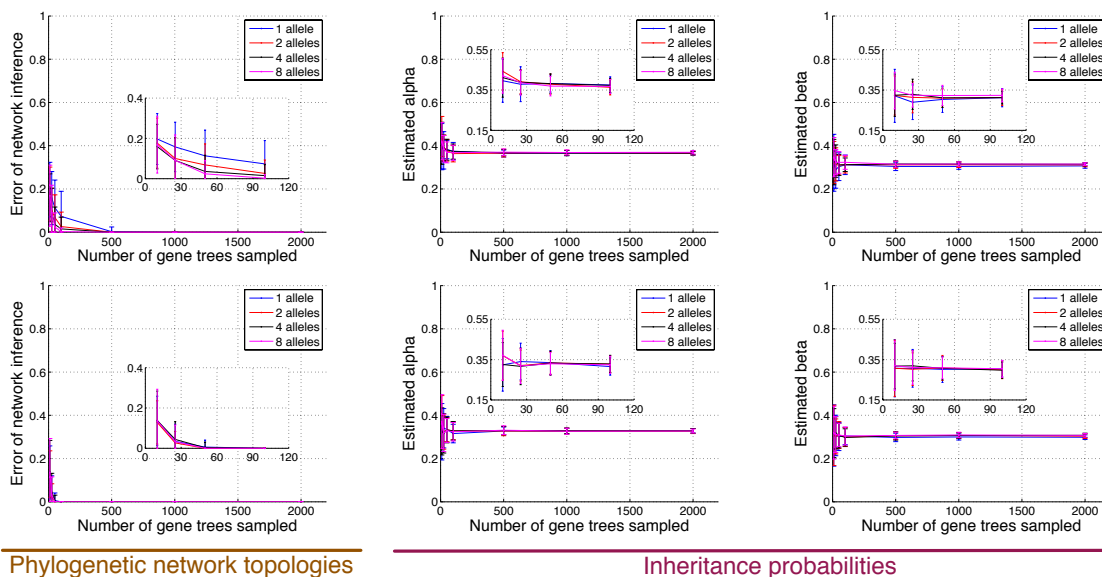


Figure 3.10: The effect of the number of alleles. Accuracy of the phylogenetic networks and inheritance probabilities estimated from gene trees simulated under Scenario **IV**, with true inheritance probabilities  $\alpha = \beta = 0.3$ , where the number of alleles sampled per species also varies. Top and bottom rows correspond to time settings 1 and 2, respectively.

affected much by the branch length settings, unlike the performance on the other three scenarios. However, in this case, the method always overestimates the inheritance probability (by about 5% hybridization), more so in the case of time setting 1. Second, in this case, the estimates of the probability  $\beta$  of the lower (closer to the leaves) hybridization are more accurate than that of the estimates of  $\alpha$ , which is unlike Scenarios **II** and **III**, where we did not observe any differences in the quality of the estimates of the two hybridization events. The reason for this is that in this scenario, some lineages, or alleles, from species D that trace different parents at the hybridization event undergo a further hybridization event, affecting the coalescence patterns towards the root. Regarding the benefit obtained by in-

creasing the number of alleles, none are observed in terms of the inheritance probability, and some are observed in terms of the phylogenetic network accuracy under time setting 1. That is, if the branches are very short, sampling two alleles, instead of one, improves the quality of the inferred network significantly. However, adding alleles beyond that does not seem to add more power, or signal, to the method. Under the other three scenarios, a single allele was already sufficient to provide highly accurate estimates. In summary, given the experimental settings I used here, there does not seem to be much benefit in sampling many alleles per species. Rather, sampling more loci per genome, particularly when the number of loci afforded is smaller than 100, provides more benefit. It is worth mentioning that the probabilistic method of [YDN12] yields very accurate estimates of the inheritance probabilities under this evolutionary scenario, even when a single allele is sampled per species (see supplementary material of [YDN12]).

### **3.4.1.3 Evaluating running time on large data sets**

To study the performance of my method in terms of efficiency, I did experiments on another sets of simulated data in which larger numbers of taxa are involved. I first generated 100 random species trees with 10, 20 and 40 taxa respectively using PhyloGen [Ram12]. In order to yield relatively similar amount of incongruence that might arise caused by ILS among the contained gene trees of the three sets of species trees, I adjusted the total heights of those species trees to 8, 16 and 32 (in coalescent units), respectively. From each species tree, I then generated random species networks with 1, 2, 3, 4 and 5 reticulation nodes respectively. When expanding a species network with  $n$  reticulation nodes to a species network with  $n + 1$  reticulation nodes, I randomly selected two existing edges in the species

network and connected their midpoints from the higher one to the lower one and then the lower one becomes a new reticulation node. For every reticulation node, I assigned random values from 0 to 1 as its inheritance probability. Finally, I simulated 25, 50, 100 and 200 gene trees respectively within the branches of each species network using the `ms` program [Hud02].

Using the input sets of gene tree topologies, I inferred phylogenetic networks assuming the knowledge of the true number of reticulation nodes. In this experiments, I started from the optimal species tree under MDC inferred by the heuristics in [TN09] and searched the network space using steepest ascent described in Section 3.3.2. When scoring a phylogenetic network, the AC-based method for computing the minimum number of extra lineages introduced in Section 3.1.2 was used. Through the combinations of various numbers of taxa and various numbers of reticulation nodes, I expect to see how the running time of our method is affected. The results are shown in Fig. 3.11. It is not surprising that the running time increases with the increase of the number of taxa and the number of reticulation nodes. But overall our method is able to finish the computations on all data sets in a reasonable amount of time. For the largest data set which has 40 taxa and 5 reticulation nodes, 75% of the computations are able to finish within 24 hours. We can see that there are many outliers which means that some data sets took much more time than others, especially for larger data sets. In fact, the running time of our method for computing the minimum number of extra lineages for a phylogenetic networks and a collection of gene trees using ancestral configurations is significantly effected by the topological features of the gene trees and phylogenetic network.

The accuracy of the inferred phylogenetic networks is given in Fig. 3.12. For a fixed

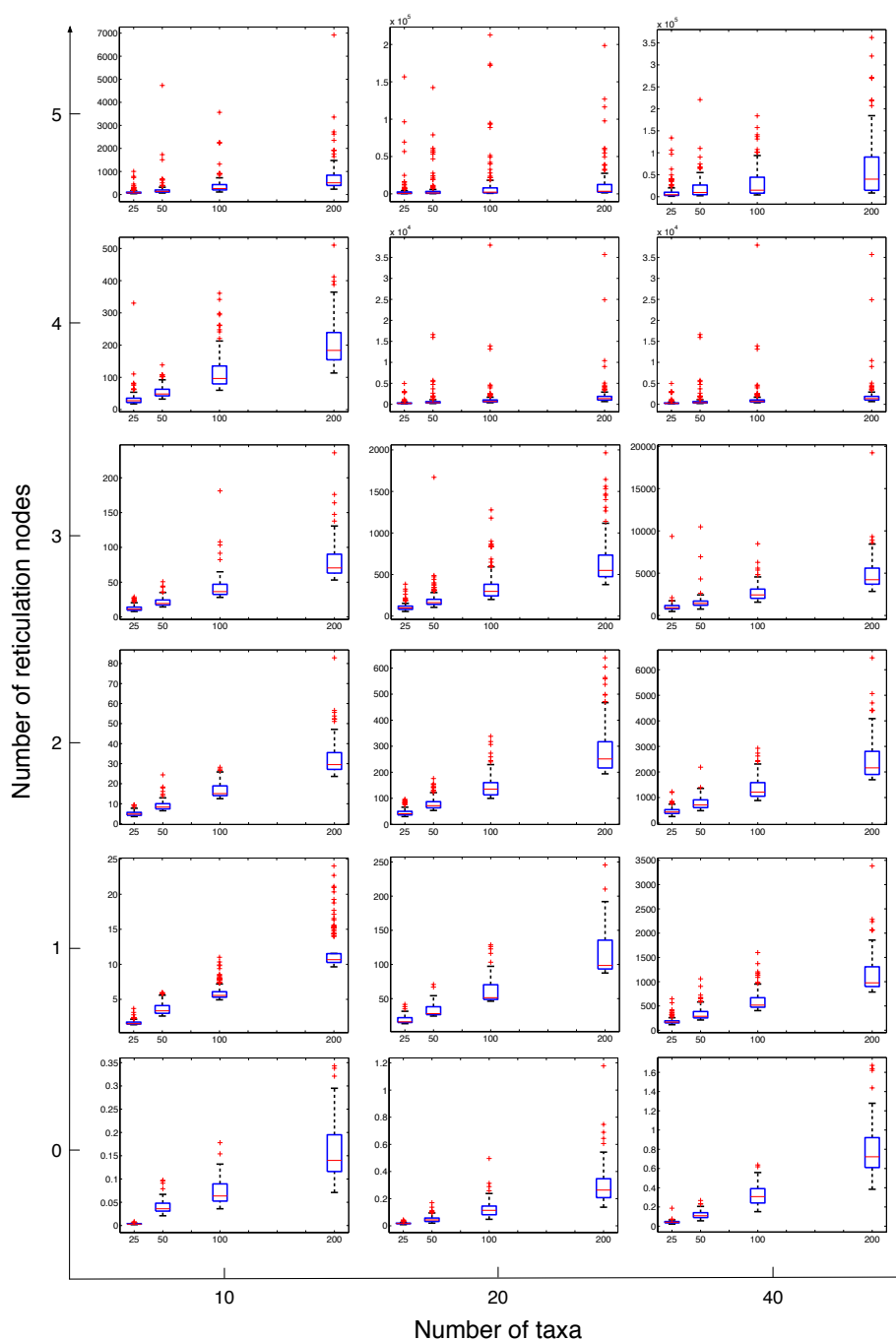


Figure 3.11: Running time of phylogenetic network inference. The three columns from left to right correspond to data sets with 10, 20 and 40 taxa, respectively. The six rows from bottom to top correspond to data sets with 0, 1, 2, 3, 4 and 5 reticulation nodes, respectively. In each sub-figure, the x-axis is the number of gene trees sampled and the y-axis is the running time in seconds.

number of taxa, the error of network inference increases with the number of reticulation nodes. It is expected because the addition of reticulation nodes increases the complexity of the phylogenetic networks. On the other hand, for a fixed number of reticulation nodes, the error of network inference decreases as the number of taxa increases. This happens because for a network with larger number of taxa, the randomly added reticulation nodes may have a higher chance to be independent of each other, which actually makes the inference easier.

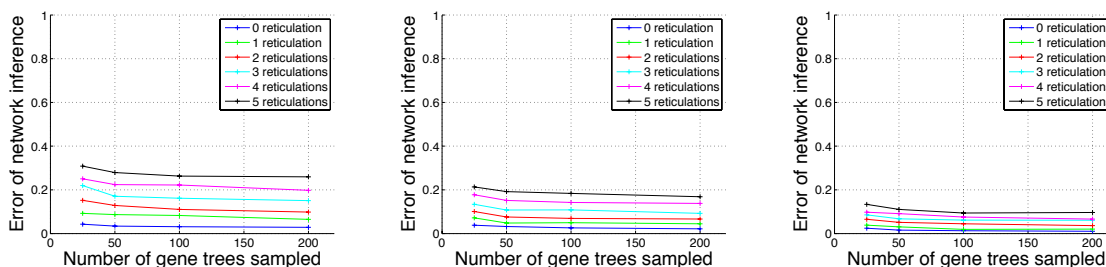


Figure 3.12: Accuracy of inferred phylogenetic networks. The three columns from left to right correspond to data sets with 10, 20 and 40 taxa, respectively.

### 3.4.1.4 Efficiency of the algorithms

#### 3.4.1.4.1 MUL-tree based method vs. AC based method

To study the efficiency of the AC based method compared to that of the MUL-tree based method, I ran both of them on synthetic data generated as follows. I first generated 100 random 24-taxon species trees using PhyloGen [Ram12], and from these I generated random species networks with 1, 2, 4, 6 and 8 reticulation nodes. When expanding a species network with  $n$  reticulation nodes to a species network with  $n + 1$  reticulation nodes, I



randomly selected two existing edges in the species network and connected their midpoints from the higher one to the lower one and then the lower one becomes a new reticulation node. Then, I simulated 10, 20, 50, 100, 200, 500 and 1000 gene trees respectively within the branches of each species network using the `ms` program [Hud02]. Since the MUL-tree method is computationally very intensive, when I run the simulation I bounded the time at 24 hours (that is, killed jobs that did not complete within 24 hours). All computations were run on a computer with a quad-core Intel Xeon, 2.83GHz CPU, and 4GB of RAM.

The results of the running time of both methods are shown in Fig. 3.13. Overall, both methods spent more time on data sets where the species networks contain more reticulation nodes. It is not surprising given the fact that adding more reticulation nodes increases the complexity of the networks in general. We can see that the speedup of the AC-based method over the MUL-tree based method also increased when the number of reticulation nodes in the species networks increased. In the best cases, the method achieves an improvement of about 5 orders of magnitude. In this figure, I only plot the results of the computations that could finish in 24 hours across all different number of loci sampled. In fact, the AC based method finished every computation in less than 3 minutes, even for the largest data set which contained species networks with 8 reticulations and 1000 gene trees. For the MUL-tree based method, out of 100 repetitions the numbers of repetitions that were able to finish in 24 hours across all different loci are 100, 100, 99, 96 and 88 for data sets containing species networks with 1, 2, 4, 6 and 8 reticulation nodes.

It is not surprising to see that for a fixed number of taxa the running time increases significantly when the number of reticulation nodes in the species networks increased. However, even for the same number of reticulation nodes, we can see that the running time still

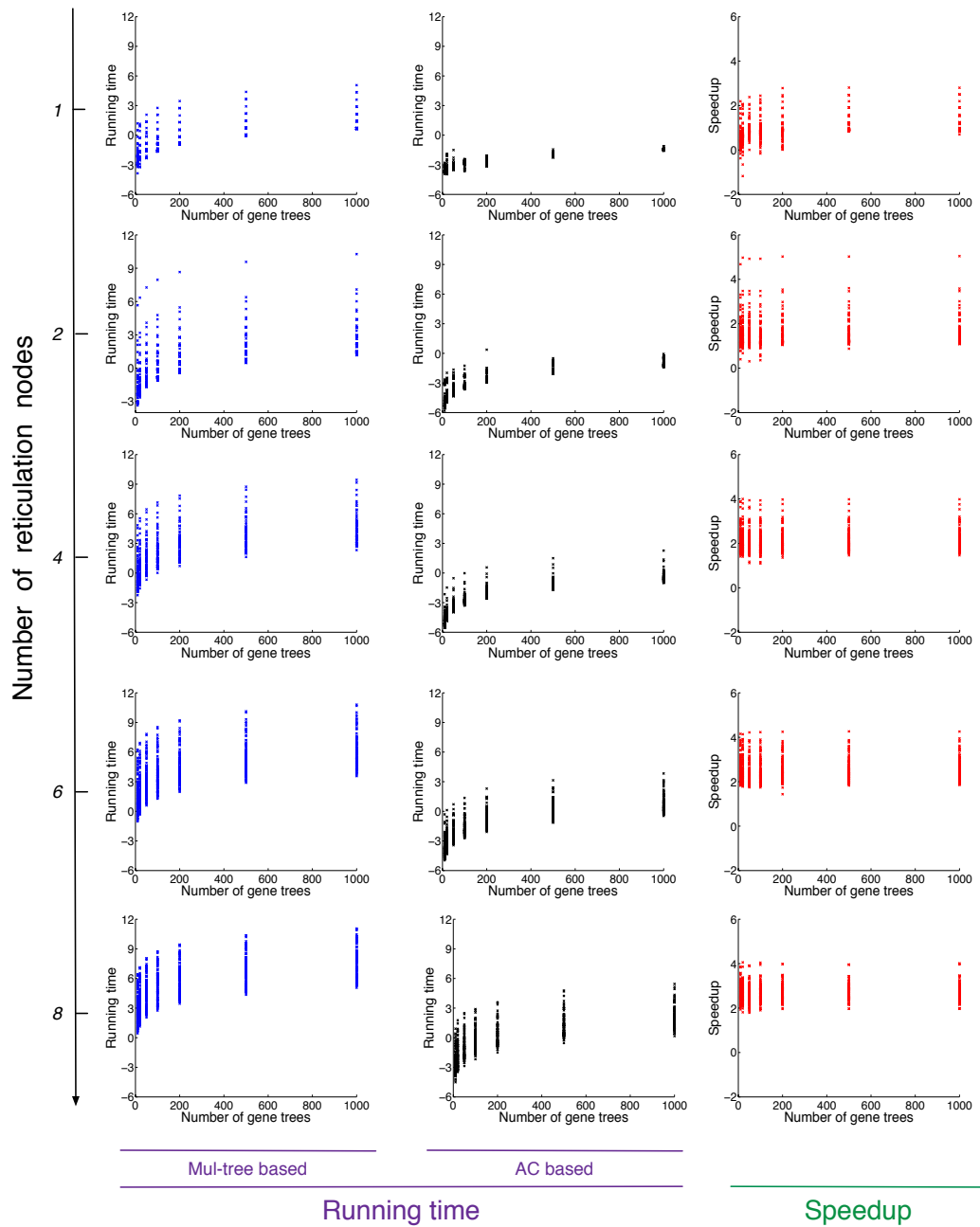


Figure 3.13: The running times (ln of number seconds) of the MUL-tree based ( $t(MUL)$ ), and AC-based ( $t(AC)$ ) methods for computing parsimonious reconciliations, as well as the speedup  $\log_{10}(t(MUL)/t(AC))$ .

differs significantly from case to case. For MUL tree based method, it depends on the number of allele mappings which is decided by the configurations of the reticulation nodes. For AC based method, I discuss in details about different factors that affect the running time of the algorithm in Section 3.4.1.4.2.

#### 3.4.1.4.2 Factors affecting the efficiency of the AC based method

There are several factors that can affect the number of configurations generated during the computation which directly dominates the running time of the algorithm. Two of the factors that affect performance are the number of leaves under a reticulation node, as well as the topology of the gene tree. I considered a “controlled” data set, where I controlled the placement of the reticulation node as well as the shapes of the gene trees. In particular, I considered three networks in Fig. 3.14, each with a single reticulation node, yet with 1, 8, and 15 leaves under the reticulation node, respectively. Further, I considered two gene trees:  $g_1$ , for which  $XL(N, g_1) = 0$ , whose topology is “contained” with each of the three networks, and  $g_2$ , whose disagreement with the three phylogenetic networks is very extensive that all coalescence events must occur above the root of the phylogenetic networks. I ran the AC based method on every pair of phylogenetic network and gene tree.

The results are listed in Table 3.3. In the case of  $g_1$ , for every articulation node  $v$  of the network,  $\mathcal{AC}_v$  has only one element  $AC$  and  $n(AC) = 1$ , resulting in short running for all three networks. However, for gene tree  $g_2$ , for every articulation node  $v$ ,  $\mathcal{AC}_v$  has only one element  $AC$  and  $n(AC)$  equals the number of leaves under  $v$ . Further, at a reticulation node, every configuration  $AC$  contributes  $2^{n(AC)}$  configurations to each of

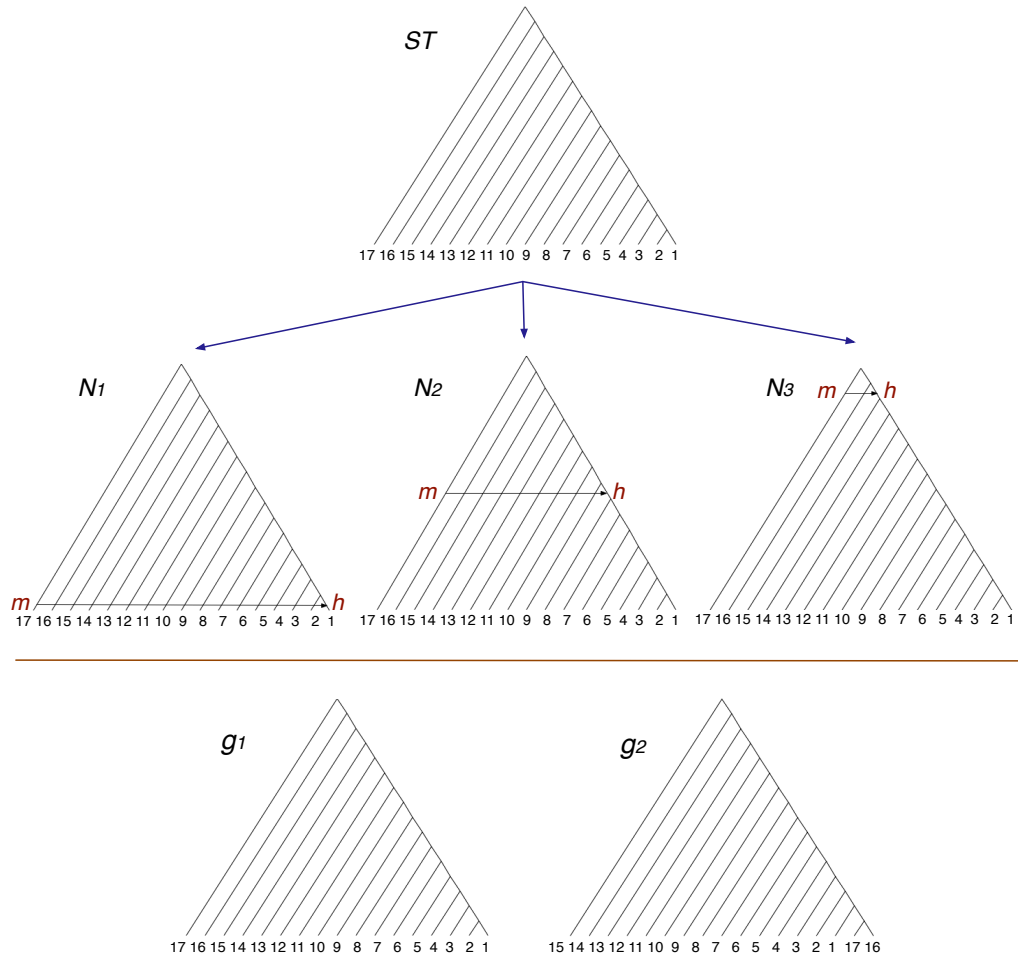


Figure 3.14: Synthetic data with controlled placements of the reticulation nodes. (Top) A species tree  $ST$ . (Middle)  $N_1$ ,  $N_2$  and  $N_3$  are three phylogenetic networks constructed by adding one reticulation edge to  $ST$  at three different locations. (Bottom) Two gene trees  $g_1$ , which is contained in all three networks, and  $g_2$ , whose coalescent events have to happen above the root of all three networks.

its parents. Therefore, the running time on  $g_2$  increased when the number of nodes under the reticulation nodes in the phylogenetic network increased. Furthermore, for  $g_2$ , I found that the number of allele mappings when using the MUL-tree based method is equal to the largest size of  $\mathcal{AC}_v$  generated during computation, unless the number of configurations

was reduced at articulation nodes. This is easy to see. For the AC based algorithm, if the number of configurations is not reduced at articulation nodes, the elements of  $\mathcal{AC}_R$ , where  $R$  is the root of the network, correspond to the exponential (in the number of reticulation nodes) number of paths that lineages could take. A similar situation arises for allele mappings. Nevertheless, since a typical gene tree involves coalescent events under the root, thus by decreasing the size of the set of ancestral configurations, the AC based algorithm improves upon the MUL-tree based algorithm in terms of efficiency. Comparing  $g_1$  and  $g_2$  we observe that for parsimony reconciliations, the more coalescent events that are allowed to occur under reticulation nodes with respect to the topology of the gene tree, the faster the AC based method is.

Table 3.3: The results of running the AC based algorithm for computing the minimum number of extra lineages given gene trees and species networks in Fig. 3.14.  $|\mathcal{AC}_h|$  is the number of configurations at the reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node that contains the largest  $AC_v$  in post-order of traversal is labeled by  $m$  in Fig. 3.14. Furthermore, the last column is the number of allele mappings if using the MUL-tree based method.

