

RICE UNIVERSITY

**Towards Accurate Reconstruction of Phylogenetic
Networks**

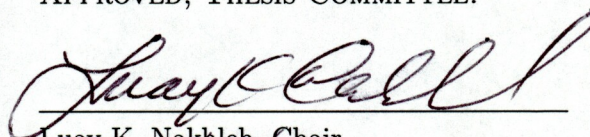
by

Hyun Jung Park

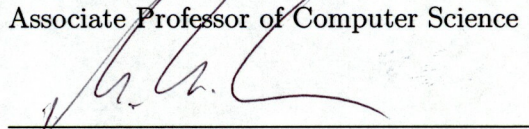
A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

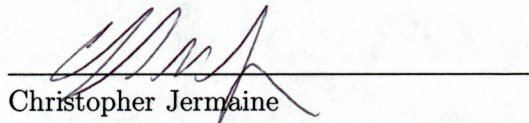
APPROVED, THESIS COMMITTEE:



Luay K. Nakhleh, Chair
Associate Professor of Computer Science



Michael Kohn
Associate Professor of Ecology and
Evolutionary Biology



Christopher Jermaine
Associate Professor of Computer Science

Houston, Texas

April, 2012

ABSTRACT

Towards Accurate Reconstruction of Phylogenetic Networks

by

Hyun Jung Park

Since Darwin proposed that all species on the earth have evolved from a common ancestor, evolution has played an important role in understanding biology. While the evolutionary relationships/histories of genes are represented using trees, the genomic evolutionary history may not be adequately captured by a tree, as some evolutionary events, such as horizontal gene transfer (HGT), do not fit within the branches of a tree. In this case, phylogenetic networks are more appropriate for modeling evolutionary histories.

In this dissertation, we present computational algorithms to reconstruct phylogenetic networks from different types of data. Under the assumption that species have single copies of genes, and HGT and speciation are the only events through the course of evolution, gene sequences can be sampled one copy per species for HGT detection. Given the alignments of the sequences, we propose systematic methods that estimate the significance of detected HGT events under maximum parsimony (MP) and maximum likelihood (ML). The estimated significance aims at addressing the issue of overestimation of both optimization criteria in the search for phylogenetic networks and helps the search identify networks with the “right” number of HGT edges. We study their performance on both synthetic and biological data sets. While the studies show very promising results in identifying HGT edges, they also highlight the issues

that are challenging for each criterion.

We also develop algorithms that estimate the amount of HGT events and reconstruct phylogenetic networks by utilizing the pairwise Subtree-Prune-Regraft (SPR) operation from a collection of trees. The methods produce good results in general in terms of quickly estimating the minimum number of HGT events required to reconcile a set of trees. Further, we identify conditions under which the methods do not work well in order to help in the development of new methods in this area.

Finally, we extend the assumption for the genetic evolutionary process and allow for duplication and loss. Under this assumption, we analyze gene family trees of proteobacterial strains using a parsimony-based approach to detect evolutionary events. Also we discuss the current issues of parsimony-based approaches in the biological data analysis and propose a way to retrieve significant estimates.

The evolutionary history of species is complex with various evolutionary events. As HGT contributes largely to this complexity, accurately identifying HGT will help untangle evolutionary histories and solve important questions. As our algorithms identify significant HGT events in the data and reconstruct accurate phylogenetic networks from them, they can be used to address questions arising in large-scale biological data analyses.

Acknowledgments

First of all, I would like to thank Professor Luay Nakhleh, my advisor. He has led me to so many interesting, important problems in bioinformatics. Also with thoughtful advice and helpful critiques, he has helped me to address these problems. Luay, thank you for your guidance and for your patience. I hope someday I will be as great a mentor as you.

I would also like to thank Professor Michael Kohn and Professor Chris Jermaine for serving on my thesis committee, for taking time to review and evaluate my dissertation, and for their helpful comments on my dissertation, my presentation and communication skills, as well as on my research and career. I would like to thank all the members in our group and Dr. Guohua Jin for helping and supporting me throughout my Ph.D. training. Dr. Jin has collaborated with me on a number of projects. Also, I thank Dr. Jill Delsigne for helping me edit this dissertation.

Finally, I thank God for everything. Also, my sincere thanks go to Soyeon Kim, my wife and Erin Park, the newest member of my family, and all my family members for always being with me and believing in me.

Contents

Abstract	i
List of Illustrations	vii
List of Tables	xvi
1 Introduction	1
1.1 Contributions of the Dissertation	2
1.2 Outline of the Dissertation	4
2 Background	8
2.1 Phylogenetic Trees	8
2.1.1 Trees and Phylogenetic Trees	8
2.1.2 The Robinson-Foulds (RF) Metric	11
2.1.3 The Subtree Prune and Regraft (SPR) Distance	11
2.2 Phylogenetic Networks	14
2.3 Species Trees, Gene Trees and Their Incongruence	17
2.3.1 HGT and Hybridization	17
2.3.2 Detecting HGT	19
2.3.3 Duplication and Loss	21
2.3.4 Detecting Duplication and Loss (DL)	22
3 Phylogenetic Networks from Gene Sequence Alignments under Maximum Parsimony (MP)	25
3.1 Introduction	25
3.2 Materials and Methods	29

3.2.1	Maximum Parsimony of Phylogenetic Networks	29
3.2.2	Inferring Well-Supported Phylogenetic Networks	32
3.2.3	Data Sets	36
3.3	Results and Discussion	38
3.3.1	Biological Data	38
3.3.2	Simulated Data	42
3.4	Conclusions	45
4	Phylogenetic Networks from Gene Sequence Alignments	
	under Maximum Likelihood (ML)	48
4.1	Introduction	49
4.2	Methods	51
4.2.1	Phylogenetic Networks and Maximum Likelihood	51
4.2.2	Information Criteria	52
5	Phylogenetic Networks from Gene Trees by Parsimony	74
5.1	Introduction	74
5.2	Background	77
5.2.1	Pairwise and Set-wise Reconciliation of Trees	77
5.3	Methods	79
5.3.1	M1: Fitting a Binomial Distribution of Pairwise Distances	79
5.3.2	M2: Combining Pairwise Solutions	82
5.3.3	MURPAR: Combining Pairwise Solutions using ILP	84
5.4	Experimental Evaluation	89
5.4.1	Experimental Setup	89
5.4.2	Results of Synthetic Data	92
5.4.3	Results of Biological Data	98
5.5	Discussion and Conclusions	99

5.5.1	Distribution of Gene Trees for Detectability	99
5.5.2	Conclusions	101
5.5.3	Future Work	102

6 On the Performance of Parsimonious Reconciliation in Detecting Duplication, Transfer and Loss 104

6.1	Background	104
6.1.1	Detecting Duplication, Transfer, and Loss (DTL)	104
6.2	Experimental Setup	108
6.2.1	Data	108
6.2.2	Species Tree of the γ -Proteobacterial Data Set	111
6.2.3	The Effect of Loss and Transfer on Species Tree Estimation	113
6.2.4	The Effect of the Cost Scheme on Reconciliation	114
6.3	Results	118
6.3.1	The Effect of Loss and Transfer on Species Tree Estimation	118
6.3.2	The Effect of the Cost Scheme on Reconciliation	120
6.4	Discussion	122
6.4.1	Conclusions	122
6.4.2	Future Work	124

7 Conclusions 128

7.1	Discussion	128
7.2	Future Directions	130

Bibliography 132

Illustrations

1.1	Overview of this dissertation. For phylogenetic network reconstruction problem, gene sequence alignments and the reconstructed gene trees from the alignments are the prominent type of data. After Chapter 2 reviews important concepts relevant to this dissertation, Chapter 3, Chapter 4, and Chapter 5 discuss methods for reconstructing phylogenetic networks. As for input data, the methods in Chapter 3 and Chapter 4 take gene sequence alignments and those in Chapter 5 take gene trees. In developing computational methods in the chapters, we assume that HGT is the only cause for the incongruence. In Chapter 6, we extend the assumption and allow for duplication and loss together with HGT. Under this assumption, we run a standard algorithm to detect evolutionary events and discuss how the detection would help biological data analyses.	7
2.1	Rooted phylogenetic tree (a) and unrooted phylogenetic tree (b) over 4 taxa a, b, c , and d . Note that the tree in (b) does not indicate an ancestor-descendant relationship.	9
2.2	Unrooted binary phylogenetic trees $T1$ and $T2$ on 5 taxa a, b, c, d, e and the edges corresponding to its nontrivial bipartitions are labeled $e1, e2, e3, e4$.	12
2.3	An rSPR operation for a rooted tree. After the original root r is extended to r' , the operation prunes taxon b and regrafts it as the new child of r'	13

2.4	Phylogenetic networks and the trees they contain. (a) A phylogenetic network with a single reticulation node h and two reticulation edges (u, h) and (v, h) with inheritance probabilities of 0.15 and 0.85, respectively. (b) One of the two trees contained within the phylogenetic network. The probability of a gene in C evolving down this tree is 0.85. (c) The other tree contained within the phylogenetic network. The probability of a gene in C evolving down this tree is 0.15.	15
2.5	Two networks (a) and (b), and their induced trees (c) and (d). Note that mathematically the networks of (a) and (b) are identical in the sense that they induce the same set of trees. However, network (a) can indicate the species tree in the network, whereas (b) is used when there is no clear species evolution. Due to their different assumptions for the species tree, the network (a) is used to represent HGT events, and (b) is for hybridization events (see Section 2.3 for more detail).	16
2.6	A horizontal gene transfer event. (a): the gene tree in red lines disagrees with the species tree shown in pipes. Because the red gene for taxon b is transferred from c , the reconstructed gene tree for the red gene would have the taxon b as a sibling of c . (b): a phylogenetic network representing the HGT on the species tree. Phylogenetic networks explicitly notate the source and the destination of the HGT event.	18
2.7	Duplication and loss events that cause gene tree-species tree incongruence. (a): the species tree of taxa a , b and c . (b): a gene tree with an incongruent evolutionary relationship to the species tree. (c): a potential scenario where the gene takes the branching order of the tree within the branches of the species tree.	22
3.1	A phylogenetic tree (a) and a phylogenetic network obtained from it by adding a horizontal edge H from edge B to edge E	30

- 3.2 (a) A scenario where none of four HGT edges identified individually in 100 bootstrap samples has good support (the recipient of each of the four edges is the same node v in the species tree). (b) When combined, thus allowing for ambiguity in pinpointing the exact source, a well-supported hypothesis of an HGT emerges. 33
- 3.3 HGT edges (in red) inferred by the MP criterion, with support values, in parentheses, computed based on Formula (3.3). Ambiguity in the source is denoted by a circle (when drawing a circle was possible) or a multi-source edge. Amborella genes are colored in red, and core eudicot genes and moss genes are colored in blue and green, respectively. Branch refinements are performed for *nad5*, *ccmFN1*, *nad5intron*, and *nad7intron* at the places marked with solid circles. 47

- 4.1 Effect of the diameter of an HGT edge on the change in the likelihood score. The diameter of an HGT edge from node x to node y in the phylogenetic network is measured as the length of the path between x and y in the underlying tree (the network without the red arrows in (a)). Each of the 12 HGT edges was assessed individually, and never in combinations in this experiment. (b) Effect of the diameter for HGTs with different diameters but with a fixed donor node (taxon 1); these results correspond to each of the 6 HGT edges involving taxa 1—8. The diameters of the HGT edges vary from 0.15, for the HGT edge from taxon 1 to taxon 3, to 0.65, for the HGT edges from taxon 1 to taxon 8, with increments of 0.1. (c) Effect of the diameter for HGTs with different diameters but with a fixed recipient node (taxon 16); these results correspond to each of the 6 HGT edges involving taxa 9—16. The diameters of the HGT edges vary from 0.15, for the HGT edge from taxon 14 to taxon 16, to 0.65, for the HGT edges from taxon 9 to taxon 16, with increments of 0.1. The case of diameter=0 corresponds to scoring the likelihood of the underlying tree given the data. 57

- 4.2 Effect of the height of an HGT edge on the change in the likelihood score. The height of an HGT edge from node x to node y in the phylogenetic network is measured as the sum of the length of the path from x to a leaf under it in the underlying tree (the network without the red arrows in (a)) and the length of the path from y to a leaf under it. Each of the 10 HGT edges was assessed individually, and never in combinations in this experiment. (b) Effect of the height for HGTs with different heights but with the recipient taxon always being a branch connected to a leaf node; these results correspond to each of the 5 HGT edges involving taxa 1–8. The heights of the HGT edges vary from 0.05, for the HGT edge from taxon 1 to taxon 4, to 0.45, for the HGT edges from taxon 1 to taxon 8, with increments of 0.1. (c) Effect of the height for HGTs with different heights but with the donor taxon always being a branch connected to a leaf node; these results correspond to each of the 5 HGT edges involving taxa 9–16. The heights of the HGT edges vary from 0.05, for the HGT edge from taxon 13 to taxon 16, to 0.45, for the HGT edges from taxon 9 to taxon 16, with increments of 0.1. The case of height=0 corresponds to scoring the likelihood of the underlying tree given the data. 58
- 4.3 Three evolutionary histories, each involving the underlying tree (black lines) and a single reticulation edge from the set of three reticulation edges 1, 2, and 3. The diameters of the three reticulation edges 1, 2, 3 are 0.5, 1.0, and 1.5, respectively. 59

4.4	The performance of ML for estimating the reticulation probabilities on data simulated with a single reticulation event. The genome size corresponds to the number of gene data sets used in the inference. Each panel contains three segments, corresponding to three different values of true reticulation probabilities: 0.1, 0.3, and 0.5. The reticulation probabilities γ_e were estimated using Eq. (4.4). The three diameters correspond to the three networks of Fig. 4.3.	61
4.5	The change in the likelihood scores as more reticulation edges are added. The true number of reticulations is 0 (all sequences were generated down the tree with no reticulations in Fig. 4.5).	65
4.6	The change in the likelihood scores as more reticulation edges are added. The true number of reticulations is 1, yet with three different diameters, as in Fig. 4.5.	72
4.7	Results on the <i>rbcL</i> gene data set. (Left) The underlying species tree, as reported in [1], with the five predicted HGT edges posited between pairs of its branches. (Right) The decrease in the AIC and BIC values as optimal HGT edges are added to the species tree. The decrease in the AIC/BIC values from HGT addition i to $i + 1$ corresponds to HGT edge H_i	73
4.8	The inferred species tree for the yeast data set in [2]. The horizontal arrow corresponds to the reticulation event inferred by our method, along with the reticulation probability.	73

5.1 Three \mathcal{X} -networks, each with two network-nodes, yet with varying degrees of redundancy. Here, $\mathcal{T}(N_1) = \{T_1\}$, $\mathcal{T}(N_2) = \{T_1, T_2\}$, and $\mathcal{T}(N_3) = \{T_1, T_2, T_3, T_4\}$, where $T_1 = ((A, (B, C)), D)$, $T_2 = (((A, B), C), D)$, $T_3 = (A, (B, (C, D)))$, and $T_4 = ((A, B), (C, D))$. Consequently, we have $\varepsilon_{N_1} = (4 - 1)/4 = 0.75$, $\varepsilon_{N_2} = (4 - 2)/4 = 0.50$, and $\varepsilon_{N_3} = (4 - 4)/4 = 0$ 80

5.2 **A phylogenetic network with four independent HGT scenarios.**
 The species tree ST in this case is the network N without the four HGT edges. The gene whose tree is GT_i ($1 \leq i \leq 3$) underwent HGT event (i), and the gene whose tree is GT_4 underwent HGT events (1) and (4). The combined effect of HGTs (1) and (4) on the gene tree topology is the same as the combined effect of HGTs (1), (2), and (3). 85

5.3 **The difference between the formulation used by MURPAR, M2 [3], and PIRN [4], and the one used by CASS [5].** For the input set of gene trees $\mathcal{T} = \{T_1, T_2, T_3\}$, CASS computes a network with a single reticulation node (N_1), since this network displays all clusters of the gene trees. However, MURPAR, M2, and PIRN compute minimal networks with two reticulation nodes, such as N_2 , since 2 is the minimum number of reticulation nodes required in a network that displays all three gene trees. 91

5.4 Performance of M1 on the 30-taxon (a) and 50-taxon (b) data sets as a function of the sample size. 93

5.5 Performance of M2 on the 30-taxon (a) and 50-taxon (b) data sets as a function of the sample size. 94

5.6 Running times (in sec.) of the three methods (PIRN, M2, and MURPAR). The numbers after the ‘/’ are the numbers of gene trees in the input. . . . 95

5.7	Numbers of reticulations estimated by each of the three methods (PIRN, M2, and MURPAR). The numbers after the ‘/’ are the numbers of gene trees in the input.	97
5.8	Inspection of over- and under-estimation of M1 as a function of the distribution deviation from 1/2 (see text for more details). Black, blue, and red dots represent correct, under-, and over-estimations, respectively, of the method. Left to right, top down: sample sizes 4, 8, 12, 16, 24, and 32 (all on 50-taxon data sets).	100
6.1	Three examples of biologically plausible HGT events, not considered in most of the parsimony-based approaches. In practice, they can occur in various combinations and further complicate the picture. . .	106
6.2	Two reconciliations for the gene tree and the species tree given in Figure 2.6. The reconciliation shown in (a) is possible under both DL and DTL models and that in (b) is possible only under DTL. (a) invokes 1 duplication and 3 losses, while (b) invokes 1 transfer and 1 loss. Note that the parsimonious mapping depends on the cost scheme. In parsimony under the cost scheme penalizing a transfer by cost 1 and a duplication by cost 2, the reconciliation of (a) requires cost 2 and that of (b) returns cost 1, for example.	107
6.3	The reconstructed gene family trees are counted by the number of strains (a) and the number of gene copies (b), respectively. In (b), we distinguish the trees with different loss amounts.	108
6.4	Species tree of the γ -proteobacterial strains estimated by DupTree [6] based on the γ -proteobacteria gene trees. Strains are colored by taxonomic order.	125

- 6.5 The RF distance values between ST and ST_i from the loss simulations. The percentage value i is given on the x-axis and the normalized RF distance value between ST_i and ST is plotted on the y-axis. (a) The average (in bars) and the standard deviation (in error bars) of the RF distance values. While the loss simulation decreases the number of leaves in a tree and sometimes removes all nodes of a tree, (b) shows the number of the trees that the simulation retains to use for the estimation, and (c) shows the percentage of retained leaves in the trees. 126
- 6.6 Results from the transfer simulations. The values on the x- and y-axis are similar as in Figure 6.5. (a) shows the normalized RF distance values between ST and ST_i , and (b) plots the average of the normalized RF distance values between gene trees and their manipulated trees with the transfers in bars and the standard deviation values in error bars. 126
- 6.7 The ratios of the duplication and transfer events across γ -proteobacteria gene trees in average (bars) and standard deviation values (error bars) based on a specific cost scheme. The sets of the values as cost schemes are denoted on the x-axis. Basically, the value before the slash (Cd) refers to the cost value for duplication, and that after the slash (Ct) refers to the cost for transfer. Schemes on the left side of the x-axis bear high costs for transfer, and those on the right bear high costs for duplication. With the cost schemes, the detections are conducted by Tofigh *et al.*'s algorithm and Algorithm 1. 127
- 6.8 The ratios of duplication of 10 gene trees of the γ -proteobacteria data by different cost schemes. The ratios of the trees of different cost values is plotted by color. Note that the ratios of the transfer of the gene trees show a complementary pattern. 127

Tables

3.1	Mitochondrial gene data sets and HGTs postulated by Bergthorsson <i>et al.</i> and those computed by the MP analysis (NEPAL). ‘donor’ denotes the group from which the gene was transferred (in all cases, the recipient is <i>Amborella</i>). ‘SH’ denotes support of the HGT events as computed by the Shimodaira–Hasegawa (SH) test and reported by Bergthorsson <i>et al.</i> (values lower than 0.05 indicate high support, and NS indicates support is not significant). The ‘b1’, ‘b2’, ‘b3’, and ‘b4’ columns correspond to the support values from Formula (3.3) for adding the first, second, third, and fourth HGT edges inferred by the MP analysis. Since adding HGT edges stops once a weakly supported edge is encountered, a ‘–’ entry under these columns indicates that adding HGT edges was stopped before. B = Bryophyte, M = Moss, E = Eudicot, and A = Angiosperm.	39
3.2	Results of the MP analysis of 20 simulated data sets. The rows are sorted by the number of true HGTs simulated for each gene data set. The ‘d1’ and ‘d2’ columns denote the distance, in terms of the number of branches on the species tree, between the source and recipient of the first and second HGT events simulated, respectively. The ‘b1’, ‘b2’, and ‘b3’ columns correspond to the support values from Formula 3.3 for adding the first, second, and third HGT edges inferred by the MP analysis. Since adding HGT edges stops once a weakly supported edge is encountered, a ‘–’ entry under these columns indicates that adding HGT edges was stopped before.	44

4.1	62
5.1	The number of estimated reticulation events and the run time (in seconds) of the methods on the five gene trees for <i>ndhF</i> , <i>phyB</i> , <i>rbcL</i> , <i>rpoC2</i> , and <i>ITS</i> in the Poaceae data set.	98

Chapter 1

Introduction

Since Darwin proposed the hypothesis that all species on the earth have evolved from a common ancestor in his famous book *On the Origin of Species by Means of Natural Selection*, placing living organisms in evolutionary relationship to others has been a primary way to understand them. Indeed, evolutionary biology has provided important insights into the mechanisms of evolution and has helped us understand the end-product of evolution, the organisms. In particular, as the advent of technology enables us to study humans in this evolutionary relationship, evolutionary biology does not stop at promoting understandings but starts answering the practical questions of finding cures for human diseases, such as inheritable genetic disorders and cancer.

A phylogeny, the model of the relationship among evolutionary units, is traditionally represented by a tree. Due to the explosion in genomic research, trees are built often based on genetic sequences. In particular, the model of genetic evolutionary history is referred to as a gene tree, and that of species evolutionary history as a species tree. When it comes to the relationship between these two trees, it is natural to expect that a gene tree is identical to the species tree. However, as multi-locus sequence data become available for an increasing number of species, it becomes clear that gene trees may be incongruent with each other. It naturally follows that a gene tree cannot represent the species tree, and the true species tree can be incongruent with a gene tree. We refer to this phenomenon as gene tree/species tree incongru-

ence. One reason for this incongruence is that the reconstructed gene tree might be incorrect due to random sampling and/or phylogenetic reconstruction errors. More importantly, evolutionary events, such as lineage sorting, gene duplication and loss, and horizontal gene transfer and hybridization, cause gene tree/species tree incongruence [7]. These events are ubiquitous in certain domains of biology and play critical roles for surviving species [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19].

When a tree cannot represent the genomic evolutionary history due to its failure to model such events, several proposed models represent the genomic evolutionary history by allowing for reticulation with speciation. One type of model focuses on representing the incompatibility between genes, but does not explicitly lead to biological interpretations. This category includes Median Networks [20], Consensus Networks [21], and Neighbor-net [22]. On the other hand, Hybridization Networks [23], Recombination Networks [24], and Evolutionary Phylogenetic Networks [25] attempt to reflect the evolutionary events rather than restricting themselves to represent incompatibility, and to include ancestral species in the model. Readers are referred to [26] for more detail. I focus on evolutionary phylogenetic networks, since they have been widely used due to their direct interpretability.

1.1 Contributions of the Dissertation

This dissertation focuses on developing computational methods for reconstructing phylogenetic networks using multi-locus data consisting of either sequences or trees. As a phylogenetic network consists of a set of reticulation edges and the underlying tree, the methods can also be used for detecting the reticulation events in species evolution. In particular, methods in Chapter 3 and 4 take the species tree and the sequence alignment as input and reconstruct phylogenetic networks by identifying the

reticulation events from them. On the other hand, methods in Chapter 5 take the set of gene trees for the reconstruction.

The first contribution of this dissertation is the algorithm for the reconstruction problem under the Maximum Parsimony (MP) criterion. Given a species tree and sequence alignments, phylogenetic networks are reconstructed by searching for potential reticulate edges under an optimization criterion and adding them to the species tree. Under this criterion, the process usually overestimates the reticulate edges and thus reconstructs an overly complex network. In order to avoid such overestimation, our method conducts a bootstrap process for the searched reticulate edges and adds only the significant ones. This method also discovers a lack of resolution in a phylogenetic-based search and suggests a way to address it.

The next method addresses the same problem under the Maximum Likelihood (ML) criterion. First, we study the behavior of the reticulations under ML. As a reticulation event between species leaves a signal in their sequences, we identify several graph-based properties of the event and the composition of the sequence carrying the signal of the event that affect the strength of the signal in likelihood score. ML is known to overfit the signal and result in overly complex networks. We systematically evaluate the performance of two information criteria, Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) under ML in addressing the overfit. Based on the simulation study, this chapter finds that the ML combined with BIC is a reasonable method for identifying the number of reticulate events.

The next class of methods is for the same problem, but it takes gene trees as the input, instead of gene sequence alignments. For the computational problem of inferring phylogenetic networks with the minimum reticulations from a collection of gene trees, we develop three algorithms. The first method infers the number of the

reticulation events based on the observation of the binomial distribution of pairwise distances of the trees. The second and the third methods reconstruct the phylogenetic network based on the aggregation of the solutions from pairwise computations. The third method formulates the problem as an instance of the *integer linear programming* (ILP) problem. Compared to the competing methods, our methods show good performance both in speed and in the minimum number of reticulation events inferred to reconcile the set of trees.

While they show good performance for the reconstruction problem, it is important to note that all methods assume reticulation events to be the only cause of the gene tree/species tree incongruence. In practice, it is possible that other evolutionary events such as duplication and loss and/or lineage sorting occur in the species set of interest, especially as the data become larger.

In the last chapter, we study the performance of parsimonious reconciliation in the evolutionary history of a γ -proteobacteria data set. In this chapter, we discuss the issues of the current parsimony-based approaches to detect the events and how to obtain significant estimates despite these issues.

1.2 Outline of the Dissertation

Chapter 2 provides a brief review of phylogenetic trees, phylogenetic networks and related concepts such as Subtree-Prune-Regraft (SPR) distance and Robinson-Foulds (RF) metric. It then describes biological processes that cause species/gene tree incongruence, the main concern of this dissertation. As discussed in the previous section, a major contribution of this dissertation consists of developing methods for reconstructing phylogenetic networks under an optimality criterion. The optimization criteria for phylogenetic networks and the overview of the current methods to infer phylogenetic

networks under these criteria will be elaborated on in the following sections.

Chapter 3 discusses a method that estimates the significance of the inferred reticulation events under MP. In this chapter, we first briefly review the MP optimality criterion for phylogenetic networks and show its tendency to overestimate the reticulation signals. Then, we introduce a statistical framework to assess the significance of the reticulate edges returned in the search. With our finding about the lack of resolution in the phylogeny-based reticulation detection, we relax the significance assessing formula and increase the sensitivity of the method. In the search under MP, this method can be used as a stopping criterion of the reticulation search and show good performance in identifying the number of potential reticulation events both on synthetic data sets and biological data sets.

In Chapter 4, we study the behavior of reticulation events under the ML optimality criterion and evaluate the performance of the widely used information criteria, AIC and BIC, in identifying reticulation events. First, we review the mathematical formula of the ML optimality criterion for phylogenetic networks, AIC, and BIC. Then, we specify how the reticulate edges are searched and added on the given species tree. With the defined search and the optimality criterion, we study the behavior of reticulation events in tree likelihood score and network likelihood score. In the study, we first characterize the graph-theoretic properties of the events and the composition of the input data that cause great changes in likelihood scores. With this characterization, we show, through extensive simulation studies, that a naive use of ML and AIC in combination with ML usually leads to overestimating reticulation events; however, BIC under ML works well as it leads to a small overestimation. We also show that the findings from the simulation study help identify reticulation events in practice in the following analysis of the yeast data set.

Chapter 5 focuses on the reconstruction problem from a collection of gene trees, rather than on sequence alignments. This chapter introduces three methods. For the first method, we define *redundancy* of a network and mathematically formulate the distribution of pairwise distances among the trees induced from the network when the *redundancy* is 0. With certain assumptions, the distribution is fitted to that of pairwise SPR distances among the input tree set and to estimate the number of reticulation events in the set. The second method is a heuristic that reconstructs a phylogenetic network with a minimum number of reticulate edges by combining pairwise solutions from the input trees. In order to reconstruct the minimal networks, it counts how many times a pairwise solution is used in the solutions and makes use of this information in selecting solutions for the problem. However, we find that the method is biased due to the multiplicity of solutions. In the third method, we formalize the problem as an instance of ILP (*integer linear programming*) problem. Both the simulation study and the comparison analysis show that these methods perform well both in speed and in accuracy.