	$g_1$			$g_2$			#allele mappings
	$ \mathcal{AC}_h $	$max \mathcal{AC} $	running time (s)	$ \mathcal{AC}_h $	$max \mathcal{AC} $	running time (s)	
$N_1$	1	2	0.011	1	2	0.016	2
$N_2$	1	2	0.013	1	256 ( $2^8$ )	0.105	256
$N_3$	1	2	0.013	1	32768 ( $2^{15}$ )	32.551	32768

Another factor that affect the efficiency of the AC based method is dependency of the reticulation nodes. To address this issue, I considered another “controlled” data set in Fig.

3.15, which contains two networks, each with 7 reticulation nodes, yet one having them all independent and the other all dependent to each other. In this case, the numbers of nodes under all reticulation nodes are the same for both networks, so the difference in running time should come from the dependency of reticulation nodes. Similar to the previous data set, two gene trees are considered here. One is “contained” and the other is very “different” from the networks. The AC based method was run on every pair of phylogenetic network and gene tree.

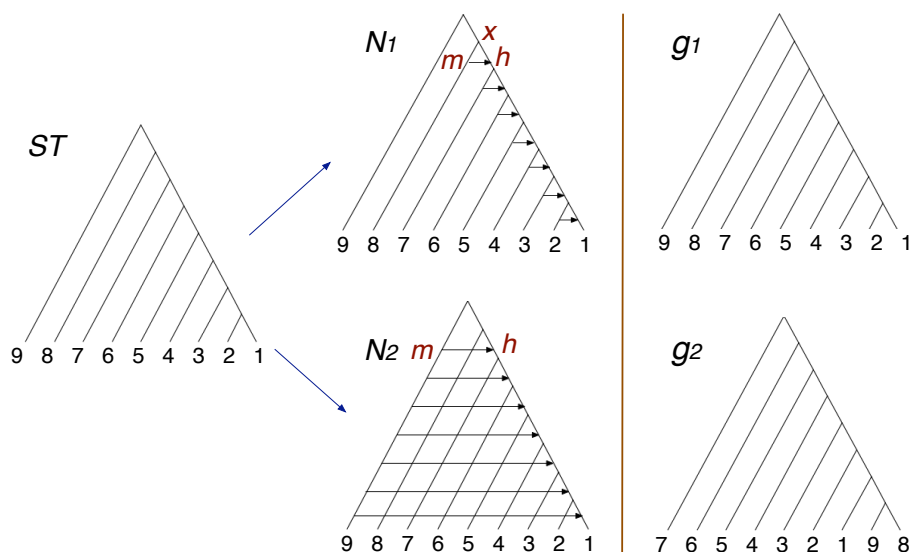


Figure 3.15: The effects of dependency of reticulation nodes in the species network and different gene tree topologies on the running time of the AC-based algorithms. (Left) A species tree  $ST$ . (Middle)  $N_1$  and  $N_2$  are two species networks constructed by adding seven reticulation edges to  $ST$  at different locations. (Right) two gene trees  $g_1$ , which is a contained tree of both  $N_1$  and  $N_2$ , and  $g_2$  whose coalescent events have to happen above the root of both species networks.

The results are listed in Table 3.4. It is not surprising to see that the running time

for  $N_1$  is much less than that for  $N_2$  since the number of configurations is reduced at articulation nodes in  $N_1$  while  $N_2$  does not have any articulation nodes except for the root. Again, the sub-network rooted at an articulation node is independent from other part of the network, so the  $\mathcal{AC}$  set for that articulation node only needs to contain a “summary” of the network. For counting the minimum number of extra lineages, it means the  $\mathcal{AC}$  set of an articulation node only needs to carry one element that has the minimum number of extra lineages. Now let us consider the highest reticulation node  $h$  in  $N_1$ . For  $g_1$ , since all lineages under node  $h$  have coalesced into one lineage,  $n(AC_h) = 1$  and it gave rise to 2 ACs to its parent node  $m$ . For  $g_2$ , no coalescent events could happen below the root, so  $n(AC_h) = 7$  which represented the 7 leaves under it and it gave rise to  $2^7$  ACs to its parent node  $m$ . Even though  $|\mathcal{AC}_m| = 128$  for  $g_2$ , after merging with the configurations coming from branch  $(x, h)$  at articulation node  $x$ , again, only the one with the minimum number of extra lineages will be kept. For both  $g_1$  and  $g_2$ , their  $\max|\mathcal{AC}|$  imply a big speedup of the AC based method comparing to the number of allele mappings 268435456 for the MUL-tree based method. In fact, species network  $N_1$  indicates another reason why the AC based algorithm is faster, which is because the existences of reticulation nodes under an articulation node  $v$  are transparent to the reticulation nodes outside the sub-network rooted at  $v$ . In contrast, when building the set of allele mappings, all possible combinations of the ways that every leaf lineage goes at every reticulation node need to be considered. On the other hand, for network  $N_2$ , there is no articulation node, so all configurations being split at every reticulation node merge back at the root. So, for  $g_2$  whose coalescent events can only occur above the root of  $N_2$ , the largest size of  $\mathcal{AC}_v$  generated during computation is equal to the number of allele mappings in the MUL-tree based method.

Table 3.4: The results of running the AC based algorithm for computing the minimum number of extra lineages given gene trees and species networks in Fig. 3.15.  $|\mathcal{AC}_h|$  is the number of configurations at the highest reticulation node  $h$  and  $\max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node that contains the largest  $AC_v$  set in post-order of traversal is labeled by  $m$  in Fig. 3.15. Furthermore, the last column is the number of allele mappings if using the MUL-tree based method.

	$g_1$			$g_2$			#allele mappings
	$ \mathcal{AC}_h $	$\max \mathcal{AC} $	running time (s)	$ \mathcal{AC}_h $	$\max \mathcal{AC} $	running time (s)	
$N_1$	1	2	0.014	1	128 ( $2^7$ )	0.039	268435456
$N_2$	874	5914	2.85	5040	40320	55.684	40320

To sum up, for the data sets of the same size (e.g., number of taxa and reticulation nodes), the running time of the AC based algorithm for computing the minimum number of extra lineages increases when there are more leaves under reticulation nodes and when the reticulation nodes are more dependent on each other. With respect to the topology of the gene tree and the species network, the more coalescent events that are allowed under reticulation nodes the faster the method is. For most cases, the AC based method is significantly much faster than the MUL-tree based one. The gain in terms of efficiency comes from avoiding allele mappings that are guaranteed to result in suboptimal reconciliations or correspond to configurations being removed at articulation nodes.



### 3.4.2 Reanalysis of a yeast (*Saccharomyces*) data set

I reanalyzed the yeast data set of [RWKC03a] using this parsimony approach [YBN13]. This data set consists of 106 loci, each present in exactly a single copy in each of seven *Saccharomyces* species, *S. cerevisiae* (*Scer*), *S. paradoxus* (*Spar*), *S. mikatae* (*Smik*), *S. kudriavzevii* (*Skud*), *S. bayanus* (*Sbay*), *S. castellii* (*Scas*), *S. kluyveri* (*Sklu*), and the out-group fungus *Candida albicans* (*Calb*). I reconstructed gene trees from sequence data using maximum parsimony with strict consensus in PAUP\* [Swo96] and Bayesian inference in MrBayes [HR01]. In each of 106 gene trees, the genes from the five species *Scer*, *Spar*, *Smik*, *Skud* and *Sbay* formed a monophyletic group. From a parsimony perspective, all coalescent events involving genes from these five species occur at or below their most recent common ancestor. Therefore, in the analysis, I only focused on the evolutionary history of these five species.

It is important to note that the gene trees used in the analysis here are not all binary. In the case where the gene trees were inferred by maximum parsimony, I used the strict consensus of all optimal trees found for each gene, which resulted in non-binary trees. In the case of Bayesian inference, I used each gene tree with its posterior probability. See Section 3.2 for how I accounted uncertainty in gene trees using these two approaches.

Using our method, I inferred the optimal species networks containing 0, 1 and 2 reticulation nodes. The resulting species networks inferred from gene trees reconstructed by maximum parsimony are shown in Fig. 3.16 along with inheritance probabilities and total number of extra lineages. The optimal species tree in Fig. 3.16A has been reported by several studies [ELP07b, RWKC03a, TN09]. The optimal species network containing one

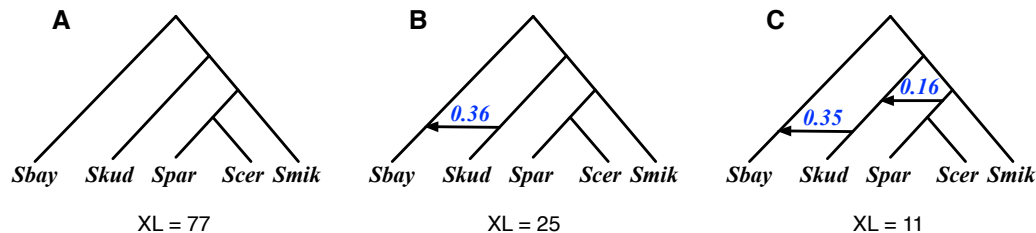


Figure 3.16: Analysis of the yeast data set, where gene trees are reconstructed using MP. Optimal species phylogenies, along with inheritance probabilities, inferred from gene trees reconstructed by maximum parsimony for the yeast data set of [RWKC03a]. (A) The optimal species tree (network with 0 reticulation nodes). (B) The optimal species network containing one reticulation node. (C) The optimal species network containing two reticulation nodes. For each species phylogeny, the total number of extra lineages (XL) is computed using Eq. (3.12) and reported.

reticulation node in Fig. 3.16B has also been proposed as an alternative evolutionary history under the stochastic framework of [BS10], the parsimony framework of [TN09] and the likelihood framework of [YDN12]. It is worth mentioning that the inheritance probability inferred by our method is almost the same as that inferred by the probabilistic approach of [YDN12]. The optimal species network with two reticulation nodes in Fig. 3.16C was not reported in any of the aforementioned studies.

For gene trees reconstructed using MrBayes, the inferred species networks are shown in Fig. 3.17. The optimal species tree in Fig. 3.17A has been reported as a very close candidate [ELP07b, TN09]. The optimal species network containing one reticulation node in Fig. 3.17B has the same topology as the one inferred from gene trees reconstructed by maximum parsimony in Fig. 3.16B, but with a slightly higher inheritance probability.

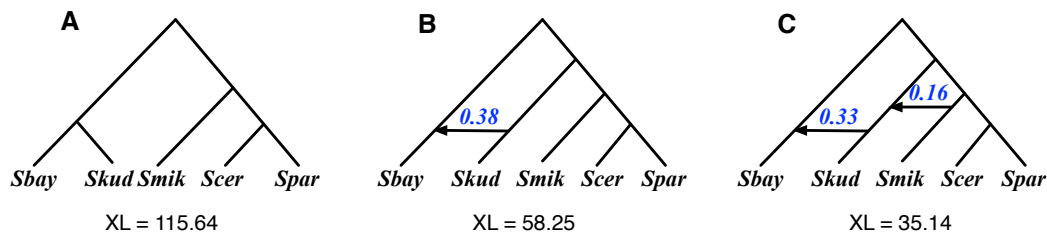


Figure 3.17: Analysis of the yeast data set, where gene trees are reconstructed using Bayesian inference. Optimal species phylogenies, along with inheritance probabilities, inferred from gene trees reconstructed by MrBayes for the yeast data set of [RWKC03a]. (A) The optimal species tree (network with 0 reticulation nodes). (B) The optimal species network containing one reticulation node. (C) The optimal species network containing two reticulation nodes. For each species phylogeny, the total number of extra lineages (XL) is computed using Eq. 3.11 and reported.

### 3.4.3 The model selection problem

A major confounding issue that arises when inferring phylogenetic network topologies is that of determining the correct number of reticulation events [Nak10]. As we observed in the yeast data set analysis, adding a single reticulation node to the optimal species tree reduces the number of extra lineages by about 70%. Further, adding an additional reticulation node to the optimal species network with a single reticulation node reduces the number of extra lineages by about a half. This is the classical model selection problem arising in the domain of phylogenetic networks: Increasing the complexity of the phylogenetic network topology by adding more reticulation nodes to it mostly improves the fit of the data. Simply minimizing the sum of the number of hybridization events and deep coalescence events does not solve the problem. Further, minimizing a weighted sum of these two numbers

raises the questions of how to weight them and whether weights are data-dependent or not.

As I pointed out above, when analyzing the simulated data, I assumed knowledge of the true number of reticulation nodes. To understand the performance of the method when this assumption is removed, I inferred phylogenetic networks with up to 4 reticulation nodes from the data I generated in Section 3.4.1.1, and explored the number of extra lineages in these inferred networks as a function of the number of reticulation nodes. The results for Scenario **III** are shown in Fig. 3.18; similar results were observed under the other scenarios.

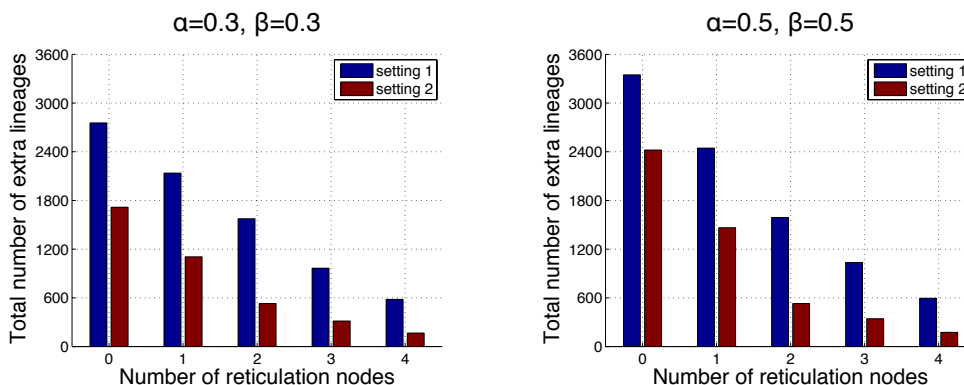


Figure 3.18: Network complexity and the number of extra lineages. The decrease in the number of extra lineages in the inferred phylogenetic network as a function of the increase in number of hybridization events inferred. The results were obtained from data pertaining to Scenario **III** in Section 3.4.1.1 under two different settings of the inheritance probabilities and two different settings of the branch lengths.

As the figure shows, the number of extra lineages of the optimal species networks keeps decreasing as more reticulation nodes are added. Thus, using the minimization of the number of extra lineages as the optimality criterion, without penalizing complexity, may

result in gross overestimation of the amount of reticulation in the data. Inspecting the data in Fig. 3.18 closely, we observe that for branch length setting 2, the decrease in the number of extra lineages is similar when going from 0 to 1 reticulation nodes and from 1 to 2 reticulation nodes, and then that decrease becomes slower. Thus, the rate of decrease in extra lineages might be helpful in determining a stopping criterion. However, the situation is more challenging under branch length setting 1, where the extent of ILS is very large, and adding reticulation nodes, though are clearly false positives, help decrease the number of extra lineages at a similar rate to that of the true reticulation nodes. In other words, using the rate of decrease in extra lineages must also account for branch length information.

As discussed above, adding an arbitrary reticulation event to a phylogenetic network might improve the number of extra lineages. A question that arises here is: Does the improvement in the number of extra lineages resulting from adding the true reticulation event differ from that of adding a different, arbitrary reticulation event? To explore this question empirically, I plotted the distributions of the number of extra lineages of all networks in the neighborhood of the optimal species tree and the optimal species network with one reticulation event of the yeast data set. The results are shown in Fig. 3.19.

The results show an interesting trend. In the case of the neighbors of the optimal species tree, the number of extra lineages in the optimal network is very different from the numbers of the other neighbors: it falls in a bin by itself, and the next larger bin corresponds to networks with more than 40 extra lineages. On the other hand, when considering all neighbors of the optimal network with a single reticulation node, the optimal network with two reticulation nodes was not a clear candidate: several other networks fall within the same bin, and over 20 other networks are within 10 extra lineages from the optimal one. These

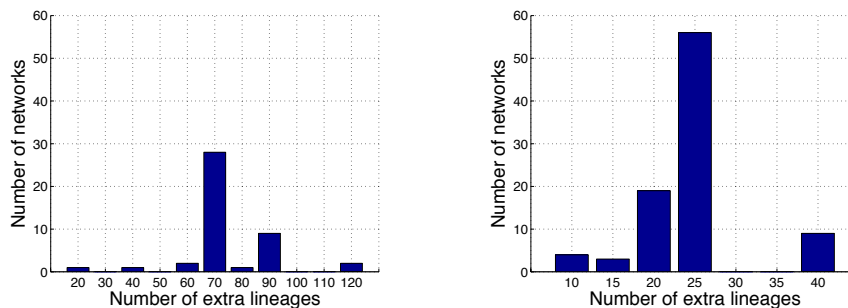


Figure 3.19: Distribution of the number of extra lineages in the neighborhood of an optimal species network. (Left) The distribution of the number of extra lineages of all networks formed from the species tree in Fig. 3.16A by adding a single reticulation edge in all possible ways. (Right) The distribution of the number of extra lineages of all networks formed from the species network in Fig. 3.16B by adding a single reticulation edge in all possible ways. All results are based on the gene trees reconstructed using maximum parsimony, and binned in ranges of size 10 (left) and size 5 (right).

results combined can be taken as strong support for the optimal network in Fig. 3.16B and as a poor support for the optimal network in Fig. 3.16C. In other words, looking at the data from this perspective, the analysis supports a single reticulation event in the yeast data set.

Last but not least, I set out to quantify the support for the two networks in Fig. 3.16 in terms of their likelihoods. To achieve this, I scored the likelihood of each network (while using the reported inheritance probabilities in the figure, but estimating branch lengths) given the data (the 106 gene tree topologies reconstructed using maximum parsimony) by the probabilistic method described in Section 4.1.1. The results, in terms of the log likelihoods and values of AIC [Aka74], AICc [BA02], and BIC [Sch78] information criteria are summarized in Table 3.5.

Table 3.5: Log likelihoods and values of three information criteria computed for the three species phylogeny candidates in Fig. 3.16.

	-log likelihood	AIC	AICc	BIC
Phylogeny of Fig. 3.16A	205	416	417	424
Phylogeny of Fig. 3.16B	157	325	326	338
Phylogeny of Fig. 3.16C	152	319	320	338

Clearly, the network with a single reticulation node results in significantly improved likelihood and values of the information criteria, whereas the improvement in these values as the second reticulation node is added is negligible. In fact, based on the BIC value, the second reticulation event does not make any difference in the fit of the data to the model. This analysis gives further support the evolutionary history in Fig. 3.16B over the other two. Moreover, this analysis illustrates how to combine the speed of a parsimony analysis with the accuracy of a probabilistic analysis to obtain a solid evolutionary history.

The materials in this section are from [YBN13].

## Chapter 4

# Probabilistic inference of phylogenetic networks

In this chapter, I propose methods for inferring phylogenetic network from a collection of gene trees under maximum likelihood. More specifically, given a phylogenetic network  $N$  and a collection of gene trees  $\mathcal{G}$ , the goal is to infer the optimal phylogenetic network  $N_{\lambda, \gamma}^*$  such that

$$N_{\lambda, \gamma}^* = \operatorname{argmax}_{N_{\lambda, \gamma}} P(\mathcal{G} | N_{\lambda, \gamma}) \quad (4.1)$$

where  $P(\mathcal{G} | N_{\lambda, \gamma})$  is the probability of observing gene trees  $\mathcal{G}$  given phylogenetic network  $N_{\lambda, \gamma}$ , which equals

$$P(\mathcal{G} | N_{\lambda, \gamma}) = \prod_{g \in \mathcal{G}} P(g | N_{\lambda, \gamma}) \quad (4.2)$$

where  $P(g | N_{\lambda, \gamma})$  denotes the probability of observing  $g$  given  $N_{\lambda, \gamma}$ . Note that we need to distinguish between cases where whether the branch lengths of gene trees are used or not.



## 4.1 Computing the probability of a gene tree given a phylogenetic network

### 4.1.1 Using only topologies of the gene trees

Given a phylogenetic network  $N_{\lambda,\gamma}$ , the probability of observing a gene tree topology  $g$  can be computed by

$$P(g|N_{\lambda,\gamma}) = \sum_{h \in H_{N_{\lambda,\gamma}}(g)} P(h|N_{\lambda,\gamma}), \quad (4.3)$$

where  $P(h|N_{\lambda,\gamma})$  is the probability of observing coalescent history  $h$  given phylogenetic network  $N_{\lambda,\gamma}$  which can be computed by

$$P(h|N_{\lambda,\gamma}) = \frac{w(h)}{d(h)} \prod_{b \in E(N_{\lambda,\gamma})} \frac{w_b(h)}{d_b(h)} \gamma_b^{u_b(h)} p_{u_b(h)v_b(h)}(\lambda_b). \quad (4.4)$$

In this equation,  $u_b(h)$  and  $v_b(h)$  denote the number of lineages enter and exit edge  $b$  of  $N_{\lambda,\gamma}$  under coalescent history  $h$ .  $p_{u_b(h)v_b(h)}(\lambda_b)$  is the probability of  $u_b(h)$  gene lineages coalescing into  $v_b(h)$  during time  $\lambda_b$  [Tav84]. And  $w_b(h)/d_b(h)$  is the proportion of all coalescent scenarios resulting from  $u_b(h) - v_b(h)$  coalescent events that agree with the topology of the gene tree [DS05b]. This quantity without the  $b$  subscript corresponds to the root of  $N$ . In Table 4.1, an example of how Eq. 4.4 is computed is given for the phylogenetic network and gene tree in Fig. 2.4.

In this section, I propose two methods for computing  $P(g|N_{\lambda,\gamma})$  the probability of observing a gene tree topology  $g$  given a phylogenetic network  $N_{\lambda,\gamma}$ . One of the two methods is based on the concept of multilabeled (MUL) tree [YDN12], and the other is based on the concept of weighted ancestral configurations [YRN13].

Table 4.1: The probabilities of all coalescent histories in Fig. 2.4. For every coalescent history  $h$ , columns from 2 to 7 list the probability of having  $h$  on every branch of the species network  $N$ , where  $t_i$  is the branch length of branch  $i$ . Branch 6 corresponds to the branch incident into the root of the species network  $N$ . A dash means no gene lineages enter that branch. Therefore, the total probability of a coalescent history is the product taken over all branches of the species network. In Fig. 2.4, coalescent events  $y$  and  $z$  can only happen above the root of  $N$ . For every coalescent history, the highlight cell shows where coalescent event  $x$  happens.

	$P(h N)$ on each branch					
	1	2	3	4	5	6
$h_1$	$g_{21}(t_1)$	$\gamma$	—	$g_{22}(t_4)$	—	$1/3$
$h_2$	$g_{21}(t_1)$	—	$1 - \gamma$	—	$g_{22}(t_5)$	$1/3$
$h_3$	$g_{22}(t_1)$	$\gamma^2 g_{21}(t_2)$	—	$g_{22}(t_4)$	—	$1/3$
$h_4$	$g_{22}(t_1)$	—	$(1 - \gamma)^2 g_{21}(t_3)$	—	$g_{22}(t_5)$	$1/3$
$h_5$	$g_{22}(t_1)$	$\gamma^2 g_{22}(t_2)$	—	$(1/3)g_{32}(t_4)$	—	$1/3$
$h_6$	$g_{22}(t_1)$	—	$(1 - \gamma)^2 g_{22}(t_3)$	—	$(1/3)g_{32}(t_5)$	$1/3$
$h_7$	$g_{22}(t_1)$	$\gamma^2 g_{22}(t_2)$	—	$g_{33}(t_4)$	—	$1/9$
$h_8$	$g_{22}(t_1)$	—	$(1 - \gamma)^2 g_{22}(t_3)$	—	$g_{33}(t_5)$	$1/9$
$h_9$	$g_{22}(t_1)$	$\gamma$	$1 - \gamma$	$g_{22}(t_4)$	$g_{22}(t_5)$	$1/9$
$h_{10}$	$g_{22}(t_1)$	$\gamma$	$1 - \gamma$	$g_{22}(t_4)$	$g_{22}(t_5)$	$1/9$

#### 4.1.1.1 An algorithm based on MUL trees

In this Section 3.1.1, I proposed an algorithm for computing the minimum number of extra lineages required to reconcile a gene tree within the branches of a phylogenetic network based on the concept of MUL tree. The insight of this method is that converting a phylogenetic network into a tree, in certain extent, enable us to make use of existing methods that work on tree topologies. This technique can also be used for computing the probability of a gene tree topology given a phylogenetic network.