Chapter 6 deals with a biological data set for γ -proteobacteria and investigates the performance of a parsimony-based approach with different parameter values. With this investigation, we discuss the issues with this approach for biological data analysis and how to strategically overcome them.

All methods presented here focus on identifying significant reticulations in the data for reconstructing accurate phylogenetic networks. Because accurately reconstructed phylogenetic networks can help conduct biological analyses, we discuss the advantages and the limitations of the methods for such analysis. First, we discuss the importance of assessing the significance of the estimates for the analysis as well as that of speed and accuracy of the method. We also discuss another benefit of the structure of our

methods, that as the methods characterize the problems as multi-layered and employ external programs for the lower-layers; improving the programs at the lower-layer can directly improve the entire process. We also discuss the limitations of the current methods' assumptions. In order to make it more helpful for biological analyses, we should study the correct values for the parameters in the model with respect to the biological data set of interest. And we also should make the model less restrictive regarding possible evolutionary events.

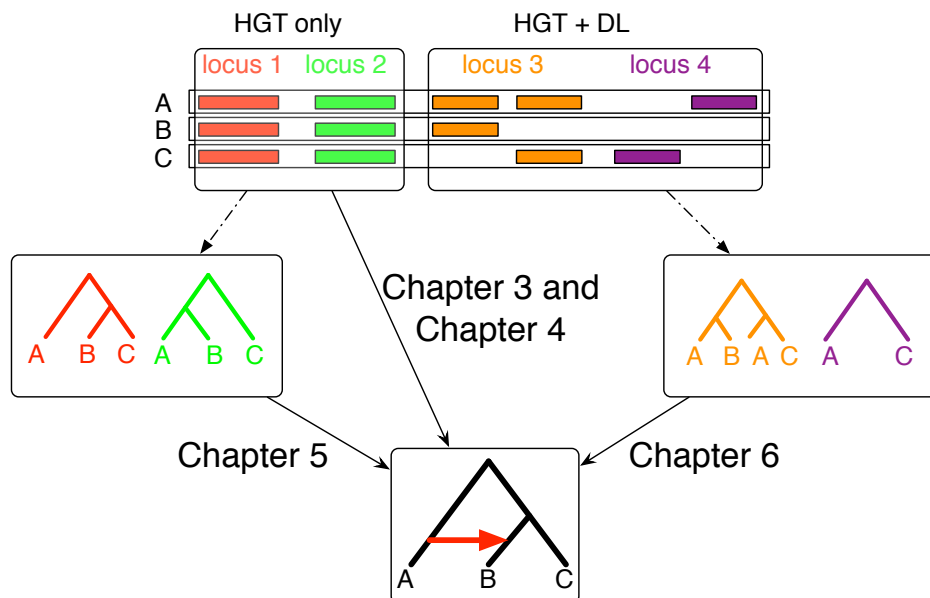


Figure 1.1 : Overview of this dissertation. For phylogenetic network reconstruction problem, gene sequence alignments and the reconstructed gene trees from the alignments are the prominent type of data. After Chapter 2 reviews important concepts relevant to this dissertation, Chapter 3, Chapter 4, and Chapter 5 discuss methods for reconstructing phylogenetic networks. As for input data, the methods in Chapter 3 and Chapter 4 take gene sequence alignments and those in Chapter 5 take gene trees. In developing computational methods in the chapters, we assume that HGT is the only cause for the incongruence. In Chapter 6, we extend the assumption and allow for duplication and loss together with HGT. Under this assumption, we run a standard algorithm to detect evolutionary events and discuss how the detection would help biological data analyses.

Chapter 2

Background

In this chapter, we review concepts and definitions relevant to this dissertation. First we introduce phylogenetic trees and discuss different types of trees and their usages. We then discuss two metrics defined to quantify the relationship between the trees, Robinson-Foulds (RF) metric and Subtree Prune and Regraft (SPR) distance, and introduce the programs for calculating them. Then, we define phylogenetic networks and their relationship to a corresponding set of trees, an important concept for the network's optimization criteria for the network. We also briefly summarize the biological processes that cause gene tree/species tree incongruence: Horizontal Gene Transfer (HGT), hybridization, duplication, and loss, along with the current theories of why these processes are important for evolution. Finally, we introduce some computational methods to detect these processes in genetic evolutionary histories.

2.1 Phylogenetic Trees

2.1.1 Trees and Phylogenetic Trees

The evolutionary history of a group of species is often depicted in the form of a tree (in the formal sense in computer science), called a species tree. Each internal node in the tree reflects a speciation event that splits the group into smaller subgroups, and leaves can be thought of as representing present-day organisms. As species evolve, their genes evolve. Since genes are also evolutionary units, the evolution of a gene is

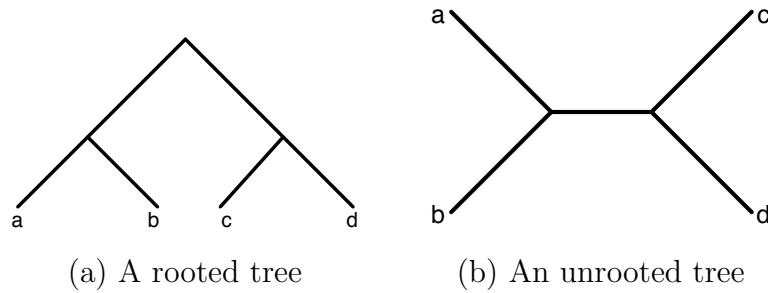


Figure 2.1 : Rooted phylogenetic tree (a) and unrooted phylogenetic tree (b) over 4 taxa $a, b, c,$ and d . Note that the tree in (b) does not indicate an ancestor-descendant relationship.

also represented by a tree, called a gene tree. When the gene tree includes duplicated gene copies, it is called a gene-family tree. Species trees and gene trees are collectively referred to as phylogenetic trees and defined as follows.

Definition 2.1. For a set of biological entities \mathcal{X} , either species or an orthologous group of genes in the species, a phylogenetic \mathcal{X} -tree, or \mathcal{X} -tree, T is an ordered pair (T, f) , where $T = (V(T), E(T))$ is a connected directed graph with no cycles with $V(T) = R(T) \cup L(T) \cup I(T)$, where

1. $R(T) \in V(G), \text{indeg}(R(T)) = 0$ ($R(T)$ is the root of T);
2. $\forall v \in L(T), \text{outdeg}(v) = 0$ ($L(T)$ is the set of leaves of T);
3. $\forall v \in I(T) = V(T) - L(T), \text{outdeg}(v) \geq 1$ ($I(T)$ is the set of the internal nodes of T);
4. $E \subseteq V(T) \times V(T)$ are the tree's edges.

The mapping $f : L(T) \rightarrow \mathcal{X}$ is a bijection to \mathcal{X} . In the trees, $x \leq_T y$ and $x <_T y$ denotes the partial order between $x, y \in V(T)$ that y is between $R(T)$ and x . Given a node $u \in V(T)$, $L(u) = \{v : v \in L(T), v \leq_T u\}$. In this sense, $L(R(T)) = L(T)$.

A phylogenetic tree can be rooted or unrooted. Tree T is rooted if there is a distinguished node, $R(T)$. In a rooted phylogenetic tree, the root corresponds to the common ancestor of all species or genes at its leaves. A rooted phylogenetic tree, therefore, shows not only the relative relationships of species but also the direction of evolution, from its root towards its leaves. An unrooted phylogenetic tree, on the other hand, only shows the relationship among species. Figure 2.1(a) shows an example of a rooted phylogenetic trees, while Figure 2.1(b) is an example of an unrooted phylogenetic tree. In this dissertation, all trees are rooted, unless explicitly stated otherwise.

See Figure 2.1 for examples of phylogenetic trees. For the sake of simplicity, in this dissertation we often call T a phylogenetic tree when the mapping f is obvious from the context.

A phylogenetic tree can also be binary or non-binary. A rooted phylogenetic tree T is called binary if $outdeg(v) = 2, \forall v \in I(T)$; otherwise, it is non-binary. If T is an unrooted tree, then it is binary if all internal nodes have degree three. A node v with $outdeg(v) > 2$ is referred to as polytomy, or a non-binary node.

It is easy to see that for a rooted, binary phylogenetic tree T on an n -element taxon set \mathcal{X} , there are exactly $n - 1$ internal nodes. The following theorem is also well known; its proof can be found in [27].

Theorem 2.1 (Number of Binary Phylogenetic Trees). *Let X be a set of n taxa. Then, the number of binary, unrooted phylogenetic trees on X is $(2n - 5)!!$, and the number of binary, rooted phylogenetic trees on X is $(2n - 3)!!$.*

A phylogenetic tree can be represented in the Newick format [28]. In this format, an \mathcal{X} -tree is represented as an instance of a hierarchical clustering of the elements in \mathcal{X} by parentheses and commas. For example, the tree in Figure 2.1(a) is written

in the Newick format as $((a, b), (c, d))$. By adding a prefix $[U]$ on the Newick representation of a random rooting of unrooted trees, it can also represent the unrooted tree. For example, a Newick representation for the trees in Figure 2.1(b) can be $[U] (a, (b, (c, d)))$.

2.1.2 The Robinson-Foulds (RF) Metric

When an internal edge in an \mathcal{X} -tree defines a bipartition of \mathcal{X} , the set of the nodes under the edge and its complement, the Robinson-Foulds (RF) distance between two \mathcal{X} -trees is the sum of the number of bipartitions that differ between them [29]. Let $\Sigma(T)$ be the set of all bipartitions defined by all edges in T . The RF distance between trees $T1$ and $T2$ is defined as

$$\frac{|\Sigma(T1) - \Sigma(T2)| + |\Sigma(T2) - \Sigma(T1)|}{2} \quad (2.1)$$

Consider the pair of trees in Figure 2.2. In the figure, the internal edges $e1, e2, e3, e4$ in $T1$ and $T2$ correspond to the bipartition $\{ab|cde\}, \{abc|de\}, \{ab|cde\}, \{abe|cd\}$, respectively. By Eq. (2.1), the RF distance between $T1$ and $T2$ is 1.

Day showed that the problem of calculating the distance between two trees has a linear complexity in the number of nodes in the trees [30].

2.1.3 The Subtree Prune and Regraft (SPR) Distance

Subtree Prune and Regraft (SPR) is a tree transforming operator that transforms the topology of a given unrooted \mathcal{X} -tree to another unrooted \mathcal{X} -tree. Given an \mathcal{X} -tree T , the transformation prunes a subtree and regrafts it back to one of its remaining places on the tree by conserving the root of the pruned subtree. The new tree is said

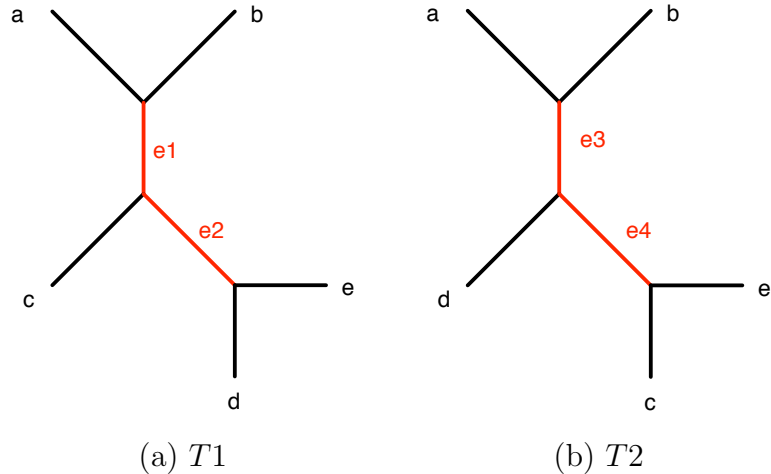


Figure 2.2 : Unrooted binary phylogenetic trees T_1 and T_2 on 5 taxa a, b, c, d, e and the edges corresponding to its nontrivial bipartitions are labeled e_1, e_2, e_3, e_4 .

to be obtained from T by an SPR operation.

An unrooted \mathcal{X} -tree is reached from another unrooted \mathcal{X} -tree by applying a series of SPR operations [31, 32]. The SPR distance between two unrooted trees is defined as the minimum number of SPR operations required to transform one to the other. The problem of computing this distance is NP-hard [33]. An SPR operation for rooted binary trees (rSPR) can be defined in a similar way, except, in order to make the rooted SPR distance a metric, we should create a new root extending from the original root and allow the pruned subtree to be regrafted onto the root [34]. Figure 2.3 shows an example of the rSPR operation. As the operator transforms the tree in (a), it first creates r' by extending from r . After that, it prunes the red branch leading to b from (a), and regrafts it onto the branch to r' to yield the tree in (b) in this example.

Because the tree transforming SPR operation simulates an HGT event, it has been widely used to detect HGT events on gene trees. In particular, in order to

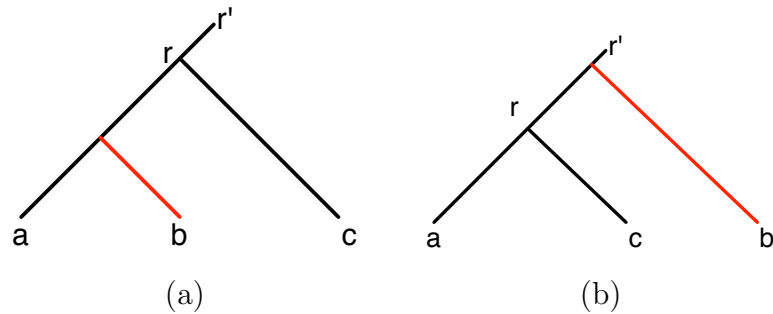


Figure 2.3 : An rSPR operation for a rooted tree. After the original root r is extended to r' , the operation prunes taxon b and regrafts it as the new child of r' .

computationally estimate the lower bound on the amount of HGT events, scholars have been interested in calculating the *SPR distance*, defined as the minimum number of SPR operations required to reconcile trees*. While there are exact methods that calculate *SPR distance*, using the distance for HGT detection raises issues including computational complexity and the possibility for gross underestimation of the number of HGT events. Computationally, the problem of computing the rooted SPR distance between two rooted binary trees is NP-hard [34]. However, there are a number of exact algorithms and heuristics that compute this distance. The exact algorithms have been developed by Bordewich and Semple [34] and Wu [35], and heuristics include LatTrans [36], EEEP [37], HorizStory [38], RIATA-HGT [39, 40]. While the algorithms perform well in practice, a more serious issue is the possibility that the SPR distance underestimates the number of reticulation events [41, 42]. In particular, Baroni *et al.* [43] reported that the underestimation can be arbitrarily large relative to the size of the leaf sets.

**rSPR distance* is defined in a similar way on the rooted trees

2.2 Phylogenetic Networks

When particular biological processes such as horizontal gene transfer occur in the species set of interest, the species' evolution might not be best represented by a phylogenetic tree but more appropriately by a phylogenetic network (see Section 2.3 for more detail). While the phylogenetic network model is general enough to allow for modeling all types of reticulate evolutionary events, such as hybrid speciation, recombination, and HGT, the semantics of the model change based on the specific evolutionary events allowed [44]. In this thesis, we focus on HGT and adopt the following phylogenetic network model.

Definition 2.2. *A phylogenetic \mathcal{X} -network, or \mathcal{X} -network, N is an ordered pair (G, f) , where*

1. $G = (V, E)$ is a directed, acyclic graph (DAG) with $V = \{r\} \cup V_L \cup V_T \cup V_N$, where
 - (a) $\text{indeg}(r) = 0$ (r is the root of N);
 - (b) $\forall v \in V_L, \text{indeg}(v) = 1$ and $\text{outdeg}(v) = 0$ (V_L are the leaves of N);
 - (c) $\forall v \in V_T, \text{indeg}(v) = 1$ and $\text{outdeg}(v) \geq 2$ (V_T are the tree-nodes of N);
 - and,
 - (d) $\forall v \in V_N, \text{indeg}(v) = 2$ and $\text{outdeg}(v) \geq 1$ (V_N are the reticulation nodes of N),
 - (e) and $E \subseteq V \times V$ are the network's edges. we distinguish between reticulation edges (edges whose heads are reticulation nodes) and tree-edges (edges whose heads are tree-nodes or leaves).
2. $f: V_L \rightarrow \mathcal{X}$ is a bijection function from V_L to \mathcal{X} .

γ can be defined for a network as follows, when it is needed. $\gamma : E_H \rightarrow [0, 1]$, where E_H is the set of reticulation edges, is the reticulation probability associated with reticulation edges, and satisfies $\gamma(e_1) + \gamma(e_2) = 1$ for every pair of edges e_1 and e_2 that share the same reticulation node at their heads.

As the name indicates, the interpretation of γ is the probability of inheritance of a gene from each of two potential parents and is estimated from the data [45, 46, 47, 48]; see Fig. 2.4 for an illustration.

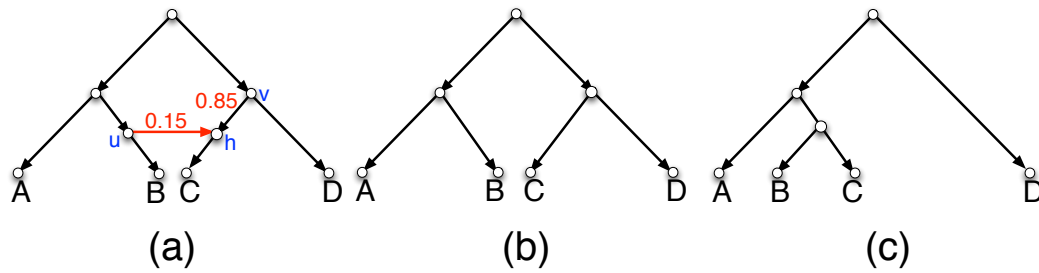


Figure 2.4 : Phylogenetic networks and the trees they contain. (a) A phylogenetic network with a single reticulation node h and two reticulation edges (u, h) and (v, h) with inheritance probabilities of 0.15 and 0.85, respectively. (b) One of the two trees contained within the phylogenetic network. The probability of a gene in C evolving down this tree is 0.85. (c) The other tree contained within the phylogenetic network. The probability of a gene in C evolving down this tree is 0.15.

A phylogenetic \mathcal{X} -tree is an \mathcal{X} -network in which $V_N = \emptyset$. While a network N represents the evolution of a set of genomes, these genomes can be partitioned into (non-recombining) regions R_1, R_2, \dots, R_k , each of which has a treelike evolutionary history, T_i . In other words, the set $\mathcal{T} = \{T_1, \dots, T_k\}$ is a subset of the set of all trees induced by the network N . More formally, $\mathcal{T} \subseteq \mathcal{T}(N)$, where $\mathcal{T}(N)$ is the set of all trees obtained as follows from N :

1. For each node of *indegree* at least 2, remove all but one of the incoming edges; and

2. For each node u of *indegree* and *outdegree* 1, remove u along with its incident edges, and add a new edge to connect u 's parent to u 's child until no such nodes remain.

For a tree $T \in \mathcal{T}(N)$, the *induction set* of T , denoted by $\eta(T)$, is the set of reticulation edges in N that are used (that is, not removed in step (1) above) to obtain tree T . Notice that $\eta(T)$ is not necessarily unique for a given tree T , as there may be more than one possible way of obtaining tree T [3].

Also note that for each network node v , there are exactly $\text{indeg}(v)$ choices, and hence, the number of induced trees is bounded by $\prod_{v \in V_N} \text{indeg}(v)$. For example, Figure 2.5(c) and (d) are two (and only two) trees induced by the network in Figure 2.5(a) or (b).

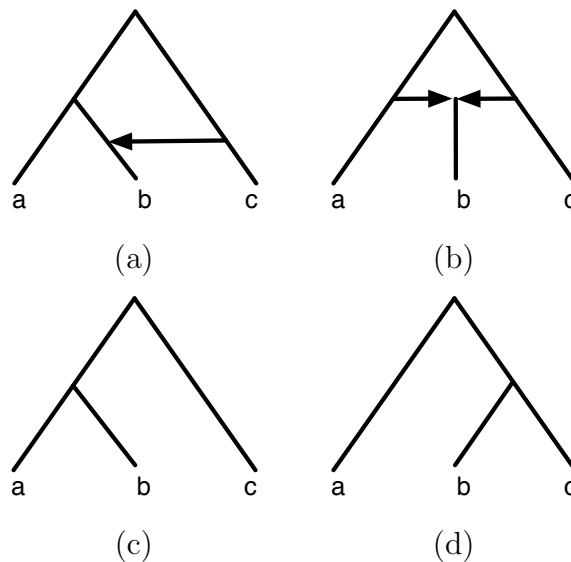


Figure 2.5 : Two networks (a) and (b), and their induced trees (c) and (d). Note that mathematically the networks of (a) and (b) are identical in the sense that they induce the same set of trees. However, network (a) can indicate the species tree in the network, whereas (b) is used when there is no clear species evolution. Due to their different assumptions for the species tree, the network (a) is used to represent HGT events, and (b) is for hybridization events (see Section 2.3 for more detail).

It is important to mention that the current definition of phylogenetic networks restricts f to be a bijection. It follows that a phylogenetic network cannot generate a tree in which there are more than one copy of a gene sampled from a species, so they do not directly account for gene duplication and loss.

2.3 Species Trees, Gene Trees and Their Incongruence

Although both species trees and gene trees take the form of phylogenetic trees, they model conceptually different evolutionary histories: a gene tree shows the evolutionary history of a single gene, while a species tree shows the evolutionary history of speciation. There are a number of biological events that cause a gene tree to differ from its containing species tree [7]. We describe in this section hybridization and HGT, and duplication and loss that occur at a species-level.

2.3.1 HGT and Hybridization

Reticulation events such as HGT and hybridization can differentiate a gene tree from its containing species tree. HGT refers to the transfer of genetic material between organisms other than vertical inheritance. HGT is believed to be rampant among bacteria [49], even between remotely related ones. Three common mechanisms through which HGT occurs are [50]

- transformation: the process in which free DNA (of a dead bacterium, for instance) is absorbed from the surrounding environment;
- conjugation: the transfer of genetic material from one bacterium to another through direct physical contact; and

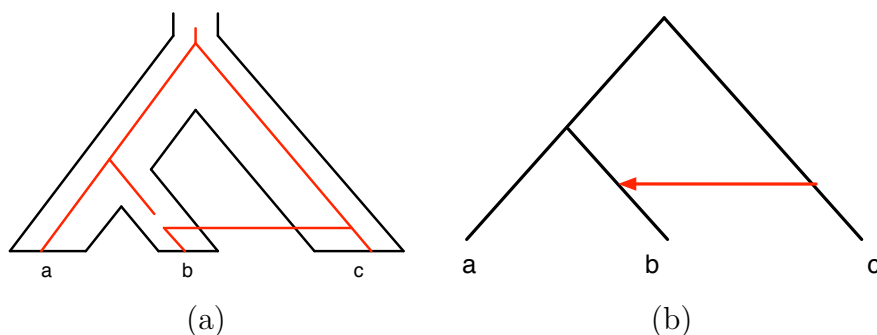


Figure 2.6 : A horizontal gene transfer event. (a): the gene tree in red lines disagrees with the species tree shown in pipes. Because the red gene for taxon b is transferred from c , the reconstructed gene tree for the red gene would have the taxon b as a sibling of c . (b): a phylogenetic network representing the HGT on the species tree. Phylogenetic networks explicitly notate the source and the destination of the HGT event.

- transduction: the process in which a bacterial virus, commonly called a phage, mediates the transfer from one bacterium to another.

We illustrate HGT from the phylogenetic point of view as in Figure 2.6. In the species tree shown in pipe in Figure 2.6(a), a and b are sister taxa. Consider the gene in red lines. Through one of the mechanisms above, the gene in species b is transferred from c , instead of inheriting it from a common ancestor with a . As a result, the evolutionary history of the gene does not agree with that of the species; as the figure shows, b and c are now sister taxa rather than a and c . Note that in order to facilitate detecting the events in the current model of the network that does not allow for duplicated genes, the HGT should be accompanied by the prior gene loss.

Hybridization can also result in species evolution that cannot be represented by a tree. In several groups of species, especially in plants and fish [51, 52], hybridization can occur between two species, where they produce a new offspring that carries genetic material from both parents. Taxon b in Figure 2.5(b) is an example of a hybrid taxon, having both a and c as its parents.

While both events can be summarized as the non-vertical transfer of genetic material, HGT and hybridization differ in the amount; HGT transfers a fraction of genetic material, whereas in hybridization, both parents contribute a similar portion of material. This difference is implicitly reflected in the networks in Figure 2.5: the network modeling HGT, (a), has a clear species tree that should explain a majority of genetic inheritance, while the network modeling hybridization, (b), does not set up a species tree. However, the difference is difficult to detect in practice. Because of this difficulty, Figure 2.5(a) and (b) are sometimes considered identical at the notation level, since they induce the same set of trees.

2.3.2 Detecting HGT

We assume that species evolution can be represented by a species tree. Since hybridization does not comply with this assumption, we will exclusively deal with HGT in this dissertation and use HGT interchangeably with reticulation. The phylogenetics community has attempted to detect HGT events using different types of data.

Sequence-based HGT Detection Algorithms that take the genome sequence of a species and detect the HGT events in the species based on the compositional characteristics, such as GC content, codon usage, and di- and tetra-nucleotide frequencies of the sequence are based on the observation that these compositional characteristics are unique to each species [53, 54, 55, 56, 57]. They are usually easier to apply, because they require only the genome of the organism under study. However, the compositional characteristics not only differ by species, but also vary by genes or their function. Also, it is not intuitive to interpret the signal of a compositional characteristic. Because they sometimes do not agree on the HGT estimates and contradict

each other [58, 59], it is not easy to tell which of the estimates are more significant and which are less significant.

Tree-based HGT Detection Another approach for HGT detection uses the trees as data and addresses the incongruences between trees. As the SPR operator simulates the HGT event, calculating the SPR distance between the species tree and a gene tree has been widely used to estimate the lower bound on the number of HGT events on the gene tree. Algorithms introduced in Section 2.1.3 are the examples. With the advent of multi-locus data for an increasing number of species, the problem of incorporating multiple gene trees has recently arisen. There are a few approaches for the problem including ours, and they will be discussed in more detail in Chapter 5. While phylogenetic-based approaches not only can detect HGT events but also can identify their placement in evolution, it is true that the performance of the approaches largely depends on the accuracy of the reconstructed gene trees and the extensiveness of the gene sampling of the trees.

HGT Detection from Trees and Sequences In order to reduce error in the gene tree reconstruction while maintaining the advantages of tree-based approaches, one can use the sequence alignments in reference to the species tree. Based on the species tree and the sequence alignments, Maximum-Parsimony (MP) and Maximum-Likelihood (ML) optimization criteria [60, 61, 62] estimate the optimality of an HGT event with respect to the sequence alignment. Also for each criterion, some efficient search algorithms for the optimal HGT events have been proposed [63, 45]. In Chapter 3 and Chapter 4, we study the performance of these optimization criteria in detecting significant HGT events and propose computational methods to estimate their statistical significance.

2.3.3 Duplication and Loss

Gene duplication is another process responsible for the species tree/gene tree incongruence. It refers to any amplification of genomic stretches containing a gene. It can occur as an error in the process of homologous recombination, a retrotransposition event, or a whole-genome duplication [64]. Since the duplicated copy is thought to be less subject to selective pressure and more freely mutable, it usually works in developing genes that specify pre-existing functions (sub-functionalization) or genes of novel functions (neo-functionalization) [65]. Also, the duplications of oncogenes are often found in the progression of many types of cancer, including breast cancer and cervical cancer [66].

After a gene is duplicated into two loci in the evolutionary history, their gene copies will evolve and descend independently of each other after the duplication point. As evolution is accompanied by loss events, it can cause the gene trees to be incongruent with the species tree. See the species tree and gene tree in Figure 2.7, for example. In the figure, (a) represents the species tree and (b) an incongruent gene tree. This incongruence can happen if the gene is duplicated into the red gene and the blue gene copy at node u ; the red gene is lost at b and c , and the blue is lost at a as illustrated in Figure 2.7(c).

Duplication and loss contribute to the gene tree-species tree incongruence in particular ways. Based on the pattern of incongruence they can make between trees, the phylogenetics community has attempted to reconcile the trees by identifying the underlying events. In this sense, as we review the algorithms to detect evolutionary events, we focus on tree-based approaches, where the species tree and the gene tree under study are fed, and the events on the gene tree are detected by reconciliation with the species tree. Formally, a reconciliation between gene tree G and the species

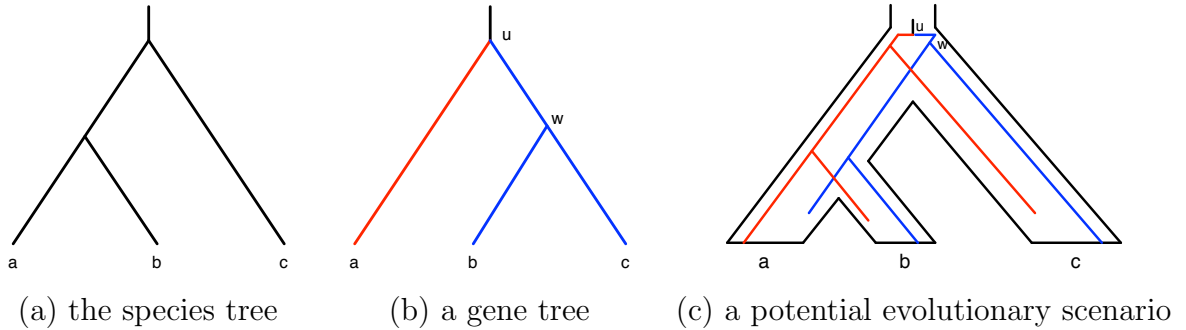


Figure 2.7 : Duplication and loss events that cause gene tree-species tree incongruence. (a): the species tree of taxa a , b and c . (b): a gene tree with an incongruent evolutionary relationship to the species tree. (c): a potential scenario where the gene takes the branching order of the tree within the branches of the species tree.

tree S corresponds to a mapping from $u \in V(G)$ to $x \in V(S)$. Since the biological validity of a mapping depends on the events the model assumes, they will be explained in detail in the following chapters.

2.3.4 Detecting Duplication and Loss (DL)

The Duplication and Loss (DL) model assumes that only duplication and loss cause the incongruence between a gene tree G and a species tree S ; a valid mapping between the trees can be defined as a map $\gamma : V(G) \rightarrow V(S)$ that satisfies the following properties:

- *order-respecting* : $u \leq_G v \Rightarrow \gamma(u) \leq_S \gamma(v)$,
- *leaf-label-respecting* : $\mathcal{X}(u) = \mathcal{X}(\gamma(u))$, $u \in L(G)$,
- *leaf-set-respecting* : $L(u) \subseteq L(\gamma(u))$.

Since an internal node $u \in V(G)$ is due to a duplication event if $L(\gamma(v)) \cup L(\gamma(w)) \neq \emptyset$, where v, w are the children of u , the count of the nodes satisfying

the condition under γ indicates the duplication cost of the mapping. Formally, with the definition of $\delta_\gamma(u)$ that is set to 1, if the internal vertex u satisfies the condition, and 0 otherwise, the duplication cost of γ , $dc(\gamma)$ can be defined as

$$dc(\gamma) = \sum_{u \in V(G)} \delta_\gamma(u) \quad (2.2)$$

The γ with the minimum $dc(\gamma)$ is obtained by Least-Common-Ancestor (LCA)-mapping M from G to S [67]. First let's define the common ancestors and the LCA. A common ancestor of set $A \subseteq L(G)$ is defined as $u \in V(S)$ such that $A \subset L(u)$. It is easy to see that there can be multiple nodes in the tree can satisfy the definition. In the set U of the common ancestors, $v \in U$, where $v \leq_S u, \forall u \in U$ is the LCA. The LCA of the set of nodes v, w is denoted as $LCA(v, w)$. With the definition of LCA , LCA -mapping $M : V(G) \rightarrow V(S)$ is defined as follows:

$$M(u) = \begin{cases} f(u) & \text{if } u \in L(T) \\ LCA(M(v), M(w)) & \text{if } u \text{ has children } v, w. \end{cases}$$

$dc(\gamma)$ corresponds to the case $L(\gamma(v)) \cup L(\gamma(w)) \neq \emptyset$. Since minimizing $L(\gamma(u)), \forall u \in V(G)$ while satisfying the three properties for a valid mapping leads to the case $L(\gamma(v)) \cup L(\gamma(w)) \neq \emptyset \forall u \in V(G)$, and M makes the valid mapping while minimizing $L(\gamma(u)), \forall u \in V(G)$, M is the most parsimonious mapping under the model, as $dc(\gamma)$ is taken as the parsimony score. Note that given γ under DL, internal nodes in G are assigned evolutionary events, either duplication or speciation. When the rates of gene duplication and loss are low, the γ from the parsimonious reconciliation appears to produce a good estimate of the duplication and loss events [68, 69, 70, 71].

However, some genes, such as the MHC gene family or the olfactory receptor genes, have high rates of duplication and loss, yielding cases where the parsimony-based approach might not work well. Model-based approaches for reconciling gene trees and species trees include those use coalescence models to describe incomplete lineage sorting [72, 73], model the events with the birth-death process [74, 75, 76, 77] and assume that the probability distribution for the number of duplication events is available for each branch [78].

Chapter 3

Phylogenetic Networks from Gene Sequence Alignments under Maximum Parsimony (MP)

This chapter presents our method for assessing significance of the reticulate edges returned in the reticulation search under MP under the assumption that reticulation is the sole cause for the gene tree/species tree incongruence with the species tree and the alignments of the gene sequences as input. Given the input, our method estimates the significance of the returns of the search, and applies the estimation to decide the stopping point of the search with the aim of addressing the overestimation issue of the optimality criterion. The experiment shows that the idea produces good performance both in the simulation study and in the biological data analysis. Note that since the parsimony score of a gene sequence alignment does not affect that of other gene sequence alignments, we conduct all experiments based on single gene sequence alignments. For multi-locus data, the parsimony score computation can go gene by gene.

3.1 Introduction

As mentioned above, phylogenetic networks have been introduced as a representation of the species evolution. In the study of phylogenetic networks, one of the primary researches has been on reconstructing the species evolutionary history using the phylogenetic network, referring to as phylogenetic network reconstruction problem. One of the most commonly used criteria for reconstructing phylogenetic trees is *maxi-*

mum parsimony (MP). Under this criterion, the phylogenetic tree that best fits a sequence data set is one that minimizes the total number of mutations over all possible tree topologies and sequence assignments to internal nodes of the tree topologies. There is a polynomial time algorithm for computing the parsimony length of a fixed phylogenetic tree leaf-labeled by a set of sequences, due to [79], while solving the MP problem (i.e., reconstructing the optimal phylogenetic tree under MP) in general is NP-hard [80, 81]. Nonetheless, several heuristics that solve the MP problem efficiently, and with high accuracy, in practice have been devised, such as the ones implemented in the phylogenetic software tool PAUP* [82].

In the early 1990's, Jotun Hein extended the maximum parsimony (MP) criterion to allow for modeling the evolutionary history of a set of sequences in the presence of recombination [83, 84]. More recently, Nakhleh *et al.* gave a mathematical formulation of the MP criterion for phylogenetic networks [60], and later studied its performance on biological as well as simulated data sets [61]. The main observation behind defining MP (and other criteria) for phylogenetic networks is that a sequence data set whose evolution involves reticulation events, such as horizontal gene transfer, can be partitioned into smaller, non-overlapping regions each of which has a treelike evolutionary history. Based on this observation, an optimal phylogenetic network under the MP criterion is one that *contains* (*induced*, or *displays*) the set of trees that best fit the evolutionary histories of the smaller regions. More formally, for a set S of sequences that can be partitioned into regions S^1, \dots, S^k , such that each region has a treelike evolutionary history, the parsimony length of a phylogenetic network N leaf-labeled by S is

$$PS(N, S) = \sum_{1 \leq i \leq k} \min_{T \in \mathcal{T}(N)} PS(T, S^i), \quad (3.1)$$

where $PS(T, S^i)$ denotes the parsimony length of region S^i on tree T , where T ranges over the set $\mathcal{T}(N)$ of all trees contained inside network N ; see [60] for more details. At the lowest level of atomicity, each region contains a single nucleotide, corresponding to the scenario where each site may evolve independently of its neighboring sites. This level of atomicity may be appropriate, for example, for analyzing single nucleotide polymorphism (SNP) data in a population, since, depending on the rate of recombination in the genomic region under study, it may be plausible to have adjacent SNPs “switching” evolutionary histories. However, in a phylogenetic study involving several species, taking each region to correspond to a single site is unrealistic, and may cause serious problems (such as adding an excessive number of reticulation events to the network so as to fit the evolution of each single site with no homoplasy). In our studies, and given that we seek to find whether a certain gene is horizontally transferred, we take each gene to be a single block. The minimization in Formula (3.1) indicates that the MP tree, among all trees contained in N , is chosen for each region, and the summation implies independence among the regions. In other words, in a phylogenetic analysis, S^1, \dots, S^k may correspond to k loci. In the discussion below, we focus exclusively on the formulations for a single locus (or, a single region).

One of the major challenges of applying the MP criterion to phylogenetic network evaluation and reconstruction is the computational complexity. As phylogenetic trees are a special case of phylogenetic networks, the problem of inferring a phylogenetic network under the MP criterion is NP-hard. Even the problem of computing the parsimony length of a *fixed* phylogenetic network is NP-hard* [85], unlike the case of trees, which is solvable in polynomial time [79]. Jin *et al.* have provided an array of

*Nonetheless, the problem of computing the parsimony length of a fixed phylogenetic network is *fixed parameter tractable*, where the parameter is the number of reticulation events (nodes of indegree 2) in the phylogenetic network.

algorithmic techniques that allow for inferring phylogenetic networks under the MP criterion in a reasonable amount of time [85, 63, 62].

A potentially more serious challenge of applying the MP criterion to phylogenetic networks concerns the overestimation of the true amount of reticulation in the evolutionary history of a sequence data set. Based on Formula (3.1), if N' is a phylogenetic network obtained by adding extra reticulation nodes to another network N , then $PS(N', S) \leq PS(N, S)$, simply because in this case we have $\mathcal{T}(N) \subseteq \mathcal{T}(N')$ (this is Observation 1 in [61]). In other words, under Formula (3.1), adding extra reticulation nodes to a phylogenetic network either leaves the parsimony length unchanged or improves it; it never makes it worse. Overestimation of the amount of reticulation in an evolutionary history, then, is inevitable under this formulation of the MP criterion. In particular, given a sequence alignment S of m sites, with site i exhibiting c_i states (e.g., $1 \leq c_i \leq 4$ for DNA), a phylogenetic network on which the evolution of each site is *homoplasy free* can be reconstructed. That is, we can infer a network N such that

$$PS(N, S) = \sum_{1 \leq i \leq m} (c_i - 1).$$

In this chapter, we focus on the horizontal gene transfer (HGT) version of the phylogenetic network reconstruction problem, in which a species tree ST and a sequence alignment of a gene S are given, and a set of edges is sought whose addition yields a network that fits the data under the MP criterion. The *ad hoc* solution to this problem that was adopted in [61] was to observe the improvements in the parsimony length as more HGT events are added, and stop the process when the improvement is below a certain threshold. Such a solution does not provide a systematic way of determining the “right” number of HGT edges. Further, it is not applicable in studies

that require a large number of analyses, such as simulation studies.

In this chapter, we address this problem in a more systematic way [86]. We propose a bootstrap method for estimating the support of an inferred HGT edge. Given a sequence alignment S , the method generates ℓ sequence alignments with the same dimensions as S by sampling (with replacement) sites from S , infers HGT edges based on the MP criterion for each sample, and finally assesses the support of each HGT edge based on its frequency in the analysis of all samples. In addition to assessing the support of the placement of an HGT edge, this method can be used to determine when to stop adding such edges.

We have implemented the method in our NEPAL software tool (available publicly at <http://bioinfo.cs.rice.edu/>), and studied its performance on both biological and simulated data sets. While our studies show very promising results, they also highlight issues that are inherently challenging when applying the maximum parsimony criterion to detect reticulate evolution. In particular, they show that the maximum parsimony criterion may not distinguish among a set of neighboring tree edges, as to which one is the true donor of a horizontal gene transfer event. In this case, we propose a relaxed version of the support formula. Further, we find that resolving non-binary nodes in the species tree, prior to the MP analysis, may help in the accuracy of the inferences made.

3.2 Materials and Methods

3.2.1 Maximum Parsimony of Phylogenetic Networks

A phylogenetic network is a rooted, directed, acyclic graph, leaf-labeled by a set of taxa, coupled with a set of temporal constraints [87]. In the case of HGT, a

phylogenetic network is obtained by adding a set of *horizontal*, or *lateral*, edges to an underlying species tree, where those horizontal edges capture the horizontal transfer events that may have occurred during the evolution of a certain gene under study. More precisely, if T is a phylogenetic (species) tree, we obtain a phylogenetic network N with a single HGT edge from tree T by selecting two edges $e_1 = (u_1, v_1)$ and $e_2 = (u_2, v_2)$ in T , splitting each of them, so that these two edges are replaced by four edges $e'_1 = (u_1, x_1)$, $e''_1 = (x_1, v_1)$, $e'_2 = (u_2, x_2)$, $e''_2 = (x_2, v_2)$, and finally a horizontal edge (x_1, x_2) is added. For example, in Figure 3.1, an HGT edge H is added in this fashion from edge B to edge E in the phylogenetic tree; the rectangular nodes in the phylogenetic network correspond to the splitting points of the two original edges B and E . It is important to note that when repeating this process to add other HGT edges, the procedure never splits a horizontal edge.

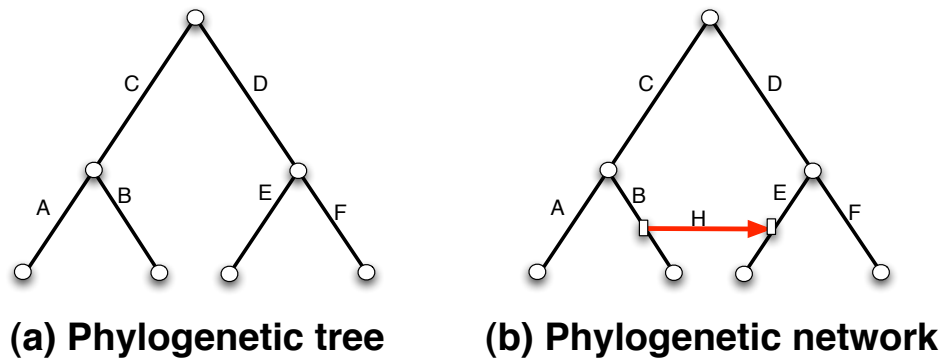


Figure 3.1 : A phylogenetic tree (a) and a phylogenetic network obtained from it by adding a horizontal edge H from edge B to edge E .

A tree is *contained* in a phylogenetic network if it can be obtained from the network by the following two steps: (1) for every node in the network, remove all but one of the edges incident into it (i.e., the edges whose head is the node under consideration);

(2) for every node u with a single parent p and a single child c , remove u and the two edges incident to it, and add a new edge from p to c (repeat this step as long as such nodes as u exist). Given a phylogenetic network N , we denote by $\mathcal{T}(N)$ the set of all trees contained inside N .

The parsimony length of a phylogenetic network N leaf-labeled by a set of sequences S is given by Formula (3.1) in the Background section, as formulated in [60]. The maximum parsimony problem in the context of phylogenetic networks is, for a given sequence alignment S , to infer the phylogenetic network N that minimizes $PS(N, S)$. In this chapter, the reticulate evolutionary events we consider are horizontal transfers on individual genes (HGT). In this context, the version of the maximum parsimony problem that we seek to solve is to find for a given (species) tree ST and a gene sequence data set S , a network N , obtained by augmenting ST with a set of HGT edges, that minimizes $PS(N, S)$.

As illustrated in Observation (1) of [61], and reviewed above, this definition of MP on phylogenetic networks does not penalize complexity of the inferred model, instead favoring networks with larger numbers of HGT edges. Two questions arise:

1. When should a method stop adding HGT edges under the MP criterion?
2. How supported are HGT edges that are inferred by the MP criterion?

Combined together, answering these two questions amounts to assessing the significance of a phylogenetic network inferred by the maximum parsimony criterion. To the best of our knowledge, neither of these two questions has been addressed in a systematic way. In the next section, we propose a bootstrap-based method for addressing both questions.

3.2.2 Inferring Well-Supported Phylogenetic Networks

Assume the HGT edge $h : X \rightarrow Y$ is inferred by the MP criterion on phylogenetic network N and sequence data set S . To assess the significance of h we generate ℓ sequence alignments, S_1, \dots, S_ℓ , with the same dimensions of S , by sampling (with replacement) sites from S , and for each sequence alignment S_i , we redo the calculation of MP on N and S_i , and record the set H_i of all optimal HGT edges inferred. Then, the bootstrap-based support of h , $S(h)$, is calculated as

$$S(h : X \rightarrow Y) = \frac{|\{i : 1 \leq i \leq \ell, \exists h_i \in H_i, h_i = h\}|}{\ell} \times 100. \quad (3.2)$$

Relaxing the Support Formula: When Ambiguity Helps Pinpointing the exact location of an HGT edge is a very hard task in practice, which would be expected to affect the support of inferred HGT. Indeed, our experimental results show that the support of an HGT edge, as given by Formula (3.2), tends to be very conservative, due to the strict requirement that h_i and h must be identical (see Results and Discussion section). From our empirical analysis of the performance of MP, we found that the major cause behind a poor support of a correctly inferred HGT edge is that “neighbors” of the source may be as good candidates as the source itself under the MP criterion. We illustrate this in Figure 3.2. In the cartoon shown in Figure 3.2(a), four HGT edges, involving edge e as the recipient, were identified individually in 100 bootstrap samples, each with the associated support (out of 100). While none of them has good support, combined they produce a well-supported hypothesis of an HGT involving the *clade*, as shown in Figure 3.2(b).

Empirically, we found that this process of introducing ambiguity in the source of an HGT edge often involves immediate neighbor edges of the source. In other words,

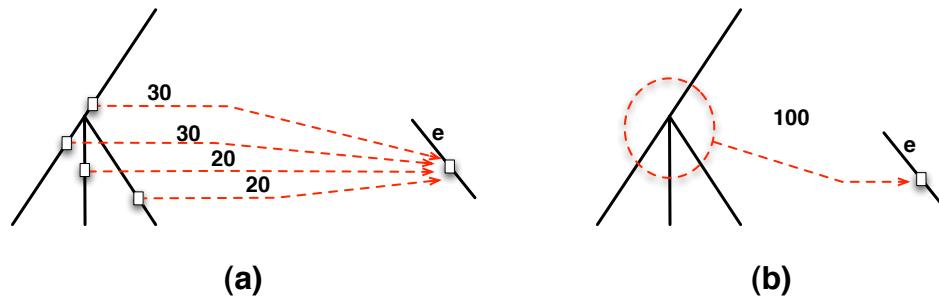


Figure 3.2 : (a) A scenario where none of four HGT edges identified individually in 100 bootstrap samples has good support (the recipient of each of the four edges is the same node v in the species tree). (b) When combined, thus allowing for ambiguity in pinpointing the exact source, a well-supported hypothesis of an HGT emerges.

we can refine Formula (3.2) of estimating the support of an edge $h : D(X) \rightarrow Y$, where $D(X)$ is a set of (neighboring) edges that correspond to potential sources, to obtain

$$S(h : X \rightarrow Y) = \frac{|\{i : 1 \leq i \leq \ell, \exists h_i = (X_i \rightarrow Y) \in H_i, X_i \in D(X)\}|}{\ell} \times 100, \quad (3.3)$$

where H_i is the set of all optimal HGT edges inferred in the i^{th} bootstrap sample. In the case when multiple best HGT edges H exist, a support value of H is computed as $\max_{h \in H}(S(h))$.

When Formula (3.3) is used on the cartoon scenario depicted in Figure 3.2(a), we obtain an HGT edge with perfect support, whose source is ambiguous, as illustrated with the dashed circle. It is important to note that in biological studies, a group of species, rather than a single specific one, is often reported as the source of a transfer event. This gives further justification for relaxing the formula. In Results and Discussion, we demonstrate the gains obtained by this relaxed formula in analyzing

the data of [88].

It is worth mentioning that while our analyses here always revealed ambiguity in the source of an HGT edge, it may be the case that for other data sets there is ambiguity in the recipient as well. In that case, Formula (3.3) can be extended by using $D(Y)$ instead and treating it in a similar fashion to the way $D(X)$ is treated. However, we did not find this to be the case in our analyses, and do not find this surprising. Replacing HGT edge $h : X \rightarrow Y$ by $h' : X' \rightarrow Y$ for $X' \in D(X)$ result in very local change to the topology of the resulting gene tree. On the other hand, replacing $h : X \rightarrow Y$ by $h' : X \rightarrow Y'$ for $Y' \in D(Y)$ results in a much greater change to the topology of the resulting gene tree (this depends on how far X and Y are in the species tree, a measure that we call “diameter” below).

As for how big of a neighborhood $D(X)$ (or, $D(Y)$) one may consider, in our analyses we found that the immediate “neighbors” of an edge are the most relevant. More precisely, if X is edge (u, v) in the underlying species tree, then $D(X)$ contains all edges emanating from either u or v , and the edge incoming into u . The reason behind defining the neighborhood $D(X)$ in this manner is that if an HGT edge $h : X \rightarrow Y$ results in improvement α to the total parsimony length, then replacing h by an edge $h' : X' \rightarrow Y$, where $X' \in D(X)$, results in an improvement to the parsimony length that is close to α .

Stopping Criterion Using the above formulas for bootstrap-based support of an HGT edge, we propose the following procedure for inferring a phylogenetic network under the maximum parsimony criterion starting from a species tree ST and a sequence alignment S of a gene of interest:

1. **Let** $N = ST$.

2. **While true**

- (a) **Compute** the set H of HGT edges, such that for each $h \in H$, $PS(N + h, S)^\dagger$ is minimum over all networks obtained by adding a single HGT edge to N .
- (b) Let $b = \max_{h \in H}(S(h))$ and $h' = \operatorname{argmax}_{h \in H}(S(h))$.
- (c) if $b > 70$
 - i. **Let** $N = N + h'$.
- (d) else
 - i. **Return** N .

In the above procedure, the network is initialized to the given species tree (Step 1). Then, the set H of all HGT edges whose addition results in the optimal improvement of the parsimony score is computed (Step 2a). If the maximum support of any edge in H exceeds 70 (out of 100), we add the edge and continue; otherwise, we stop adding edges (Step 2c). Hillis and Bull [89] showed that bootstrap values $\geq 70\%$ usually correspond to the “real” clade with very high probability, and this value has been widely accepted as an indication of good support [90]. Below we show that the value 70, as a threshold, works well in practice for the support of HGT edges.

If more than a single locus is involved in the analysis, then we have, as discussed above, a set of sequence alignments S^1, S^2, \dots, S^k , each corresponding to an individual locus. If these loci have evolved independently, then analyzing each of them individually, using the methodology described above, is sufficient. This may result, for example, in an HGT edge $h : X \rightarrow Y$ that has high support based on the analysis

[†]We write $N + h$ to denote the phylogenetic network resulting from adding HGT edge h to phylogenetic network N .

of locus i and low support based on the analysis of a different locus, j . This is not contradictory, since the support of an HGT edge is dependent on the data, and the support should be reported for each HGT edge and each locus independently. Now, let us consider the case when, for example, two loci i and j are depended (e.g., they are linked). In this case, one could concatenate the sequences from both loci and consider the resulting “supergene” as a single locus in the analysis. This, of course, requires determining if two loci are linked, a question whose treatment is beyond the scope of this chapter. Nevertheless, we conjecture that analyzing each gene independently, even when the independence assumption does not hold, may be a safe choice, particularly if enough sites are available for each locus.

3.2.3 Data Sets

We have implemented the method described above in the NEPAL tool, which is available publicly at <http://bioinfo.cs.rice.edu/>. Using species trees and sequence alignments of genes from biological and simulated data, we studied the performance of our method in identifying the amount of HGT as well as location of those HGT events in the data sets.

Biological Data We studied 20 out of the 31 mitochondrial gene data sets, which were collected from 29 diverse land plants and analyzed in [88]. These are *cox2*, *nad2*, *nad3*, *nad4(ex4)*, *nad4(exons)*, *nad5*, *nad6*, *nad7*, *atp1*, *atp8*, *ccmB*, *ccmC*, *ccmFN1*, *cox3*, *nad1*, *rpl16*, *rps19*, *sdh4*, and three introns *nad2intron*, *nad5intron* and *nad7intron*. We used a species tree for the data set based on information at NCBI

(<http://www.ncbi.nih.gov>) and analyzed the entire data set with both seed and

non-seed plants together. For each gene data set, we restricted the species tree to those species for which the gene sequence was available. We compared HGTs we have identified with the result of Bergthorsson *et al.* It is important to note that in their analyses, Bergthorsson *et al.* focused only on genes that were horizontally transferred to the (mitochondrial genome of) *Amborella*; in other words, they did not consider HGT events that may not have involved *Amborella*.

Simulated Data We used PhyloGen [91] to generate two 50-taxon species trees ST_1 and ST_2 under the birth-death model. More precisely, we used the following settings for the PhyloGen tool:

```
birthdeath birth=1 death=0 extant=50
generate replicates=2
```

For each species tree, we simulated ten DNA sequence alignments of length 1000 under the Kimura 2-Parameter model, involving HGT events, using the tool of [92]. To achieve this, we used the following settings for the tool:

```
nb_genes      10
diameter      1.  1.
sampling      100  100
seq_type      DNA
seq_length    1000 1000
total_rho     0
total_tau     1
total_rho_prime 0
alpha_l       1.
```

```

alpha_s          0.5
subst_model      K80
subst_rates     2

```

We modified the tool of [92] so that it also prints the actual HGT events it simulates. We label the 20 generated gene data sets as $GS_{1,1}, \dots, GS_{1,10}, GS_{2,1}, \dots, GS_{2,10}$. The actual number of HGT events involved in each of the genes is reported in Table 3.3.2.

3.3 Results and Discussion

We have analyzed the biological and synthetic data by applying the procedure given above, to assess the confidence of the postulated HGT edges and determine the number of HGT events by the confidence. For our experiments, we generated 100 sequence alignments by sampling sites with replacement from the original alignment, in all cases for the biological and simulated data analysis.

3.3.1 Biological Data

The numerical results of analyzing the 20 gene data sets of [88] are given in Table 3.1, while the inferred phylogenetic networks with strong support for the inferred HGT events for 13 of the gene data sets are shown in Figure 3.3 (for the other 7 data sets, our method did not identify any HGTs). The three columns under the header [88] in Table 3.1 correspond to the number of HGTs postulated by Bergthorsson *et al.*, the donor group, and support value for each postulated HGT event, as calculated by the test of [93], respectively.

Bergthorsson *et al.* reported the groups of species to which the donor(s) of horizontally transferred genes belong, rather than the specific donor. In particular, they

Gene	Bergthorsson <i>et al.</i> [88]			MP analysis					
	#HGTs	donor	SH	b1	b2	b3	b4	#HGTs	F?
cox2	3	M	<0.001	100	38	—	—	1	Y
		E	NS						Y
		E	NS						—
nad2	2	M	<0.001	100	62	—	—	1	Y
		E	NS						Y
nad4(exons)	1	M	<0.001	99	98	44	—	2	Y
nad4(ex4)	1	E	NS	58	—	—	—	0	Y
nad5	2	M	<0.001	100	95	84	35	3	Y
		A	0.025						Y
nad6	1	B	<0.001	100	26	—	—	1	Y
nad7	2	M	<0.001	99	64	—	—	1	Y
		E	NS						Y
atp1	1	E	0.001	98	33	—	—	1	Y
atp8	1	E	0.008	75	38	—	—	1	Y
ccmB	1	E	NS	39	—	—	—	0	Y
ccmC	1	E	0.03	68	—	—	—	0	Y
ccmFN1	1	E	0.004	80	86	37	—	2	Y
cox3	1	A	NS	69	—	—	—	0	N
nad1	1	E	<0.001	100	88	25	—	2	Y
rpl16	1	E	NS	46	—	—	—	0	Y
rps19	1	E	0.003	100	61	—	—	1	Y
sdh4	1	E	NS	35	—	—	—	0	Y
nad2intron	1	M	—	66	—	—	—	0	Y
nad5intron	1	M	—	97	41	—	—	1	Y
nad7intron	1	M	—	100	67	—	—	1	Y

Table 3.1 : Mitochondrial gene data sets and HGTs postulated by Bergthorsson *et al.* and those computed by the MP analysis (NEPAL). ‘donor’ denotes the group from which the gene was transferred (in all cases, the recipient is *Amborella*). ‘SH’ denotes support of the HGT events as computed by the Shimodaira–Hasegawa (SH) test and reported by Bergthorsson *et al.* (values lower than 0.05 indicate high support, and NS indicates support is not significant). The ‘b1’, ‘b2’, ‘b3’, and ‘b4’ columns correspond to the support values from Formula (3.3) for adding the first, second, third, and fourth HGT edges inferred by the MP analysis. Since adding HGT edges stops once a weakly supported edge is encountered, a ‘—’ entry under these columns indicates that adding HGT edges was stopped before. B = Bryophyte, M = Moss, E = Eudicot, and A = Angiosperm.

focused on four groups: Bryophytes, Moss, Eudicots, and Angiosperms. For the recipient, the authors only focused on *Amborella*. Of the 25 HGT events that Bergthorsson *et al.* postulated, 13 were supported, 9 unsupported, and 3 (the 3 intron data sets) had no reported support.