The algorithm consists of three steps. The first two are exactly the same as the ones used in this Section 3.1.1, which are converting the phylogenetic network into MUL tree, and creating allele mappings. Once the phylogenetic network  $N_{\lambda, \gamma}$  is converted into MUL tree  $T_{\lambda', \gamma'}$  and the set of all allele mappings is produced (a straightforward computational task, yet results in a number of allele mappings that is exponential in a combination of the number of alleles sampled and the number of reticulation nodes), the probability of observing gene tree topology  $g$  is found by summing the probability of  $g$  given the MUL tree over all possible allele mappings. Then, the probability of observing gene tree topology  $g$  is found by summing over all possible allele mappings:

$$P(g|N_{\lambda, \gamma}) = \sum_{f \in \mathcal{F}_{T, g}} P(g|T_{\lambda', \gamma'}, f). \quad (4.5)$$

In this equation, the  $P(g|T_{\lambda', \gamma'}, f)$  term accounts for all coalescent histories of a given mapping, which, when combined with the summation over all allele mappings, accounts for all coalescent histories within the branches of a phylogenetic network. This formulation naturally gives rise to a likelihood setup for estimating the parameters of a reticulate evolutionary history from a collection of gene trees described by their topologies.

To complete the framework, I now provide a formula for  $P(g|T_{\lambda',\gamma'}, f)$ , which is the probability of a gene tree given a MUL tree and an allele mapping. Similar to the issue raised in Fig. 3.5, special attention needs to be paid to sets of branches in the MUL tree that correspond to single branches in the phylogenetic network, since coalescence events within these branches are not independent. Additionally, each branch in the MUL tree may have an inheritance probability associated with it that is neither 0 nor 1, and must be accounted for in computing the probabilities. Accounting for these two cases gives rise to

$$P(g|T_{\lambda',\gamma'}, f) = \sum_{h \in H_{T,f}(g)} \frac{w(h)}{d(h)} \prod_{b=1}^{n-2} \gamma_b'^{v_b(h)} P_b'(h), \quad (4.6)$$

where the  $P_b'(h)$  terms are symbolic quantities, that do not individually evaluate to any value. Instead, they play a role in simultaneously computing the probability along pairs of branches in the MUL tree that share a single source branch in the phylogenetic network. More formally, let  $b' = (u, v)$  be a branch in  $N_{\lambda,\gamma}$ . Recall that given the mapping  $\phi$  from the branches of  $T$  to the branches of  $N_{\lambda,\gamma}$ , the pre-image (or, inverse image)  $\phi^{-1}(b')$  is the set of all branches in  $T$  that map to  $b'$  under  $\phi$ . That is,  $\phi^{-1}(b') = \{e \in E(T) : \phi(e) = b'\}$ , where  $E(T)$  is the set of  $T$ 's branches. Then, I define

$$u_{b'}(h) = \sum_{b \in \phi^{-1}(b')} u_b(h) \quad \text{and} \quad v_{b'}(h) = \sum_{b \in \phi^{-1}(b')} v_b(h). \quad (4.7)$$

This equation states that the number of lineages  $u_{b'}(h)$  that enters (working backward in time) branch  $b'$  in the phylogenetic network equals the sum of the numbers of lineages that enter all branches of the MUL tree that map to branch  $b'$ . The number of lineages  $v_{b'}(h)$  that exists branch  $b'$  is defined similarly. In Figure 3.5, the number of lineages that enters branch  $b' = (h, w)$  in the phylogenetic network equals the sum of the number of lineages

that enter branch  $b_1 = (h_1, w_1)$  and the number of lineages that enter branch  $b_2 = (h_2, w_2)$  in the MUL tree.

Then, I use the following equation to evaluate the probability in Equation (4.6):

$$\prod_{b \in \phi^{-1}(b')} P'_b(h) = \frac{1}{d_{b'}(h)} p_{u_{b'}(h)v_{b'}(h)}(\lambda_{b'}) (u_{b'}(h) - v_{b'}(h))! \prod_{b \in \phi^{-1}(b')} \frac{w_b(h)}{(u_b(h) - v_b(h))!}, \quad (4.8)$$

where  $d_{b'}(h)$  is computed using the formula in [DS05b], with  $u_{b'}(h)$  and  $v_{b'}(h)$  as parameters. In the example of branches  $b'$ ,  $b_1$  and  $b_2$  that I just illustrated, Equation (4.8) states that  $P'_{b_1}(h)P'_{b_2}(h)$  evaluates to

$$\frac{1}{d_{b'}(h)} p_{u_{b'}(h)v_{b'}(h)}(\lambda_{b'}) (u_{b'}(h) - v_{b'}(h))! \frac{w_{b_1}(h)}{(u_{b_1}(h) - v_{b_1}(h))!} \frac{w_{b_2}(h)}{(u_{b_2}(h) - v_{b_2}(h))!}.$$

The term  $p_{u_{b'}(h)v_{b'}(h)}(\lambda_{b'})$  gives the probability that  $u_{b'}(h)$  lineages coalesce into  $v_{b'}(h)$  lineages within time  $\lambda(b')$ . The term

$$[(u_{b'}(h) - v_{b'}(h))! \prod_{b \in \phi^{-1}(b')} (w_b(h)/(u_b(h) - v_b(h))!)]$$

corresponds to the quantity  $w_{b'}(h)$  in [DS05b]. Finally, the term

$$\prod_{b \in \phi^{-1}(b')} (w_b(h)/(u_b(h) - v_b(h))!)$$

is the number of restrictions for the ordering of coalescent events within branch  $b'$ .

Similar to the algorithm for computing the minimum number of extra lineages of a gene tree given a phylogenetic network based on MUL tree, the number of allele mappings dominates the running time of this algorithm. See Section 3.1.1 for an analysis.

#### 4.1.1.2 An algorithm based on weighted ancestral configurations

The concept of *weighted ancestral configuration* (AC) is introduced in Section 3.1.2 for algorithm for counting the minimum number of extra lineages. It can also be used here to

compute the probability of observing a gene tree topology given a phylogenetic network. The same definition of AC is used, but with weight  $w$  in  $AC = (B, a, w)$  being interpreted as  $p(AC)$ , which denotes the cumulative probability of the extant gene lineages in  $AC$  coalescing into the present gene lineages in  $AC$  from time 0. Accordingly, the two main operations that occur as the algorithm proceed bottom-up need to change for the parts where weights are involved:

- At a reticulation node, when splitting  $AC = (B, a, w)$  into all possible pairs of  $AC_1 = (B_1, a_1, w_1)$  and  $AC_2 = (B_2, a_2, w_2)$ , weights should be updated as  $w_1 = w$  and  $w_2 = 1$ .
- At an internal tree node, when merging two compatible ACs  $AC_1 = (B_1, a_1, w_1)$  and  $AC_2 = (B_2, a_2, w_2)$  into  $AC_2 = (B, a, w)$ , weight  $w$  should be set to  $w_1 \cdot w_2$ .

**Lemma 1** *Let  $B$  be a set of gene lineages entering branch  $b$  of network  $N$  with branch length  $\lambda_b$ . Then the probability of observing a set of gene lineages  $B^+$  leaving branch  $b$  is*

$$p_t(B, B^+, b) = p_{|B|, |B^+|}(\lambda_b) \frac{w_b(B, B^+)}{d_b(B, B^+)}, \quad (4.9)$$

where  $p_{|B|, |B^+|}(\lambda_b)$  is the probability that  $|B|$  gene lineages coalesce into  $|B^+|$  gene lineages within time  $\lambda_b$ ,  $w_b(B, B^+)$  is the number of ways that coalescent events can occur along edge  $b$  to coalesce  $B$  into  $B^+$  with respect to the gene tree topology, and  $d_b(B, B^+)$  is the number of all possible orderings of  $|B| - |B^+|$  coalescent events.

**Observation 2** *Let  $AC = (B, a, w)$  be a configuration entering an edge  $b$  and  $AC^+ = (B^+, a^+, w^+)$  be a configuration that  $AC$  coalesced into when leaving  $b$ . Then  $w^+ = w \cdot p_t(B, B^+, b)$ .*

I define a function called **CreateCACsForProb** which takes a gene tree  $g$ , an edge  $b = (u, v)$  of the network  $N$  and a set of ACs  $\mathcal{AC}_v$  that enter edge  $b$ , and returns a set of all possible ACs  $\mathcal{AC}_{(u,v)}$  that exit edge  $b$ . See Alg. 5 for details. Recall that  $Coal(B, g)$  in the function denotes the set of all sets of lineages that a set of lineages  $B$  could coalesce into with respect to the topology of gene tree  $g$ . If no coalescent events occur on edge  $b$ ,  $|\mathcal{AC}_v| = |\mathcal{AC}_{(u,v)}|$ ; otherwise,  $|\mathcal{AC}_v| < |\mathcal{AC}_{(u,v)}|$ .

---

**Algorithm 5: CreateCACsForProb.**

---

**Input:** Gene tree  $g$ , an edge  $b = (u, v)$ , a set of ACs  $\mathcal{AC}_v$

**Output:** A set of ACs  $\mathcal{AC}_{(u,v)}$

**foreach**  $(B, a, w) \in \mathcal{AC}_v$  **do**

**foreach**  $B^+ \in Coal(B, g)$  **do**

        Compute  $w^+$  using Rule 2;

**if**  $\exists (B', a', w') \in \mathcal{AC}_{(u,v)}$  where  $B' = B^+$  and  $a' = a$  **then**

$w' \leftarrow w' + w^+$ ;

**else**

$\mathcal{AC}_{(u,v)} \leftarrow \mathcal{AC}_{(u,v)} \cup (B^+, a, w^+)$ ;

**return**  $\mathcal{AC}_{(u,v)}$ ;

---

The algorithm for calculating the probability of observing a gene tree  $g$  given a species network  $N$  is given in Alg. 6. The basic idea is similar to the parsimony method described in Section 3.1.2, where ancestral configuration sets are being built when traversing the network bottom-up and the final probability can be obtained after the configuration set for the root has been constructed. To better illustrate this process, an example is given in Fig.

4.1 which shows the sets of all ACs constructed during the executions.

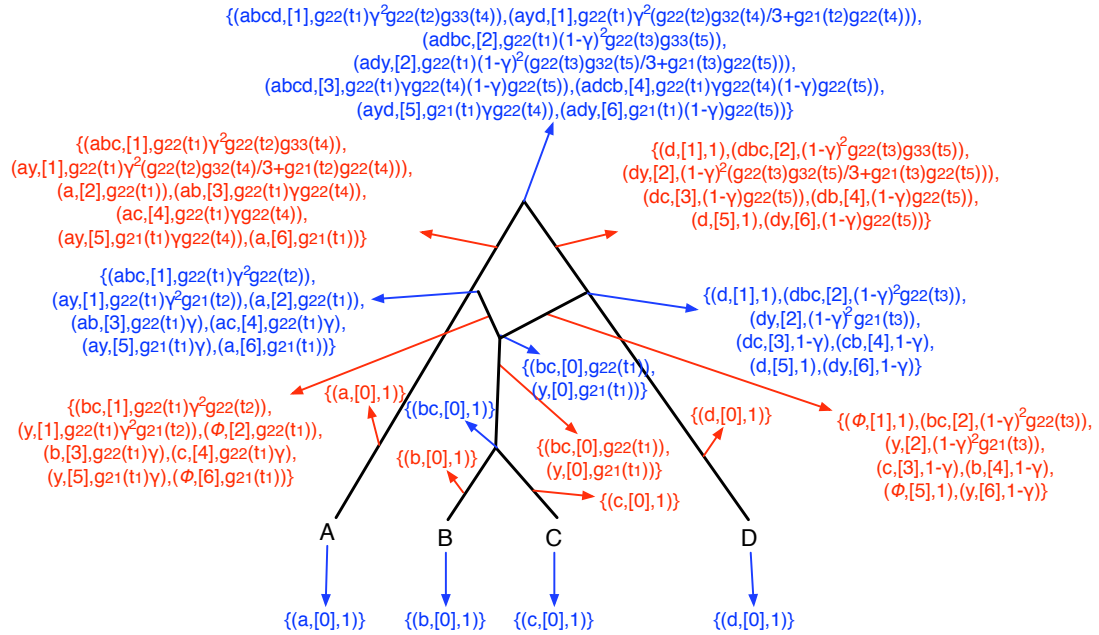


Figure 4.1: The ancestral configurations that result during the computations given phylogenetic network and gene tree  $((a, d), (b, c))$  in Fig. 2.4 under the probabilistic approach. Configurations in blue represent configurations generated for nodes and configurations in red represent configurations generated for branches. Curly braces and commas are removed from the ACs for compactness (e.g.,  $ady$  is the set  $\{a, d, y\}$ ). The branch length of branch  $i$  ( $i = 1, \dots, 5$ ) is represented by  $t_i$ .

It is important to note that although this algorithm is much more efficient than the MUL tree based one in simulation study (see Section 4.4.1.2.1), the running time of this algorithm can still be exponential for some data sets, as the complexity of the problem is open and conjectured to be NP-hard.



**Algorithm 6: CalProb.**

**Input:** Phylogenetic network  $N$  including topology, edge lengths and inheritance probabilities, gene

tree  $g$

**Output:**  $P(g|N)$

**while** traversing the nodes of  $N$  in post-order **do**

**if** node  $v$  is a leaf, whose parent is  $u$  **then**

$\mathcal{AC}_v \leftarrow \{(B, a, 1)\}$  where  $B$  is the set of leaves in  $g$  sampled from the species associated

    with  $v$  and  $a$  is a vector of  $q$  0's;

$\mathcal{AC}_{(u,v)} \leftarrow \text{CreateCACsForProb}(g, (u, v), \mathcal{AC}_v)$ ;

**else if** node  $v$  is a reticulation node, who has child  $w$ , and two parents  $u_1$  and  $u_2$  **then**

$\mathcal{AC}_v \leftarrow \mathcal{AC}_{(v,w)}$ ;

$S_1 \leftarrow \emptyset$ ;

$S_2 \leftarrow \emptyset$ ;

**foreach**  $AC \in \mathcal{AC}_v$  **do**

        Split  $AC$  in every possible way into pairs of ACs, and for each pair, add one AC to  $S_1$

        and the other AC to  $S_2$  ;

**foreach**  $(B, a, w) \in S_1$  **do**

$w \leftarrow w \cdot \gamma_{(u_1,v)}^{|B|}$  ;

$\mathcal{AC}_{(u_1,v)} \leftarrow \text{CreateCACsForProb}(g, (u_1, v), S_1)$ ;

**foreach**  $(B, a, w) \in S_2$  **do**

$w \leftarrow w \cdot \gamma_{(u_2,v)}^{|B|}$  ;

$\mathcal{AC}_{(u_2,v)} \leftarrow \text{CreateCACsForProb}(g, (u_2, v), S_2)$ ;

**else if** node  $v$  is an internal tree node, who has two children  $w_1$  and  $w_2$  **then**

**foreach** pair  $(AC_1, AC_2)$  of compatible ACs in  $\mathcal{AC}_{(v,w_1)} \times \mathcal{AC}_{(v,w_2)}$  **do**

        Merge  $AC_1$  and  $AC_2$  and add the resulting AC to  $\mathcal{AC}_v$ ;

**if** node  $v$  is an internal tree node, who has a parent  $u$  **then**

$\mathcal{AC}_{(u,v)} \leftarrow \text{CreateCACsForProb}(g, (u, v), \mathcal{AC}_v)$ ;

**else**

        Let  $B_R$  be the root lineage of the gene tree  $g$  ;

**return**  $\sum_{(B,a,w) \in \mathcal{AC}_v} w \cdot p_t(B, B_R, +\infty)$ ;

### 4.1.2 Using both topologies and branch lengths of the gene trees

Given a species tree, the method for computing the probability of observing a gene tree with branch lengths was introduced in [RY03]. In this section, I propose the first method for computing this probability when the given species phylogeny is a network [YDLN14]. Note that for this method both the gene tree and the phylogenetic network need to be ultrametric.

Given a phylogenetic network  $N_{\lambda,\gamma}$ , the probability of observing a gene tree  $g_{\lambda'}$  can be calculated by

$$P(g_{\lambda'}|N_{\lambda,\gamma}) = \sum_{ht \in H_{N_{\lambda,\gamma}}(g_{\lambda'})} P(ht|N_{\lambda,\gamma}), \quad (4.10)$$

where  $P(ht|N_{\lambda,\gamma})$  is the probability of observing coalescent history with respect to coalescence times  $ht$  given phylogenetic network  $N_{\lambda,\gamma}$ . For an edge  $b = (u, v) \in E(N_{\lambda,\gamma})$ , I define  $T_b(ht)$  to be  $\{\tau_{g_{\lambda'}}(w) : w \in ht^{-1}(b)\} \cup \{\tau_{N_{\lambda,\gamma}}(v)\}$  in an increasing order, so  $T_b(ht)$  contains a list of ordered coalescence times of the coalescent events in  $ht$  that occur on branch  $b$  plus the time of node  $v$  of  $N$ , where  $T_b(ht)_k$  stands for the  $k_{th}$  element in  $T_b(ht)$ . Furthermore, I denote by  $u_b(ht)$  the number of gene lineages entering edge  $b$  and  $v_b(ht)$  the number of gene lineages leaving edge  $b$  under  $ht$ . Then the probability of observing a coalescent history with respect to coalescence time  $ht$  can be calculated as follows:

$$P(ht|N_{\lambda,\gamma}) = \prod_{b=(u,v) \in E(N_{\lambda,\gamma})} \left[ \prod_{k=1}^{|T_b(ht)|-1} f_c(u_b(ht) - k + 1, T_b(ht)_{k+1} - T_b(ht)_k) \right. \\ \left. \times \frac{1}{\binom{u_b(ht)-k+1}{2}} \right] \times f_n(v_b(ht), \tau_{N_{\lambda,\gamma}}(u) - T_b(ht)_{|T_b(ht)|}) \times \gamma_b^{u_b(ht)} \quad (4.11)$$

which is taken as a product of the probabilities of  $ht$  on every branch of  $N_{\lambda,\gamma}$ , where  $f_c(j, t)$  is the probability of  $j$  gene lineages waiting for time  $t$  to coalesce into  $j - 1$  gene lineages

which can be computed as  $f_c(j, t) = \binom{j}{2} e^{-\binom{j}{2}t}$  [Kin82a, Kin82b],  $1/\binom{u_b(ht)-k+1}{2}$  is the probability of a particular pair of gene lineages among  $u_b(ht) - k + 1$  lineages coalescing, and  $f_n(j, t)$  is the probability of no coalescent events happening among  $j$  gene lineages for time  $t$  which can be computed as  $f_n(j, t) = e^{-\binom{j}{2}t}$  [Kin82a, Kin82b]. After substituting these terms, Eq. 4.11 becomes

$$P(ht|N_{\lambda,\gamma}) = \prod_{b=(u,v) \in E(N_{\lambda,\gamma})} \left[ \prod_{k=1}^{|T_b(ht)|-1} e^{-\binom{u_b(ht)-k+1}{2}(T_b(ht)_{k+1}-T_b(ht)_k)} \right] \times e^{-\binom{v_b(ht)}{2}(\tau_{N_{\lambda,\gamma}}(u)-T_b(ht)|_{T_b(ht)})} \times \gamma_b^{u_b(ht)}. \quad (4.12)$$

Table 4.2 shows how the probabilities of observing  $ht_1$  and  $ht_2$  in Fig. 2.5 are calculated according to Eq. 4.12, respectively.

Table 4.2: The probabilities of all coalescent histories with respect to coalescence times in Fig. 2.5. For every  $ht$ , columns from 2 to 7 list the probability of having  $ht$  on every branch of the species network  $N_{\lambda,\gamma}$ . Branch 6 corresponds to the branch incident into the root of the species network. A dash means no gene lineages enter that branch. Therefore, the total probability of a coalescent history with respect to coalescence times is the product taken over all branches of the species network. In Fig. 2.5, coalescent events  $y$  and  $z$  can only happen above the root of  $N_{\lambda,\gamma}$ . For every  $ht$ , the highlight cell shows where coalescent event  $x$  happens.

	Probability of each branch					
	1	2	3	4	5	6
$ht_1$	$e^{-\eta_4}$	$\gamma^2 e^{-(\eta_3-\eta_4)}$	—	$3e^{-(\tau_3-\eta_3)} e^{-(\eta_1-\tau_3)}$	1	$3e^{-(\tau_2-\eta_1)} e^{-(\tau_1-\tau_2)}$
$ht_2$	$e^{-\eta_4}$	—	$(1-\gamma)^2 e^{-(\eta_2-\eta_4)}$	1	$3e^{-(\tau_3-\eta_2)} e^{-(\eta_1-\tau_3)}$	$3e^{-(\tau_2-\eta_1)} e^{-(\tau_1-\tau_2)}$

Similar to the technique used in Section 3.1.1 and Section 4.1.1.1,  $P(g_{\lambda'}|N_{\lambda,\gamma})$  can be

computed using the concept of MUL tree. More specifically, I first convert the phylogenetic network to a MUL tree. Then under every allele mapping, I compute the set of coalescent histories with respect to coalescence times and use Eq. 4.12 to compute the probability of every coalescent history. Note that again special attention needs to be paid to the sets of edges in MUL tree that come from the same edge in the original network.

Here, I will focus on the algorithm for computing  $P(g_{\lambda'} | N_{\lambda, \gamma})$  based on weighted ancestral configurations, since it is faster than the one based on MUL tree. The definition and operations of the weighted ancestral configuration follow the one used in algorithm for computing the probability of observing a gene tree topology given a phylogenetic network in Section 4.1.1.2.

I define a function called **alg:CreateCACsForProbUsingBL** (see Alg. 7), which takes a gene tree  $g_{\lambda'}$ , a branch  $b = (u, v)$  of the phylogenetic network and a set of ACs  $\mathcal{AC}_v$  that enter branch  $b$  as input, and returns the set of ACs  $\mathcal{AC}_{(u,v)}$  that exit branch  $b$  according to the topology of the gene tree with respect to the coalescence times.  $Coal(B, g)$  in the function denotes the set of all sets of lineages that a set of lineages  $B$  could coalesce into with respect to the topology of gene tree  $g$ . Suppose there should be some coalescent event happening on branch  $b$ , which means there is a node  $w$  in  $g_{\lambda'}$  where  $\tau_{N_{\lambda, \gamma}}(v) \leq \tau_{g_{\lambda'}}(w) < \tau_{N_{\lambda, \gamma}}(u)$  and  $L_w \subseteq L_v$ , where  $L_w$  is the set of taxa under node  $w$  in  $g_{\lambda'}$  and  $L_v$  is the set of taxa under node  $v$  in  $N_{\lambda, \gamma}$ . Note that some ACs in  $\mathcal{AC}_v$  here may not contain all gene lineages that are required for coalescent event represented by  $w$  to occur. In this case, they will be removed. And for those ACs in  $\mathcal{AC}_v$  that contribute to  $\mathcal{AC}_{(u,v)}$ , there is a 1-1 correspondence between them and those in  $\mathcal{AC}_{(u,v)}$ , because once time constraints are imposed it is deterministic what coalescent events would occur for gene lineages in an AC

on a branch. And therefore,  $|\mathcal{AC}_v| \leq |\mathcal{AC}_{(u,v)}|$ . Note that if node  $v$  is a reticulation node,  $|\mathcal{AC}_v|$  here represents the set of ACs that about to enter branch  $(u, v)$  after splitting.

The main idea of the algorithm is similar to the one in Section 3.1.2 for computing the minimum number of extra lineages of a gene tree topology given a phylogenetic network and to the one in Section 4.1.1.2 for computing the probability of observing a gene tree topology given a phylogenetic network. We basically traverse the network in a bottom-up fashion while building ancestral configuration sets for nodes being visited, and the final probability can be obtained once the ancestral configuration set has been constructed for the root. The complete algorithm is given in Alg. 8. Besides, I used the same technique introduced in 3.1.2 to reduce the number of configurations at articulation nodes of the network. Again, the running time of this algorithm is still exponential for some data sets, as the complexity of the problem is open and conjectured to be NP-hard.