The ‘b1’, ‘b2’, ‘b3’, and ‘b4’ columns under the MP analysis in Table 3.1 correspond to the support values from Formula (3.3) for adding the first, second, third, and fourth HGT edges inferred by the MP analysis. Since adding HGT edges stops once a weakly supported edge is encountered, a dash entry under these columns indicates that adding HGT edges was stopped before. The ‘#HGTs’ lists the number of HGT edges inferred based on the support using the threshold value 70 (see discussion above of the choice of this threshold). In other words, it is the count of non-dash entries minus one in the bootstrap-value columns. The ‘F?’ column lists in each row whether the HGT postulated by Bergthorsson *et al.* and reported in that row was also found by the MP analysis. The row in gray refers to the case where the HGT postulated by Bergthorsson *et al.* was found by the MP analysis, but with support smaller than 70 (the support of the edge was 68).

Of the 13 HGTs reported in [88] with high support according to the [93] test, the MP analysis with bootstrap supports identified 12, missing one HGT for *ccmC* that has a support value of 0.03 by SH test. While the MP analysis postulated the right HGT edge from the Eudicot group to *Amborella* (in the sense that the edge resulted in the best improvement in the parsimony length), the bootstrap-based support for this edge was 68, which is lower than the threshold of 70. It is worth mentioning that the SH test reports the weakest support for this case compared to other cases (that are not ‘NS’). Further, from the perspective of the parsimony length of the resulting network, postulating the HGT edge for this gene only improves the parsimony length

by 6. In other words, this edge has very low support based on all three criteria: parsimony length improvement, bootstrap-based support, and the SH test.

The three HGT edges postulated by Bergthorsson *et al.* for the intron data sets, and which had no support values based on the SH test reported, were all identified by the MP analysis. The HGT edge from the Moss group for the *nad2intron* gene is not well supported, while the HGT edges for the *nad5intron* and *nad7intron* data sets are both strongly supported.

Of the other 9 HGT events reported by the authors with no significant support based on the SH test, the MP analysis identifies seven HGT edges, missing the other two. The identified seven HGTs were all from the Eudictots to *Amborella*, and they were in the *cox2*, *nad2*, *nad4(ex4)*, *nad7*, *ccmB*, *rpl16*, and *sdh4* data sets. However, none of them is strongly supported according to the bootstrap-based analysis, which is consistent with the SH test results.

In four data sets, the MP analysis identified HGT edges in addition to those reported in [88]. However, none of these edges involved *Amborella*. One possible explanation for why these edges were not reported in [88] is because the authors focused only on HGT events involving *Amborella*. Another explanation may be the inaccuracy of the parsimony criterion as raised in the preceding section.

Figure 3.3 shows the phylogenetic networks of 13 of the 20 biological data sets. Each of the HGTs in the networks is marked as ‘Hi’ representing the *i*-th HGT identified by the MP analysis, and labeled with a bootstrap support value. We used the relaxed bootstrap support value, as given by Formula 3.3, in 10 out of 13 cases for locating the clade of the source of an HGT since it is hard to identify their exact locations. In 7 cases, clades of the source locations are identified and represented with circles in the figure. Among these 7 cases, *atp1*, *cox2*, *nad5*, *rps19*, *nad5intron*, and

nad7intron have very high bootstrap support (above 97) for the transfers to *Amborella* from clades of their source locations. In *cox2*, all three locations inside the circle of ‘H1’ are identified as equally good sources of an HGT with perfect support. Others show significantly improved bootstrap supports when the sources are identified as a clade instead of an exact location. In some cases, multiple branches with individual bootstrap values labeled are used instead of a clade for identifying more precise source locations of the HGTs. In these cases, the relaxed bootstrap values are marked after the joint points of the branches. Transfers identified by MP but not well supported are not shown in the networks in Figure 3.3. Refinements, marked with solid circles, are performed for unresolved branches in *nad5*, *ccmFN1*, *nad5intron*, and *nad7intron*, based on MP scores. The MP scores for these four datasets are improved from 927 to 909, from 234 to 227, from 688 to 650, and from 950 to 900 with the marked refinements.

3.3.2 Simulated Data

The numerical results of analyzing the synthetic gene data sets are given in Table 3.3.2. The columns under the ‘true HGTs’ list the number of HGT edges added by the tool of [92], and $d1$ and $d2$ denote the distance, in terms of the number of branches on the species tree, between the source and recipient of each of the HGT events simulated. When no HGT events are simulated, neither value is provided, and when only one is simulated, only $d1$ is specified. The reason for computing these values is to study the performance of the MP criterion on data sets with varying HGT event *diameters* (the distance between source and recipient), as we hypothesize that as the *diameter* becomes smaller, the performance of the MP analysis may become poorer. An entry with value p^* indicates that the diameter is p , but that the event

is from a branch to another branch that is its descendant in the species tree. While this seemingly contradicts temporal constraints (e.g., that the source and recipient co-exist in time), such a scenario can be explained through extinction or incomplete taxon sampling of taxa; see [87].

Under the ‘MP analysis’, we report the support of the inferred edges as before (‘b1’, ‘b2’ and ‘b3’), the number of HGT edges detected (‘#HGTs’), and whether the true ones were found (‘F1?’ and ‘F2?’), respectively. In this case, a dash entry in the support value columns indicates that the support was not calculated since it was determined already to stop adding HGT edges (i.e., the support for a preceding entry was already < 70).

In this case, for each $GS_{i,j}$ ($i \in \{1, 2\}$ and $1 \leq j \leq 10$), if there are m true HGTs, we report the support value of the best $m + 1$ HGTs inferred by the MP analysis, even if the bootstrap-based stopping criteria indicated stopping the addition of HGT edges at a value smaller than m . The rows in gray refers to the cases where the bootstrap-based approach failed to stop with the right amount of HGT.

The results show that when the number of true HGTs, as simulated in the data, is 0, the MP analysis detected no reticulation (or, HGTs) in the data, as the support for adding the first HGT edge is < 70 in all cases with one exception of ($GS_{1,9}$). For the cases where the true number of HGTs is 1, there are only two cases where according to the bootstrap-based support no HGTs were postulated, while the correct number of HGTs was postulated in the other eight cases.

It is interesting to note that all cases in which the bootstrap-based method fails to determine the right number of HGT edges have small diameter values. The bootstrap underestimated the true number of HGTs in two cases ($GS_{2,2}$ and $GS_{2,8}$), inferring incorrectly that the number of HGTs is 0. The horizontal transfer in $GS_{2,2}$ and

Gene	true HGTs			MP analysis					
	#HGTs	d1	d2	#HGTs	F1?	F2?	b1	b2	b3
<i>GS</i> _{1.2}	0	—	—	0	—	—	15	—	—
<i>GS</i> _{1.3}	0	—	—	0	—	—	43	—	—
<i>GS</i> _{1.7}	0	—	—	0	—	—	46	—	—
<i>GS</i> _{1.9}	0	—	—	1	—	—	84	45	—
<i>GS</i> _{2.1}	0	—	—	0	—	—	27	—	—
<i>GS</i> _{2.7}	0	—	—	0	—	—	47	—	—
<i>GS</i> _{2.9}	0	—	—	0	—	—	33	—	—
<i>GS</i> _{2.2}	1	2	—	0	N	—	55	19	—
<i>GS</i> _{2.8}	1	3*	—	0	N	—	24	33	—
<i>GS</i> _{1.1}	1	4	—	1	Y	—	100	47	—
<i>GS</i> _{1.6}	1	10	—	1	Y	—	95	45	—
<i>GS</i> _{1.10}	1	8	—	1	Y	—	76	39	—
<i>GS</i> _{2.3}	1	7	—	1	Y	—	100	60	—
<i>GS</i> _{2.4}	1	4	—	1	Y	—	77	38	—
<i>GS</i> _{2.5}	1	6	—	1	Y	—	100	37	—
<i>GS</i> _{2.6}	1	9	—	1	Y	—	100	47	—
<i>GS</i> _{2.10}	1	9	—	1	Y	—	100	15	—
<i>GS</i> _{1.4}	2	3*	5	2	Y	Y	100	77	13
<i>GS</i> _{1.5}	2	4*	7	2	Y	Y	100	100	41
<i>GS</i> _{1.8}	2	5*	7	2	Y	Y	100	98	19

Table 3.2 : Results of the MP analysis of 20 simulated data sets. The rows are sorted by the number of true HGTs simulated for each gene data set. The ‘d1’ and ‘d2’ columns denote the distance, in terms of the number of branches on the species tree, between the source and recipient of the first and second HGT events simulated, respectively. The ‘b1’, ‘b2’, and ‘b3’ columns correspond to the support values from Formula 3.3 for adding the first, second, and third HGT edges inferred by the MP analysis. Since adding HGT edges stops once a weakly supported edge is encountered, a ‘—’ entry under these columns indicates that adding HGT edges was stopped before.

$GS_{2,8}$ have diameters 2 and 3, respectively, which are the lowest values among all the simulated data sets. The small diameter of a transfer indicates that the transfer occurred from a branch to another that is almost its immediate sibling or descendant in the species tree. These cases are very hard for the MP criterion to detect, since it detects other HGT edges as yielding the best improvement to the parsimony score. This highlights a fundamental drawback of the MP criterion which is that the HGT edge resulting in the best improvement to the parsimony score is not necessarily the true one. This is not surprising, since MP suffers from similar issues even for reconstructing trees. The second HGT postulated by the MP analysis of the *nad5* gene differs from that reported in [88] for this very reason: the MP analysis identifies an edge that improves the parsimony score more than the one reported by Bergthorsson *et al.* (the one from *Angiosperm* to *Amborella*).

In the three simulated data sets with two true HGTs, the tool of [92] added one of the two edges from a branch to one of its (not immediate) descendants, making a very hard case for the bootstrap-based support method to detect. However, the MP analysis correctly identifies both HGT edges, and with very high support, in all three cases.

3.4 Conclusions

In this chapter, we revisited the maximum parsimony criterion for inferring phylogenetic networks. In previous studies, the criterion was shown to provide very promising results on both biological and simulated data. However, previous work did not provide the means to assess the significance of the number of reticulation events estimated nor the location of the inferred events.

We proposed a systematic measure to serve as a *stopping rule* to the otherwise

“overestimating-by-definition” criterion, and demonstrated their performance on 20 empirical data sets and 20 simulated data sets. From the result, it has been shown that bootstrap measure provided very accurate results in general. Further, we found that there are some boundary cases under which the MP criterion performs poorly. Finally, we point out that the bootstrap-based support formula that we presented here can be applied with any method that uses the gene sequences to infer HGT edges, such as maximum likelihood [45].

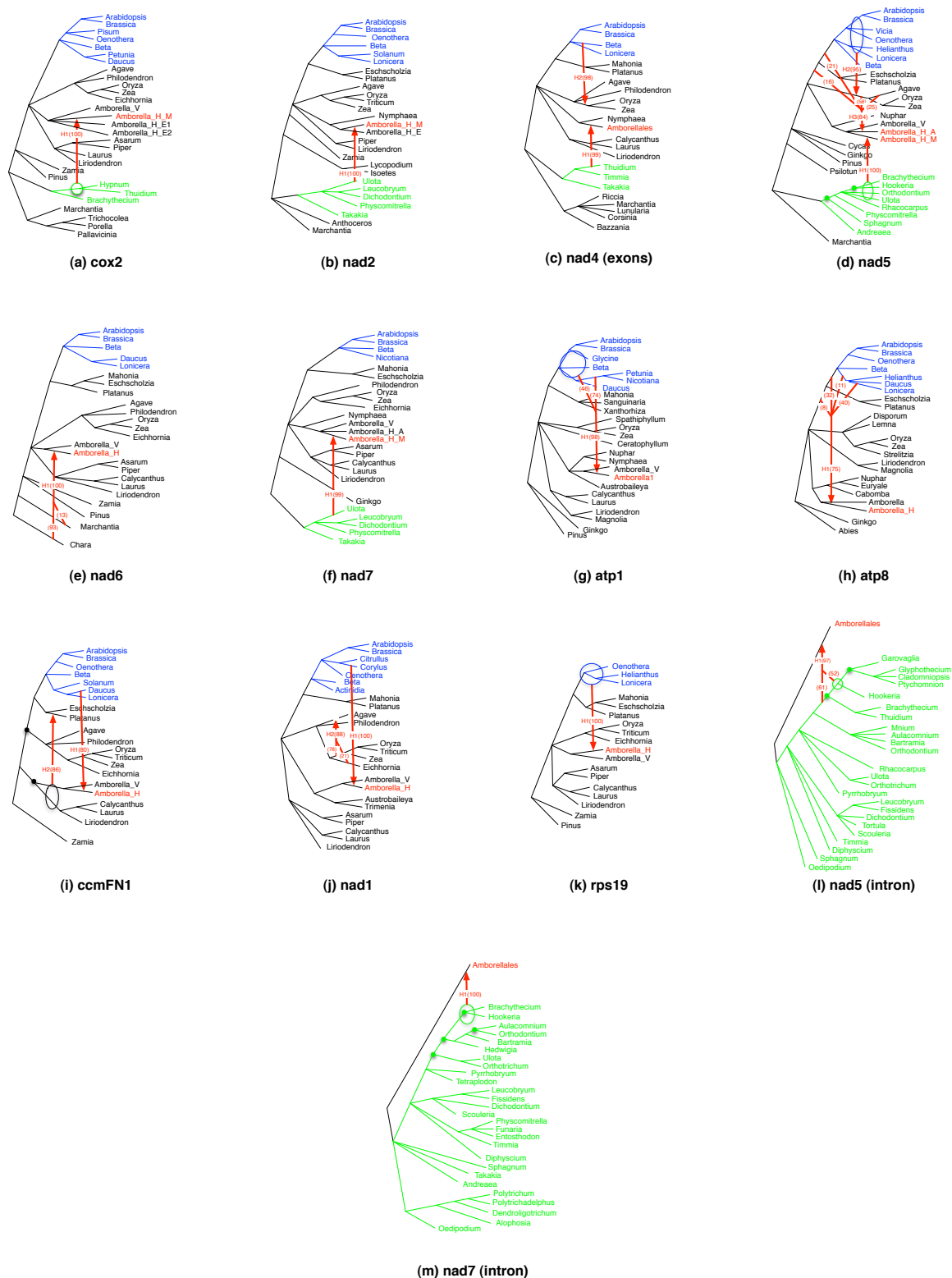


Figure 3.3 : HGT edges (in red) inferred by the MP criterion, with support values, in parentheses, computed based on Formula (3.3). Ambiguity in the source is denoted by a circle (when drawing a circle was possible) or a multi-source edge. Amborella genes are colored in red, and core eudicot genes and moss genes are colored in blue and green, respectively. Branch refinements are performed for *nad5*, *ccmFN1*, *nad5intron*, and *nad7intron* at the places marked with solid circles.

Chapter 4

Phylogenetic Networks from Gene Sequence Alignments under Maximum Likelihood (ML)

In the previous chapter, we develop an algorithm that assesses the significance of the reticulate edges under MP, and study the performance of the MP in detecting the reticulate events. In this chapter, we study the performance of the maximum-likelihood (ML) for the detection [94]. A maximum likelihood (ML) model has been proposed for this case and accounts for both mutation within a genomic region and reticulation across the regions. However, the performance of this model in terms of inferring information about reticulate evolution and properties that affect this performance have not been studied. In this chapter, we study the effect of the evolutionary diameter and height of a reticulation event on its identifiability under ML. We find both of them, particularly the diameter, have a significant effect. Further, we find that the number of genes (which can be generalized to the concept of “non-recombining genomic regions”) that are transferred across a reticulation edge affects its detectability. Last but not least, a fundamental challenge with phylogenetic networks is that they allow an arbitrary level of complexity, giving rise the model selection problem. We investigate the performance of two information criteria, the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC), for addressing this problem. We find that BIC performs well in general for controlling the model complexity and preventing ML from grossly overestimating the number of reticulation events. As more genomic data become available, accurate models for inferring evo-

lutionary histories that involve reticulations become essential, particularly in light of the increasing evidence of the significant role these evolutionary events play. Our findings establish ML as a good criterion for this task, yet highlight significant issues that must be accounted for when interpreting results obtained under this criterion.

4.1 Introduction

In a seminal paper, Maddison proposed a likelihood framework for inferring species trees by simultaneously accounting for evolutionary events within loci (that is, mutations at the nucleotide level) and across loci (that is, gene tree incongruence) [7]. The post-genomic era has highlighted and further stressed the need for inference under such a framework, as analyses of different data sets have revealed varying degrees of incongruence among gene trees; e.g., [95, 96, 97, 98, 99, 100]. All these analyses have focused on *deep coalescence* as the source of gene tree incongruence.

Another source of incongruence that has long been acknowledged by biologists and that is being increasingly revealed by phylogenomic analyses is *reticulate*, or, non-treelike, evolutionary events. For example, evidence shows that bacteria may obtain a large proportion of their genetic diversity through the acquisition of sequences from distantly related organisms, via horizontal gene transfer (HGT) [101, 102, 103, 13, 9, 104, 105, 106]. Furthermore, evidence of widespread HGT in plants has also emerged [14, 88, 15]. Interspecific recombination is believed to be ubiquitous among viruses [107, 108]. Hybrid speciation is a major evolutionary mechanism in plants, and it is also seen in groups of fish and frogs [109, 110, 111, 112, 113, 114]. Further, hybridization is believed to play an important role in speciation and evolutionary innovation in several groups of plant and animal species [115, 116].

When reticulate evolutionary events occur among species, the species phylogeny

takes the shape of a *network*, which is a rooted, directed, acyclic graph that extends the evolutionary tree model by incorporating non-vertical inheritance of genetic material [44]. Jin *et al.* restricted the maximum likelihood framework of Maddison [7] to the case where gene tree incongruence is exclusively due to horizontal gene transfer (HGT) events, thus providing a maximum likelihood formulation of the problem of inferring phylogenetic networks from sequence data. [45]

While the maximum likelihood (ML) formulation of Jin *et al.* showed a good performance in inferring reticulations on synthetic and biological data sets [45], it is not clear what parameters affect the performance of ML in general. We hypothesize the diameter of the reticulate evolutionary event (e.g., the distance between the source and donor of an HGT event) plays an important role in the detectability of such an event. Further, as more complex networks (that is, ones with more reticulations) necessarily fit the data better than simpler ones, it is important to address the overfitting issue [44]. In this chapter, we conduct simulation studies to assess the effect of the evolutionary diameter on the identifiability of reticulation events. Further, we investigate the performance of two commonly used information criteria for controlling for the complexity in inferred phylogenetic networks, namely the Akaike Information Criterion, or AIC, [117] and the Bayesian Information Criterion, or BIC, [118]. These criteria have been used for model selection in molecular phylogenetics and their performance has been assessed [119, 120]. Further, these criteria have been used in the context of phylogenetic networks recently to distinguish between reticulation events and incomplete lineage sorting [47, 121]. However, none of these works aimed at studying the performance of these criteria for the problem.

Our results show that ML performs well in terms of estimating reticulation probabilities, and less so in determine the location, or placement, of reticulation edges.

They also show that the diameter, reticulation probability, and number of gene data sets used combined have a significant effect on the performance. We find that BIC, and to a lesser extent AIC, performs very well in terms of model selection and preventing ML from grossly overestimating the amount of reticulation. Our analysis of two biological data sets that involve more complex evolutionary scenarios also demonstrate good performance of ML.

4.2 Methods

4.2.1 Phylogenetic Networks and Maximum Likelihood

Given a collection R_1, R_2, \dots, R_k of n genomic regions, and set $\mathcal{S} = \{S_1, S_2, \dots, S_k\}$, where S_i is the sequence alignment corresponding to region R_i , the likelihood function, as proposed in [45], is given by

$$L(N, \gamma | \mathcal{S}) = \prod_{S_i \in \mathcal{S}} \left[\sum_{T \in \mathcal{T}(N)} [\mathbf{P}(S_i | T) \cdot \mathbf{P}(T | N, \gamma)] \right], \quad (4.1)$$

where $\mathbf{P}(S_i | T)$ represents the tree likelihood score, and $\mathbf{P}(T | N, \gamma)$ is the probability of observing gene tree T , given phylogenetic network N and the reticulation probability γ . Notice that if we relax our constraint on the allowed evolutionary events, and allow for *deep coalescence* for example, then the term $\mathcal{T}(N)$ in Eq. (4.1) is replaced by the set \mathcal{T} of all possible trees on set \mathcal{X} of taxa, to obtain the original formula proposed by Maddison[7] (in addition to the γ factor, which was not accounted for by Maddison).

To compute the likelihood function, as given by Eq. (4.1), the term $\mathbf{P}(T | N, \gamma)$ is

computed as

$$\mathbf{P}(T|N, \boldsymbol{\gamma}) = \prod_{e \in \eta(T)} \gamma(e).$$

The maximum likelihood framework for inferring reticulation evolutionary histories from a set \mathcal{S} for loci amounts to identifying the phylogenetic network N (topology and branch lengths) along with the reticulation probabilities vector $\boldsymbol{\gamma}$ that maximize Eq. (4.1).

4.2.2 Information Criteria

Given a phylogenetic network N , it can be augmented into a phylogenetic network N' , by adding further reticulation nodes and edges. By definition of the set of trees contained within a network, we obtain the relationship $\mathcal{T}(N) \subseteq \mathcal{T}(N')$. Using this relationship in conjunction with Eq. (4.1), we obtain $L(N, \boldsymbol{\gamma}|\mathcal{S}) \leq L(N', \boldsymbol{\gamma}'|\mathcal{S})$, where $\boldsymbol{\gamma}'$ is the reticulation probabilities associated with phylogenetic network N' . In other words, augmenting the network results, in most cases, in a better fit of the data, and never in a worse fit [44]. Based on this observation, a phylogenetic network inference procedure that seeks the network that maximizes Eq. (4.1) without accounting for network complexity (in terms of the number of reticulation nodes) would produce unrealistic evolutionary histories with large numbers of reticulation events.

To address this issue, we explore in this chapter two information criteria, the Akaike Information Criterion, or AIC [117] and the Bayesian Information Criterion, or BIC [118], which are widely used for model selection problems. The AIC criterion is defined as

$$AIC = 2K - 2 \ln L, \tag{4.2}$$

where K is the number of parameters in the model, and L is the likelihood of the

estimated model.

BIC [118] measures the balance between goodness-of-fit and the noise based on the following formula:

$$BIC = K \ln n - 2 \ln L, \quad (4.3)$$

where K , L , and n are defined as above. When using these criteria, the model with the smallest value is sought.

In our context, K corresponds to the number of the branches of the network, L is given by Eq. (4.1), and n is the total number of genes in the sequence data.

Searching the phylogenetic network space

In this chapter, we are concerned with the performance of the ML criterion in terms of the number of reticulations it estimates, rather than in terms of speed. As such, we implemented an exhaustive search procedure that starts from an initial tree T , and then searches all networks obtained from T by adding a single reticulation node, identifying an optimal network N_1 , then all networks obtained from N_1 by adding a single reticulation node, etc. To add a reticulation node to a network (or tree), the procedure picks a pair of edges (u_1, v_1) and (u_2, v_2) , subdivide each edge into two edges of equal length (each of the two edges is half the length of the original edge that was subdivided), such that we have (u_1, x_1) , (x_1, v_1) , (u_2, x_2) , and (x_2, v_2) , and finally, it adds a horizontal edge between x_1 and x_2 (in either direction). It is important to note that in this procedure, when the pair of edges is picked for adding a reticulation node, cycles are excluded, as well as reticulation edges between two tree edges emanating from the same node (“sibling edges”).

Using this procedure, we analyzed the species tree (which is a network with 0 reticulation nodes), all networks with 1 reticulation nodes, all networks with 2 reticulation

nodes, etc. For each number of reticulation nodes, we maintain the network with the optimal value for the information criterion. Then, a network with $k + 1$ reticulation nodes is always formed by adding a single reticulation edge to the optimal network with k reticulation nodes. That is, the set of all networks with $k + 1$ reticulation nodes is not generated “from scratch” by adding $k + 1$ reticulation nodes in all possible ways to the initial tree T ; rather, it is generated by adding a single reticulation node, in all possible ways, to the optimal network with k reticulation nodes. In other words, we build the network model using forward selection with reticulation nodes as variables, rather than an exhaustive model building. Even though the feature selection approach inherits its own issues, it has been shown to provide good results [86, 45].

For each phylogenetic network, we also need to compute the reticulation probabilities γ that optimize Eq. (4.1). For this purpose, we used a grid search with values for each reticulation probability in the set $\{0.05, 0.1, \dots, 0.5\}$. Finally, to compute the probabilities $\mathbf{P}(S_i|T)$ in Eq. (4.1), we used the dnaml program packaged in Phylip [122].

To put it all together, given a phylogenetic network N with h reticulation nodes, we identify the optimal phylogenetic network N' with $h + 1$ reticulation nodes using the equation

$$(e^*, \gamma^*) = \operatorname{argmax}_{(e, \gamma)} L(N', \gamma | \mathcal{S}), \quad (4.4)$$

where (e, γ) range over all possible ways of adding a reticulation edge e with reticulation probability $\gamma \in \{0.05, 0.1, \dots, 0.5\}$ to produce phylogenetic network N' that differs from N by a single reticulation node. Here, the vector γ of reticulation probabilities includes those of phylogenetic network N and the reticulation probability γ of the new reticulation edge e .

Once the pair (e^*, γ^*) is identified, the phylogenetic network N' is obtained by adding reticulation edge e^* to N , with its reticulation probability γ^* .

Results

In this section, we investigate the effects of topological properties of reticulation events on the performance of an ML approach to phylogenetic network inference. Further, we study the performance of ML in terms of estimating the reticulation probabilities from sequence data, and then investigate how the three information criteria perform in terms of estimating the number of reticulation events in a data set.

For the synthetic data we analyze here, we used the PhyloGen program [91] to generate species trees under a birth-death model. Each species tree was then used to generate gene trees with HGT events using the tool of Galtier [92]. Since Galtier's tool does not give information about the actual HGT events simulated, we modified the tool so that it produces such information. We then used the Seq-gen tool [123] to simulate the evolution of DNA sequence data sets, each of length 100 sites, down each of the gene trees, using the K80 model with transition/transversion ratio of 2. As we modified the experimental setup to investigate each of the questions, we describe below the details of the remaining steps of the simulation setup that are specific to each study.

Effect of the diameter and height of reticulation events

Consider a set \mathcal{S} of k independent sequence alignments, each of which evolved down a (species) tree T . That is, the evolutionary history of \mathcal{S} is reticulation-free. Now, consider evaluating, under maximum likelihood a hypothesis that involves a single reticulation event along with its associated probability γ ; i.e., a phylogenetic network

N that induces the two trees, T and T' , where T' differs from T by the placement of a subtree due to a hypothesized reticulation. Under the maximum likelihood framework, the change in the likelihood of the model is

$$\mathbf{P}(\mathcal{S}|N, \gamma) - \mathbf{P}(\mathcal{S}|T) = \gamma [\mathbf{P}(\mathcal{S}|T') - \mathbf{P}(\mathcal{S}|T)].$$

This quantity is non-negative whenever $P(\mathcal{S}|T') \geq P(\mathcal{S}|T)$. That is, under the maximum likelihood framework, if an arbitrary tree T' has a higher likelihood than the true tree T on which the sequences evolved, the ML framework would end up inferring reticulation events, even though the true evolutionary history is reticulation-free. The question we investigate first is: what factors might affect the performance of ML in this case? We hypothesize that the diameter of a reticulation event (that is, the length of the path along the underlying species tree between the donor and host nodes) and height (that is, the sum of the lengths of the paths from the donor and host nodes to the farthest leaves under them, respectively) play a role in the performance of ML. To investigate this question, we conducted the following experiment. We simulated the evolution of 100 sequence alignments, S_1, S_2, \dots, S_{100} down the 16-taxon tree T shown in Fig. 4.1(a), and then calculated $P(S_i|T')$, for $1 \leq i \leq 100$, where T' is one of the 12 trees that differ from T by a single subtree prune and regraft (SPR) move, with varying diameters, as shown with the arrows across the tree T in the figure.