## 4.2 Handling gene tree uncertainty

Gene tree topologies may have uncertainty when they are estimated from sequence data. In Section 4.2, I proposed ways to handle gene trees with uncertainty when parsimony approach is used to infer species phylogeny. In this section, I proposed a way for incorporating the uncertainty of gene trees into my probabilistic framework [YDN12, YDLN14].

Assume there are  $k$  loci under analysis, and for each locus  $i$ , a Bayesian analysis of the sequence alignment returns a collection of gene trees  $g_1^i, \dots, g_q^i$ , along with their associated posterior probabilities  $p_1^i, \dots, p_q^i$  ( $p_1^i + \dots + p_q^i = 1$ ). Now, let  $\mathcal{G}$  be the set of all distinct tree topologies computed on all  $k$  loci, and for each  $g \in \mathcal{G}$  let  $p_g$  be the sum of posterior

---

**Algorithm 7: CreateCACsForProbUsingBL.**


---

**Input:** a gene tree  $g_{\lambda'}$ , an edge  $(x, y) \in E(N_{\lambda, \gamma})$ , a set of ACs  $\mathcal{AC}_y$

**Output:** a set of ACs  $\mathcal{AC}_{(x,y)}$

Let  $V_g$  be the set of internal nodes of  $g_{\lambda'}$  ordered by their heights increasing;

$\mathcal{AC}_{(x,y)} \leftarrow \emptyset$ ;

**foreach**  $(B, a, w) \in AC_y$  **do**

$t \leftarrow \tau_{N_{\lambda, \gamma}}(y)$ ;

$B^+ \leftarrow B$ ;

$p \leftarrow \lambda_{(x,y)}^{|B|}$ ;

**foreach**  $v \in V_g$  **do**

**if**  $\tau_{N_{\lambda, \gamma}}(y) \leq \tau_{g_{\lambda'}}(v) < \tau_{N_{\lambda, \gamma}}(x)$  **then**

Let  $L_v$  be the set of taxa under node  $v$  in  $g_{\lambda'}$ ;

Let  $L_B$  is the set of taxa that coalesce into  $B$ ;

**if**  $L_v \subseteq L_B$  **then**

$p \leftarrow p \cdot e^{-\binom{|B^+|}{2}(\tau_{g_{\lambda'}}(v)-t)}$ ;

$t \leftarrow \tau_{g_{\lambda'}}(v)$ ;

Apply the coalescent event represented by  $v$  to  $B^+$  and the resulting  $B^+$  contains one less lineages;

**else if**  $L_v \cap L_B \neq \emptyset$  **then**

$p \leftarrow 0$ ;

Break;

**if**  $p \neq 0$  **then**

**if**  $|B^+| \neq 1$  **then**

$p \leftarrow p \cdot e^{-\binom{|B^+|}{2}(\tau_{N_{\lambda, \gamma}}(x)-t)}$ ;

$\mathcal{AC}_{(x,y)} \leftarrow \mathcal{AC}_{(x,y)} \cup (B^+, a, w \cdot p)$ ;

**return**  $\mathcal{AC}_{(x,y)}$ ;

---

---

**Algorithm 8: CalProbUsingBL.**


---

**Input:** Phylogenetic network  $N_{\lambda,\gamma}$ , gene tree  $g_{\lambda'}$

**Output:**  $P(g_{\lambda'}|N_{\lambda,\gamma})$

**while** traversing the nodes of  $N$  in post-order **do**

**if** node  $v$  is a leaf, whose parent is  $u$  **then**

$\mathcal{AC}_v \leftarrow \{(B, a, 1)\}$  where  $B$  is the set of leaves in  $g_{\lambda'}$  sampled from the species associated with  $v$  and  $a$  is a vector of  $q$  0's;

$\mathcal{AC}_{(u,v)} \leftarrow \text{CoalACs}(g_{\lambda'}, (u, v), \mathcal{AC}_v)$ ;

**else if** node  $v$  is a reticulation node, who has child  $w$ , and two parents  $u_1$  and  $u_2$  **then**

$\mathcal{AC}_v \leftarrow \mathcal{AC}_{(v,w)}$ ;

$S_1 \leftarrow \emptyset$ ;

$S_2 \leftarrow \emptyset$ ;

**foreach**  $AC \in \mathcal{AC}_v$  **do**

        Split  $AC$  in every possible way into pairs of ACs, and for each pair, add one to

$S_1$  and the other to  $S_2$  ;

$\mathcal{AC}_{(u_1,v)} \leftarrow \text{CreateCACsForProbUsingBL}(g_{\lambda'}, (u_1, v), S_1)$ ;

$\mathcal{AC}_{(u_2,v)} \leftarrow \text{CreateCACsForProbUsingBL}(g_{\lambda'}, (u_2, v), S_2)$ ;

**else if** node  $v$  is an internal tree node, who has two children  $w_1$  and  $w_2$  **then**

**foreach** pair  $(AC_1, AC_2)$  of compatible ACs in  $\mathcal{AC}_{(v,w_1)} \times \mathcal{AC}_{(v,w_2)}$  **do**

        Merge  $AC_1$  and  $AC_2$  and add the resulting AC to  $\mathcal{AC}_v$ ;

**if** node  $v$  is an internal tree node, who has a parent  $u$  **then**

$\mathcal{AC}_{(u,v)} \leftarrow \text{CoalACs}(g_{\lambda'}, (u, v), \mathcal{AC}_v)$ ;

**else**

        Create a virtual node  $r'$  with height  $+\infty$ ;

$\mathcal{AC}_{(r',v)} \leftarrow \text{CoalACs}(g_{\lambda'}, (r', v), \mathcal{AC}_v)$ ;

**return**  $\sum_{(B,a,w) \in \mathcal{AC}_{(r',r)}} w$ ;

---

probabilities associated with all gene trees computed over all loci whose topology is  $g$ .

Thus,  $p_g = \sum_{i=1}^k p_g^i$  and  $\sum_{g \in \mathcal{G}} p_g = k$ . Then, Eq. 4.2 becomes

$$P(\mathcal{G}|N_{\lambda,\gamma}) = \prod_{g \in \mathcal{G}} [P(g|N_{\lambda,\gamma})]^{p_g}. \quad (4.13)$$

Note that if  $p_j^i = 1$  or 0 for each  $i$  and  $j$ , then Eq. 3.11 is equivalent to Eq. 3.2. I additionally allow the  $p_j^i$  terms to be between 0 and 1 (and therefore  $p_g$  to be non-integer values) in order to reflect uncertainty in the estimated gene trees.

In the case where a maximum parsimony analysis is conducted to infer gene trees on the individual loci, a different treatment is necessary, since for each locus, all inferred trees are equally optimal. For locus  $i$ , let  $g$  be the strict consensus of all optimal gene tree topologies found. Then, Eq. 4.2 can be replaced by either

$$P(\mathcal{G}|N_{\lambda,\gamma}) = \prod_{g \in \mathcal{G}} \max_{g' \in b(g)} P(g'|N_{\lambda,\gamma}) \quad (4.14)$$

or

$$P(\mathcal{G}|N_{\lambda,\gamma}) = \prod_{g \in \mathcal{G}} \sum_{g' \in b(g)} P(g'|N_{\lambda,\gamma}) \quad (4.15)$$

Notice that under this Eq. 4.15, a completely unresolved gene tree  $g$  (that is, a star) would have probability 1, regardless of the phylogenetic network  $N_{\lambda,\gamma}$ .

### 4.3 Inferring a phylogenetic network

Given a collection of gene trees, in order to find the optimal phylogenetic network under maximum likelihood, I used the same method for searching the space of phylogenetic networks described in Section 3.3. During the search, for each network topology proposed, we need to optimize its branch lengths and inheritance probabilities which I described below.



### 4.3.1 Optimizing branch lengths and inheritance probabilities of a phylogenetic network

In this section, I describe that given a collection of gene trees  $\mathcal{G}$  how to find the optimal branch lengths  $\lambda^*$  and inheritance probabilities  $\gamma^*$  for a given network topology  $N$ , where  $\lambda^*, \gamma^* = \operatorname{argmax}_{\lambda, \gamma} P(\mathcal{G}|N_{\lambda, \gamma})$ . I will discuss the two cases where gene trees  $\mathcal{G}$  have and do not have branch lengths separately [YDLN14].

#### 4.3.1.1 Using only topologies of gene trees

A heuristic for finding the optimal branch lengths for a given species tree topology was introduced in [Wu12]. Here, I am using the same method but in the case of phylogenetic networks I am optimizing not only branch lengths but also inheritance probabilities. In particular, an initial value of likelihood is first calculated with every branch length initialized to be 1.0 and inheritance probability initialized to be 0.5. Then the elements in  $[\lambda, \gamma]$  are optimized one by one separately using the well-known Brent's method [Bre73]. More specifically, while the Brent's method is varying the value of one element in  $[\lambda, \gamma]$  in order to find a local optimum, the values of all other elements are fixed. After the local optimum is found, the element is replaced by this new value and then the Brent's method moves to the next element for optimization. Updating all elements in  $[\lambda, \gamma]$  once is called a round. After each round of optimization, I compare the new likelihood score of  $P(\mathcal{G}|N_{\lambda', \gamma'})$ , where  $\lambda'$  and  $\gamma'$  are newly updated in this round, with the one after previous round. If the improvement is smaller than some predetermined threshold or some predetermined maximum number of rounds is reached,  $\lambda'$  and  $\gamma'$  are claimed to be optimum and the process

terminates. Since elements in  $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$  are optimized one by one separately, it is not guaranteed to find the global optimal solution. To avoid getting stuck at local optimum, the order of which the elements in  $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$  are being optimized is shuffled at every round. In practice, I see very accurate estimates in the simulation study. All parameters used in this optimization process, including those for the Brent's method, are listed in Chapter 5.

Given a phylogenetic network  $N$  and a gene tree  $g$ , note that the set of coalescent histories  $H_{N_{\boldsymbol{\lambda}, \boldsymbol{\gamma}}}(g)$  remains the same no matter how the branch lengths  $\boldsymbol{\lambda}$  and inheritance probabilities  $\boldsymbol{\gamma}$  of  $N$  change, so as the set of ancestral configurations themselves at every node of  $N$ . Hence, while varying the value of  $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$  in order to maximize  $P(\mathcal{G}|N_{\boldsymbol{\lambda}, \boldsymbol{\gamma}})$ , there is no need to recompute the ancestral configuration sets after computing them once. Instead, what we need to do is only to update the probabilities of ancestral configurations if necessary. More specifically, when the length of a branch  $(u, v)$  changes, only the probabilities of ancestral configurations at the ancestor nodes of  $u$  and node  $u$  itself need to be updated. Similarly, when the inheritance probabilities of two branches incident into a reticulation node  $v$  change, only the probabilities of ancestral configurations at the ancestor nodes of  $v$  need to be updated. Since the method of calculating  $P(\mathcal{G}|N_{\boldsymbol{\lambda}, \boldsymbol{\gamma}})$  needs to be called many times when optimizing  $\boldsymbol{\lambda}$  and  $\boldsymbol{\gamma}$  of a phylogenetic network  $N$  given  $N$  and gene trees  $\mathcal{G}$ , doing computation from scratch only once significantly improves the efficiency due to the fact that only updating the corresponding probabilities in the afterward computations is trivial in terms of running time, especially for large data sets. However, since all ancestral configurations need to be saved in order to allow us to avoid computing them again, as a tradeoff, a lot of memory is required.

### 4.3.1.2 Using both topologies and branch lengths of gene trees

As I mentioned, the set of coalescent histories of a gene tree  $g$  within the branches of a phylogenetic network  $N$  does not change with the branch lengths or inheritance probabilities of  $N$  if only the topology of gene tree  $g$  is considered. In other words, given gene tree  $g$  without branch lengths, a phylogenetic network  $N$  with any values of branch lengths and inheritance probabilities is valid. In theory, a branch length can be anything in  $\mathbb{R}^+ \cup \{0\}$  and inheritance probability can be anything in  $[0, 1]$ . That is why every element in  $[\lambda, \gamma]$  can be optimized independently, as described in Section 4.3.1.1.

However, it is no longer the case if both topologies and branch lengths of the gene trees need to be taken into account. There are two reasons for that. First, as I mentioned in Section 4.1.2, calculating the probability of observing a gene tree  $g_{\lambda'}$  given a phylogenetic network  $N_{\lambda, \gamma}$  requires  $N_{\lambda, \gamma}$  to be ultrametric (so as  $g_{\lambda'}$ ). However, optimizing the branch lengths of  $N_{\lambda, \gamma}$  separately does not guarantee the ultrametricity of  $N_{\lambda, \gamma}$ . Second, even if some  $\lambda$  makes  $N_{\lambda, \gamma}$  an ultrametric phylogenetic network, it is still considered invalid if  $g_{\lambda'}$  could not be reconciled within the branches of  $N_{\lambda, \gamma}$  with that set of branch lengths. As I mentioned before, if a set of taxa  $L$  coalesced at time  $t$  in  $g_{\lambda'}$ , the height of the most recent common ancestor node of any two taxa in  $L$  in the phylogenetic network  $N_{\lambda, \gamma}$  must be lower than  $g_{\lambda'}$ .

Now I describe how I deal with these two issues. Let  $\mathcal{G}_t$  be a collection of gene trees with branch lengths. First, in order to guarantee the ultrametricity, instead of optimizing branch lengths and inheritance probabilities of phylogenetic network  $N_{\lambda, \gamma}$ , I optimize the height of every internal node of  $N_{\lambda, \gamma}$  and inheritance probabilities, denoted by  $\phi$  of  $N$ .

Second, in order to ensure that the phylogenetic network  $N$  with  $\phi$  is valid for every gene tree in  $\mathcal{G}_t$ , I compute a vector  $\phi_{upper}$  from  $\mathcal{G}_t$  such that every element in  $\phi_{upper}$  is an upper bound of the corresponding element in  $\phi$ . I use  $\phi_{upper}(v)$  to denote the upper bound of the height of node  $v$ . Now I describe how  $\phi_{upper}$  is computed. I denote the coalescence time of  $x$  and  $y$  in gene tree  $g_{\lambda'}$  by  $t_{g_{\lambda'}}(x, y)$ , where  $x$  and  $y$  are two leaves of  $g_{\lambda'}$ . So  $t_{g_{\lambda'}}(x, y)$  is equal to the height of the most recent common ancestor node of  $x$  and  $y$  in  $g_{\lambda'}$ . For every pair of taxa  $x$  and  $y$  in phylogenetic network  $N$ , I first find their minimum coalescence time in gene trees  $\mathcal{G}_t$ , denoted by  $t_{\mathcal{G}_t}(x, y)$ , such that  $t_{\mathcal{G}_t}(x, y) = \min_{g_{\lambda'} \in \mathcal{G}_t} t_{g_{\lambda'}}(x, y) - \varepsilon$ . Let  $L_v$  represents the set of taxa that are reachable from  $v$  in phylogenetic network  $N$ . Then, for every internal node  $u$  of  $N$ , if it is a tree node,  $\phi_{upper}(u) = \min\{t_{\mathcal{G}_t}(x, y) : x \in L_{v_1} - L_{v_2}, y \in L_{v_2} - L_{v_1}\}$ , where  $v_1$  and  $v_2$  are two child nodes of  $u$ . If  $L_{v_1} - L_{v_2} = \emptyset$  or  $L_{v_2} - L_{v_1} = \emptyset$ ,  $\phi_{upper}(u)$  is set to  $+\infty$ . If it is a reticulation node,  $\phi_{upper}(u) = \phi_{upper}(v)$  where  $v$  is the child node of  $u$ .

The heuristic I devised for optimizing  $\phi$  and  $\gamma$  is as follows. I first compute  $\phi_{upper}$  from the gene trees  $\mathcal{G}_t$ , and then initialize  $\gamma$  to be all 0.5 and  $\phi$  to be some value according to  $\phi_{upper}$ . More specifically, I first set  $\phi$  to be the same as  $\phi_{upper}$ . Then I traverse the nodes of  $N$  in post-order. For every node  $v$  being visited, if  $v$  is a leaf, the height of it is set to 0. Otherwise, if  $v$  is not the root,  $\phi(v)$  has to be some value between  $maxD(v)$  and  $minA(v)$  where  $maxD(v) = \max\{\phi(w) : w \text{ is a descendant node of } v\}$  and  $minA(v) = \min\{\phi(u) : u \text{ is an ancestor node of } v\}$  to ensure the related branch lengths will not be negative. So if  $\phi(v) \geq minA(v)$ ,  $\phi(v)$  is reset to  $maxD(v) + (minA(v) - maxD(v))/(dDiff + 1)$  where  $dDiff$  is the difference between the depth of node  $v$  and the minimum depth of its parent nodes. If  $\phi(v) = +\infty$ ,  $\phi(v)$  is reset to  $maxD(v) + 1$ . After

initializing  $\phi$  and  $\gamma$ , an initial value of likelihood is calculated. Then the iterative process for optimization itself is similar to what is described in Section 4.3.1.1 where the elements in  $[\phi, \gamma]$  are optimized one by one separately using Brent's method. It is important to note that when  $\phi(v)$  is being varied for optimization for a node  $v$  in  $N$ , its value needs to stay in  $(\max D(v), \min\{\min A(v), \phi_{upper}(v)\})$  to make sure (i) it will not result in negative branch lengths; (ii) all  $\mathcal{G}_t$  can be reconciled within the branches of  $N_{\lambda, \gamma}$ . Again, the global optimum is not guaranteed from this heuristic.

Given a phylogenetic network  $N_{\lambda, \gamma}$  and a gene tree  $g_{\lambda'}$ , the set of coalescent histories  $H_{N_{\lambda, \gamma}}(g_{\lambda'})$  may change with  $\lambda$ . So during the optimization,  $P(\mathcal{G}_t | N_{\lambda, \gamma})$  needs to be computed from scratch every time  $\lambda$  is changed. On the other hand, much less memory is required compared to the optimization in Section 4.3.1.1, because no ancestral configurations need to be stored and also the number of coalescent histories is usually much smaller when both topologies and branch lengths of the gene trees need to be considered.

## 4.4 Performance

### 4.4.1 Simulation study

To study the performance of the criterion and the method in terms of the accuracy of the inferred branch lengths and inheritance probabilities, as well as the identifiability issue, I did intensive simulation studies [YDN12]. Also, I compared the efficiency between the MUL-tree based method and AC based method [YRN13].

#### 4.4.1.1 Identifiability of hybridization using gene tree topologies

To study the power of the likelihood approach at identifying hybridization events using gene tree topologies only, I did intensive simulation study. More specifically, I evolved gene trees within the branches of phylogenetic networks, while varying branch lengths and inheritance probabilities, and investigated two questions: (1) how much data (gene trees) is needed to obtain accurate inference of the parameters (branch lengths and/or inheritance probabilities)? (2) are the parameters always identifiable? To answer these two questions, I investigated six different phylogenetic network topologies in Fig. 4.2 that involved single reticulation scenario (Scenario **VI**), two reticulation scenarios (dependent (Scenario **I**) and independent (Scenario **IV** and **V**), and cases with extinctions involving the species that hybridize (Scenario **II** and **III**). Further, I varied the inheritance probabilities associated with the hybridization events in the phylogenetic networks. For Scenario **I** and **II**, I considered  $(\alpha, \beta) \in \{(0.0, 0.5), (0.3, 0.3), (0.5, 0.0), (0.5, 0.5), (0.5, 1.0)\}$ . For Scenario **III** and **VI**, I considered  $\alpha \in \{0.0, 0.3, 0.5\}$ . For Scenario **IV** and **V**, I considered  $(\alpha, \beta) \in \{(0.0, 0.5), (0.3, 0.3), (0.5, 0.5)\}$ . The rationale for selecting the three values 0.0, 0.3, and 0.5 is that they represent no hybridization, "skewed" hybridization (different genetic contributions of the two parents to the hybrid), and perfect hybridization (equal genetic contributions of the two parents to the hybrid). Finally, to vary the extent of deep coalescence within each of the four evolutionary histories, I considered two settings for branch lengths (are measured in coalescent units): setting 1, in which  $t_1 = t_2 = t_3 = t_4 = 1.0$ , and setting 2, in which  $t_1 = t_2 = t_3 = t_4 = 2.0$ . All reticulation branches have length 0. As the extent of ILS increases as branches become shorter, I expect setting 1 to provide

more challenging data for the method.

Using each combination of phylogenetic network, inheritance probabilities, and branch length setting, I used the `ms` program [Hud02] to generate 10, 20, 50, 100, 500, 1000 and 2000 gene trees within the branches of the phylogenetic networks. To obtain statistically significant results, I generated 100 data sets per parameter setting and evaluated the performance as averaged over these 100 data sets, for each point in the parameter space. In these experiments, a single allele per species per gene was sampled.

#### 4.4.1.1.1 Accuracy of inference

In this section, I studied the performance of the method in terms of estimating the branch lengths and inheritance probabilities when no extinction events were involved in the parents of hybrid populations in the phylogenetic networks. For the four scenarios **I**, **IV**, **V**, and **VI**, the parameters (branch lengths and inheritance probabilities) are identifiable, and we focused on the accuracy of our method for inferring these parameters from samples of gene trees that were simulated as discussed in the previous section. That is, given a sample  $\mathcal{G}$  of gene tree topologies, and a phylogenetic network topology  $N$ , I solved

$$\lambda^*, \gamma^* \leftarrow \operatorname{argmax}_{\lambda, \gamma} P(\mathcal{G} | N_{\lambda, \gamma}), \quad (4.16)$$

In this experiment, instead of using the heuristics in Section 4.3.1.1 to optimize these parameters, I used grid search. More specifically, to infer the inheritance probabilities I used a grid search of values between 0 and 1 with step length of 0.01, and for the branch lengths I used a grid search of values between 0.1 and 4.0 with step length of 0.1.

The results are shown in Figs. 4.3—4.6 below. We can see that both inheritance prob-

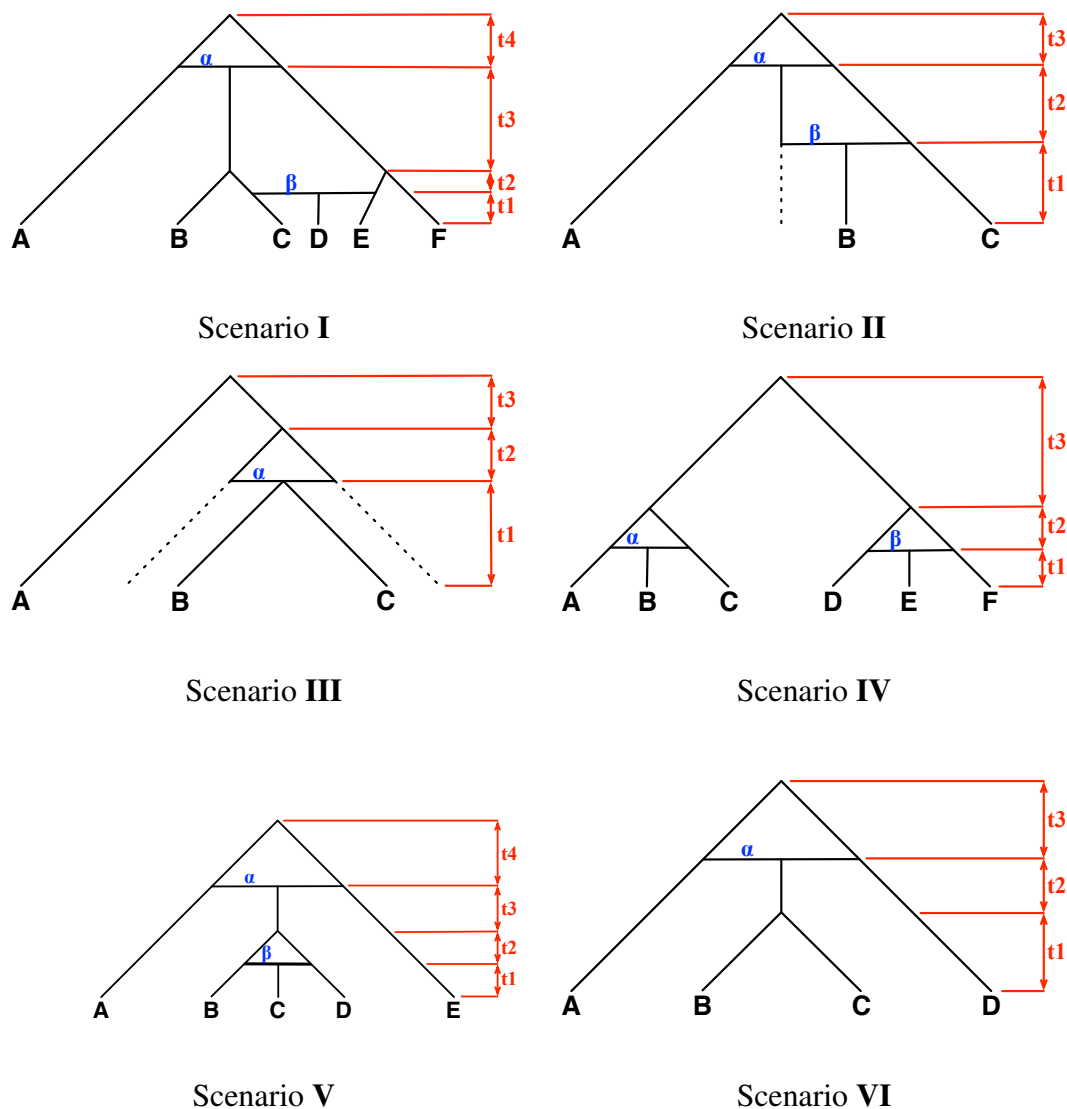


Figure 4.2: Phylogenetic networks depicting different hybridization/divergence/extinction scenarios. The  $\alpha$  and  $\beta$  parameters denote the proportions (or, probabilities) of alleles that are inherited from the “left” parents of the network-nodes ( $1 - \alpha$  and  $1 - \beta$  denote the proportions of the alleles that are inherited from the “right” parents of the nodes).

abilities and branch lengths can be estimated with very high accuracy provided that no extinction events were involved in the parents of hybrid populations. Further, this accuracy can be achieved even when using the smallest number of gene trees I used in the study,



which is 10. Under these settings, estimates using my framework seemed to converge quickly to the true values.