The results show that as the diameter of a falsely postulated reticulation event increases, the probability of the data on that tree decreases compared to the probability on the true tree. Consequently, if the ML criterion errs in inferring reticulation events, it may introduce reticulation events between very closely related taxa. Or, put differently, reticulation events of very low diameter that are inferred by ML may

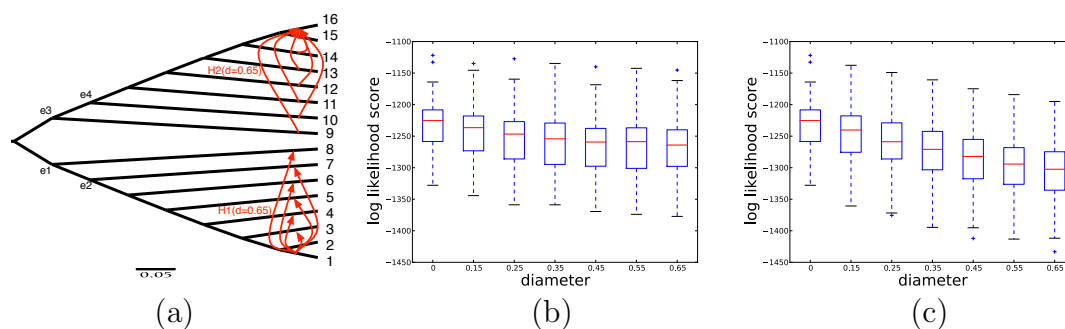


Figure 4.1 : Effect of the diameter of an HGT edge on the change in the likelihood score. The diameter of an HGT edge from node x to node y in the phylogenetic network is measured as the length of the path between x and y in the underlying tree (the network without the red arrows in (a)). Each of the 12 HGT edges was assessed individually, and never in combinations in this experiment. (b) Effect of the diameter for HGTs with different diameters but with a fixed donor node (taxon 1); these results correspond to each of the 6 HGT edges involving taxa 1–8. The diameters of the HGT edges vary from 0.15, for the HGT edge from taxon 1 to taxon 3, to 0.65, for the HGT edges from taxon 1 to taxon 8, with increments of 0.1. (c) Effect of the diameter for HGTs with different diameters but with a fixed recipient node (taxon 16); these results correspond to each of the 6 HGT edges involving taxa 9–16. The diameters of the HGT edges vary from 0.15, for the HGT edge from taxon 14 to taxon 16, to 0.65, for the HGT edges from taxon 9 to taxon 16, with increments of 0.1. The case of diameter=0 corresponds to scoring the likelihood of the underlying tree given the data.

not be well supported.

It is important to note that when the host is kept fixed, while changing the donor node to increase diameter (Fig. 4.1(c)), the effect on the decrease of the model likelihood is more than when the donor node is kept fixed and the host node changes (Fig. 4.1(b)). These results combined show that for short diameters where ML may make wrong inferences, the chances are higher that the error involves the placement of the donor node. In general, and beyond the ML framework, one may have more confidence in inference about the recipient than the donor, since in data sets involving bacteria for example, it is very easy to imagine that the true donor is not sampled

in the data set given the challenges with sampling bacterial data and the very large population size.

For our second experiment, we generated data as above, yet scored the probabilities of the sequence data on trees that differ from the true underlying tree in a single reticulation event that varies across trees in terms of its height; see Fig. 4.2. Unlike the diameter, the height does not seem to have much of an effect on the probabilities beyond the decrease as compared to the probability of the sequences on the true tree (height 0).

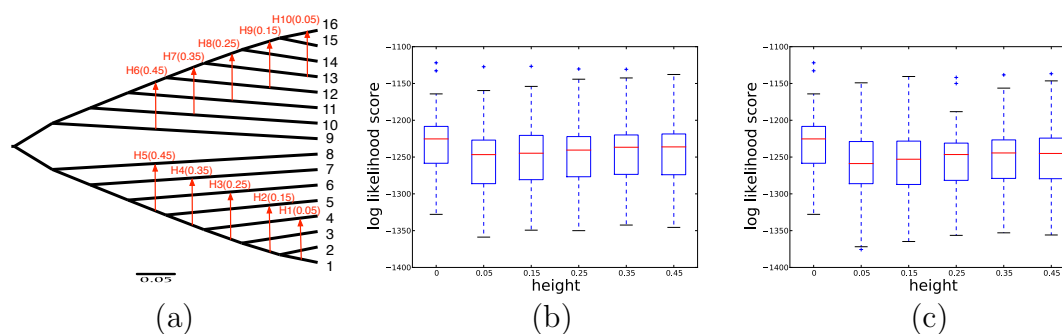


Figure 4.2 : Effect of the height of an HGT edge on the change in the likelihood score. The height of an HGT edge from node x to node y in the phylogenetic network is measured as the sum of the length of the path from x to a leaf under it in the underlying tree (the network without the red arrows in (a)) and the length of the path from y to a leaf under it. Each of the 10 HGT edges was assessed individually, and never in combinations in this experiment. (b) Effect of the height for HGTs with different heights but with the recipient taxon always being a branch connected to a leaf node; these results correspond to each of the 5 HGT edges involving taxa 1—8. The heights of the HGT edges vary from 0.05, for the HGT edge from taxon 1 to taxon 4, to 0.45, for the HGT edges from taxon 1 to taxon 8, with increments of 0.1. (c) Effect of the height for HGTs with different heights but with the donor taxon always being a branch connected to a leaf node; these results correspond to each of the 5 HGT edges involving taxa 9—16. The heights of the HGT edges vary from 0.05, for the HGT edge from taxon 13 to taxon 16, to 0.45, for the HGT edges from taxon 9 to taxon 16, with increments of 0.1. The case of height=0 corresponds to scoring the likelihood of the underlying tree given the data.

Performance of ML in determining the placement and probability of reticulation

In our second set of experiments, we set out to investigate how the ML performs in terms of identifying the location of a reticulation edge as well as the reticulation probability that indicates the fraction of genes that were transferred across that edge. We considered three independent evolutionary scenarios, each involving a single reticulation edge of a certain diameter, as shown in Fig. 4.3. All three reticulation edges

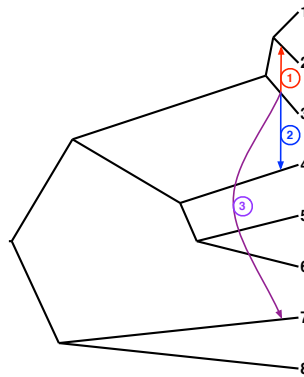


Figure 4.3 : Three evolutionary histories, each involving the underlying tree (black lines) and a single reticulation edge from the set of three reticulation edges 1, 2, and 3. The diameters of the three reticulation edges 1, 2, 3 are 0.5, 1.0, and 1.5, respectively.

have the same height and agree on the donor node, yet differ in terms of host node, and consequently the diameter. Each network of the three resulting network contains exactly two trees:

- Network N_1 , which is formed by adding only reticulation edge 1 to the underlying tree T ; this network contains the two trees T and T_1 , where T_1 differs from T only by placing taxon 2 as a sister taxon of 3.
- Network N_2 , which is formed by adding only reticulation edge 2 to the underlying-

ing tree T ; this network contains the two trees T and T_2 , where T_2 differs from T only by placing taxon 4 as a sister taxon of 3.

- Network N_3 , which is formed by adding only reticulation edge 3 to the underlying tree T ; this network contains the two trees T and T_3 , where T_3 differs from T only by placing taxon 7 as a sister taxon of 3.

To answer the two questions, we generated sequence data as follows: For a reticulation probability γ associated with the reticulation edge in network N_i , we evolved $(1 - \gamma)n$ gene sequence alignments down tree T , and γn gene sequence alignments down the tree T_i . In our experiment, we used reticulation probabilities $\gamma \in \{0.1, 0.3, 0.5\}$ and “genome size” $n \in \{10, 20, 40, 80\}$. For each combination of parameter values, we generated 50 data sets and performance inference of reticulation edges and their probabilities on all of them.

To investigate how ML performs in terms of estimating the reticulation probability, we fixed all elements of the model and only inferring the reticulation probability. That is, in this part, we assumed knowledge of the correct placement of the reticulation edge, and inferred the value of its associated γ using Eq. (4.4) (in this case, the equation identifies γ while e is known). The results are shown in Fig. 4.4.

There are several points to make. The diameter of the reticulation edge has a great effect on the accuracy of the estimated probabilities. For the largest diameter ($d = 1.5$), the ML criterion estimates the correct value of γ in almost all 50 cases, regardless of the true value of γ . It is important to note, though, that even for this diameter value, increasing the genome size (number of genes) reduces the variance in the estimates.

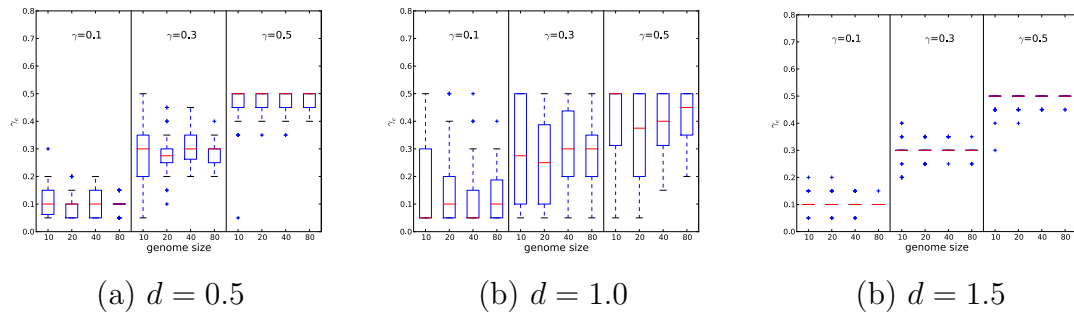


Figure 4.4 : The performance of ML for estimating the reticulation probabilities on data simulated with a single reticulation event. The genome size corresponds to the number of gene data sets used in the inference. Each panel contains three segments, corresponding to three different values of true reticulation probabilities: 0.1, 0.3, and 0.5. The reticulation probabilities γ_e were estimated using Eq. (4.4). The three diameters correspond to the three networks of Fig. 4.3.

For the smallest diameter, we observe an accurate estimate of the reticulation probability on average, yet with larger variance across the 50 data sets. In this case as well, increasing the number of genes reduces the variance. Further, for larger values of γ , the estimates become more accurate in general.

The data sets with medium diameter do not fit the trend very well in that show worse performance than the other diameters and not much improvement as the number of genes increases.

For studying the performance of ML in terms of placing the postulated reticulation edges, we used the data generated as described above along with the underlying (species) tree, as shown in Fig. 4.3, and inferred a single reticulation edge for each data set, by using Eq. (4.4) and the network search procedure. Suppose that network N with a single reticulation edge was inferred from data generated down network N_i from Fig. 4.3. Since both networks N and N_i have the same underlying (species) tree, checking whether the inferred reticulation edge agrees in terms of placement with the

true one is equivalent to checking whether the other tree T' (besides T) induced by N is identical to the tree T_i (the one induced by N_i in addition to T). However, rather than returning a 0/1 value, we quantify the symmetric difference between the two sets of bipartitions induced by T' and T_i . The results are summarized in Table 4.1. A value of 0 in the table indicates correct inference of the placement of the reticulation edge and the larger the value in the table the worse the predicted placement.

Table 4.1 : The accuracy of the placement of the inferred reticulation edge in terms of the symmetric difference between the true and inferred gene trees with a single reticulation event (see text for more details). The genome size corresponds to the number of gene data sets used in the inference. The three diameters correspond to the three networks of Fig. 4.3.

Diameter		Genome size			
		10	20	40	80
$\gamma=0.1$	0.5	0.6	0	0	0
	1	2.3	2.6	1.2	0.3
	1.5	5.6	5.7	5.6	5.5
$\gamma=0.3$	0.5	0	0	0	0
	1	1.2	0.1	0	0
	1.5	5.0	3.6	2.3	1.7
$\gamma=0.5$	0.5	0	0	0	0
	1	0.2	0	0	0
	1.5	3.0	3.2	1.5	0

The results show a very strong effect of the diameter of the true reticulation event on the postulated placement of the inferred one. Holding the reticulation probability and genome size constant, we observe a significant increase in the error as the diameter increases. For example, when using 10 genes and with reticulation probability of 0.1, the error in the placement of the reticulation event increases from 0.6 for diameter

0.5 to 5.6 for diameter 1.5. The same trend holds across all parameter values. This result indicates that confidence in the placement of an inferred reticulation event based on ML decreases as the diameter of the inferred event increases. On the more positive side, and with the exception of diameter 1.5 and reticulation probability of 0.1, increasing then number of genes drastically improves the accuracy of the placement. It is not surprising that for $\gamma = 0.1$, the error is high even for a large number of genes, since in this case the signal for reticulation is very low. For example, in the case of 10 genes, the evolutionary history of only a single gene involves the reticulation edges; recovering this edge is very hard in this case.

These results highlight an important issue in detecting reticulations using ML. If reticulation is a hybridization or hybrid speciation event, where a large number of genes may be exchanged or transferred across a reticulation edge (that is, a high value of γ), then ML would perform very well in terms of identifying the proportion of genes that were transferred horizontally, as well as the actual location of the reticulation (however, see Discussion section about the issue of incomplete taxon sampling). In the case of horizontal gene transfer in prokaryotes, on the other hand, a very small number of genes (or even a fraction of a gene) may be transferred across a reticulation edge; in this case, not much confidence can be assigned to the placement of the reticulation edge, especially if it has a large evolutionary diameter. However, horizontal gene transfer in microbial evolution seems to occur more often between closely related lineages than between distantly related ones [124].

Model selection under ML and the performance of information criteria

Now that we have explored the effect of diameter on the performance of ML in terms of estimating the placement of reticulation edges along with their associated

probabilities, we turn our attention to a most crucial issue with this model, as well as with phylogenetic networks in general, namely model selection. Here, we will investigate how ML does in estimating the correct number of reticulation edges and how, when augmented with information criteria, it performs. Let us denote by $L(i)$ the highest likelihood score of all phylogenetic networks with i reticulation edges for a given data set. Then, the AIC criterion selects a phylogenetic network with i reticulation edges over a phylogenetic network with $i - 1$ edges only when

$$(2K - 2 \ln L(i - 1)) - (2(K + 1) - 2 \ln L(i)) < 0.$$

Simplifying this inequality yields $\ln L(i) - \ln L(i - 1) > 1$. That is, whenever a network with i reticulation edges improves the likelihood score by at least one point, over a phylogenetic network with $i - 1$ reticulations, the i th edge would be selected under AIC, resulting in more complex network. This is equivalent to

$$\frac{L(i)}{L(i - 1)} > e.$$

Similarly, for the BIC, a phylogenetic network with i reticulation edges is selected over a phylogenetic network with $i - 1$ reticulation edges whenever

$$(K \ln n - 2 \ln L(i - 1)) - ((K + 1) \ln n - 2 \ln L(i)) > 0,$$

which is equivalent to $\ln L(i) - \ln L(i - 1) > \ln n/2$ or

$$\frac{L(i)}{L(i - 1)} > \sqrt{n}.$$

Based on these thresholds, we use 1 as the penalty term of AIC and $\ln n/2$ as the penalty term of BIC (since in the results we show below we explore the difference, rather than ratio, of the likelihood scores). In the experiments we now discuss, we focus on the quantity $L(i) - L(i - 1)$ as we add more reticulation edges, and compare it to the AIC and BIC penalty terms.

In our first experiment, we set out to investigate how both criteria perform when the data set has no reticulations. We used an experimental setup as above, where we generated 50 sequence data sets based on the (species) tree of Fig. 4.3 with genome sizes of $n = \{10, 20, 40, 80\}$ genes. We then applied our search procedure to identify the best first, second, third, and fourth reticulation edges to add, and compared the changes in likelihood scores, $L(i) - L(i - 1)$ to the penalty terms of both information criteria. The results are shown in Fig. 4.5.

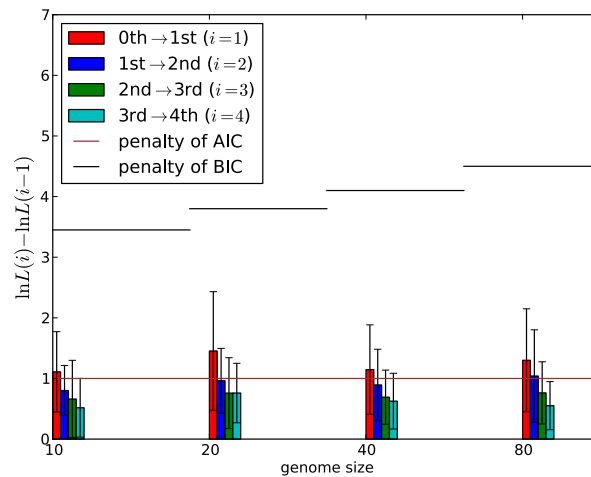


Figure 4.5 : The change in the likelihood scores as more reticulation edges are added. The true number of reticulations is 0 (all sequences were generated down the tree with no reticulations in Fig. 4.5).

As the results show, the estimated number of reticulation edges under both criteria

is always correct (0), except for a few cases when AIC estimates a single reticulation event. Notice that without either of the two criteria, the likelihood improvement is positive whenever any of the four reticulation edges are added. In other words, when no reticulations have occurred, both criteria, and particularly BIC, do a very good job at model selection, whereas ML with no penalty term would grossly overestimate the amount of reticulation.

We now turn our attention to the case of a single reticulation, yet with three different diameters and three different reticulation probabilities, as shown in Fig. 4.5. The results are shown in Fig. 4.6. The data used here is the same that we used to obtain the results in Fig. 4.4 and Table 4.1 above.

The results highlight several issues. For a very small diameter, the change in the likelihood score always exceeds the penalty term of AIC and is always smaller than that of BIC, resulting in accurate estimates based on BIC and overestimates based on AIC. As the diameter increases, to 1, BIC has a very good performance for the larger reticulation probabilities, but underestimates for the case of $\gamma = 0.1$. However, in this case, increasing the number of genes used to 40 or 80 gives BIC the necessary signal to make an accurate estimation. In the case of a diameter of 0.5, BIC almost always incorrectly predicts 0 reticulations, except when 80 genes are used and $\gamma = 0.5$.

Unlike BIC, AIC performs better at higher diameters, but that is because the change in likelihood scores become smaller and do not exceed the penalty term.

These results, combined with those from Fig. 4.5, indicates that inspecting both the change in the likelihood score itself, as well as the information criteria value may be valuable in determining, for real data sets, the true number of reticulations. An important trend to notice also is that the improvement in the likelihood score decreases when overestimated reticulations are added. Further, the reticulation probability has

a clear effect on the performance: the higher the probability, the higher the improvement of the likelihood score becomes, especially as compared to the improvements when overestimating. This again points to the conclusion that it is easier to detect hybridization or hybrid speciation events, where many genes support a reticulation edge, than horizontal gene transfer events involving very small number of genes.

Results on biological data sets

Unlike synthetic data, where the full evolutionary history is known, biological data sets pose several challenges, including the often unknown evolutionary history. In this section, we analyze two data sets. The first is a 15-taxon dataset of plastids, cyanobacteria, and proteobacteria, which is a subset of the dataset considered in [1] and for which multiple HGT events were conjectured by the authors. For this dataset, we obtained the species (organismal) tree which was reported in [1]. The species tree is based on 16S rRNA and other evidence and is shown in Fig. 4.7. We analyzed the rubisco gene *rbcL* of these 15 organisms. The gene dataset consists of 15 aligned amino acid sequences, each of length 532 (the alignment is available from <http://www.life.umd.edu/labs/delwiche/alignments/rbcLgb7-95.distrib.txt>).

Based on both the AIC and BIC criteria, we infer five HGT events, which agree with the hypotheses of Delwiche and Palmer [1] as well as the findings under both maximum parsimony and maximum likelihood analyses of Jin *et al.* [125, 45]. A major difference between this analysis and the previous computational analyses is that the information criteria systematically determined the number of HGT edges (Fig. 4.7), whereas in the other analyses the number was determined by an *ad hoc* inspection of the trends of the maximum parsimony and maximum likelihood scores. It is important

to mention that in this analysis, we did not infer the reticulation probabilities, but rather set them to 0.5, since only one gene data set was used and estimating the probabilities is not possible from such a data set.

For the second data set, we reanalyzed the yeast data set of [2], which is composed 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). In our analysis, we focused only on the first five species, since various studies have indicated the possibility of a hybridization within this group. From the sequence alignments, we reconstructed the species tree (topology and branch lengths) using the branch-and-bound algorithm in PAUP [82] through the following command:

```
BEGIN PAUP;
  [1] NJ;
  [2] LSCORE 1/ BASEFREQ=empirical
  TRATIO=estimate RATES=gamma SHAPE=estimate;
  [3] SET CRITERION=like;
  [4] LSET BASEFREQ=empirical TRATIO=previous
  RATES=gamma SHAPE=previous;
  [5] BANDB;
END;
```

The reconstructed species tree, which agrees in terms of topology with the one inferred in [2], is shown Fig. 4.8.

Bloomquist and Suchard used a Bayesian framework to analyze the data set and found that 37 of the 106 genes supported the reticulation event [126], which amounts to a hybridization probability of 0.34. Yu used a maximum likelihood framework that

estimates hybridization in the presence of incomplete lineage sorting [121]. When using 106 gene tree topologies estimated under maximum parsimony, their method estimated a hybridization probability of 0.34, which is identical to that of [126]. While the maximum likelihood framework we investigate here does not account for incomplete lineage sorting, it still produced a hypothesis of a reticulation event that agrees with the other studies (Fig. 4.8), with a lower estimate of the number of loci involved in this reticulation event (it estimated that about 5 of the 106 loci were involved in the hybridization). This further emphasizes the good performance of the maximum likelihood framework, even on a data set for which evolutionary events other than reticulation have been hypothesized.

Discussion

In this chapter, we studied the performance of ML for identifying reticulation events from sequence data, based on the formulation given in Eq. (4.1). We showed through simulation studies that the evolutionary diameter, and to a lesser extent, the height of a reticulation edge affects the performance in terms of estimating the reticulation probability (which reflects the proportion of genes transferred across a reticulation edge) and postulating a placement for the reticulation edge. We showed that increasing the number of genes improves the performance as well. We then investigated the performance of two information criteria, AIC and BIC, and found that BIC in general performs well in terms of model selection and preventing ML from overestimating the number of reticulation edges. Both AIC and BIC produced reasonable results on two biological data sets.

It is important to stress again that the framework, as given by Eq. (4.1), that we investigated here assumes reticulation as the only source of heterogeneity in the

evolution of the sequence data. However, in practice, other events may take place and the model needs to be modified accordingly. In particular, if events such as *deep coalescence* were allowed in the model, then the evolutionary history of a genomic region may take the form of a tree that is not in the set $\mathcal{T}(N)$ as we defined it above. Rather, every possible tree topology can now appear in the set $\mathcal{T}(N)$, and the probability of each tree can be assessed under models such as the coalescent. Work on accounting for both reticulation and *deep coalescence* simultaneously is emerging [46, 47, 48, 121], but dealing with it is beyond the scope of this chapter.

Another issue that is of extreme significance when dealing with reticulation is taxon sampling. As we showed above, the location of the donor node has a significant impact on the detectability of a reticulation edge. When analyzing data sets in practice, particularly prokaryotic data, it may easily be the case that the true donor of the horizontally transferred is not in the data set being analyzed. Therefore, beyond our findings here about the power of ML to infer the placement of a reticulation edge, one has to be cautious about interpreting the placement of a computationally inferred reticulation edge.

A third issue is that while the term reticulation encompasses all types of evolutionary events that are not vertical, there is a clear distinction between, for example, the exchange of a genomic regions through homologous recombination in bacteria and a hybrid speciation event that gives rise to a new species in plants. The amount of genetic material transferred across a reticulation edge in the latter case is way much larger than that of in the former. In a phylogenomic study involving thousands of gene families, identifying a reticulation edge that might have been used in the transfer of a single gene might be confounded by the overwhelming vertical signal supported by the remaining genes. Consequently, more confidence can be associated

with inferences in cases where a large number of genes support a reticulation edge.

When gene trees are estimated with confidence, one can replace Eq. (4.1) by

$$L(N, \gamma | \mathcal{T}) = \prod_{T_i \in \mathcal{T}} \mathbf{P}(T_i | N, \gamma),$$

where T_i is the gene tree for gene i . In this case, a method for estimating the term $\mathbf{P}(T_i | N, \gamma)$ is required. Yu recently devised such a method [121]. We identify comparing this approach to the one we used here as a future research task. Further, in the work of [121], the authors also gave a method to account for uncertainty in the estimated gene trees in set \mathcal{T} , which we will explore as well.

Finally, we showed in this manuscript that if the improvement ratio in the likelihood score by adding a reticulation edge is beyond e and \sqrt{n} for AIC and BIC, respectively, then adding the reticulation edge would be supported. This result can be further pursued in two directions. First, mathematical results can be derived, for specific models of sequence evolution, to establish analytically conditions under which ML would support a reticulation edge, and equivalently, when AIC and BIC would result in overestimation. Second, these results can be utilized for devising efficient algorithmic techniques for identifying reticulation edges whose addition result in significant improvement, as opposed to exhaustively searching the space of all possible reticulation edges, which is infeasible for large values of n .

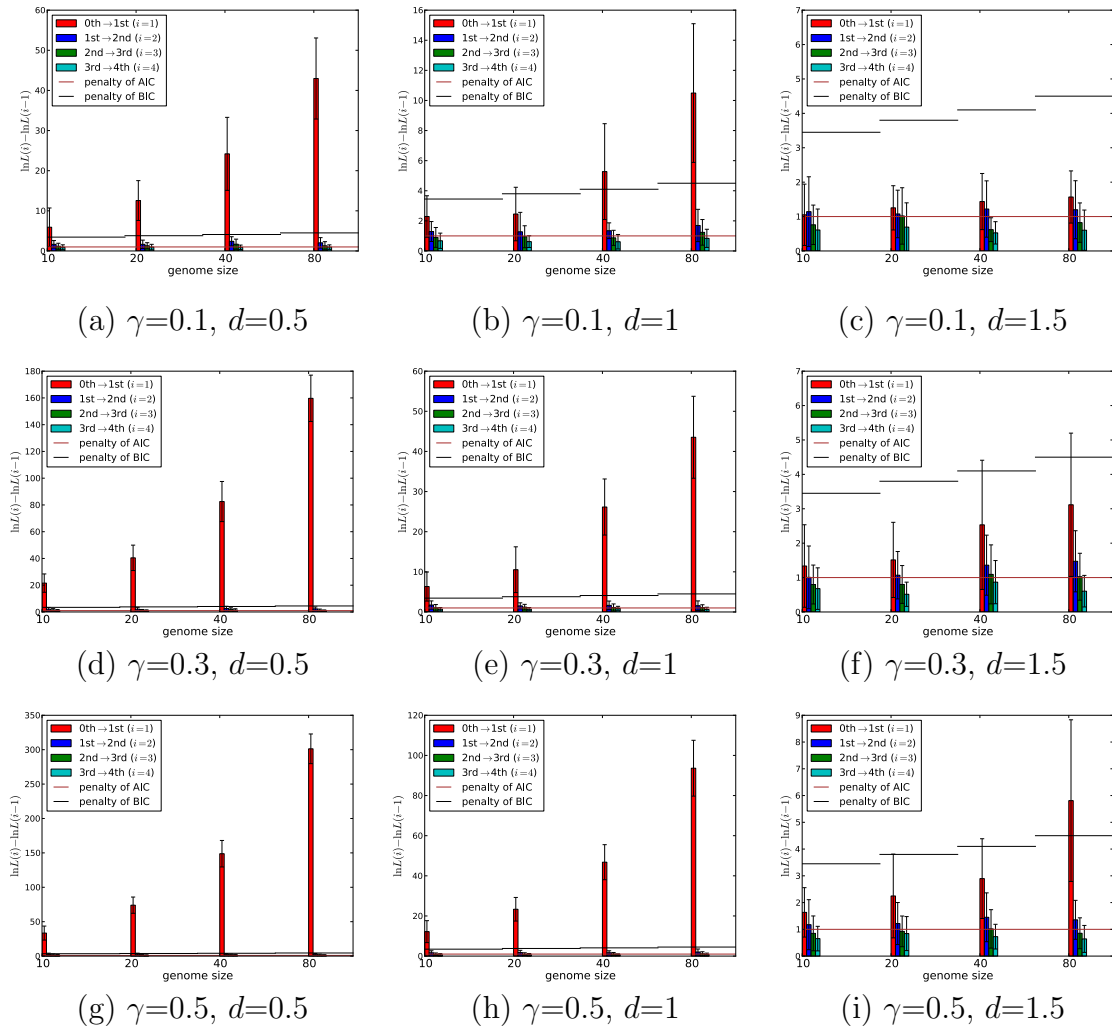


Figure 4.6 : The change in the likelihood scores as more reticulation edges are added. The true number of reticulations is 1, yet with three different diameters, as in Fig. 4.5.