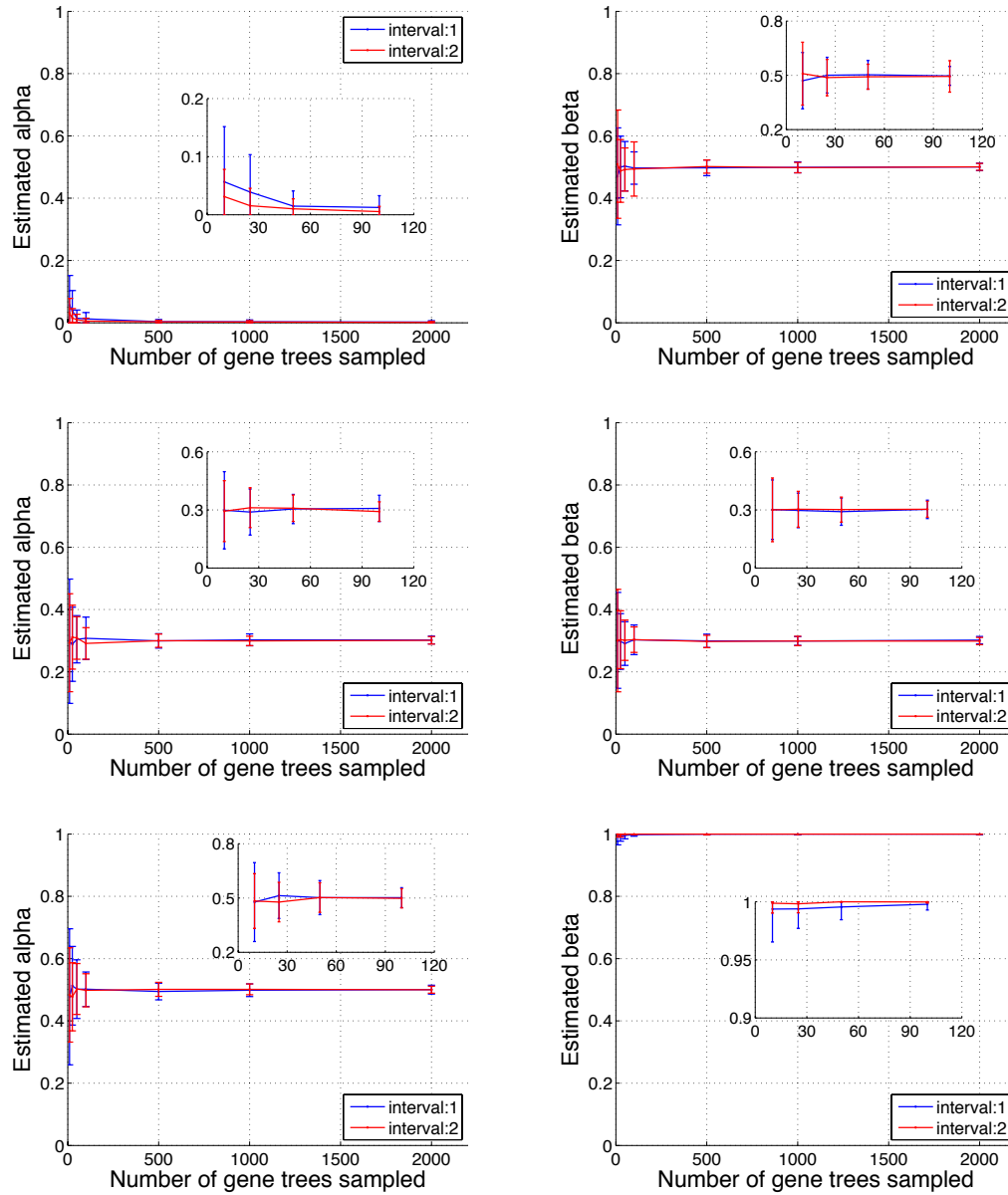


Figure 4.3: Estimates of  $\alpha$  and  $\beta$  on Scenario I. Rows from top to bottom correspond to true  $(\alpha, \beta)$  values of  $(0.0, 0.5)$ ,  $(0.3, 0.3)$ , and  $(0.5, 1.0)$ , respectively.

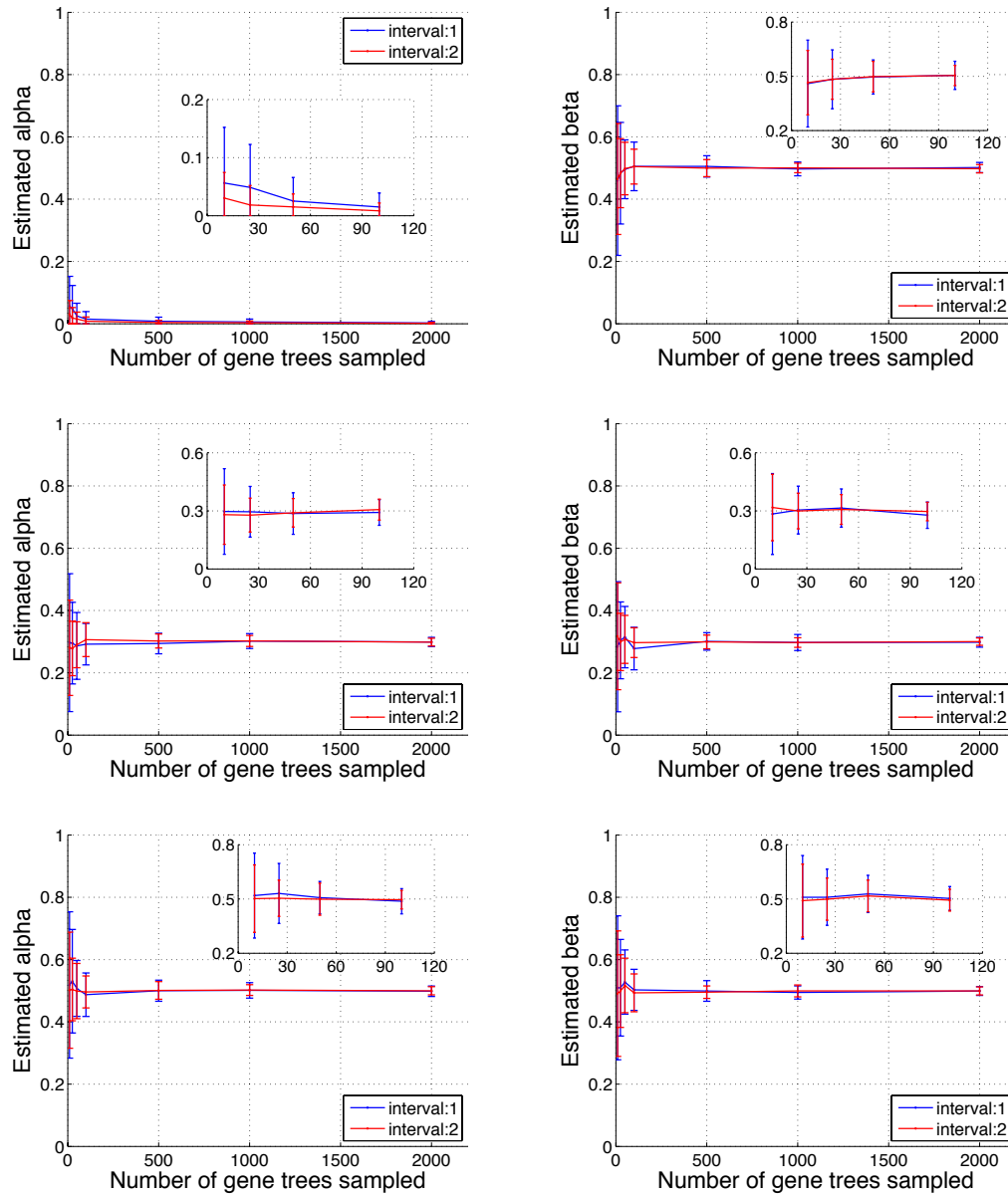


Figure 4.4: Estimates of  $\alpha$  and  $\beta$  on Scenario IV. Rows from top to bottom correspond to true  $(\alpha, \beta)$  values of  $(0.0, 0.5)$ ,  $(0.3, 0.3)$ , and  $(0.5, 0.5)$ , respectively.

#### 4.4.1.1.2 Identifiability

In this section, I investigated the performance of the method, as well as identifiability issues, when phylogenetic signal from at least one of the species involved in the hybridization

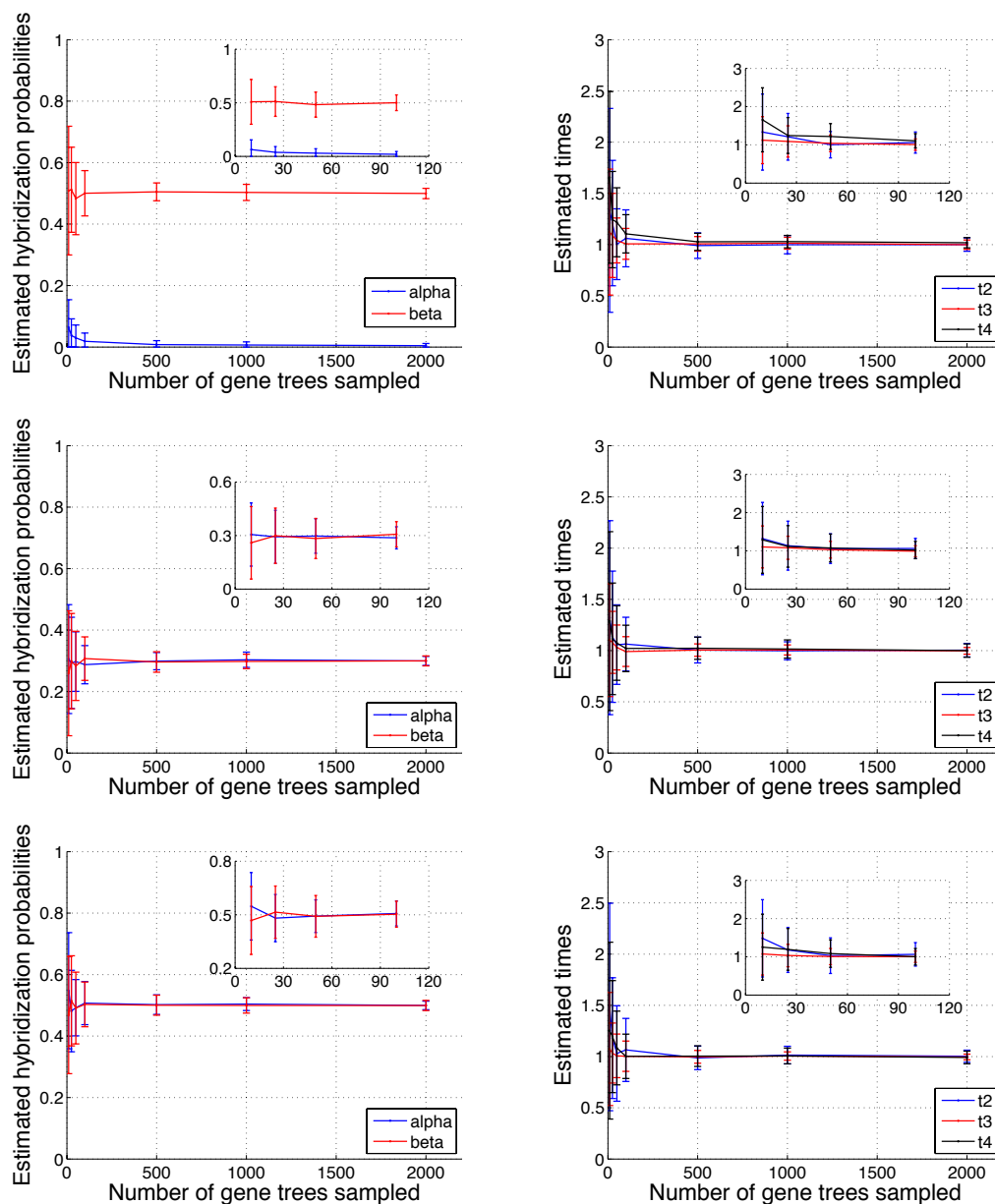


Figure 4.5: Estimates of  $\alpha$ ,  $\beta$ ,  $t_2$ ,  $t_3$ , and  $t_4$  on Scenario V. Rows from top to bottom correspond to true  $(\alpha, \beta)$  values of  $(0.0, 0.5)$ ,  $(0.3, 0.3)$ , and  $(0.5, 0.5)$ , respectively. All plots correspond to true values of  $t_1 = t_2 = t_3 = t_4 = 1.0$ .

is completely lost. I used the same inference procedure described in previous section to optimize branch lengths and inheritance probabilities of Scenario II and Scenario III.

The results of Scenario II is shown in Fig. 4.7. We can see that if the correct (true) val-

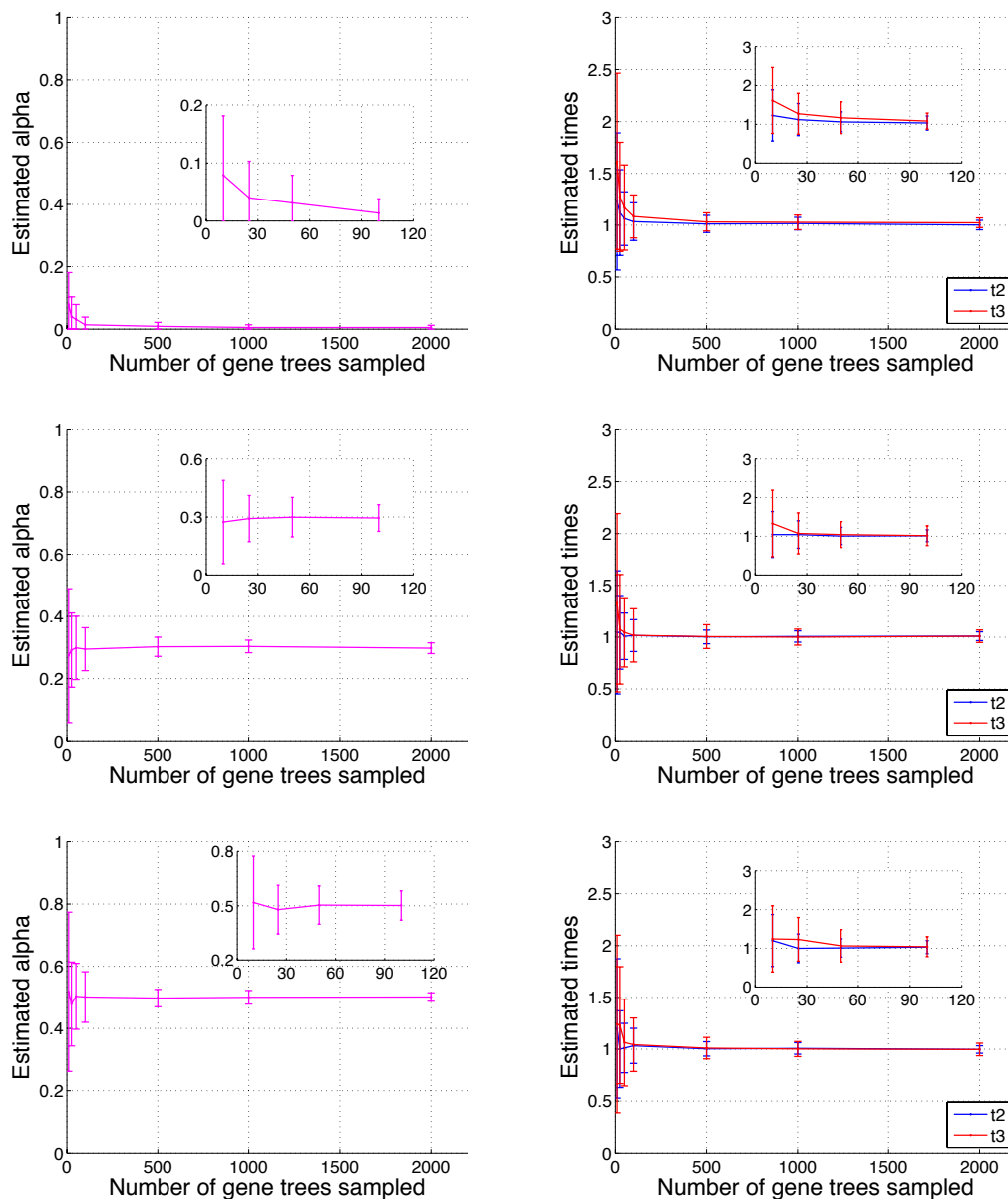


Figure 4.6: Estimates of  $\alpha$ ,  $t_2$ , and  $t_3$  on Scenario VI. Rows from top to bottom correspond to true  $\alpha$  values of 0.0, 0.3, and 0.5, respectively. All plots correspond to true values of  $t_2 = t_3 = 1.0$ .

ues of branch lengths are used, the inheritance probabilities are identifiable, and converge to the true values. However, unlike the cases that did not involved extinctions, a larger number of gene trees is now required to obtain an accurate estimate (while there are only three

possible gene tree topologies, a large number of gene trees need be sampled in order for the three topologies frequencies to be informative). The time setting 1 where  $t_2 = t_3 = 1$  amounts to a large extent of deep coalescence events, which blurs the phylogenetic signal, and results in slight over- or under-estimation of the inheritance probabilities.

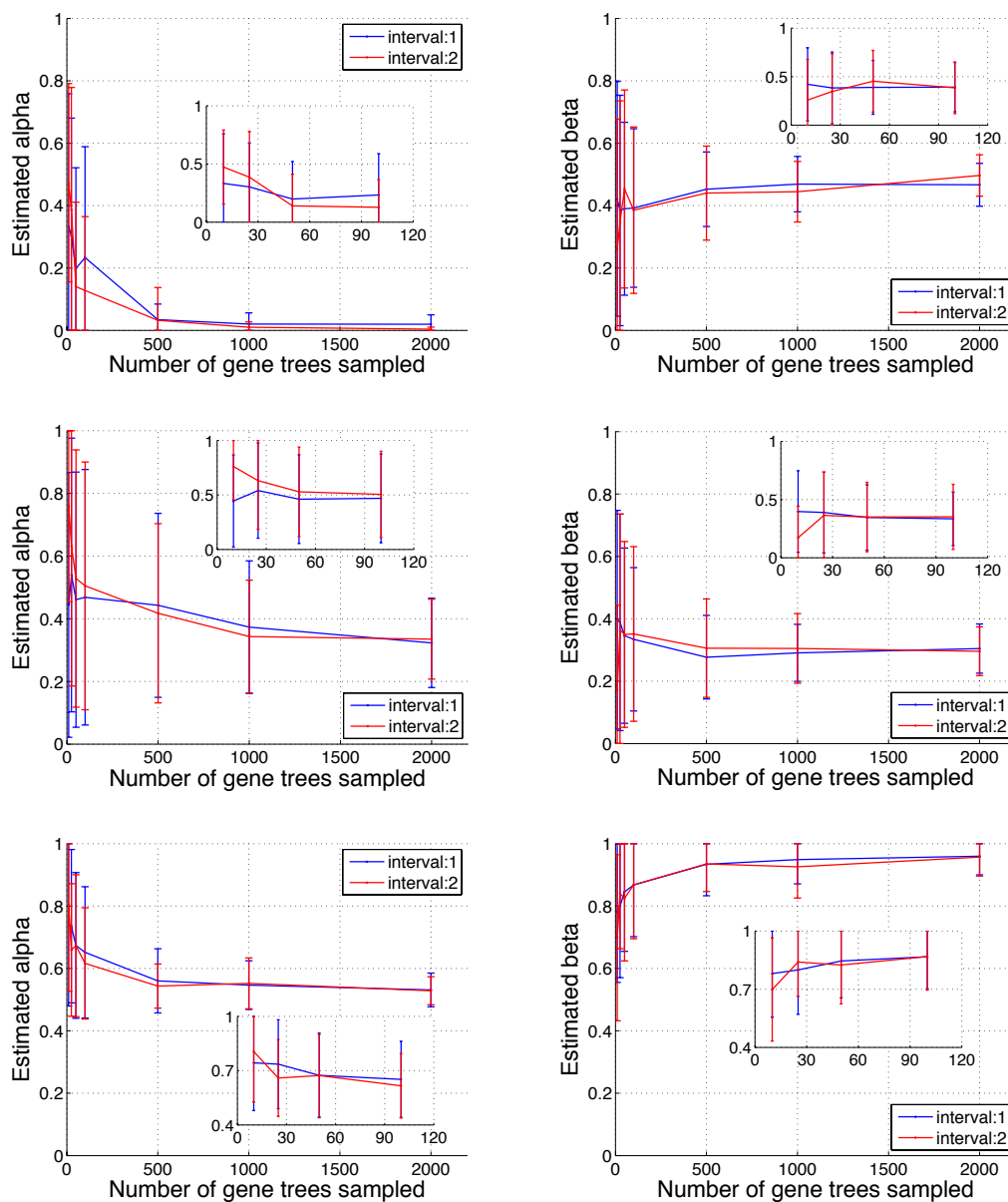


Figure 4.7: Estimates of  $\alpha$  and  $\beta$  on Scenario II. Rows from top to bottom correspond to true  $(\alpha, \beta)$  values of  $(0.0, 0.5)$ ,  $(0.3, 0.3)$ , and  $(0.5, 1.0)$ , respectively.

However, if both branch lengths and inheritance probabilities are to be estimated, then issues of unidentifiability arise, as I now show. Consider the phylogenetic network depicted by Scenario **II** in Fig. 4.2. Let  $\lambda$  be the branch lengths vector with  $\lambda_1 \equiv t_1 = s$ ,  $\lambda_2 \equiv t_2 = p$ , and  $\lambda_3 \equiv t_3 = q$ , and let  $\gamma$  be the inheritance probabilities vector with  $\gamma_1 \equiv \alpha = a$  and  $\gamma_2 \equiv \beta = b$ . For a given set  $\mathcal{G}$  of gene trees, other vectors  $\lambda'$  and  $\gamma'$  can be obtained such that

$$P(\mathcal{G}|N_{\lambda,\gamma}) = P(\mathcal{G}|N_{\lambda',\gamma'}), \quad (4.17)$$

by setting the branch lengths arbitrarily to  $t_1 = s'$ ,  $t_2 = p'$ ,  $t_3 = q'$ , and then setting the inheritance probabilities as follows

$$\alpha = -\frac{(e^{p'} - 1)(e^q - 1)abe^{p+q}}{(e^{q'} - 1)(e^{p+q} - be^{p+p'+q'} - e^{p'+q'} + be^{p'+q'})} \quad (4.18)$$

and

$$\beta = -\frac{(e^{p+q} - be^{p+p'+q'} - e^{p'+q'} + be^{p'+q'})e^{-(p+q)}}{e^{p'} - 1}. \quad (4.19)$$

For example, if I use  $p = 2.0$ ,  $q = 2.0$ ,  $a = 0.5$ ,  $b = 0.5$ ,  $p' = 1.7$ ,  $q' = 1.7$ , and then set  $\alpha = 0.9088149157446168$  and  $\beta = 0.29101947060819205$  (based on the above two formulas), then the same probability of any set of gene trees on the phylogenetic network of Scenario **II** in Fig. 4.2 can be obtained.

If I sample two alleles per species B (and a single or more alleles per each of the two species A and C), this lack of identifiability case disappears, since now the number of gene tree topologies is greater than the number of parameters being estimated. However, in practice, the value of  $t_1$  does affect the identifiability of the parameter values, since the larger it is, the higher the probability that the two alleles sampled from B would coalesce

and give a signal similar to that provided by a single allele. This point is illustrated by the results shown in Fig. 4.8.

To produce these results, I parameterized the phylogenetic network of Scenario **II** above with two different sets of values:

- network1:  $t_2 = t_3 = 2.0, \alpha = \beta = 0.5$ .
- network2:  $t_2 = t_3 = 1.7, \alpha = 0.9088149157446168$  and  $\beta = 0.29101947060819205$ .

As discussed above, the probability of each of the three gene tree topologies  $((a, b), c)$ ,  $((a, c), b)$ , and  $((b, c), a)$ , is the same under both networks. However, now consider the case where two alleles from B are sampled. In this case, there are 15 different gene tree topologies, which can be grouped into 9 categories, where all gene tree topologies within the same category have identical probabilities, regardless of the species phylogeny:

1.  $(b_2, ((b_1, c), a))$  and  $(b_1, ((b_2, c), a))$
2.  $(b_1, (c, (b_2, a)))$  and  $(b_2, (c, (b_1, a)))$
3.  $(c, (b_1, (b_2, a)))$  and  $(c, (b_2, (b_1, a)))$
4.  $((b_1, c), (b_2, a))$  and  $((b_2, c), (b_1, a))$
5.  $(a, (b_2, (b_1, c)))$  and  $(a, (b_1, (b_2, c)))$
6.  $(a, (c, (b_1, b_2)))$
7.  $(b_1, (b_2, (a, c)))$  and  $(b_2, (b_1, (a, c)))$
8.  $(c, (a, (b_1, b_2)))$

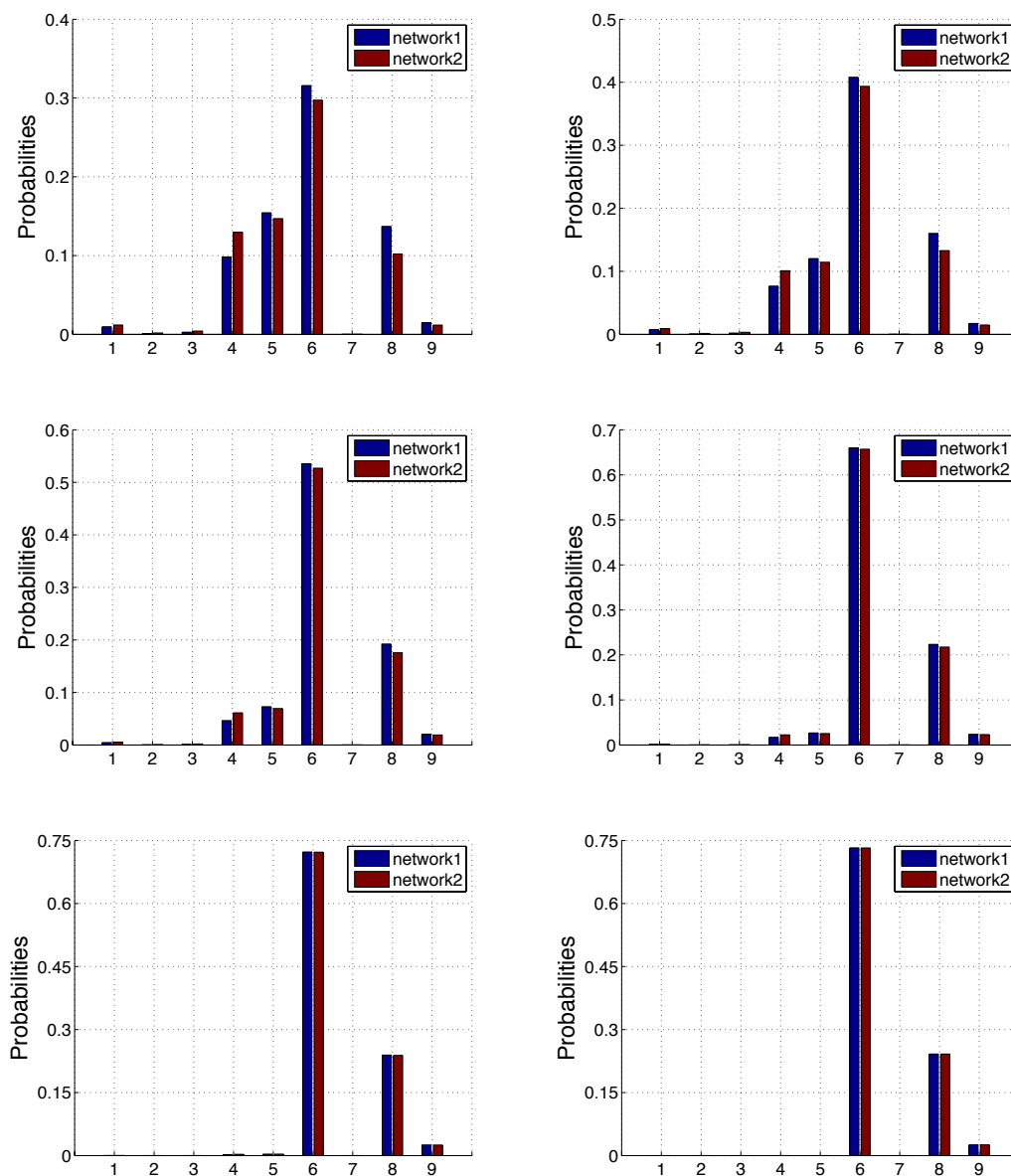


Figure 4.8: The probabilities of the 9 different gene tree topologies (when a single allele is sampled from each of two species A and C, and two alleles are sampled from species B) on the two phylogenetic networks obtained by parameterizing the values of  $\alpha$ ,  $\beta$ ,  $t_2$  and  $t_3$  differently for Scenario II; see text. Left to right, top to bottom:  $t_1 = 0.25, 0.5, 1.0, 2.0, 4.0,$  and  $8.0$ , respectively.



9.  $((b_1, b_2), (a, c))$ 

The probabilities of each of these 9 gene tree topologies (I choose one gene tree topology per category), as a function of the value of  $t_1$  are shown in Fig. 4.8.

Clearly, the two networks exhibit the gene tree topologies with different probabilities, when  $t_1 = 0.25$ . However, the gap between the probabilities starts closing as the value of  $t_1$  increases. When  $t_1 = 4.0$  or  $8.0$ , the gaps are too small to be even observed in any realistic data set (of a few thousand loci). At these branch lengths, the three topologies with non-negligible probabilities are the ones of categories 6, 8, and 9, which have the two alleles of B coalesce before either of them coalesce with alleles of the other two species.

In other words, while sampling two alleles from B help ameliorate the identifiability issue, a relatively large sample (in terms of the number of loci) needs to be used, and the time between hybridization and the subsequent divergence must not be too large, for methods to uniquely identify the parameter values.

Furthermore, in the special case where  $\alpha = 0.0$ , a phylogenetic tree, with appropriate branch lengths can be found, to fit the data exactly with the same probability that the phylogenetic network would. Consider the phylogenetic network  $N$  in Fig. 4.9(left), which reflects Scenario **II** in Fig. 4.2 in the case where  $\alpha = 0.0$ .

Let  $\lambda$  be the branch lengths vector with  $\lambda_1 \equiv t_1$ ,  $\lambda_2 \equiv t_2$ , and  $\lambda_3 \equiv t_3$ , and let  $\gamma$  be the hybridization probabilities vector with  $\gamma_1 \equiv \beta$ . Now, consider the phylogenetic tree  $T$  in Fig. 4.9(right). Then, if I set  $t$  as a function of  $\beta$ ,  $t_2$ , and  $t_3$ , as follows:

$$t(\beta, t_2, t_3) = -\ln(\beta e^{t_2} + 1 - \beta) + t_2 + t_3, \quad (4.20)$$



and

$$\frac{\partial t}{\partial t_3} = 1. \quad (4.23)$$

Clearly,

$$\lim_{t_2 \rightarrow \infty} \frac{\partial t}{\partial t_2} = 0. \quad (4.24)$$

Now consider the phylogenetic network of Scenario **III** in Fig. 4.2. In this case, both species involved in the hybridization are extinct. Surprisingly, the results in Fig. 4.11 show that if the correct (true) values of branch lengths is used, the inheritance probability  $\alpha$  is identifiable, and can be estimated with high accuracy as the number of gene trees sampled increases.

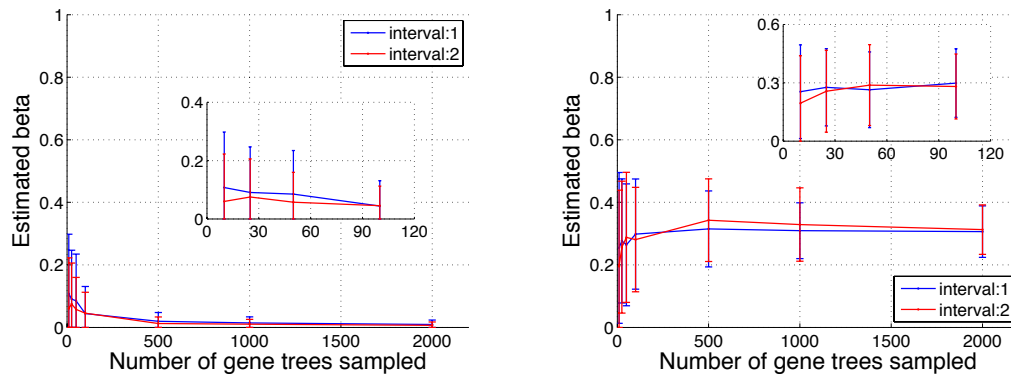


Figure 4.11: Estimates of  $\alpha$  on Scenario **III**. (Left)  $\alpha = 0.0$ ; (right)  $\alpha = 0.3$ .

However, if both branch lengths and inheritance probability are to be estimated, then issues of unidentifiability arise, as I now show. Let  $\lambda$  be the branch lengths vector with  $\lambda_1 \equiv t_1 = s$ ,  $\lambda_2 \equiv t_2 = p$ , and  $\lambda_3 \equiv t_3 = q$ , and let  $\gamma$  be the hybridization probabilities vector with  $\gamma_1 \equiv \alpha = a$ . For a given set  $\mathcal{G}$  of gene trees, other vectors  $\lambda'$  and  $\gamma'$  can be obtained such that

$$P_{N,\lambda,\gamma}(\mathcal{G}) = P_{N,\lambda',\gamma'}(\mathcal{G}), \quad (4.25)$$

by setting the inheritance probability arbitrarily to  $\alpha = a'$  and the branch lengths arbitrarily to  $t_1 = s'$ ,  $t_3 = q'$ , and

$$t_2 = -\ln \frac{2a'e^{p+q}(a'-1) + 2ae^{p+q'}(1-a) + 2ae^{q'}(a-1) + e^{q'}}{e^{q'}(2a'^2 + 1 - 2a')} + p + q - q'. \quad (4.26)$$

For example, if I use  $p = 1.0$ ,  $q = 2.0$ ,  $a = 0.8$ ,  $a' = 0.1$ ,  $q' = 1.8$ , and then set  $p' = 1.050498643$  (based on the above formula), the same probability of any set of gene trees on the phylogenetic network of Scenario **III** in Fig. 4.2 can be obtained.

Furthermore, a phylogenetic tree, with appropriate branch lengths can be found, to fit the data exactly with the same probability that the phylogenetic network would. Let  $\lambda$  be the branch lengths vector with  $\lambda_1 \equiv t_1$ ,  $\lambda_2 \equiv t_2$ , and  $\lambda_3 \equiv t_3$ , and let  $\gamma$  be the inheritance probabilities vector with  $\gamma_1 \equiv \alpha$ . Now, consider the phylogenetic tree  $T$  in Fig. 4.9(right). Then, if I set  $t$  as a function of  $\alpha$ ,  $t_2$ , and  $t_3$ , as follows:

$$t(\alpha, t_2, t_3) = -\ln(2\alpha^2 + 2\alpha e^{t_2} - 2\alpha^2 e^{t_2} + 1 - 2\alpha) + t_2 + t_3 \quad (4.27)$$

then,

$$P_{N,\lambda,\gamma}(\mathcal{G}) = P_{T,t}(\mathcal{G}) \quad (4.28)$$

for any set  $\mathcal{G}$  of gene trees. See Fig. 4.12 for values of  $t(\alpha, t_2, t_3)$ .

This result shows that as  $t_2$  increases, the value of  $t$  becomes unaffected by  $t_2$ , and that increasing  $t$  proportionally to the increase in  $t_3$  always maintains identical probabilities of gene trees under both the phylogenetic network of Scenario **III** and the phylogenetic tree in Fig. 4.9, as reflected by the derivatives:

$$\frac{\partial t}{\partial t_2} = 1 - \frac{1}{1 + \frac{1}{e^{t_2}} \left( \frac{1}{2\alpha(1-\alpha)} - 1 \right)} \quad (4.29)$$

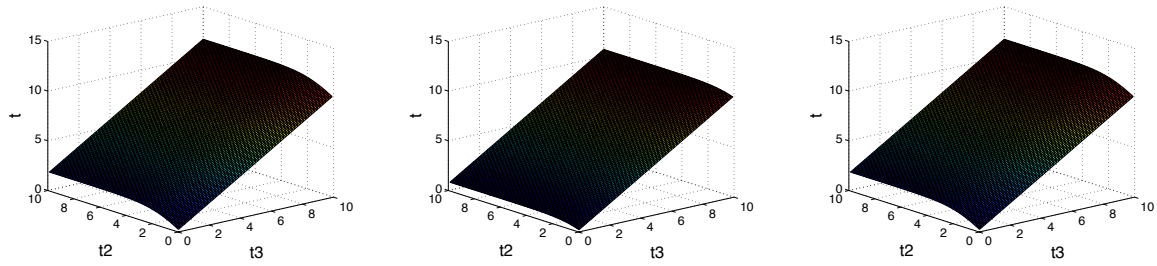


Figure 4.12: Values of  $t(\alpha, t_2, t_3)$  based on Equation (4.27); from left to right:  $\alpha = 0.1$ , 0.5, and 0.9, respectively.

and

$$\frac{\partial t}{\partial t_3} = 1. \quad (4.30)$$

Clearly,

$$\lim_{t_2 \rightarrow \infty} \frac{\partial t}{\partial t_2} = 0. \quad (4.31)$$

#### 4.4.1.2 Efficiency of the algorithms

##### 4.4.1.2.1 MUL-tree based method vs. AC based method

To study the efficiency of the AC based method compared to that of the MUL-tree based method for computing the probability of observing a gene tree topology given a phylogenetic network, I ran both methods on the same datasets I used for compare the efficiency between the AC based method and MUL-tree based method for computing the minimum number of extra lineages of a gene tree and a phylogenetic network in Section 3.4.1.4.1. All computations that could not finish in 8 hours were killed.

Part of the running times of the AC based algorithm are shown in Fig. 4.13. Only the results of the computations that could finish successfully in 8 hours across all loci were

plotted. We can see that the number of data points in the figure decreased significantly when the number of reticulation nodes in the species networks increased. In fact, out of 100 repetitions, the numbers of repetitions that finished the computations successfully across all different loci are 99, 96, 84, 54 and 32 for data sets containing species networks with 1, 2, 4, 6 and 8 reticulation nodes respectively. Those computations failed not only because of the 8 hours time limit. Part of them are due to memory issues: the number of configurations generated during the computation in order to cover all the possible coalescence patterns that could arise is huge. And the increase in the number of reticulation nodes in the species network might result in a very large increase in the number of configurations.

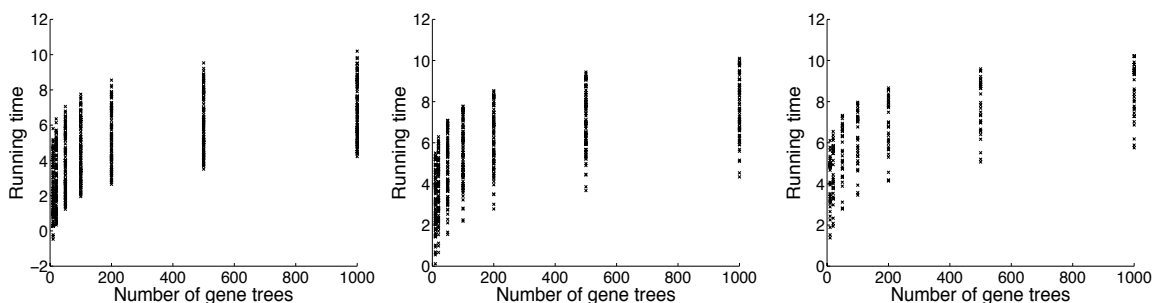


Figure 4.13: The running time (ln of number of seconds) of the AC based algorithm for computing the probability of gene tree topologies given a species network on the data sets described in Section 3.4.1.4.1. The columns from left to right correspond to data sets containing species networks with 1, 4 and 8 reticulation nodes, respectively.

On the other hand, for the MUL-tree based method, none of the computations finished within the time limit. Then I run the MUL-tree based method on a smallest dataset which contains only one gene tree and a species network with only one reticulation node for up to 24 hours, but still failed. In contrast, the AC based method only needed 0.4 seconds on the

same data set which implies a speedup of at least 5 orders of magnitude.

#### 4.4.1.2.2 Factors affecting the efficiency of the AC based method

From Fig. 4.13, we can see that the running time of the AC based method differed significantly from case to case, even for data sets of the same size (same size of gene tree sets and same reticulation nodes in the networks). I found that There are several factors that can affect the number of configurations generated during the computation which directly dominates the running time of the algorithm. Two of the factors that affect performance are the number of leaves under a reticulation node, as well as the topology of the gene tree. I considered a “controlled” data set in Fig. 3.14, the same one which I used to investigate the factors that affects the efficiency of the AC based method for parsimony approach (see Section 4.4.1.2.2 for a detailed description of this data set). I ran the AC based method on every pair of phylogenetic network and gene tree.

The results are listed in Table 4.3. First of all, for both  $g_1$  and  $g_2$ , the running time of the algorithm increased when there are more nodes under the reticulation nodes in the phylogenetic network, which is the same as we see for parsimony method (see Table 3.3). Furthermore, for the same network, the running time on  $g_2$  is always faster than that on  $g_1$ , which implied that gene trees all of whose coalescent events have to happen above the root result in shorter computation times than other gene trees. This is in stark contrast to the parsimony method where such gene trees take the longest running time (see Table 3.3). For the MUL-tree based method, the probability is computed by summing up the probabilities of all coalescent histories in MUL-tree under all allele mappings. However, for most

cases, the number of coalescent histories is much larger than the number of configurations generated [Wu12]. That is part of the reason why the AC-based algorithm outperforms the MUL-tree based algorithm for computing the probability in terms of efficiency.

Table 4.3: The results of running the AC based algorithm for computing the probability of gene tree topologies given gene trees and species networks in Fig. 3.14.  $|\mathcal{AC}_h|$  is the number of configurations at the reticulation node  $h$  and  $max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node  $v$  in post-order of traversal that contains the largest  $AC_v$  set is labeled by  $m$  in Fig. 3.14. Furthermore, the last column is the number of valid allele mappings if using the MUL-tree based method.

	$g_1$			$g_2$			#allele mappings
	$ \mathcal{AC}_h $	$max \mathcal{AC} $	running time (s)	$ \mathcal{AC}_h $	$max \mathcal{AC} $	running time (s)	
$N_1$	1	16	0.075	1	2	0.019	2
$N_2$	8	813	0.526	1	256 ( $2^8$ )	0.232	256
$N_3$	15	98286	617.845	1	32768 ( $2^{15}$ )	34.968	32768

To study the third factor that impacts performance which is the dependency of the reticulation nodes in the phylogenetic network (roughly, how many of them fall on a single path to the root), another “controlled” data set is considered, the one in Fig. 3.15. Again, I ran the AC based method on every pair of gene tree and phylogenetic network in the figure. The results are shown in Table 4.4. Basically, we see the same trend as the one we see in Section 3.4.1.4.2 for parsimony approach, where the more dependent to each other the reticulation nodes in the phylogenetic network are, the more ancestral configurations are generated during the computation, which means the more running time it needs.



Table 4.4: The results of running the AC based algorithm for computing the probability of gene tree topologies given gene trees and species networks in Fig. 3.15.  $|\mathcal{AC}_h|$  is the number of configurations at the highest reticulation node  $h$  and  $\max|\mathcal{AC}|$  is the maximum number of configurations generated at a node during computation. The first node  $v$  in post-order of traversal that contains the largest  $AC_v$  set is labeled by  $m$  in Fig. 3.15. Furthermore, the last column is the number of valid allele mappings if using the MUL-tree based method.

	$g_1$			$g_2$			#allele mappings
	$ \mathcal{AC}_h $	$\max \mathcal{AC} $	running time (s)	$ \mathcal{AC}_h $	$\max \mathcal{AC} $	running time (s)	
$N_1$	7	274	0.26	1	128 ( $2^7$ )	0.124	268435456
$N_2$	9928	146433	1336.494	5040	40320	57.418	40320

#### 4.4.2 Reanalysis of a yeast (*Saccharomyces*) data set

I reanalyzed the yeast data set of [RWKC03b] using the method described in Section 4.1.1 [YDN12]. The data set consists of 106 loci, each with a single allele sampled from seven *Saccharomyces* species *S. cerevisiae* (*Scer*), *S. paradoxus* (*Spar*), *S. mikatae* (*Smik*), *S. kudriavzevii* (*Skud*), *S. bayanus* (*Sbay*), *S. castellii* (*Scas*), *S. kluyveri* (*Sklu*), and the outgroup fungus *Candida albicans* (*Calb*) (see Section 3.4.2 for the analysis using parsimony approach on the same data set). Given that there is no indication of coalescences deeper than the MRCA of *Scer*, *Spar*, *Smik*, *Skud*, and *Sbay* [TN09], I focused only on the evolutionary history of these five species. I inferred gene trees using Bayesian inference in MrBayes [HR01] and using maximum parsimony with strict consensus in PAUP\* [Swo96].

The species tree that has been reported for these five species, based on the 106 loci,

is shown in Fig. 4.14A [RWKC03b]. Further, additional studies inferred the tree in Fig. 4.14B as a very close candidate for giving rise to the 106 gene trees, under the coalescent model [ELP07a, TN09]. Notice that the difference between the two trees is the placement of *Skud*, which flags hybridization as a possibility. Indeed, the phylogenetic network topologies in Fig. 4.14C-D have been proposed as an alternative evolutionary history, under the stochastic framework of [BS10], as well as the parsimony framework of [YTDN11].