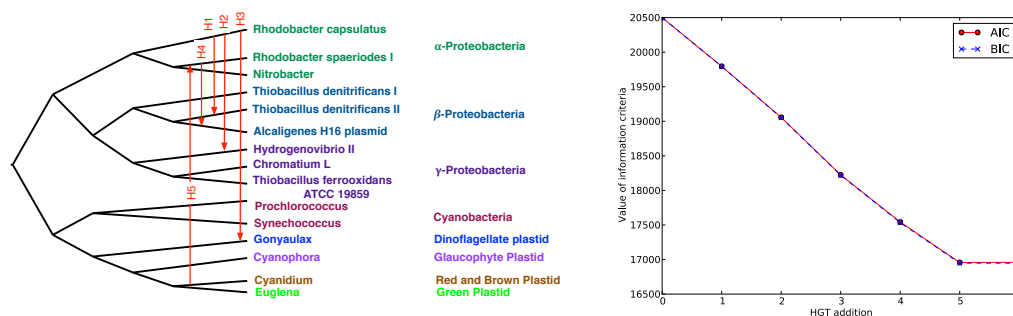


Figure 4.7 : Results on the *rbcL* gene data set. (Left) The underlying species tree, as reported in [1], with the five predicted HGT edges posited between pairs of its branches. (Right) The decrease in the AIC and BIC values as optimal HGT edges are added to the species tree. The decrease in the AIC/BIC values from HGT addition i to $i + 1$ corresponds to HGT edge H_i .

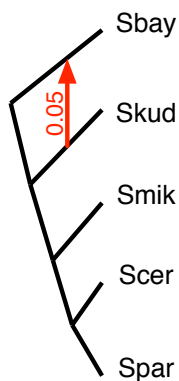


Figure 4.8 : The inferred species tree for the yeast data set in [2]. The horizontal arrow corresponds to the reticulation event inferred by our method, along with the reticulation probability.

Chapter 5

Phylogenetic Networks from Gene Trees by Parsimony

So far, we have discussed the reconstruction problem with a set of sequence alignments as the input. As data, gene trees contain less noise and more error in general in comparison to the gene sequence. There are a body of researches focusing on reconstructing gene trees, either with branch lengths or without them, from various types of data (e.g. [127, 128, 129, 82, 122]). In this chapter, we introduce the computational methods addressing the phylogenetic network problem from a collection of the reconstructed gene trees.

5.1 Introduction

With the availability of whole-genome data from an increasingly large number of organisms, particularly prokaryotic ones, evolutionary studies are faced with a large number of *gene trees** in a given study. Therefore, it is imperative to develop computational techniques that simultaneously analyze a large number of trees, and combine them into networks. Clearly, the problem is NP-hard when the *SPR distance* is used, since it is NP-hard for a pair of trees. Huson and Rupp [130] proposed a method for summarizing a collection of gene trees using *cluster networks*, which differ from the phylogenetic network model we address here. Beiko and Ragan [131] discussed

*In this context, the term “gene tree” applies to an evolutionary tree of any non-recombining genomic region; i.e., it is not limited to trees on (protein-coding) gene regions.

aggregating inferred HGT events from pairwise tree comparisons, and discussed three strategies for this task; yet, they did not implement the strategies, nor did they study their performance. Iersel *et al.* [5] developed the CASS method, which is an efficient algorithm for inferring a minimal phylogenetic network that contains all the clusters of taxa displayed by the input gene trees, but not necessarily the input gene trees themselves. Further, Wu [4] recently introduced the PIRN algorithm for obtaining lower and upper bounds on the amount of reticulations necessary for reconciling a set of input gene trees. Finally, we introduced two methods for estimating the amount of reticulation, as well as inferring a phylogenetic network, from a collection of gene trees [3, 132]. Both of our methods are based on obtaining estimates for the set of trees from pairwise distance calculations. Note that all methods given above operate based on two main assumptions about the input gene trees: (1) the trees are accurate (that is, we ignore incongruence among the trees due to error in the gene tree inference), and (2) reticulation is the only biological cause of the gene tree incongruence. While these two assumptions may be violated in practice, we believe that they can be used to obtain a quick analysis, after which a careful inspection of the reticulations can follow up. In this sense, we believe that the algorithms should guarantee speed as well as accuracy. The results show that our approaches of combining pairwise reconciliations to obtain a solution for the entire set of trees, though *ad hoc*, showed good performance in both senses. Therefore, these strict assumptions notwithstanding, we believe the methods make a significant contribution.

In this chapter, we aim at inferring phylogenetic networks with the minimum number of reticulation events that reconcile a collection of gene trees. We present two heuristic algorithms, one that is based on the observation of a binomial distribution of the pairwise distances of a collection of trees contained in a network, and

the second is based on agglomerating pairwise solutions to obtain a global, hopefully minimal, solution for all trees [3]. Further, we discover the bias of the second algorithm resulting from the multiplicity of the pairwise solutions. In order to calculate the unbiased solutions, we define the problem more formally and provide an integer linear programming (ILP) solution for it [132]. It is important to mention that this makes an improvement on the previous algorithm not only in accuracy, but also in speed. Finally, we study the performance of the methods, and compare it to other methods, on synthetic data sets and one biological data set. The results show that our methods are fast in practice, and that they produce accurate estimates of the phylogenetic network. With the results, the simulation study also highlights conditions under which the methods' performances become not as good. This characterization is particularly important, since it may help develop more accurate methods for this problem.

The rest of the chapter is organized as follows. In Section 5.2, we give the definition of the problems of interest and discuss the optimization criteria for the problems. In Section 5.3, we present three heuristic algorithms that infer the number of the reticulation events among the input gene trees. Note that the second and the third, the ILP-based improvement to the second, infer the species tree as well as the locations of the events. We conduct experiments to evaluate the performance of the algorithms in Section 5.4, in which the results show that our algorithms perform better than the competing algorithms in speed and/or accuracy. Section 5.5 concludes the chapter with final remarks and some directions for future research.

5.2 Background

The main focus of the chapter is to infer phylogenetic networks from a set of gene trees under a certain set of assumptions, which will be given at the end of this section. Under the assumptions, the true phylogenetic network should display all the given gene trees, and likewise the gene trees should be contained in the network. As described in the process 2.2, it is straightforward to induce the set $\mathcal{T}(N)$, given an \mathcal{X} -network N , though this computation may be expensive, since $|\mathcal{T}(N)| = O(2^{|V_N|})$. The more relevant problem in the context of inferring phylogenetic networks is that of estimating an \mathcal{X} -network from a subset[†] of its induced trees, since this amounts to inferring the (reticulate) evolutionary history of a set of organisms.

5.2.1 Pairwise and Set-wise Reconciliation of Trees

A main problem of interest is the following [44].

Problem 5.1. (SET-WISE HGT INFERENCE)

Input: *A collection of gene trees $\mathcal{G} = \{GT_1, \dots, GT_k\}$, each modeling the evolutionary history of a genomic region of a set \mathcal{X} of taxa.*[‡]

Output: *A phylogenetic network N with the smallest number of reticulation nodes (a minimal network) such that $\mathcal{G} \subseteq \mathcal{T}(N)$.*

[†]It is highly unlikely for a biological data set to exhibit all trees induced by the network; in practice, the set of trees exhibited by the different genomic regions is a small subset of all possible trees induced by the network.

[‡]It is important to note that while we focus on collections of trees that have the same leaf labels, in practice gene trees may differ in their leaf labels (e.g., due to sampling, gene duplication, gene loss, etc.).

Obviously, if all trees in \mathcal{T} are identical, the problem is trivial since N would be the tree in \mathcal{T} . Otherwise, the problem is hard. When the input consists of exactly two trees, we refer to this as the PAIRWISE HGT INFERENCE problem.

Given that there is a very large number of \mathcal{X} -networks N such that $\mathcal{T} \subseteq \mathcal{T}(N)$, the main issue in this domain is to define a criterion Φ and seek the \mathcal{X} -network (or, set of \mathcal{X} -networks) that is optimal under Φ , given the set \mathcal{T} of trees. A natural parsimony criterion to define is to minimize the number of network-nodes in N . In other words, we seek the network (or set of networks) N such that (1) $\mathcal{T} \subseteq \mathcal{T}(N)$, and (2) N has the minimum number of network-nodes among all \mathcal{X} -networks satisfying (1). While the “true” phylogenetic network may not necessarily be a parsimonious one, this criterion yields plausible networks in many realistic cases (although it is easy to show examples of cases in which this criterion results in networks with numbers of network-nodes that are arbitrarily smaller than the true number [44]). In particular, this criterion can be viewed as a way to estimate a lower bound on the amount of reticulation in the data. With this criterion for the reconstruction, a solution to a PAIRWISE HGT INFERENCE problem with $\mathcal{T} = \{T_1, T_2\}$ is to compute the *SPR distance* [32] between the two trees, denoted by $d_{SPR}(T_1, T_2)$, and take it as the estimate of the number of network-nodes in the \mathcal{X} -network N that induced both trees in \mathcal{T} .

Based on the discussion, we set the reconstruction problem to estimate an \mathcal{X} -network, with the minimum number of network-nodes, that induces a given set of trees \mathcal{T} in this chapter. This problem is NP-hard, given that is NP-hard for a pair of trees [34].

We will now make two assumptions that we will use throughout this chapter [§].

[§]Unlike the other two assumptions given above, these are not necessarily kept in other approaches

First, a solution to the PAIRWISE HGT INFERENCE problem is obtained by solving for $SPR(T_1, T_2)$. In other words, we will take a smallest set Ξ of Subtree Prune and Regraft (SPR) moves that transform T_1 to T_2 , and obtain N by adding Ξ to T_1 . Second, in the PAIRWISE HGT INFERENCE problem, we will assume that the first tree is a species tree ST . In this problem, we will assume that a species tree is given, so that the distance from each tree in \mathcal{G} to the species tree is computed. We discuss below a potential solution to the problem when a species tree is not given as a part of the input.

5.3 Methods

5.3.1 M1: Fitting a Binomial Distribution of Pairwise Distances

As we show below, our investigation of simulated data sets indicates that, in practice, one factor that may affect the hardness of the problem is the *redundancy* in the network, which we define as follows.

Definition 5.1. *The redundancy of an \mathcal{X} -network N with set V_N of network-nodes is $\varepsilon_N = (2^{|V_N|} - |\mathcal{T}(N)|)/2^{|V_N|}$.*

Figure 5.1 illustrates the concept of redundancy. In a non-redundant \mathcal{X} -network N ($\varepsilon_N = 0$), each tree in $\mathcal{T}(N)$ is uniquely induced by the network, whereas in a redundant network ($\varepsilon_N > 0$), some trees may be induced in multiple ways. An upper bound on ε_N for an \mathcal{X} -network with h network-nodes is $1 - 1/2^h$, in which case the network induces a single tree and, considering topology alone, none of the reticulation events may be detectable.

Let $V_N = \{v_1, \dots, v_h\}$ be the set of all network-nodes in an \mathcal{X} -network N , and for each two edges incoming into a node $v_i \in V_N$, let one be labeled l (for *left*) and

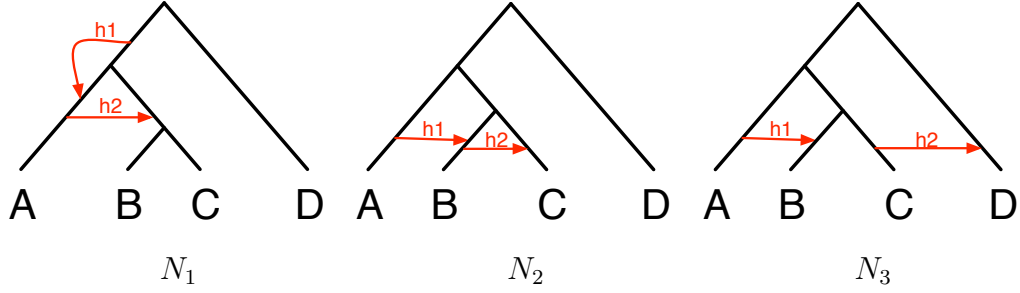


Figure 5.1 : Three \mathcal{X} -networks, each with two network-nodes, yet with varying degrees of redundancy. Here, $\mathcal{T}(N_1) = \{T_1\}$, $\mathcal{T}(N_2) = \{T_1, T_2\}$, and $\mathcal{T}(N_3) = \{T_1, T_2, T_3, T_4\}$, where $T_1 = ((A, (B, C)), D)$, $T_2 = (((A, B), C), D)$, $T_3 = (A, (B, (C, D)))$, and $T_4 = ((A, B), (C, D))$. Consequently, we have $\varepsilon_{N_1} = (4 - 1)/4 = 0.75$, $\varepsilon_{N_2} = (4 - 2)/4 = 0.50$, and $\varepsilon_{N_3} = (4 - 4)/4 = 0$.

the other be labeled r (for *right*). Further, let $T \in \mathcal{T}(N)$ be a tree induced by the network. A *displaying vector* of T , denoted by $d(T)$ is an element of $\{l, r\}^h$, where $d(T)[i]$ denotes the label of the edge incoming into v_i that was retained to induce the tree T . We have the following two lemmas and ensuing theorem.

Lemma 5.1. *Let N be an \mathcal{X} -network. Then, $d(T)$ is unique for every tree $T \in \mathcal{T}(N)$ iff $\varepsilon_N = 0$.*

Lemma 5.2. *Let $\mathcal{D} = \{l, r\}^h$. Then $|\{\{d_1, d_2\} : d_1, d_2 \in \mathcal{D}, HD(d_1, d_2) = q\}| = \binom{h}{q} 2^{h-1}$, where $HD(d_1, d_2)$ denotes the Hamming distance between the two binary vectors d_1 and d_2 .*

Theorem 5.1. *Let N be an \mathcal{X} -network with h network-nodes, and assume $d_{SPR}(T_1, T_2) = HD(d(T_1), d(T_2))$ for every $T_1, T_2 \in \mathcal{T}(N)$. If $\varepsilon_N = 0$ then $|\{\{T_1, T_2\} : T_1, T_2 \in \mathcal{T}(N), d_{SPR}(T_1, T_2) = q\}| = \binom{h}{q} 2^{h-1}$.*

Theorem 5.1 implies that when there is no redundancy in the network, and given that we do not know the actual displaying vectors of the trees, we can use the *SPR distance* as a proxy to the Hamming distance of the displaying vector, and expect a

binomial distribution of the pairwise distances. This, in turn, naturally gives rise to the following approach for estimating the minimum number of reticulations required in a phylogenetic network to reconcile a set \mathcal{T} of trees:

1. Compute all pairwise *SPR distances* over the set \mathcal{T} of trees, and let Q be the distribution of these distances.
2. Denoting by P_m the distribution $\binom{m}{q}2^{m-1}$ for $1 \leq q \leq m$, find the value m that minimizes $KL(Q|P_m)$, where KL is the Kullback-Leibler distance [133]
 $KL(g|f) = \sum_q f(q) \ln \frac{f(q)}{g(q)}$.

The way we compute the value of m in Step (2) in the above procedure is by starting from

$$m = \max\{\lceil \log_2 |\mathcal{T}| \rceil, \max_{T_1, T_2 \in \mathcal{T}} d_{SPR}(T_1, T_2)\} \quad (5.1)$$

and incrementing m as long as $KL(Q|P_m)$ decreases. The rationale behind Equation ((5.1)) is that the \log_2 of the number of trees in the given set is a lower bound on the number of reticulations, and so is the maximum pairwise *SPR distance* over all trees in the set.

Obviously, the conditions of Theorem 5.1 may not hold in practice. In particular, it may be that some or all of the following issues arise when analyzing a data set:

1. It may be that for some pairs of trees $T_1, T_2 \in \mathcal{T}(N)$, $d_{SPR}(T_1, T_2) < HD(d(T_1), d(T_2))$.
 In this case, the distribution of the pairwise distances may be skewed to the left. A potential alternative for considering the minimum number of SPR moves is to take a stochastic approach that simulates random walks, using SPR moves, in the tree space [134].
2. The (unknown) network N may have $\varepsilon_N > 0$. Here, the frequencies of some

pairwise distances may be lower than the true frequencies (which are the ones based on P_m).

3. The given set of trees \mathcal{T} does not contain all trees induced by the (unknown) network N . Here, not enough data points may be available for reliably estimating the true distribution Q .

Nevertheless, we show below, through extensive simulations, that this heuristic provides good estimates of the number of network-nodes required for a network to reconcile a given set of trees. From the next section on, we refer to this method M1.

5.3.2 M2: Combining Pairwise Solutions

While the approach in the previous section is aimed at estimating only the minimum number of reticulations needed in a phylogenetic network to reconcile a set of trees, the approach we present here is aimed at estimating minimal sets of actual SPR moves (obviously, the sizes of such sets can be taken as estimates of the amount of reticulation). Note that both this approach and its improvement in the next section can be taken as an upper-bound on the minimum number of reticulation nodes, whereas the estimates of the first approach cannot necessarily be taken neither as an upper-bound nor as a lower-bound. Actually, the result shows that that of the first approach works usually as a lower-bound, in the sense that it usually underestimates.

The general outline of the method we propose here for estimating a set of SPR moves to reconcile a set of trees \mathcal{T} is simple (similar to the *greedy approach* for aggregating inferred HGT events in [131]):

1. For each pair of trees in \mathcal{T} , identify a minimal set of SPR moves that reconcile them.

2. Combine the set of solutions identified in Step (1).

There are two main issues that need to be addressed for this approach to work in practice. First, for a given pair of trees, there may be multiple minimal sets of SPR moves that reconcile them [135]. In this case, we need the pairwise SPR computation to return all, or a large number, of these minimal solutions. We make use of the modified version of RIATA-HGT [39, 40], as implemented in PhyloNet [136], to compute multiple minimal solutions. The second issue is two-fold: (a) Given a set of minimal sets of SPR moves for each pair of trees, how do we find a global minimal set of SPR moves that covers at least one minimal set for each pair? (b) Once the (global) minimal set is computed, how do we obtain a network from it?

In the case of the horizontal gene transfer detection problem, usually a species tree ST is given, in addition to the set of trees \mathcal{T} . In this case, the pairwise computations should be conducted only between ST and every tree in \mathcal{T} , but not between pairs of trees in \mathcal{T} . Then, the global set of SPR moves computed by the procedure above is posited on the tree ST . In the case where no “backbone” tree, such as ST , is given, we propose to use each of the k trees in \mathcal{T} as a backbone tree against each all SPR computations are conducted, and choose the tree in \mathcal{T} that results in the smallest set of SPR moves.

We use this idea in the heuristic M2 below. Let ST be an (species) \mathcal{X} -tree and $\mathcal{T} = \{T_1, \dots, T_k\}$ be a collection of (gene) \mathcal{X} -trees. Further, let Z be the set of all possible SPR moves that can be defined on ST (the cardinality of Z is quadratic in the number of leaves in ST [32]). For each tree $T_i \in \mathcal{T}$, let $SPR(ST, T_i) = \{S_i^1, \dots, S_i^{w_i}\}$ be the set of minimal sets of SPR moves that transform ST into T_i . Our task is to find a minimal set $z \subseteq Z$ such that for every $1 \leq i \leq k$, there exists $1 \leq \ell_i \leq w_i$ such that $S_i^{\ell_i} \subseteq z$. In other words, we seek a minimal set z of SPR moves that cover

at least one minimal “solution” for each gene tree. Clearly, each tree in \mathcal{T} can be obtained by applying a subset (or all) of the SPR moves in z to ST . This is a hard problem, and we solve it heuristically, as described in the following algorithm.

ALGORITHM M2

1. For each gene tree $T_i \in \mathcal{T}$
 - 1.1. initialize count: $c(r) = 0$ for every SPR move r in Z ;
 - 1.2. for each gene tree $T_j \in \mathcal{T}$ and $T_j \neq T_i$
 - 1.2(a). compute $SPR(T_i, T_j)$;
 - 1.3. for each SPR move r , compute count $c(r) = |\{j | r \in \text{solution } s \text{ and } s \in SPR(T_i, T_j)\}|$;
 - 1.4. for each gene tree $T_j \in \mathcal{T}$ and $T_j \neq T_i$
 - 1.4(a). for each solution $s \in SPR(T_i, T_j)$, compute count $c(s) = \sum_k \{c(r_k) | r_k \in s\}$;
 - 1.4(b). choose a solution s , $\hat{SPR}(T_i, T_j) = \{s | c(s) \geq c(s') \text{ for all } s' \neq s, s' \in SPR(T_i, T_j)\}$;
 - 1.5. compute the union $R_i = \bigcup_{T_j \in \mathcal{T}, T_j \neq T_i} \{s | s \in \hat{SPR}(T_i, T_j)\}$;
2. choose $\mathcal{R} = R_l$ such that $|R_l| = \min_i (|R_i| | 1 \leq i \leq k)$ along with the corresponding tree $T_l \in \mathcal{T}$.

In the next section on, this method will be referred to as M2.

5.3.3 MURPAR: Combining Pairwise Solutions using ILP

In the last section, M2 heuristically derives

$$SPR(ST, \mathcal{G}) = \bigcup_{gt \in \mathcal{G}} SPR(ST, gt) \quad (5.2)$$

as an estimate of the solution of the *Set-wise HGT Inference* problem. An advantage of this approach is that fast exact algorithms and heuristics exist for obtaining $SPR(ST, gt)$, as described above, and taking the union of pairwise reconciliations is very simple computationally. Indeed, the algorithm yields good performance on simulated data sets [3]. Nonetheless, a careful inspection of the algorithm raises some issues that need to be resolved for accurate estimates.

First, it is possible that the optimal solutions for $SPR(ST, gt)$ do not lead to the optimal solution for $SPR(ST, \mathcal{G})$, and cause overestimation in a “global” estimate of reticulations. For example, consider the HGT scenarios shown in Fig. 5.2. If we take

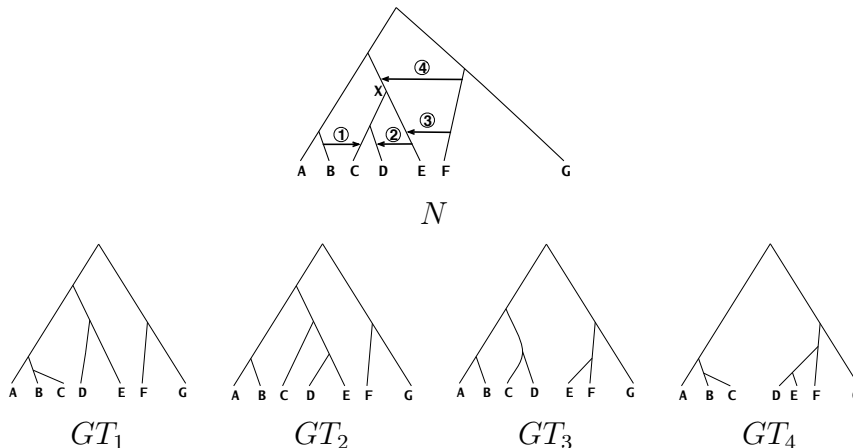


Figure 5.2 : **A phylogenetic network with four independent HGT scenarios.** The species tree ST in this case is the network N without the four HGT edges. The gene whose tree is GT_i ($1 \leq i \leq 3$) underwent HGT event (i), and the gene whose tree is GT_4 underwent HGT events (1) and (4). The combined effect of HGTs (1) and (4) on the gene tree topology is the same as the combined effect of HGTs (1), (2), and (3).

the union of the four pairwise solutions computed by $SPR(ST, GT_i)$, for $1 \leq i \leq 4$, we obtain a set of four HGT edges: $\Xi_1 = \{[B \rightarrow C], [E \rightarrow D], [F \rightarrow E], [F \rightarrow X]\}$. However, a smallest solution for the SET-WISE HGT INFERENCE problem given the species tree ST and the four gene trees contains three HGT edges, which is the set

$\Xi_2 = \{[B \rightarrow C], [E \rightarrow D], [F \rightarrow E]\}$. In this case, the HGT edge $[F \rightarrow X]$ is not needed, since its effect can be simulated by the two HGT edges $[E \rightarrow D]$ and $[F \rightarrow E]$, once the HGT edge $[B \rightarrow C]$ is applied. However, notice that in this case, the solution that truly reflects what happened is Ξ_1 , since the HGT event denoted by $[F \rightarrow X]$ did occur, even though its effect on the topologies of gene trees can be simulated by the other three HGT edges. In other words, while the union of pairwise solutions may not provide a smallest global solution, it may provide a solution that is closer to the true HGT scenarios that took place at the genomic level. Further, under these scenarios, where a smallest solution is a proper subset of the union of pairwise solutions, post-processing via gradual elimination of members of the union can yield a smallest solution. However, it is not guaranteed that the smallest solution is a subset of the union of pairwise reconciliations.

Second, $SPR(ST, gt)$ may not be unique; in fact, the number of possible solutions to the Pairwise HGT Inference problem can be exponential in the size of a solution [135]. To account for this issue, we need to consider all solutions, or a large number of them when obtaining all is computationally infeasible, for each pair of trees. Without accounting for multiple solutions, methods such as M2 [3] that agglomerate pairwise solutions would obtain biased estimates.

A third issue that requires special attention is the following: while solutions to the PAIRWISE HGT INFERENCE problem may be acyclic (that is, the inferred HGT events, when added to the species tree, do not result in cycles), taking the union of solutions cannot guarantee acyclicity. When this occurs, the solution is not a phylogenetic network as given by Definition 2.2.

As mentioned above, the former method M2 [3] uses the approach given by Eq. (5.2), but it does not address the last two issues raised above. For the first

issue, let ST be a species tree and $\mathcal{G} = \{GT_1, \dots, GT_k\}$ be a collection of gene trees. Also let $S_i = \{s_{i,1}, s_{i,2}, \dots, s_{i,m_i}\}$ be the set of all optimal pairwise solutions on the pair $\langle ST, GT_i \rangle$. Then, M2 counts the frequency with which each potential reticulation edge appears throughout $s_{i,j}$, $1 \leq i \leq k, 1 \leq j \leq m_i$, and calculates the set of reticulation edges such that 1) it covers at least one solution for all trees, and 2) it maximizes its frequency value, in the assumption that the frequency reflects how often a reticulation edge is used in each solution and maximizing the frequency would result in a smallest set of reticulation edges.

However, with multiple solutions (e.g., obtained by using RIATA-HGT [39, 40]), a reticulation edge occurring multiple times in $s_{i,j}, \forall 1 \leq j \leq m_i$, contributes more to the frequency than those occurring once. As a result, a solution would be biased towards the edges occurring multiple times in solutions. In this section, we propose our heuristic MURPAR (Multi-tree Reconciliation using PAirwise Reconciliations) for addressing these issues. With the definition of $S = \bigcup_{1 \leq i \leq k, 1 \leq j \leq m_i} s_{i,j}$ MURPAR seeks the smallest set $S' \subseteq S$ that satisfies the property $[\forall 1 \leq i \leq k, \exists s_{i,j} \in S_i \text{ s.t. } s_{i,j} \subseteq S']$. MURPAR solves this problem using integer linear programming (ILP). We define binary variables as follows:

- (A) $B_s, \forall s \in S$. B_s will take value 1 if SPR move s is selected as an element of S' , and 0 otherwise.
- (B) $P_{ij}, \forall 1 \leq i \leq k, \forall 1 \leq j \leq m_i$. P_{ij} will take value 1 if all SPR moves in the optimal pairwise solution s_{ij} are selected, and 0 otherwise.