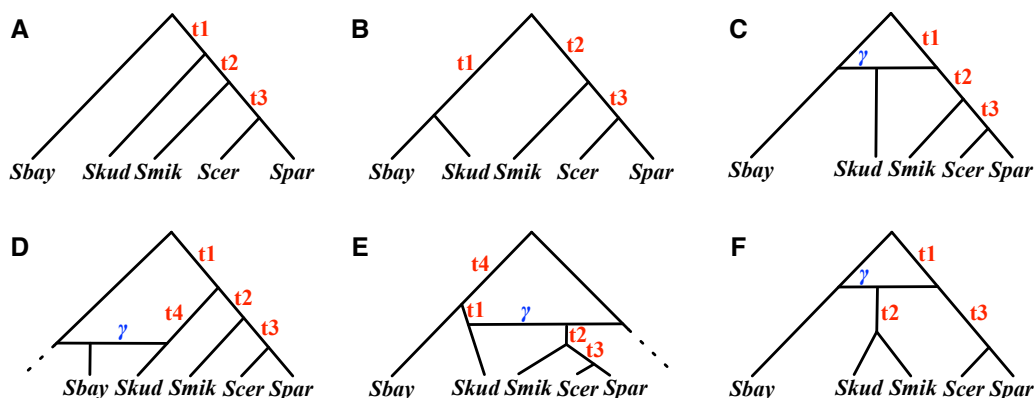


Figure 4.14: Various hypotheses for the evolutionary history of a yeast data set. (A) The species tree for the five species *Sbay*, *Skud*, *Smik*, *Scer*, and *Spar*, as proposed in [RWKC03b], and inferred using a Bayesian approach [ELP07a] and a parsimony approach [TN09]. (B) A slightly suboptimal tree for the five species, as identified in [ELP07a, TN09]. (C)—(E) The three phylogenetic networks that reconcile both trees in (A) and (B), and which we reported as equally optimal evolutionary histories under a parsimony criterion in [YTDN11]. (F) A phylogenetic network that postulates *Smik* and *Skud* as two sister taxa whose divergence followed a hybridization event.

Using the 106 gene trees, I estimated the times  $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$  and  $\gamma$  for the six phylogenies in Fig. 4.14 that maximize the likelihood function (I used a grid search of values between

0.05 and 4, with step length of 0.05 for branch lengths, and values between 0 and 1 with step length of 0.01 for  $\gamma$ ). Table. 4.5 lists the values of the parameters computed using Eq. 4.13 on the gene trees inferred by MrBayes and Table. 4.6 lists the values of the parameters computed using Eq. 4.14 on the gene trees inferred by PAUP\*, as well as the values of three information criteria, AIC [Aka74], AICc [BA02] and BIC [Sch78], in order to account for the number of parameters and allow for model selection.

Table 4.5: Parameter values estimated for the six phylogenies in Fig. 4.14, as well as the values of three information criteria, using gene tree topologies inferred by a Bayesian analysis (using MrBayes).

Species phylogeny	$t_1$	$t_2$	$t_3$	$t_4$	$\gamma$	$-\ln L$	AIC	AICc	BIC
Fig. 4.14A	0.05	0.85	2.05	N/A	N/A	284	575	576	583
Fig. 4.14B	0.2	0.85	2.05	N/A	N/A	276	559	560	567
Fig. 4.14C	0.4	0.65	2.05	N/A	0.59	274	556	556	567
Fig. 4.14D	2.95	0.7	2.1	0.85	0.5	247	504	504	517
Fig. 4.14E	0.6	0.05	2.05	0.2	0.0	276	563	564	577
Fig. 4.14F	0.9	0.05	2.15	N/A	0.27	325	659	659	669

Out of the 106 gene trees (using either of the two inference methods), roughly 100 trees placed *Scer* and *Spar* as sister taxa, which potentially reflects the lack of deep coalescence involving this clade (and is reflected by the relatively large  $t_3$  values estimated). Roughly 25% of the gene trees did not show monophyly of the group *Scer*, *Spar*, and *Smik*, thus indicating a mild level of deep coalescence involving these three species (and reflected by

Table 4.6: Parameter values estimated for the six phylogenies in Fig. 4.14, as well as the values of three information criteria, using gene tree topologies inferred by maximum parsimony (using PAUP\*).

Species phylogeny	$t_1$	$t_2$	$t_3$	$t_4$	$\gamma$	$-\ln L$	AIC	AICc	BIC
Fig. 4.14A	0.3	1.25	3.6	N/A	N/A	205	416	417	424
Fig. 4.14B	0.2	1.35	3.6	N/A	N/A	208	423	423	431
Fig. 4.14C	1.1	1.05	3.6	N/A	0.34	188	384	385	395
Fig. 4.14D	3.45	1.15	3.6	3.05	0.34	157	325	326	338
Fig. 4.14E	0.3	1.25	3.6	N/A	1.0	205	420	421	434
Fig. 4.14F	1.55	0.05	3.7	N/A	0.18	252	512	512	523

the relatively small  $t_2$  values estimated). However, a large proportion of the 106 gene trees indicated incongruence involving *Skud*. This pattern is reflected by the very low estimates of the time  $t_1$  on the two phylogenetic trees in Fig. 4.14. On the other hand, analysis under the phylogenetic network models of Fig. 4.14C-D indicates a larger divergence time, with substantial extent of hybridization. These latter hypotheses naturally result in a better likelihood score. When accounting for model complexity, all three information criteria indicated that these two phylogenetic network models with extensive hybridization and larger divergence time between *Sbay* and the (*Smik*,(*Scer*,*Spar*)) clade provide better fit for the data. Further, while both networks produced identical hybridization probabilities, the network in Fig. 4.14D had much lower values of the information criteria than those of the network in Fig. 4.14E. The networks in Fig. 4.14E-F have lower support (under all

measures) than the other four phylogenies. In summary, our analysis gives higher support for the hypothesis of extensive hybridization, a low degree of deep coalescence, and long branch lengths than to the hypothesis of a species tree with short branches and extensive deep coalescence. It is worth mentioning that while the three networks in Fig. 4.14C-E were reported as equally optimal under a parsimonious reconciliation [TN09], my new framework can distinguish among the three, and identifies the network in Fig. 4.14D as best, followed by the one in Fig. 4.14C (the network of Fig. 4.14E is found to be a worse fit than either of the two species tree candidates).

#### 4.4.3 Analysis of a house mouse (*Mus musculus*) data set

I used this maximum likelihood approach to analyze a data set of house mouse (*Mus musculus*) genomes [YDLN14]. In this data set, I used two *Mus musculus domesticus* samples from [SLM<sup>+</sup>12b], which represent one population from France (in the Massif Central) and another population from Germany (in the vicinity surrounding Cologne and Bonn). I also used three *Mus musculus musculus* samples obtained from [SLM<sup>+</sup>12b, DYS<sup>+</sup>12, YWD<sup>+</sup>11], and which represent a population in Czechoslovakia (Studenec) [SLM<sup>+</sup>12b], another population in Kazakhstan (Almaty) [SLM<sup>+</sup>12b], and a third population from China (Urumqi in Xinjiang Province) [DYS<sup>+</sup>12, YWD<sup>+</sup>11].

Staubach *et al.* [SLM<sup>+</sup>12b] found substantial genome-wide evidence of subspecific introgression in all four populations, amounting to 3% of the genome in the two *M. m. d.* populations (one from France and the other from Germany), 4% in an *M. m. m.* population from Kazakhstan, and 18% of an *M. m. m.* population from the Czech Republic. However,

it is important to note that the method HAPMIX [PTP<sup>+</sup>09], which was used in [SLM<sup>+</sup>12b], does not explicitly account for ILS.

My study included all of the samples in the study of [SLM<sup>+</sup>12b]. Furthermore, My study included additional samples from an *M. m. m.* population from China [YWD<sup>+</sup>11] that were not used in the study of [SLM<sup>+</sup>12b]. In total, 20,639 local phylogenies were used (see how gene trees were reconstructed from genome-wide sequence data in [YDLN14]) in the analysis. From the reconstructed gene trees, I inferred the optimal phylogenetic networks with 0, 1, 2 and 3 reticulation nodes, respectively, using the method described in Section 4.1.1, 4.3.1.1 and 3.3.2 (only topologies of gene trees were used). For each of them, the search was run 50 times and top 5 networks were saved. All other parameters were set to their default values as listed in Section 5. Since all five populations under analysis are closely related, about 40% of the reconstructed local trees were partially, not fully, resolved. When likelihood scores of phylogenetic networks are calculated using these trees, I used Eq. 4.15 to account for uncertainties. The inferred networks are shown in Fig. 4.15.

Furthermore, to account for model complexity, I calculated the values of three information criteria, AIC, AICc and BIC, as well as the error of cross-validation, for the optimal inferred networks with the number of reticulation nodes from 0 to 3 respectively. More specifically, I did 10-fold cross-validation and only binary gene trees in the validation sets were used to calculate the error. The results are given in Table 4.7.

We can see that there is a significant improvement in a phylogenetic network with a single reticulation over no reticulations, a significant improvement in a phylogenetic network with two reticulations over a single reticulation for both three information criteria and

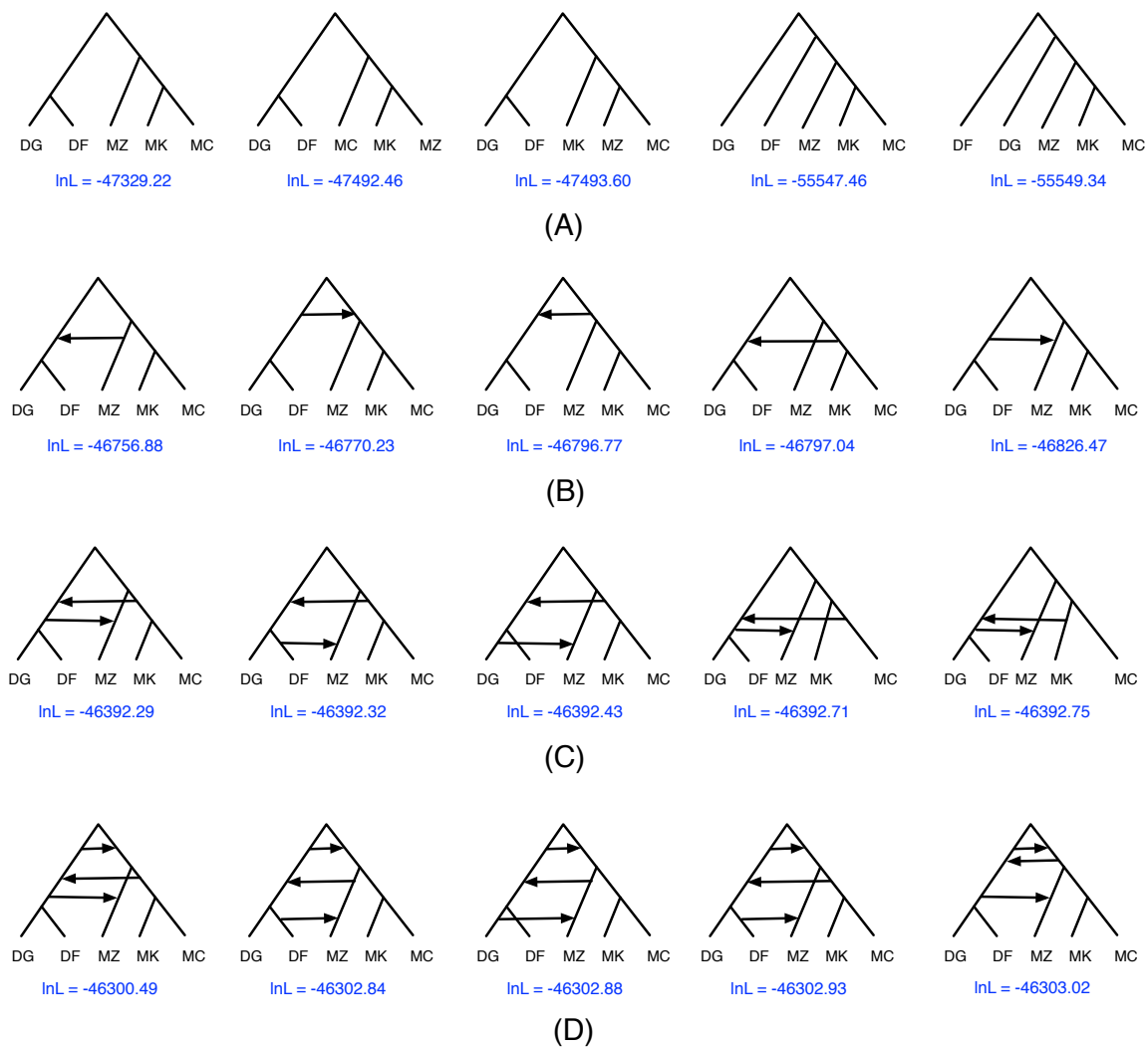


Figure 4.15: The inferred phylogenetic networks of the *M. musculus* dataset. The rows from top to bottom contain top 5 phylogenetic networks with 0, 1, 2 and 3 reticulation nodes, respectively. In each row, networks are listed from left to right with an decreasing value of log likelihood shown under each of them.

cross-validation. However, when I continued the search for the optimal network with three reticulations, I found that the improvement gained by considering a third reticulation event was insignificant based on the information criteria, and that there was no improvement

Table 4.7: The results of information criteria and cross validation of the optimal inferred species networks of the *M. musculus* dataset.  $N(k)$  refers to the optimal inferred species network with  $k$  reticulation nodes.

	lnL	AIC	AICc	BIC	Error of cross-validation
$N(0)$	-47329	94664	94664	94688	$7.69 \times 10^{-5}$
$N(1)$	-46756	93527	93527	93583	$5.36 \times 10^{-5}$
$N(2)$	-46392	92806	92806	92893	$4.03 \times 10^{-5}$
$N(3)$	-46300	92635	92635	92754	$4.13 \times 10^{-5}$

at all based on cross-validation. I thus called the optimal phylogenetic network with two reticulations as my hypothesis for the evolutionary history of this set of genomes. Since the likelihood scores of top five networks with 2 reticulations are all very close (see Fig. 4.15C), they can be somehow summarized as shown in Fig. 4.16, with the inferred branch lengths and inheritance probabilities. The phylogenetic network is not ultrametric, and it is worth emphasizing that the branch lengths are given in coalescent units. Thus, the lack of ultrametricity could be due to different population sizes or, to a lesser degree, different generation times.

The results I obtained differ from [SLM<sup>+</sup>12b] not only in terms of the number of populations involved, but also by accounting for the evolutionary history of the populations involved. I consider the percentages of the genome with introgressed origin reported by [SLM<sup>+</sup>12b] to be over-estimates since introgression involving an ancestral population that later split into more than one extant population would be multiply reported for each extant population in the case of [SLM<sup>+</sup>12b]. On the other hand, the same percentages would



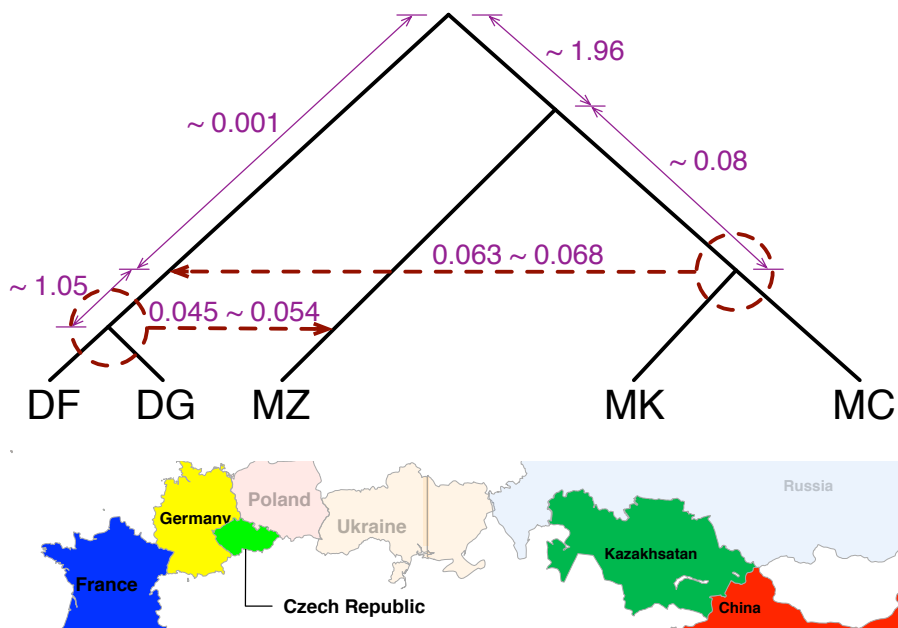


Figure 4.16: The optimal phylogenetic network inferred on the house mouse (*Mus musculus*) data set. A single individual was sampled from each of five populations: *M.m. domesticus* from France (DF), *M.m. domesticus* from Germany (DG), *M.m. musculus* from the Czech Republic (MZ), *M.m. musculus* from Kazakhstan (MK), and *M.m. musculus* from China (MC). The analysis found multiple, almost equally optimal, phylogenetic networks with two reticulation events. These multiple networks all agreed on the recipient populations, but disagreed on the donor populations. One hybridization (the top dashed horizontal arrow) involves the MRCA of DF and DG as a recipient population, yet seems to have involved MK, MC, or their MRCA as the donor population. The second hybridization (the bottom dashed horizontal arrow) involves MZ as a recipient population, yet seems to have involved DF, DG, or their MRCA as the donor population. Branch lengths in coalescent units (on the tree branches) and inheritance probabilities (on the horizontal edges) are shown.

be under-estimated in the case where admixed populations were used in place of the non-admixed reference populations required by HAPMIX, as [SLM<sup>+</sup>12b] did by using putatively introgressed mouse samples to construct the reference populations. Notably, my new methodology does not require the use of non-admixed reference populations.

I hypothesize that the more recent introgression event in Figure 4.16 is due to gene flow from secondary contact where the ranges of the two *M. musculus* subspecies overlapped, roughly at the border between Germany and the Czech Republic. The biological interpretation of the more ancient introgression event is less clear. I conjecture that the event is related to gene flow during and after subspecific divergence. Further study may provide important clues to the mechanistic basis of the evolution of subspecies in *M. musculus* and the process of speciation itself.

## 4.5 Parametric Bootstrap

With the increasing interest in reconstruction of phylogenetic trees, in order to evaluate how confident one should be in a reconstructed phylogeny, bootstrapping has been widely used for decades since it was first proposed as a method for obtaining confidence limits on phylogenies [Fel85]. Here, I employ parametric bootstrap evaluate the confidence of the edges in an inferred phylogenetic network.

In Fig. 4.17, I illustrate given a collection of gene trees how to infer a phylogenetic network with parametric bootstrap. Basically, after a phylogenetic network  $N$  is inferred from the original set of gene tree  $\mathcal{G}$ , within the branches of  $N$  I first simulate  $k$  sets of gene trees independently, each of which has the same size as  $\mathcal{G}$ , where  $k$  is some pre-

specified number which should be large enough to get a precise estimate of bootstrap value (in PhyloNet, the default value of  $k$  is 100). Then from each simulated set of gene trees, a phylogenetic network is inferred using the same method and settings as the one used to obtain the original phylogenetic network  $N$  from gene trees  $\mathcal{G}$ . Finally, by comparing the  $k$  inferred phylogenetic networks with  $N$ , I am able to obtain the support of every edge in  $N$ .

This parametric bootstrap works for maximum likelihood (ML) inference using both gene trees with and without branch lengths. In PhyloNet, when only the topologies of gene trees were used, I implemented my own simulator to generate topologies of gene trees from a given phylogenetic network. And when both the topologies and branch lengths of gene trees were needed, an external software `ms` [Hud02] was called to simulate gene trees with branch lengths.

The bootstrap value of an edge in the inferred phylogenetic network  $N$  is calculated as the proportion of networks in  $N_1, \dots, N_k$  that contain the same edge. I consider edge  $b_1$  in network  $N_1$  and edge  $b_2$  in network  $N_2$  to be the same if they satisfy the following two conditions:

- $b_1$  and  $b_2$  induce the same set of softwired clusters [HRS10],
- $b_1$  and  $b_2$  are either both tree edges or both reticulation edges.

The materials in this section are from [YDLN14].

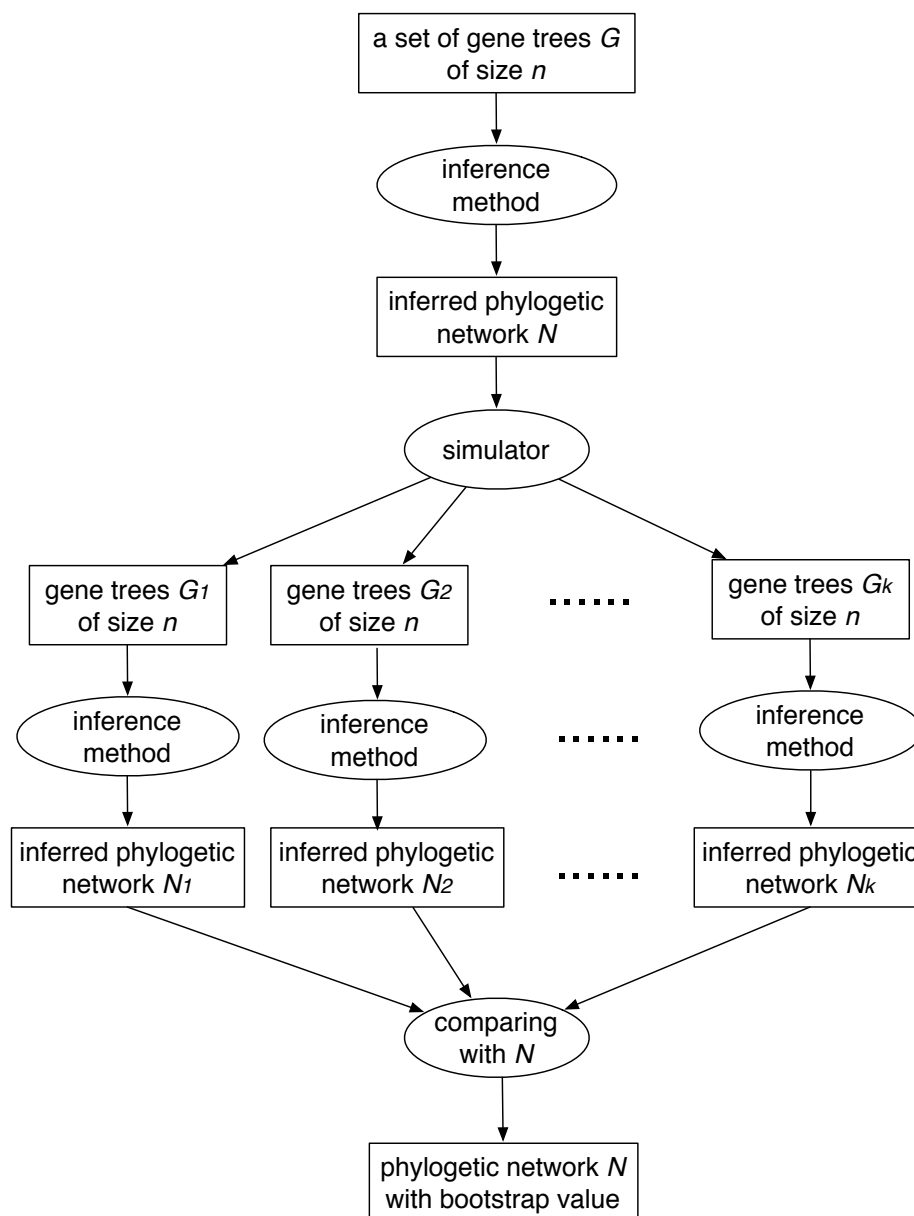


Figure 4.17: Illustration of parametric bootstrap to assess the significance of the edges in the inferred phylogenetic network.

## Chapter 5

# Usage of PhyloNet to infer phylogenetic networks

I have implemented all methods discussed in this thesis in PhyloNet [TRN08], which is an open-source software package for phylogenetic network inference and analysis.

The *inferNetwork\_ML* functionality infers a phylogenetic network from a collection of gene trees. It takes a collection of gene trees and the maximum number of reticulations and returns optimal inferred phylogenetic networks along with branch lengths and inheritance probabilities. There are many parameters for the users to specify; See Table 5.1 for details.

The materials in this section are from [YDLN14].

Table 5.1: The usage of command *inferNetwork\_ML* in PhyloNet. The first two parameters are mandatory and all others are optional.

InferNetwork_ml (gt1 [, gt2...]) numReticulations [-a taxaMap] [-bl] [-b threshold] [-s startingNetwork] [-n numNetReturned] [-h {s1 [, s2...]}] [-w (w1,w2,w3,w4)] [-f maxFailure] [-x numRuns] [-m maxNetExamined] [-d maxDiameter] [-p (rel,abs)] [-r maxRounds] [-t maxTryPerBr] [-i improveThreshold] [-l maxBL] [-pl numProcessors] [-di]		
Parameter	Illustration	Default
( <i>gt</i> <sub>1</sub> [, <i>gt</i> <sub>2</sub> . . .])	Comma delimited list of gene tree identifiers.	-
<i>numReticulations</i>	Maximum number of reticulations to add to the species network.	-
-a <i>taxaMap</i>	Gene tree / species network taxa association.	-
-bl	Use the branch lengths of the gene trees for the inference.	No
-b <i>threshold</i>	Gene trees bootstrap threshold. Edges of gene trees whose bootstrap values are under it will be contracted.	100
-s <i>startingNetwork</i>	The network to start search from.	MDC tree
-n <i>numNetReturned</i>	Number of top optimal networks to return.	1
-h { <i>s</i> <sub>1</sub> [, <i>s</i> <sub>2</sub> . . .]}	A set of specified hybrid species. The size of this set equals the number of reticulation nodes in the inferred network.	-
-w ( <i>w</i> <sub>1</sub> , <i>w</i> <sub>2</sub> , <i>w</i> <sub>3</sub> , <i>w</i> <sub>4</sub> )	The weights of operations ( $\delta_1$ , $\delta_2$ , $\delta_3$ , $\delta_4$ ) for network arrangement during the network search.	(0.15, 0.15, 0.2, 0.5)
-f <i>maxFailure</i>	The maximum number of consecutive failures before the search terminates.	100
-x <i>numRuns</i>	The number of runs of the search.	10
-m <i>maxNetExamined</i>	Maximum number of network topologies to examine during the search in each run.	$+\infty$
-d <i>maxDiameter</i>	Maximum diameter to make an rearrangement during network search.	$+\infty$
-p ( <i>rel</i> , <i>abs</i> )	The original stopping criterion of Brents algorithm for optimizing branch lengths and inheritance probabilities of a network.	(0.01, 0.001)
-r <i>maxRound</i>	Maximum number of rounds to optimize branch lengths and inheritance probabilities for a network topology.	100
-t <i>maxTryPerBr</i>	Maximum number of trial per branch in one round to optimize branch lengths and inheritance probabilities for a network topology.	100
-i <i>improveThreshold</i>	Minimum threshold of improvement to continue the next round of optimization of branch lengths and inheritance probabilities.	0.001
-l <i>maxBL</i>	Maximum branch lengths considered during optimization.	6
-pl <i>numProcessors</i>	Number of processors if you want the computation to be done in parallel.	1
-di	Output the Rich Newick string of the inferred network that can be read by Dendroscope [HS12].	No

# Chapter 6

## Conclusions and future work

In this work, I devised the first parsimony and likelihood criteria for the inference of phylogenetic networks in the presence of ILS, along with new algorithms for the inference. Both methods are general enough to allow for multiple hybridizations, multiple alleles per species, and arbitrary divergence patterns following hybridization. For each of them, I proposed a way of handling uncertainty in gene tree topologies when gene trees are estimated from sequences. Furthermore, for the likelihood approach, I used information criteria and cross-validation to account for the model selection issue. Also, I employed parametric bootstrap to evaluate the confidence of the inferred species phylogenies.

I studied the performance of the algorithms in extensive simulation studies. For the likelihood approach, it showed very good performance in terms of identifying the location of hybridization events, as well as estimating the proportions of genes inherited through hybridization. I also discussed the identifiability of phylogenetic networks from gene tree topologies. As for the parsimony approach, it also estimated accurate phylogenetic network topologies along with inheritance probabilities, and in addition, it showed good per-

formance in terms of efficiency of handling large data sets in the experiments. Furthermore, for both likelihood and parsimony approaches, I investigated the factors (i.e., the topologies of gene trees and species networks) that affect the efficiencies of the algorithms.

My work allows, for the first time, systematic phylogenomic analyses of data sets where hybridization is suspected. Thus, it allows us now to revisit existing analyses and conduct new ones with richer evolutionary models and inference methods. In particular, using the new methods, I reanalyzed two biological data sets, a data sets of yeast (*Saccharomyces*) genomes and another of house mouse (*Mus musculus*) genomes, and found support for hybridization in both of them.

Finally, I have implemented all the algorithms in our open-source, publicly available PhyloNet software package.

## 6.1 Future work

I now discuss four main directions for future work.

First is to improve the scalability of the methods to larger data sets. Nowadays, with the improvement of the DNA sequencing techniques, more and more data sets are being available to biologists. Our work provides, for the first time, a general framework for inferring species phylogenies in the presence of both hybridization and incomplete lineage sorting. However, in practice, it is not feasible for our methods to deal with large data sets, say data sets with hundreds of taxa. There are two main challenges here. One is the huge space of the phylogenetic networks during the search. A straightforward idea is to narrow the search space, for example by using biological and geographical knowledge about the



species. The second challenge is that evaluating a candidate network is time-consuming, especially for the probabilistic approach in which computing the likelihood of the network requires optimizing the branch lengths and inheritance probabilities of that network. An alternative way is to use heuristics instead of doing exact computation. By solving these two challenges, our methods could scale up to larger data sets.

Another problem is to relax the two assumptions of our methods. First assumption is that each locus is independent, which is a very common assumption in this area and almost all methods for inferring species phylogenies are developed under this assumption. To ensure this, loci are usually sampled far away from each other such that they can be considered independent. However, since independence is not guaranteed, it is important that dependence among loci can be taken into account when species phylogenies are inferred from them. Another assumption is that gene trees have already been estimated from sequences, which allows us to focus only on maximizing the probability of observing gene trees. This assumption is made assuming that gene trees were reconstructed without error, which can be avoided. So it is important that we compute the likelihood of a species phylogeny directly from gene sequences. Maddison [Mad97] gave the formula as follows

$$\prod_{\text{loci}} \sum_{\text{possible gene trees}} [P(\text{sequences}|\text{gene tree}) \cdot P(\text{gene tree}|\text{species phylogeny})]. \quad (6.1)$$

Now we are not assuming that gene trees have been reconstructed, so, for every locus, every possible gene tree must be considered. For each of them,  $P(\text{sequences}|\text{gene tree})$  and  $P(\text{gene tree}|\text{species phylogeny})$  are calculated separately. The former one can be computed by [Fel81], and we have already devised methods for the latter. Therefore, to have the complete likelihood model, which does the computation from gene sequences, it is just a

matter of of implementation.

Last but not least, our methods infer species phylogenies assuming hybridization and incomplete lineage sorting are the only two factors that cause gene tree incongruence. However, other factors may be at play, like gene duplication and loss. So it is important to go beyond these two.

# Bibliography

- [Aka74] H. Akaike, *A new look at the statistical model identification.*, IEEE Trans Automat Contr **19** (1974), 716–723.
- [BA02] K.P. Burnham and D.R. Anderson, *Model selection and multi-model inference: a practical-theoretic approach.*, 2nd ed., Springer Verlag, New York, 2002.
- [Bre73] R.P. Brent, *Algorithms for minimization without derivatives*, Prentice- Hall, Englewood Clifts, New Jersey, 1973.
- [BS10] E.W. Bloomquist and M.A. Suchard, *Unifying vertical and nonvertical evolution: A stochastic ARG-based framework*, Systematic Biology **59** (2010), no. 1, 27–41.
- [BvIJ<sup>+</sup>13] Eric Bapteste, Leo van Iersel, Axel Janke, Scot Kelchner, Steven Kelk, James O McInerney, David A Morrison, Luay Nakhleh, Mike Steel, Leen Stougie, and James Whitefield, *Networks: expanding evolutionary thinking*, Trends in Genetics **29** (2013), no. 8, 439–441.

- [CHW<sup>+</sup>09] K. A. Cranston, B. Hurwitz, D. Ware, L. Stein, and R. A. Wing, *Species trees from highly incongruent gene trees in rice*, *Syst. Biol.* **58** (2009), 489–500.
- [DR06] J. H. Degnan and N. A. Rosenberg, *Discordance of species trees with their most likely gene trees*, *PLoS Genet.* **2** (2006), 762–768.
- [DR09] J.H. Degnan and N.A. Rosenberg, *Gene tree discordance, phylogenetic inference and the multispecies coalescent*, *Trends in Ecology and Evolution* **24** (2009), no. 6, 332–340.
- [DS05a] J. H. Degnan and L. A. Salter, *Gene tree distributions under the coalescent process*, *Evolution* **59** (2005), 24–37.
- [DS05b] J.H. Degnan and L.A. Salter, *Gene tree distributions under the coalescent process*, *Evolution* **59** (2005), 24–37.
- [DYS<sup>+</sup>12] John Didion, Hyuna Yang, Keith Sheppard, Chen-Ping Fu, Leonard McMillan, Fernando de Villena, and Gary Churchill, *Discovery of novel variants in genotyping arrays improves genotype retention and reduces ascertainment bias*, *BMC Genomics* **13** (2012), no. 1, 34.
- [ELP07a] S. V. Edwards, L. Liu, and D. K. Pearl, *High-resolution species trees without concatenation*, *PNAS* **104** (2007), 5936–5941.
- [ELP07b] ———, *High-resolution species trees without concatenation*, *Proc. Natl. Acad. Sci. U. S. A* **104** (2007), 5936–5941.

- [EM12] Anders Eriksson and Andrea Manica, *Effect of ancient population structure on the degree of polymorphism shared between modern human populations and ancient hominins*, *Proceedings of the National Academy of Sciences* **109** (2012), no. 35, 13956–13960.
- [Fel81] J. Felsenstein, *Evolutionary trees from gene frequencies and quantitative characters: finding maximum likelihood estimates*, *Evolution* **35** (1981), 1229–1242.
- [Fel85] ———, *Confidence limits on phylogenies: an approach using the bootstrap*, *Evolution* **39** (1985), 783–791.
- [GKB<sup>+</sup>10] Richard E. Green, Johannes Krause, Adrian W. Briggs, Tomislav Maricic, Udo Stenzel, Martin Kircher, Nick Patterson, Heng Li, Weiwei Zhai, Markus Hsi-Yang Fritz, Nancy F. Hansen, Eric Y. Durand, Anna-Sapfo Malaspinas, Jeffrey D. Jensen, Tomas Marques-Bonet, Can Alkan, Kay Prüfer, Matthias Meyer, Hernn A. Burbano, Jeffrey M. Good, Rigo Schultz, Ayinuer Aximu-Petri, Anne Butthof, Barbara Hber, Barbara Hffner, Madlen Siegemund, Antje Weihmann, Chad Nusbaum, Eric S. Lander, Carsten Russ, Nathaniel Novod, Jason Affourtit, Michael Egholm, Christine Verna, Pavao Rudan, Dejana Brajkovic, eljko Kucan, Ivan Guic, Vladimir B. Doronichev, Liubov V. Golovanova, Carles Lalueza-Fox, Marco de la Rasilla, Javier Fortea, Antonio Rosas, Ralf W. Schmitz, Philip L. F. Johnson, Evan E. Eichler, Daniel Falush, Ewan Birney, James C. Mullikin, Montgomery Slatkin, Rasmus Nielsen,

- Janet Kelso, Michael Lachmann, David Reich, and Svante Pbo, *A draft sequence of the Neandertal genome*, *Science* **328** (2010), no. 5979, 710–722.
- [HDH<sup>+</sup>11] A. Hobolth, J. Dutheil, J. Hawks, M. Schierup, and T. Mailund, *Incomplete lineage sorting patterns among human, chimpanzee, and orangutan suggest recent orangutan speciation and widespread selection*, *Genome Research* **21** (2011), 349356.
- [HOLM06] K.T. Huber, B. Oxelman, M. Lott, and V. Moulton, *Reconstructing the evolutionary history of polyploids from multilabeled trees*, *Molecular Biology and Evolution* **23** (2006), no. 9, 1784–1791.
- [HR01] J. P. Huelsenbeck and F. Ronquist, *MRBAYES: Bayesian inference of phylogenetic trees*, *Bioinformatics* **17** (2001), 754–755.
- [HRS10] D.H. Huson, R. Rupp, and C. Scornavacca, *Phylogenetic networks: Concepts, algorithms and applications*, Cambridge University Press, New York, 2010.
- [HS12] D.H. Huson and C. Scornavacca, *Dendroscope 3: an interactive tool for rooted phylogenetic trees and networks.*, *Systematic Biology* **61** (2012), 1061–7.
- [Hud83] R. R. Hudson, *Testing the constant-rate neutral allele model with protein sequence data*, *Evolution* **37** (1983), 203–217.
- [Hud02] ———, *Generating samples under a Wright-Fisher neutral model of genetic variation*, *Bioinformatics* **18** (2002), 337–338.

- [JML09] S. Joly, P. A. McLenachan, and P. J. Lockhart, *A statistical approach for distinguishing hybridization and incomplete lineage sorting*, *Am. Nat.* **174** (2009), no. 2, E54–E70.
- [JSO12] G. Jones, S. Sagitov, and B. Oxelman, *Statistical inference of allopolyploid species networks in the presence of incomplete lineage sorting*, arXiv (2012), 1208.3606.
- [Kin82a] J. F. C. Kingman, *The coalescent*, *Stochast. Proc. Appl.* **13** (1982), 235–248.
- [Kin82b] ———, *On the genealogy of large populations*, *J. Appl. Prob.* **19A** (1982), 27–43.
- [Kub09] L. S. Kubatko, *Identifying hybridization events in the presence of coalescence via model selection*, *Syst. Biol.* **58** (2009), no. 5, 478–488.
- [KWK08] Chih-Horng Kuo, John P. Wares, and Jessica C. Kissinger, *The Apicomplexan whole-genome phylogeny: An analysis of incongruence among gene trees*, *Mol. Biol. Evol.* **25** (2008), no. 12, 2689–2698.
- [LYK<sup>+</sup>09] L. Liu, L. L. Yu, L. Kubatko, D. K. Pearl, and S. V. Edwards, *Coalescent methods for estimating phylogenetic trees*, *Mol. Phylogenet. Evol.* **53** (2009), 320–328.
- [Mad97] W. P. Maddison, *Gene trees in species trees*, *Syst. Biol.* **46** (1997), 523–536.

- [MK09] C. Meng and L. S. Kubatko, *Detecting hybrid speciation in the presence of incomplete lineage sorting using gene tree incongruence: A model*, *Theor. Popul. Biol.* **75** (2009), no. 1, 35–45.
- [MR12] M.L. Moody and L.H. Rieseberg, *Sorting through the chaff, nDNA gene trees for phylogenetic inference and hybrid identification of annual sunflowers (*Helianthus sect Helianthus*)*, *Molecular Phylogenetics And Evolution* **64** (2012), 145–155.
- [Nak10] L. Nakhleh, *Evolutionary phylogenetic networks: models and issues*, *The Problem Solving Handbook for Computational Biology and Bioinformatics* (L. Heath and N. Ramakrishnan, eds.), Springer, New York, 2010, pp. 125–158.
- [Nak13] Luay Nakhleh, *Computational approaches to species phylogeny inference and gene tree reconciliation*, *Trends in Ecology & Evolution* **28** (2013), no. 12, 719–728.
- [Nei86] M. Nei, *Stochastic errors in DNA evolution and molecular phylogeny*, *Evolutionary Perspectives and the New Genetics* (H. Gershowitz, D. L. Rucknagel, and R. E. Tashian, eds.), Alan R. Liss, New York, 1986, pp. 133–147.
- [Nei87] ———, *Molecular evolutionary genetics*, Columbia University Press, New York, 1987.



- [NWL04] L. Nakhleh, T. Warnow, and C.R. Linder, *Reconstructing reticulate evolution in species—theory and practice*, Proc. 8th Ann. Int’l Conf. Comput. Mol. Biol. (RECOMB04), 2004, pp. 337–346.
- [PIME06] D. A. Pollard, V. N. Iyer, A. M. Moses, and M. B. Eisen, *Widespread discordance of gene trees with species tree in Drosophila: evidence for incomplete lineage sorting*, PLoS Genet. **2** (2006), 1634–1647.
- [PTP<sup>+</sup>09] Alkes L. Price, Arti Tandon, Nick Patterson, Kathleen C. Barnes, Nicholas Rafaels, Ingo Ruczinski, Terri H. Beaty, Rasika Mathias, David Reich, and Simon Myers, *Sensitive detection of chromosomal segments of distinct ancestry in admixed populations*, PLoS Genet **5** (2009), e1000519.
- [Ram12] A. Rambaut, *Phylogen v1.1*, <http://tree.bio.ed.ac.uk/software/phylogen/> (2012).
- [Ros02] N. A. Rosenberg, *The probability of topological concordance of gene trees and species trees*, Theor. Pop. Biol. **61** (2002), 225–247.
- [Ros07] N.A. Rosenberg, *Counting coalescent histories*, Journal of Computational Biology **14** (2007), 360–377.
- [RWKC03a] A. Rokas, B. L. Williams, N. King, and S. B. Carroll, *Genome-scale approaches to resolving incongruence in molecular phylogenies*, Nature **425** (2003), 798–804.

- [RWKC03b] A. Rokas, B.L. Williams, N. King, and S.B. Carroll, *Genome-scale approaches to resolving incongruence in molecular phylogenies*, *Nature* **425** (2003), 798–804.
- [RY03] B. Rannala and Z. Yang, *Bayes estimation of species divergence times and ancestral population size using dna sequences from multiple loci.*, *Genetics* **164** (2003), 1645–1656.
- [RY08] B. Rannala and Z. Yang, *Phylogenetic inference using whole genomes*, *Annu. Rev. Genomics Hum. Genet.* **9** (2008), 217–231.
- [Sch78] G.E. Schwarz, *Estimating the dimension of a model.*, *Annals of Statistics* **6** (1978), 461–464.
- [SLM<sup>+</sup>12a] F. Staubach, A. Lorenc, P.W. Messer, K. Tang, D.A. Petrov, and D. Tautz, *Genome patterns of selection and introgression of haplotypes in natural populations of the house mouse (*mus musculus*)*, *PLoS Genetics* **8** (2012), e1002891.
- [SLM<sup>+</sup>12b] Fabian Staubach, Anna Lorenc, Philipp W. Messer, Kun Tang, Dmitri A. Petrov, and Diethard Tautz, *Genome patterns of selection and introgression of haplotypes in natural populations of the house mouse (*Mus musculus*)*, *PLoS Genet* **8** (2012), no. 8, e1002891.
- [SWCL05] J. Syring, A. Willyard, R. Cronn, and A. Liston, *Evolutionary relationships among *Pinus* (*Pinaceae*) subsections inferred from multiple low-copy nuclear loci*, *American Journal of Botany* **92** (2005), 2086–2100.

- [Swo96] D. L. Swofford, *PAUP\*: Phylogenetic analysis using parsimony (and other methods)*, 1996, Sinauer Associates, Sunderland, Massachusetts, Version 4.0.
- [Taj83] F. Tajima, *Evolutionary relationship of DNA sequences in finite populations*, *Genetics* **105** (1983), 437–460.
- [Tak89] N. Takahata, *Gene genealogy in three related populations: consistency probability between gene and population trees*, *Genetics* **122** (1989), 957–966.
- [Tav84] S. Tavaré, *Line-of-descent and genealogical processes, and their applications in population genetics models*, *Theor. Pop. Biol.* **26** (1984), 119–164.
- [The12] The Heliconious Genome Consortium, *Butterfly genome reveals promiscuous exchange of mimicry adaptations among species*, *Nature* **487** (2012), no. 7405, 94–98.
- [TKS<sup>+</sup>12] S. Takuno, T. Kado, R.P. Sugino, L. Nakhleh, and H. Innan, *Population genomics in bacteria: A case study of staphylococcus aureus*, *Molecular Biology and Evolution* 29:797809 **29** (2012), 797809.
- [TN09] C. Than and L. Nakhleh, *Species tree inference by minimizing deep coalescences*, *PLoS Computational Biology* **5** (2009), no. 9, e1000501.
- [TN10] ———, *Inference of parsimonious species phylogenies from multi-locus data by minimizing deep coalescences*, *Estimating Species Trees: Practical and Theoretical Aspects* (L.L. Knowles and L.S. Kubatko, eds.), Wiley-VCH, 2010, pp. 79–98.

- [TR11] C. Than and N. Rosenberg, *Consistency properties of species tree inference by minimizing deep coalescences*, *Journal of Computational Biology* **18** (2011), no. 1, 1–15.
- [TRIN07] C. Than, D. Ruths, H. Innan, and L. Nakhleh, *Confounding factors in HGT detection: Statistical error, coalescent effects, and multiple solutions*, *J. Comput. Biol.* **14** (2007), no. 4, 517–535.
- [TRN08] C. Than, D. Ruths, and L. Nakhleh, *PhyloNet: a software package for analyzing and reconstructing reticulate evolutionary relationships*, *BMC Bioinformatics* **9** (2008), no. 1, 322.
- [TSIN08] C. Than, R. Sugino, H. Innan, and L. Nakhleh, *Efficient inference of bacterial strain trees from genome-scale multi-locus data*, *Bioinformatics* **24** (2008), i123–i131, *Proceedings of the 16th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB '08)*.
- [WAD<sup>+</sup>09] M.A. White, C. Ane, C.N. Dewey, B.R. Larget, and B.A. Payseur, *Fine-scale phylogenetic discordance across the house mouse genome*, *PLoS Genetics* **5** (2009), e1000729.
- [Wu12] Y. Wu, *Coalescent-based species tree inference from gene tree topologies under incomplete lineage sorting by maximum likelihood.*, *Evolution* **66** (2012), 763–775.

- [YBN13] Y. Yu, R.M. Barnett, and L. Nakhleh, *Parsimonious inference of hybridization in the presence of incomplete lineage sorting*, *Systematic Biology* **62** (2013), no. 5, 738–751.
- [YDLN14] Y. Yu, J. Dong, K. Liu, and L. Nakhleh, *Probabilistic inference of reticulate evolutionary histories*, Under Review (2014).
- [YDN12] Y. Yu, J.H. Degnan, and L. Nakhleh, *The probability of a gene tree topology within a phylogenetic network with applications to hybridization detection.*, *PLoS Genetics* **8** (2012), no. 4, e1002660.
- [YRN13] Y. Yu, N. Ristic, and L. Nakhleh, *Fast algorithms and heuristics for phylogenomics under ILS and hybridization.*, *BMC Bioinformatics* **14** (2013), no. Suppl 15, S6.
- [YTDN11] Y. Yu, C. Than, J.H. Degnan, and L. Nakhleh, *Coalescent histories on phylogenetic networks and detection of hybridization despite incomplete lineage sorting*, *Systematic Biology* **60** (2011), no. 2, 138–149.
- [YWD<sup>+</sup>11] Hyuna Yang, Jeremy R. Wang, John P. Didion, Ryan J. Buus, Timothy A. Bell, Catherine E. Welsh, Francois Bonhomme, Alex Hon-Tsen Yu, Michael W. Nachman, Jaroslav Pialek, Priscilla Tucker, Pierre Boursot, Leonard McMillan, Gary A. Churchill, and Fernando Pardo-Manuel de Villena, *Subspecific origin and haplotype diversity in the laboratory mouse*, *Nat Genet* **43** (2011), no. 7, 648–655.

- [YWN11a] Y. Yu, T. Warnow, and L. Nakhleh, *Algorithms for MDC-based multi-locus phylogeny inference*, The 15th Annual International Conference on Research in Computational Molecular Biology (RECOMB) **LNBI 6577** (2011), 531–545.
- [YWN11b] ———, *Algorithms for MDC-based multi-locus phylogeny inference: Beyond rooted binary gene trees on single alleles*, *Journal of Computational Biology* **18** (2011), no. 11, 1543–1559.