Finally, the ILP program is:

$$\text{minimize } \sum B_s$$

subject to

- $P_{ij} = [\wedge_{y \in s_{ij}} B_y], \forall 1 \leq i \leq k, \forall 1 \leq j \leq m_i$
- $[\vee_{1 \leq j \leq m_i} P_{ij}] = 1, \forall 1 \leq i \leq k$

Here, \wedge and \vee represent logical ‘and’ and logical ‘or’, respectively. All these constraints can be turned into linear constraints by introducing auxiliary variables as follows:

- $a = (b_1 \wedge b_2 \wedge \dots \wedge b_p)$, where all variables are binary, can be turned into the linear inequalities $-1 \leq 2b_1 + 2b_2 + \dots + 2b_p - 2pa \leq 2p - 1$.
- $(b_1 \vee b_2 \vee \dots \vee b_p) = 1$, where all variables are binary, can be turned into the linear inequality $b_1 + b_2 + \dots + b_p \geq 1$.

When a species tree ST is not given, we repeat MURPAR with $GT_i, 1 \leq i \leq k$ as the species tree and choose the smallest S'_i as S' .

As discussed above, the solution thus far may result in cyclic graphs, which are not phylogenetic networks. To address this issue, MURPAR post-processes the results to avoid the solutions with cycles using a straightforward cycle detection algorithm. If a minimal solution is found to have a cycle, MURPAR skips it and inspects the next minimal solution (minimal in terms of the number of reticulation nodes). While all solution candidates found by MURPAR may have cycles, and thus MURPAR returns no solution, we found through extensive simulations that this was never the case. Similarly, it was shown in [137] and confirmed in [138] that cycles may not be a major concern for reticulation detection algorithms when run on real data sets or synthetic data sets that are generated under realistic models.

5.4 Experimental Evaluation

In this section, we evaluate our algorithms in two subsections. First, we systematically study the performance of M1 and M2 on simulated data sets. And we compare the performance of M2 and MURPAR with another competing tool, on both simulated data sets and a biological data set. Since M1 returns only the numbers, it is excluded for the second experiment.

5.4.1 Experimental Setup

Data Simulations were conducted on 30- and 50-taxon phylogenies. For 30-taxon data sets, 10 random trees were generated using PHYL-O-GEN tool [91] as “species tree” under birth-death model, and 5 horizontal gene transfer events were simulated between pairs of branches on the species trees using Galtier’s tool [92]. The simulation of horizontal gene transfer were conducted individually 10 times on each species tree, so totally 100 networks are generated from the simulation. Since Galtier’s tool does not provide the details of simulated transfer events, we modified the tool to have it report the simulated transfers that it added. From the set of 32 gene trees contained in each network, 4, 8, 12, 16, 24, and 32 gene trees were randomly sampled and used as input to the methods.

For 50-taxon data sets, the same procedure as above was applied with the two differences: (1) the number of horizontal gene transfer events simulated was 10, so the sampling was made over 1024 gene trees, (2) and the 30 times of sampling process was repeated for each sample size to generate input data, so that statistically significant results are obtained.

The second evaluation runs M2 [3], MURPAR [132] and PIRN [4] both on the simulated data sets and on a biological data. For biological data, we used the Poaceae

data set, which was originally sequenced by the Grass Phylogeny Group, and was used to test both CASS [5] and PIRN [4]. Binary trees were constructed for six loci: *ITS*, *ndhF*, *phyB*, *rbcL*, *rpoC* and *waxy* [139]. Since the gene trees had different sets of leaves, we selected the gene trees for *ndhF*, *phyB*, *rbcL*, *rpoC2* and *ITS*, and restricted them to 14 leaves that they have in common.

Methods and Accuracy Measures All of our methods are based on solving the PAIRWISE HGT INFERENCE problem. M1 runs the exact method of Wu [35], SPRDist, that returns the exact *rSPR distance*, since it only requires the *rSPR distance*. However, M2 and MURPAR cannot directly utilize the method, because they require the placement of the HGT estimates. So we employ RIATA-HGT method [39, 40], as implemented in PhyloNet [136] to solve the PAIRWISE HGT INFERENCE problem and obtain the locations of the multiple optimal HGT events. Other pairwise inference tools including SPRIT [140] were tested as well, but results were almost identical; therefore, all results are reported based on the pairwise solutions obtained by either SPRDist or RIATA-HGT. We used the GLPK ILP Solver to solve the ILP formulation of MURPAR.

Although we introduce CASS [5] as an algorithm for the problem, we do not use it for comparison. As indicated by the authors in [5], while CASS computes a minimal network N from an input set \mathcal{C} of clusters of a set of gene trees \mathcal{T} , it is not guaranteed that $\mathcal{T} \subseteq \mathcal{T}(N)$. More formally, if \mathcal{C} is the set of all clusters of taxa displayed by the trees in \mathcal{T} , the network N computed by CASS is the minimal network that displays all clusters in \mathcal{C} . It is important to note that if a network N displays all clusters of a set of trees, N does not necessarily display all the trees themselves. It is easy to see that if N is minimal for \mathcal{C} and N' is minimal for \mathcal{T} (\mathcal{C} is the set of all clusters

of trees in \mathcal{T}), then the number of reticulation nodes in N is smaller than or equal to that in N' , because the problem with \mathcal{C} is less restrictive than that with \mathcal{T} . An illustration of this issue is given in Fig. 5.3.

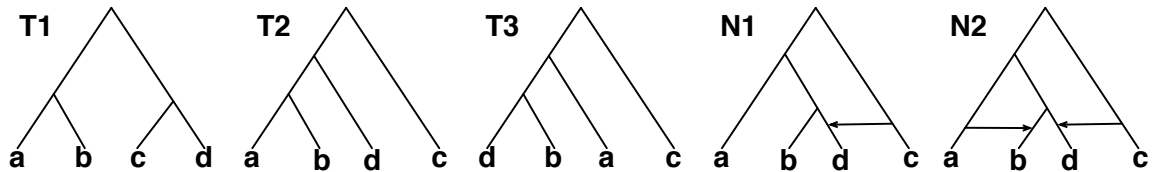


Figure 5.3 : **The difference between the formulation used by MURPAR, M2 [3], and PIRN [4], and the one used by CASS [5].** For the input set of gene trees $\mathcal{T} = \{T1, T2, T3\}$, CASS computes a network with a single reticulation node ($N1$), since this network displays all clusters of the gene trees. However, MURPAR, M2, and PIRN compute minimal networks with two reticulation nodes, such as $N2$, since 2 is the minimum number of reticulation nodes required in a network that displays all three gene trees.

The evaluation for the first part measures *detected* number of reticulations from the methods, and *detectable* from the input tree set. When M1 or M2 is run on a collection $\mathcal{T} = \{T_1, \dots, T_k\} \subseteq \mathcal{T}(N)$ induced by network N , we record the number of reticulations that the method computed; we call this number the *detected* number of reticulations. Now, if network N was generated with 5 or 10 HGTs, this does not necessarily mean that the collection \mathcal{T} of trees will have all trees to allow for detecting 5 or 10 HGTs, respectively. For example, consider the collection \mathcal{T} that has only trees whose (pairwise) *SPR distance* is 1. In this case, the number of detectable HGTs is 1, and not 5 (or 10). Therefore, for each such collection \mathcal{T} , we compute (exhaustively) the smallest subset of HGTs in N that can reconcile all trees in \mathcal{T} ; we call this number the *detectable* number of reticulations (notice that this is not necessarily the smallest number of reticulations needed to reconcile all trees in \mathcal{T} ; computing this number would be prohibitive). The accuracy of the methods is considered better

as the difference between *detectable* and *detected* numbers of reticulations becomes smaller.

The second part evaluates the performance by comparing their return values. Since they all return the lower-bound of the number of network-nodes required to reconcile the input trees, the values can directly be used for comparison. In parsimony, the smaller the value is, the better the corresponding approach is. In this comparison, M1 is excluded, because its return is not by the same measure. We will refer to this as accuracy. Besides accuracy, we also assess the run time of the methods, which is important when they are used as a preprocessing unit for the following analysis. We checked the results from both 30-taxon and 50-taxon data sets for the second part, and confirm that they show the same trend. But we only visualize the result of 30-taxon data sets.

5.4.2 Results of Synthetic Data

For the first evaluation, we show the results only for the 30- and 50-taxon data sets. Figure 5.4 shows the difference between *detectable* and *detected* numbers of the reticulation events, of M1, while Figure 5.5 shows the accuracy value of M2.

In the case of 30 taxa, sampled trees are selected from the network that contains ($2^5 =$) 32 trees. The simulation generates 100 networks as described, and we choose 80 of them for the first evaluation. There are 80 such networks, each of which was sampled with sample sizes of 2, 4, 8, 12, 16, and 24. Therefore, at each point of the x-axis in Figures 5.4(a) and 5.5(a), the result shows the distribution of the number of reticulation events for 80 gene tree sets. Figures 5.4(b) and 5.5(b) show the results for the sampled gene tree sets from 100 networks of 50 taxa with 10 HGTs. Each of these networks contains up to 1024 different trees. From them, sampled gene tree

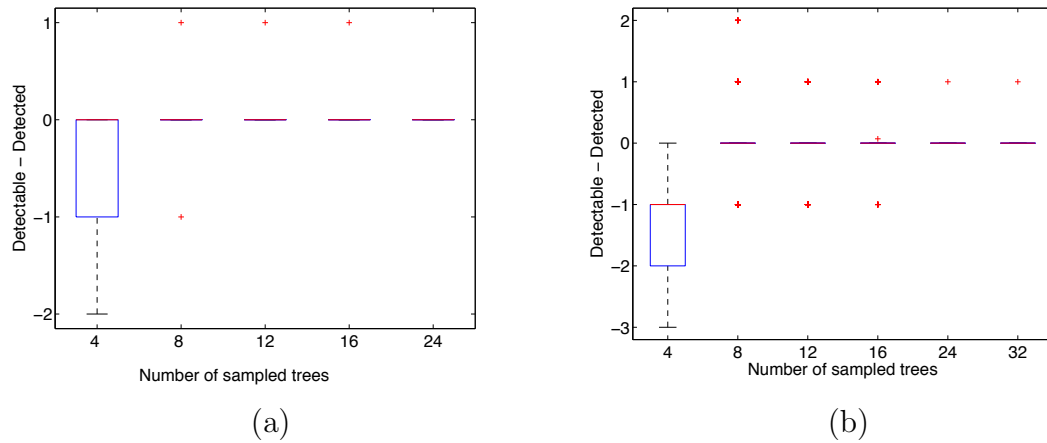


Figure 5.4 : Performance of M1 on the 30-taxon (a) and 50-taxon (b) data sets as a function of the sample size.

sets are chosen with sample sizes of 4, 8, 12, 16, 24, and 32. For each of the sampling sizes and each of the networks, we sampled 30 times. Therefore, at each point of the x-axis in Figures 5.4(b) and 5.5(b), the result is the distribution of the number of reticulation events for 3000 different gene tree sets.

As the figures show, both methods perform very well on the 30-taxon data sets, with the median different between detectable and detected numbers of reticulations, for both methods, falling at zero. In the case of M1, there is an improvement in the accuracy as the sample size increases, as is evident from the lack of outliers and the convergence to the median value of 0. This is because, as the sample size increases, the data points become much denser so that fitting the binomial distribution becomes easier. Nonetheless, even for very sparse samples (sizes 4 and 8), the method still performs very well, as shown in Figure 5.4(a). M2, on the other hand, does not show clear improvement with increased sample size; to the contrary, more outliers emerge as the sample size increases (Figure 5.5(a)). One reason behind this is that as the sample size increases and the SPR move sets become larger, a more careful handling

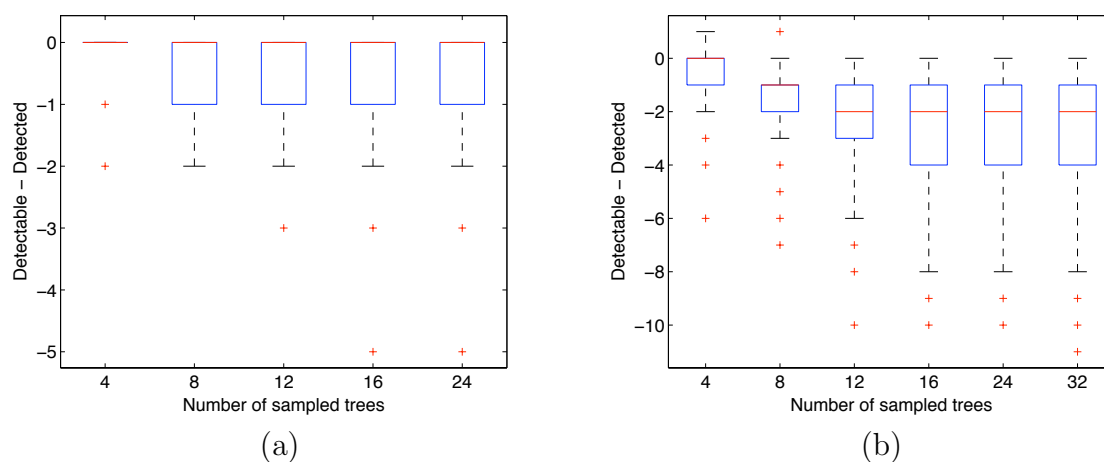


Figure 5.5 : Performance of M2 on the 30-taxon (a) and 50-taxon (b) data sets as a function of the sample size.

of the union of those sets is required than we employ in our heuristics. In some sense, this problem becomes similar to the Inclusion-Exclusion principle, where one has to avoid double-counting.

For the 50-taxon data sets, both methods also perform well, particularly M1. Even though both methods tend to overestimate the amount of reticulation in these cases (as shown by the negative values in Figures 5.4(b) and 5.5(b)), the under-estimation is very mild on average. It is worth mentioning that the results in Figure 5.5(b) come from much smaller sampled gene tree sets (less than 4%). From the results shown in Figures 5.5(a) and 5.5(b), sampling with size of 2, or only given a pair of gene trees, is not sufficient to estimate true number of reticulation events. For sampling with the sizes larger than 2, the results are very close to the true number of reticulation events (5 in Figure 5.5(a) and 10 in Figure 5.5(b)) in most cases, having a difference of up to 2. M2 tends to overestimate the number of reticulation events. In the worst case, the estimated results could double the actual number of reticulation events. However, the median of the distribution and the results for most cases have a converging trend

when the sampling size increases.

For the second experiment, we use all 100 networks, unlike the previous experiment, and assess the running times of the methods; It is important for a method to maintain reasonable computation speed in order to be of general application for large-scale data analysis, especially when it might be used as a preprocessing step for the following analysis, as mentioned above. Through the experimental validation, we record the run times of the 100 cases for the methods by sample size in Fig. 5.6. In interpreting the figure, we focus on two properties regarding the computation speed, the overall run time and the frequency of outliers in time.

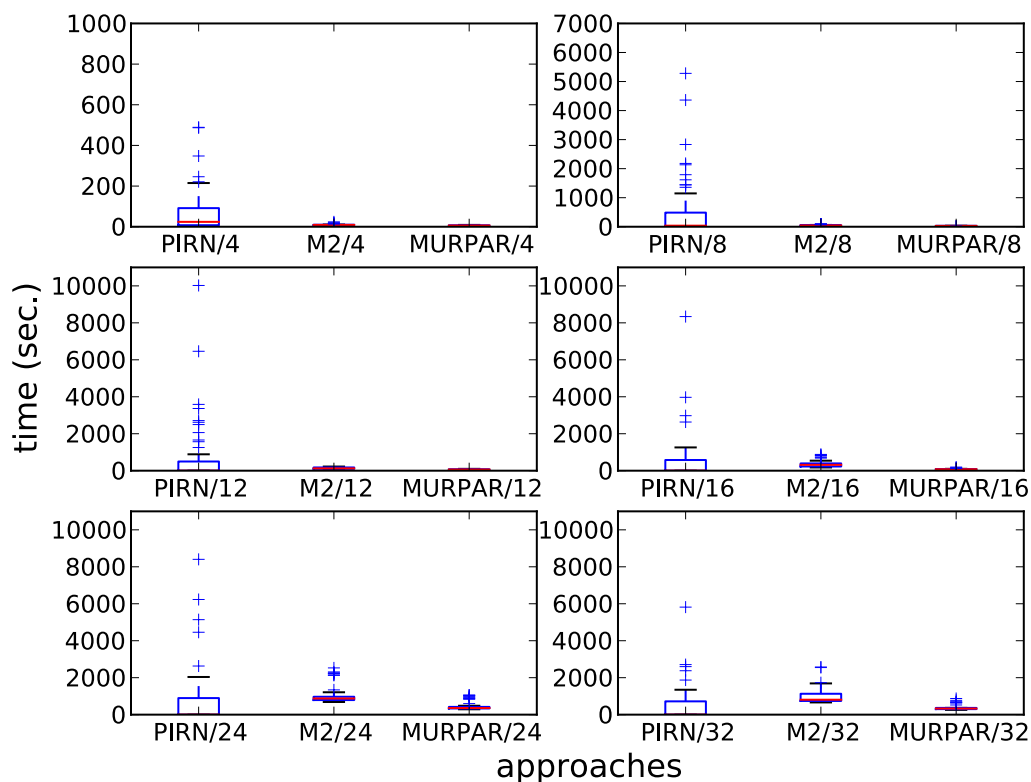


Figure 5.6 : Running times (in sec.) of the three methods (PIRN, M2, and MURPAR). The numbers after the '/' are the numbers of gene trees in the input.

In terms of the overall run time, it is clear that MURPAR runs fast across varying sample sizes. Compared with M2, the gain in speed of MURPAR is mainly attributed to the use of the ILP solver, since they share the underlying computation structure. The figure also shows that the gap between MURPAR and PIRN closes as the number of trees increases. However, the outliers in the case of PIRN are still much slower than those of MURPAR and M2. This point requires further elaboration. Holding the input size (in terms of the number of gene trees) fixed, the variance in the speed of MURPAR is very similar, whereas that of PIRN is very large. This somehow indicates dependence of the PIRN on the structure of the problem, and lack of dependence of MURPAR on such a structure. It may be that the smaller the number of gene trees, the fewer the constraints, and hence the larger the space that PIRN explores. In the case of MURPAR, the pairwise solutions constrain the search space significantly, giving the method gains in speed.

The numbers of reticulation nodes estimated by each of the methods are shown in Fig. 5.7. It is worth mentioning that even though the true networks were produced by adding 5 HGT edges, the true number of reticulations may be smaller, depending on the size of the input, since, for example, when only 4 gene trees are sampled, some HGT events may not be observable. Even though network N has $m > 1$ reticulation nodes, this does not necessarily mean that the collection \mathcal{T} of trees, with $|\mathcal{T}| \geq 2$ will have all trees to allow for detecting m . For example, consider the collection \mathcal{T} that has only a pair of trees whose *SPR distance* is 1. In this case, the number of detectable HGTs is 1, and not m . However, the number of detectable HGTs is expected to increase as more trees are given, and the results in the figure satisfy the expectation.

As more trees are sampled as the input, a naive method for the reticulation estima-

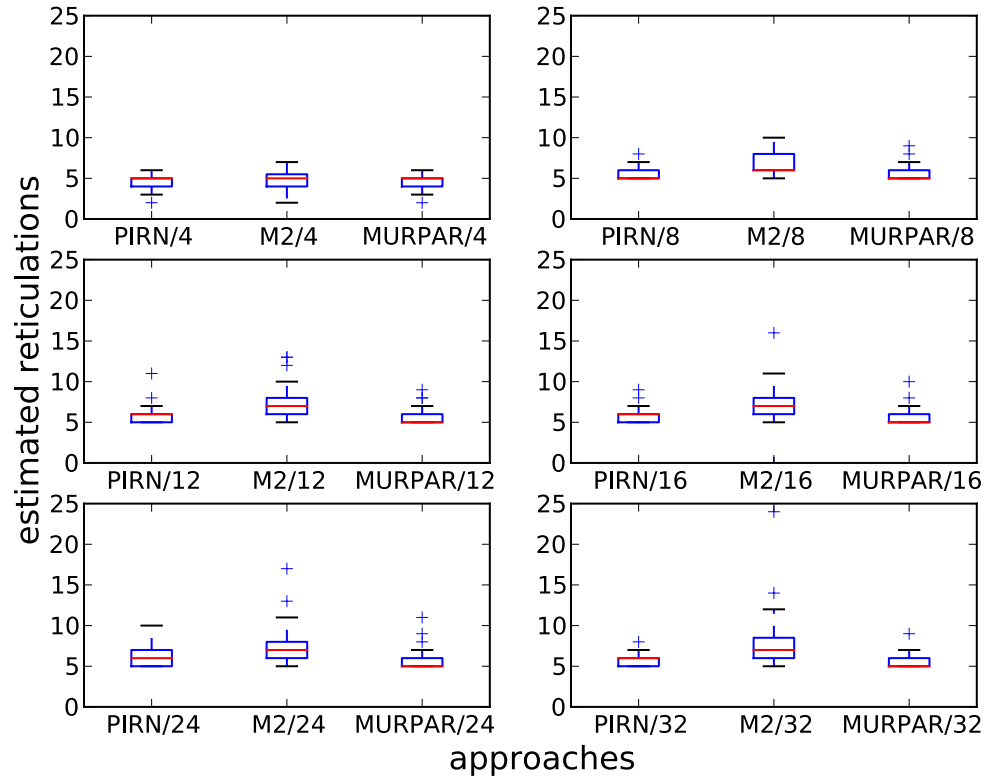


Figure 5.7 : Numbers of reticulations estimated by each of the three methods (PIRN, M2, and MURPAR). The numbers after the ‘/’ are the numbers of gene trees in the input.

tion would estimate more reticulations simply because the size of the data increases. M2 follows the expectation, in that the estimation gets larger with more trees. However, MURPAR overcomes this problem, even though it is based on the same idea. Rather, both the estimations of PIRN and MURPAR hardly increase with the input size. Between them, it is clear that as more trees are given, the estimate becomes more accurate in MURPAR than in PIRN, particularly for inputs of sizes ≥ 12 . Even though the maximum difference between PIRN and MURPAR is one reticulation event on average, this difference gets larger for larger data sets.

5.4.3 Results of Biological Data

In order to evaluate how well the methods perform for biological data, we ran M2, MURPAR, and PIRN on the five gene trees of the *Poaceae* data set (see Section 5.4.1 for details of this data set). Table 5.1 reports the estimated number of reticulations and the amount of time taken by the methods. Notice that we also ran PIRN in coarse mode (CoarsePIRN) on them (which is a faster, yet less accurate, version of PIRN).

Table 5.1 : The number of estimated reticulation events and the run time (in seconds) of the methods on the five gene trees for *ndhF*, *phyB*, *rbcL*, *rpoC2*, and *ITS* in the *Poaceae* data set.

Approaches	#Reticulations	Time (sec.)
PIRN	13	2143
M2	14	16
MURPAR	14	8
CoarsePIRN	16	58

PIRN identifies the lowest estimate of the number of reticulations, but it took the longest time to obtain the estimate. On the other hand, M2 and MURPAR obtained estimates that are higher by just one reticulation event, two and three orders of magnitude faster, respectively. In other words, MURPAR and M2 produce very accurate results within very short amounts of time. Between MURPAR and M2, the difference is negligible on this size of data. However, it is easy to see that the difference will grow with the increase of the input data, and MURPAR will be preferred in large-scale data analysis. For PIRN, we also ran it in coarse mode. Notice that while PIRN in coarse mode is much faster than PIRN, and is comparable to M2 in terms of speed, the estimates it produces are higher than the other three methods.

Considering the run times they take to work on the small input data (5 trees on 14 species), it is clear that PIRN would run the slowest in large-scale data analyses.

5.5 Discussion and Conclusions

5.5.1 Distribution of Gene Trees for Detectability

In an attempt to help in the development of new methods for the SET-WISE problem, we set out to investigate the effect of the actual sample of trees on the performance of the methods, particularly M1, since it is sensitive to the distribution of pairwise distances. For each actual network-node (reticulation event) in a simulated network N , roughly half of the trees in $\mathcal{T}(N)$ use one parent, whereas the other half use the other parent. We hypothesize that the detectability of a reticulation node is easy when half of the gene trees give signal about one of its parents, while the other half give signal about its other parent. In Figure 5.8, we plot the performance of **Method 1** on the 50-taxon data sets, as a function of the deviation of the trees in a sample from the balanced coverage of each reticulation event (written as “distribution deviation from 1/2” on the x-axis). Clearly, there is a correlation between the deviation from a balanced coverage of reticulation events by the trees in a sample and the estimation trend: over-estimations occur at lower deviation from balanced coverage, followed by correct estimation at higher deviations, and finally under-estimations occurring at the highest deviation from balanced coverage. We do not have a clear answer to why this is the case, but this leads to an interesting question about the effect of the balance of trees in a set on the detectability of reticulations.

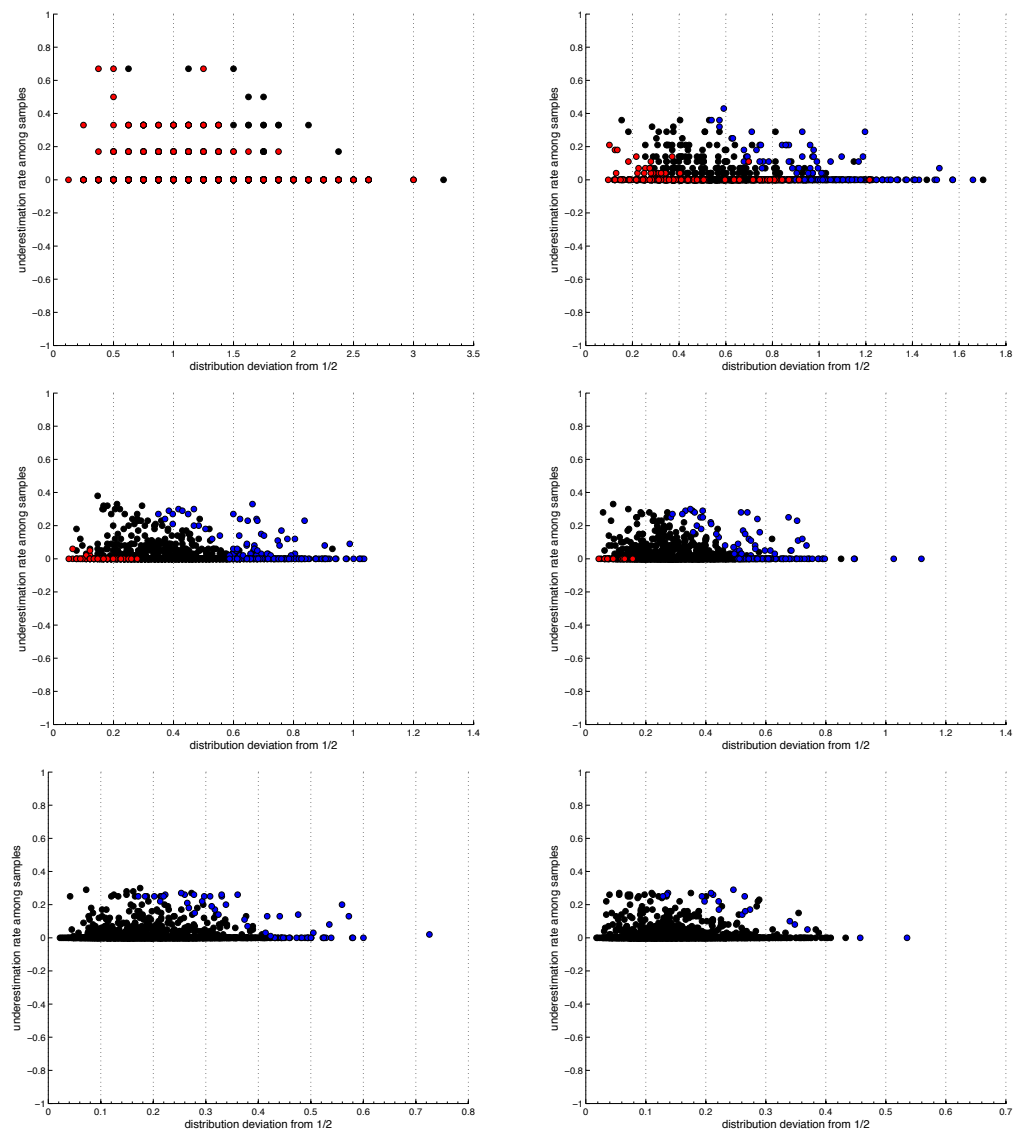


Figure 5.8 : Inspection of over- and under-estimation of M1 as a function of the distribution deviation from 1/2 (see text for more details). Black, blue, and red dots represent correct, under-, and over-estimations, respectively, of the method. Left to right, top down: sample sizes 4, 8, 12, 16, 24, and 32 (all on 50-taxon data sets).

5.5.2 Conclusions

The increasing availability of whole-genome and multi-locus data has highlighted the need for computational tools that enable phylogenomic analyses. One such analysis entails comparing gene trees in a group of organisms, identifying their differences, and using this information to elucidate the evolutionary mechanisms that acted on the organisms during the course of their evolution. In prokaryotic organisms, it is widely believed that horizontal gene transfer (HGT) is ubiquitous, and that it plays an important role in genomic diversification.

Mathematically, the *subtree prune and regraft*, or SPR, distance between a pair of trees has been commonly used as a proxy for a lower bound on the number of HGT events, or reticulations. As a result, a wide array of mathematical results and computational tools have been developed around this distance. Nonetheless, most of these results and tools apply to a pair of trees, which is a shortcoming, particularly for phylogenomic studies involving many trees.

In this chapter, we addressed the problem of estimating the amount of reticulation that is detectable in a collection of gene trees, assuming all incongruence among the trees is due to reticulate evolution (i.e., ruling out any other discord processes, such as incomplete lineage sorting, gene duplication/loss, etc.). We provided two algorithmic strategies for this task, both of which showed promising results in simulations.

And then, we extend M2 for accuracy and time using ILP, and introduced MURPAR, a method for inferring a phylogenetic network from a collection of gene trees, under the same assumption. While MURPAR is not guaranteed to compute a minimal network, it produces an upper bound on the minimum number of reticulations required to reconcile all gene trees in the input. Performance analysis on both synthetic and biological data sets shows that the MURPAR method is both accurate and

fast. Further, MURPAR’s run time does not vary much within the same sample size, and has fewer outliers than other methods.

The idea of employing pairwise reconciliations in reconciling a set of gene trees has added advantages in that pairwise reconciliations can be computed in parallel or in a distributed fashion, thus speeding up the overall computation, and improvements to pairwise reconciliation methods will automatically translate into improvement of the MURPAR method. Direct interpretability of the results from the direct relationship of the solutions between SET-WISE HGT INFERENCE and PAIRWISE HGT INFERENCE is another advantage of MURPAR that it is easy to identify the dynamics between any gene trees in the tree set from the estimates of SET-WISE HGT INFERENCE.

5.5.3 Future Work

Our main task for future research is to apply these strategies to biological data, not only to assess the performance of the methods, but also to better understand the reticulate evolution in biology. However, in addition to the multiplicity of gene trees, biological data poses another serious issue for our approach to address: while our methods are based on solving *SPR distance* problem, which requires the trees to be on the same leaf set, the gene trees in practice are on different numbers of leaves. In particular, incomplete taxon sampling and disparity in sequence coverage for different organisms may result in “missing” genes for some organisms. Biologically, gene duplication and loss may result in multiple or no copies of certain genes in some organisms. Further, a horizontal gene transfer event from outside of the group of organisms under study may give rise to genes that are present in some, but not all, of the organisms. Figure 6.3 illustrates the heterogeneity of the gene trees in terms of the number of gene copies. Last but not least, HGT events across genes may not be

independent, as a single HGT event may transfer a large genomic region that contains multiple genes, as proposed by [141]. All these issues need to be addressed in order to facilitate phylogenomic study; otherwise, analyses would have to be restricted to a small fraction of the genomic data, rendering their results and conclusions unreflective of the true, global picture [142].

Chapter 6

On the Performance of Parsimonious Reconciliation in Detecting Duplication, Transfer and Loss

So far, we have focused on developing computational methods that detect particular evolutionary events by identifying significant estimates. However, we have not discussed how such computational methods can actually be used in biological data analysis. We employ a computational method that operates under an extensive model to determine the issues the computational method needs to address to be useful for biological data analysis. Note that these issues are not specific to the method, but are obstacles that must be addressed by most computational methods. We address these issues by identifying the significance of estimates.

6.1 Background

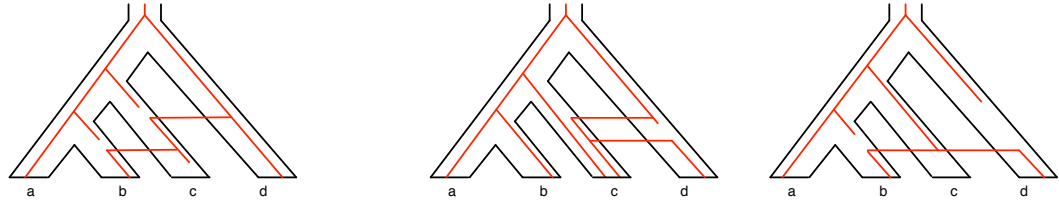
6.1.1 Detecting Duplication, Transfer, and Loss (DTL)

Computational methods to detect evolutionary events are developed under the assumption of specific evolutionary events. Section 2.3.2 introduces several methods that detect HGT events, assuming that only HGTs occur in the data. All our methods proposed in this dissertation fall under this category. Section 2.3.4 reviews several methods that detect duplication and loss events, assuming that only duplication and loss are cause for incongruence. However, as the current biological data involve many

loci from many species, the data may involve both HGT and duplication and loss. Then, it is easy to see that an algorithm assuming only one of the events loses accuracy in detecting events. In response to the need to take both transfer and duplication/loss into account, the Duplication-Transfer-Loss (DTL) model has been proposed. Several approaches, both model-based and parsimony-based approaches, have been proposed under the model [74, 143, 144]; however, they have some difficulties for direct use in large-scale biological data analysis, in that appropriate values of the parameters of the models are not well-studied and they generally run slowly. Since parsimony-based approaches require fewer parameters and run fast, we use a parsimony-based approach for analysis.

While parsimonious reconciliations under the DTL model detect evolutionary events by mapping gene trees to the species tree as in the DL model, they are more difficult to calculate. Given gene tree G and the species tree S under DL, the *order-respecting* property matches nodes under $u \in V(G)$ only with nodes under $u \in \gamma(u)$ (see Section 2.3.4 for more detail) and reduce the search space for matches effectively as the match goes from $R(G)$ and $R(S)$ to their leaves. However, DTL does not provide this property. As an HGT occurs between separate, but contemporary, species for a gene, it can break the confinement of the species barrier, and the corresponding internal edge in the gene tree does not occur along species tree edges. As a result, the search space to map the transfer is not reduced as the reconciliation progresses, causing the computation to be demanding. Another complication of DTL reconciliation comes from the transfers involving the incomplete sampling (or gene loss), degenerate transfers, and/or simultaneous transfers as in Figure 6.1. These cases complicate the picture as genes are repeatedly transferred. However, most parsimony-based approaches disregard them (see [145], for example), even though they are biologically

probable.



(a) incomplete sampling/gene loss (b) degenerate transfer (c) multiple transfers

Figure 6.1 : Three examples of biologically plausible HGT events, not considered in most of the parsimony-based approaches. In practice, they can occur in various combinations and further complicate the picture.

A mapping γ under DTL can be used to identify the evolutionary events associated with internal nodes of a gene tree. We denote those internal nodes representing speciation events Σ , those representing duplication events Δ , and those representing transfer events Θ , with respect to the γ . Additionally, we collect tree edge (u, v) , $u \in \Theta$, v represents the destination of the transfer Ξ . Observing that when v, w are the children of u in G , $\gamma(u) \leq_S \gamma(v)$ and $\gamma(u) \leq_S \gamma(w)$ and assuming that at least one of $\gamma(v)$ and $\gamma(w)$ is a descendant of u , γ assigns the nodes in G to Σ , Δ , or Θ by the following logic in parsimony

1. $u \in \Sigma$ only if $\gamma(u) = LCA(\gamma(v), \gamma(w))$ and $L(\gamma(v)) \cap L(\gamma(w)) = \emptyset$.
2. $u \in \Delta$ only if $\gamma(u) = LCA(\gamma(v), \gamma(w))$ and $L(\gamma(v)) \cap L(\gamma(w)) \neq \emptyset$.
3. $u \in \Theta$ if and only if $(u, v) \in \Xi$.
4. $(u, v) \in \Xi$ if and only if $\gamma(u)$ is incomparable to $\gamma(v)$.

Since a mapping γ can produce multiple valid mappings and the corresponding sets of $\{\Sigma, \Delta, \Theta\}$ for a pair of the species tree and a gene tree, parsimony selects

6.2 Experimental Setup

6.2.1 Data

We analyze a biological data set composed of γ -proteobacterial strains downloaded from eggNOG [146] on July 11th, 2011. The data set consists of 3086 gene families on 134 γ -proteobacterial strains, including model organisms such as *E. coli* and *Vibrionaceae*.

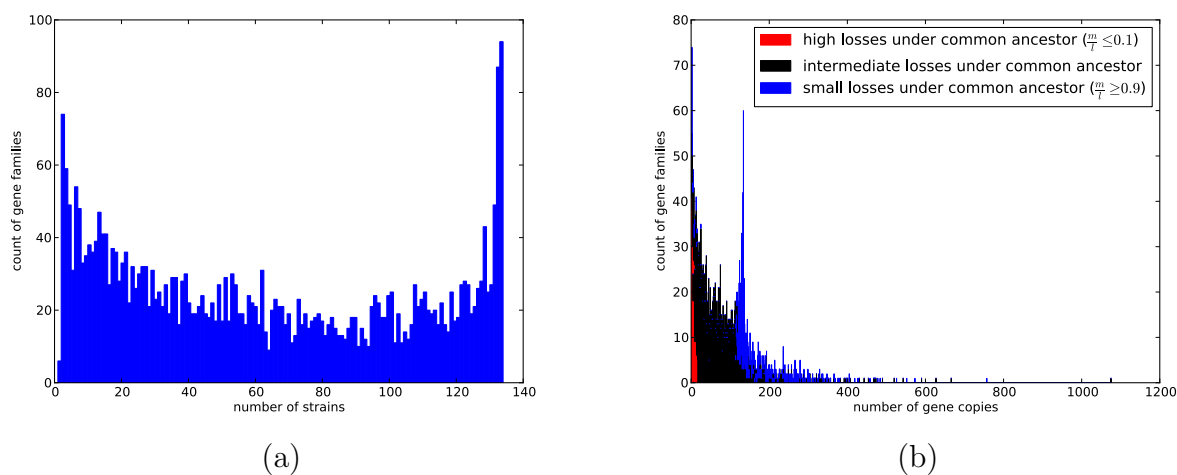


Figure 6.3 : The reconstructed gene family trees are counted by the number of strains (a) and the number of gene copies (b), respectively. In (b), we distinguish the trees with different loss amounts.

eggNOG generates multiple sequence alignments for the gene families [147] and builds trees by taking the consensus of the reconstructed maximum-likelihood trees from the sequences and their bootstrapped duplicates [148]. From the gene family trees downloaded from eggNOG, Figure 6.3 shows the frequency of the trees with specific numbers of strains and gene copies, respectively. The gene trees are heterogeneous both in terms of the strains and the number of gene copies each species has. However, it is important to note that the number of copies in the trees does

not necessarily indicate gene loss or duplication, since there are other processes that can result in these situation. Because bacterial species are believed to undergo massive transfer events, it is likely that the sampled γ -proteobacteria data set includes some genes transferred from out of the sample, which would yield small gene trees without loss. Since both transfer from outside and loss can yield small gene trees, in order to check loss and duplication in the data set distinguished especially from the transfers from outside, we distinguished the trees by the amount of species loss from the common ancestor in Figure 6.3 (b). If there is no transfer from outside, then we expect a tree that contains many leaves to have few species losses and a tree with few species to have many losses, assuming that all evolved from the same root. However, if there are trees of few species with small losses, that have a common ancestor lower than the root of the species tree, either a selection process selectively kills the gene in all species outside the common ancestor in the species tree, or the gene tree has not evolved from the root of the species tree but was transferred from outside at that point in the species evolution, affecting only the species that developed after that transfer. Assuming that loss generally happens only by chance, the trees of small species with small losses are taken as a sign of transfers from outside the sample.

In order to make use of this observation in analysis, we calculate the following measure for each tree. Given a gene family tree G of m species, we locate the common ancestor of the species of G on the species tree. Suppose that there are l species under the common ancestor on the species tree. If $\frac{m}{l} \leq 0.1$, then we count this as a high loss under the common ancestor, and if $\frac{m}{l} \geq 0.9$, then this counts as a small loss under the common ancestor, since $l - m$ represents the amount of species loss on the species tree under the common ancestor. It is important to note that $l - m$ does not directly indicate the number of actual loss events, as calculating the actual loss events

should involve the detection of duplication and transfer events. In the figure, large red bars show small numbers of gene copies, and large blue bars show a large numbers of them, if there is no transfer from outside. The red bars for a small number of gene copies indicate the trees in which genes are lost widespread, and they are taken as a potential sign of loss. On the other hand, the blue bars for a large number of gene copies, especially those with more than 134 gene copies, are taken as a potential sign of duplication. On the other hand, the blue bars for small number of gene copies might indicate transfers from the outside.

Besides duplication and loss, a body of research based on molecular and phylogenetic analyses identifies the abundance of transfer in prokaryotic evolution, including γ -proteobacterial evolution [149, 150, 151], even though it is controversial how pervasive it actually is [141, 152, 153, 154].

Putting all of them together, it is clear that γ -proteobacterial evolution contains duplication, loss, and transfer events, for which there are both model-based and parsimony-based approaches. However, due to the lack of knowledge for parameters in the models and their slow running time, model-based approaches are not yet suited for large-scale data analysis. We therefore employ a parsimony-based approach, since it requires fewer parameters and quickly analyzes large-scale data.

Tofigh *et al.* proposes a dynamic-programming (DP) algorithm that returns the cost of the most parsimonious mapping based on a cost scheme [145]. While it guarantees consideration of all possible transfer events based on a cost scheme of the cost for duplication and transfer, several algorithms have been proposed to address its issues [155, 156, 157, 158, 159, 160] (1) to incorporate the cost value for loss and (2) to account for the possible time-constraint violations as a solution to the algorithm (Readers are referred to [161] for more detail). However, more study is

required to make the extensions beneficial for analysis. The cost for loss has not been well-characterized for most evolutionary history until now, so do algorithms taking this parameter into account not yet directly improve analysis. Algorithms of the second type of extension mainly focus on restricting time-inconsistent transfers, defined as those whose sources and destinations are within a time window on the dated species tree. As the extensions restrict the transfers, they reduce the search space, effectively prohibiting time-inconsistent transfers. However, given the difficulty to obtain accurately timed species trees, it is possible that this extension, when combined with an incorrectly timed species tree, will reduce the search space in an incorrect way, excluding solutions of interest. To avoid this issue, we run Tofigh *et al.*'s algorithm for analysis.

As the parsimony-based approach's computation depends on the species tree and cost scheme, we investigate the reliability of the estimated species tree and the effect of the cost scheme on event detection. Note that the result of the investigation would easily transmit to understanding other parsimonious reconciliation algorithms, since most of them share the same computational structure as Tofigh *et al.*'s algorithm.

6.2.2 Species Tree of the γ -Proteobacterial Data Set

Since a species tree represents the evolutionary history of species, correctly estimated species trees play a crucial role in addressing important problems for the species, including the detection of evolutionary events.

However, no algorithm has been proposed to infer species trees under DTL. Instead, when it is assumed that transfers are rare in a data set compared to duplication and loss, species tree is estimated by minimizing only duplication and loss. In this case, a transfer would be interpreted as a duplication event, which adds false loss

events on the species tree's edges ranging from the source and the destination of the transfer. However, if data set actually has few transfers, then the correct branching order of the species tree would not be disrupted.

Assuming that transfer are rare in γ -proteobacterial evolution, the species tree for the γ -proteobacteria data set can be estimated under DL using DupTree [6]. Figure 6.4 shows the estimated species tree. The tree is quite different from the NCBI taxonomy, as the RF distance value between them is 57. Given that the fully resolved tree in Figure 6.4 is composed of 132 internal edges (bipartitions), around half of the bipartitions in each tree are not found in the other. In comparison to the NCBI taxonomy, it is not clear which bipartitions in the estimated tree are correct representations of speciation events. However, it is worth mentioning that the estimated species tree has many bipartitions in common with that reconstructed from a large set of protein families [162].

In order to reconstruct a more accurate species tree, the current estimation should consider some issues. First, an evolutionary process involving duplication, loss, and transfer does not necessarily happen in parsimony. This naturally motivates the need to develop a model-based approach that can account for the non-parsimony nature of the process. However, to be useful in analysis, developing a model-based approach requires not only defining a model but also studying appropriate values for the parameters, which is beyond the scope of this chapter. Secondly, it is difficult to verify how much duplication, loss, and transfer occur in the data. We must first determine how robust the estimated tree is against evolutionary events in order to answer the question of what would happen if genetic transfer greatly affects the estimate. In the next section, we explore the effect of evolutionary events on the current species tree estimation.

6.2.3 The Effect of Loss and Transfer on Species Tree Estimation

We can determine the effect of evolutionary events on the current species tree estimation by simulating additional evolutionary events and evaluating changes in the species tree. These simulations will allow us to estimate the performance of the current parsimonious reconciliation algorithm based on the species tree. Transfer and duplication/loss evolutionary events cause gene tree-species tree incongruence at the species level. However, duplication should be accompanied by loss in order to cause the incongruence, and without loss, it does not skew the estimation. In this chapter, we focus on simulating loss and transfer, not duplication. The simulation is based on the simple hypothesis of loss and transfer: loss can happen at any node in the tree, removing all nodes under it, and transfer can happen between any species, as long as it does not violate the time constraint of the previous transfers.

A simulation of loss and transfer events on a gene family tree is given a percentage value. With this value, we first determine the number of events as the ceiling of the product of the percentage value and the number of the nodes of the tree. For example, the number of the events for a tree of 200 nodes, either internal nodes or leaves, is 2, with 0.01 (1%) as the percentage value. With the number of events determined, loss is simulated by (1) randomly selecting the number of non-root nodes on the tree, whether an internal node or a leaf, (2) removing all nodes under them, and (3) refining the tree. The tree refinement process first locates an internal edge with a single child and then removes the edge after connecting its parent to its children, until no such edge remains in the tree. As for the transfer simulation, the number of SPR moves are applied to the tree, one at a time, while they are prohibited from going to their ancestor. This guarantees that the simulated transfers do not result in a cycle. After simulating a given percentage i of loss or transfer events on each gene

tree, DupTree can estimate the species tree ST_i based on them. In order to obtain statistical significance, the simulation process is repeated 30 times for a percentage value to generate 30 ST_i s. The 30 ST_i s are compared with ST , the species tree estimated from the original gene trees, by RF distance.

6.2.4 The Effect of the Cost Scheme on Reconciliation

In order to investigate the parsimonious detection of evolutionary events under DTL, we use Tofigh *et al.*'s algorithm in this section for two reasons: as discussed in Chapter 2, the extensions are not very useful without knowing their parameters, and most of them have not been well-characterized. Secondly, since most of the parsimonious reconciliation algorithms share the computational structure of Tofigh *et al.*'s algorithm, we expect that observations for this algorithm can apply to other parsimonious reconciliation algorithms.

The appropriate cost values for duplication and transfer, known as the cost scheme, are crucial to accurately detect evolutionary events, since the computation calculates the parsimony score based on the cost scheme. However, the cost for duplication and transfer has neither been applied to biological data sets nor studied systematically with simulated data sets, so it is difficult to figure out the appropriate cost scheme for a specific data set.

We study how the cost scheme affects the performance of the algorithm by running the algorithm on the gene trees with different cost schemes. Given gene tree G and species tree S , Tofigh *et al.*'s algorithm returns the cost of the most parsimonious mapping under DTL using speciation, duplication, and transfer based on a given cost scheme. In particular given $u \in V(G)$ and $x \in V(S)$, speciation and duplication match u 's children with a node under x , and transfer matches the children with a

node incomparable with x . Since the algorithm does not return evolutionary events, we develop an algorithm returning the events associated with the parsimonious mapping. Developing such an algorithm, especially one that operates without losing its dynamic programming efficiency, is not a trivial task. The algorithm should address the issue of multiple optimal scenarios and cases where a transfer is associated with multiple candidates for a parsimonious source. Algorithm 1 is an efficient algorithm that returns the optimal set of evolutionary events under a cost scheme in $O(|V(G)| \cdot \log|V(S)|)$, provided that the annotations are given to each node in the gene tree by Tofgh *et al.*'s algorithm.

As Tofgh *et al.*'s algorithm navigates all the combinations of $u \in V(G)$ and $x \in V(S)$ given a gene tree G and a species tree S , we first annotate the gene tree nodes with the events it is associated with, either speciation, duplication, or transfer. In particular when v, w are u 's children and y, z are x 's children, we specify the speciation case where v is mapped with z and w is mapped with y as '*Speciation_{vz}*', and the speciation case where v is mapped with y and w is mapped with z as '*Speciation_{vy}*'. In the same sense, '*Transfer_v*' labels u when v is due to transfer, and '*Transfer_w*' labels u when w is due to transfer. Since the parsimonious reconciliation assumes that at least one of the children does not involve transfer, the cases represent all possibilities. Note that since different event assignments can yield the same optimal score, u can carry multiple annotations. For the purpose of retrieving the sources of transfers, two other values are maintained for (u, x) , $u \in V(G), x \in V(S)$, $tFromOut[u, x]$ and $outSrc[u, x]$. In particular, $tFromOutside[u, x] \in \{ 'yes', 'no' \}$ indicates the location between two incomparable places in the tree, either x 's ancestors ('*yes*') or the sibling of x ('*no*'), where the match originates. $outSrc[u, x]$ points to the ancestor or the parsimonious matching of the transfer in the sibling, depending on the corresponding

value of $tFromOutside$. Using the annotations and the two values for each mapping, Algorithm 1 identifies Σ, Δ, Θ and the source of the transfers with respect to the γ .

While it is straightforward to annotate u in Tofigh *et al.*'s algorithm, the search for the source of the transfers should be explained in more detail. We can update $outside[u, y]$ in Tofigh *et al.*'s algorithm with the minimum value from either $below[u, z]$, where z is the sibling of y or $outside[u, x]$ where x is the parent of y ; if the value comes from $below[u, z]$, $tFromOutside[u, x]$ is set to 'no' and $outSrc[u, y]$ is set to z , and if the value comes from $outside[u, x]$, $tFromOutside[u, x]$ is set to 'yes' and $outSrc[u, y]$ is set to x . The key observation to detect the source is that the parsimonious mapping of (u, x) in $outside$ comes from a sibling of an ancestor of u on the species tree. As Algorithm 1 detects all events under $u \in V(G)$, all the events are detected by invoking Algorithm 1 with $R(G)$ and $R(S)$.

Even though the algorithm keeps a single node as the source of a transfer in order to keep the correspondence between a transfer and its source, it is true that there can be multiple candidates for the parsimonious source for a transfer. However, the complexity of retrieving all parsimonious candidates as the source is the same as our algorithm, since they are found as the search goes up to $R(S)$.

Algorithm 1: The algorithm retrieving the evolutionary events under u

Input: node $u \in V(G)$, node $x \in V(S)$
Output: Σ, Δ, Θ and a set of nodes $SRC \subseteq V(S)$ detected as a src of $t \in \Theta$

if $u.event$ is notated as *Duplication* **then**
 $v, w \leftarrow$ children of u
 $\Sigma_{d1}, \Delta_{d1}, \Theta_{d1}, dummy = \text{Algorithm 1}(v, x, S)$
 $\Sigma_{d2}, \Delta_{d2}, \Theta_{d2}, dummy = \text{Algorithm 1}(w, x, S)$
 $\Sigma_D = \Sigma_{d1} + \Sigma_{d2}, \Delta_D = \Delta_{d1} + \Delta_{d2}, \Theta_D = \Theta_{d1} + \Theta_{d2}, SRC_D = \{x\}$

else if $u.event$ is notated as *Speciation_{vz}* **then**
 $v, w \leftarrow$ children of u and $y, z \leftarrow$ children of x
 $\Sigma_{s1}, \Delta_{s1}, \Theta_{s1}, dummy = \text{Algorithm 1}(v, z, S)$
 $\Sigma_{s2}, \Delta_{s2}, \Theta_{s2}, dummy = \text{Algorithm 1}(w, y, S)$
 $\Sigma_S = \Sigma_{s1} + \Sigma_{s2} + \{u\}, \Delta_S = \Delta_{s1} + \Delta_{s2}, \Theta_S = \Theta_{s1} + \Theta_{s2}, SRC_S = \{x\}$

else if $u.event$ is notated as *Speciation_{vy}* **then**
 $v, w \leftarrow$ children of u and $y, z \leftarrow$ children of x
 $\Sigma_{s1}, \Delta_{s1}, \Theta_{s1}, dummy = \text{Algorithm 1}(v, y, S)$
 $\Sigma_{s2}, \Delta_{s2}, \Theta_{s2}, dummy = \text{Algorithm 1}(w, z, S)$
 $\Sigma_S = \Sigma_{s1} + \Sigma_{s2} + \{u\}, \Delta_S = \Delta_{s1} + \Delta_{s2}, \Theta_S = \Theta_{s1} + \Theta_{s2}, SRC_S = \{x\}$

else if $u.event$ is notated as *Transfer_v* **then**
 $v, w \leftarrow$ children of u
 $\Sigma_{t1}, \Delta_{t1}, \Theta_{t1}, SRC_{t1} = \text{Algorithm 1}(w, x, S)$
 while $tFromOutside[v, x] == 'yes'$ **do**
 $x = outSrc[v, x]$
 end
 $\Sigma_{t2}, \Delta_{t2}, \Theta_{t2}, dummy = \text{Algorithm 1}(v, x, S)$
 $\Sigma_T = \Sigma_{t1} + \Sigma_{t2}, \Delta_T = \Delta_{t1} + \Delta_{t2}, \Theta_T = \Theta_{t1} + \Theta_{t2} + \{v\}, SRC_T = SRC_{t1}$

else if $u.event$ is notated as *Transfer_w* **then**
 $v, w \leftarrow$ children of u
 $\Sigma_{t1}, \Delta_{t1}, \Theta_{t1}, SRC_{t1} = \text{Algorithm 1}(v, x, S)$
 while $tFromOutside[w, x] == 'yes'$ **do**
 $x = outSrc[w, x]$
 end
 $\Sigma_{t2}, \Delta_{t2}, \Theta_{t2}, dummy = \text{Algorithm 1}(w, x, S)$
 $\Sigma_T = \Sigma_{t1} + \Sigma_{t2}, \Delta_T = \Delta_{t1} + \Delta_{t2}, \Theta_T = \Theta_{t1} + \Theta_{t2} + \{w\}, SRC_T = SRC_{t1}$

else
 if x is not a leaf **then**
 $y, z \leftarrow$ children of x
 if $u.b[y] < u.b[z]$ **then**
 $\Sigma_X, \Delta_X, \Theta_X, SRC_X = \text{Algorithm 1}(u, y, S)$
 else
 $\Sigma_X, \Delta_X, \Theta_X, SRC_X = \text{Algorithm 1}(u, z, S)$
 end
 else
 $SRC_X = \{x\}$
 end

end
return $\Sigma_D + \Sigma_S + \Sigma_T + \Sigma_X, \Delta_D + \Delta_S + \Delta_T + \Delta_X, \Theta_D + \Theta_S + \Theta_T + \Theta_X, SRC_D + SRC_S + SRC_T + SRC_X$

6.3 Results

6.3.1 The Effect of Loss and Transfer on Species Tree Estimation

In the last section, we determined that the current species tree estimation assumes that transfer does not affect greatly the species tree estimation. In order to estimate the reliability of the species tree reconstructed under this assumption, we conduct a simulation study where species trees, ST_i s, are estimated from the gene trees on which loss and transfer events are imposed. The ST_i s are compared with the species tree ST estimated in the same way from the original gene trees by RF distance, and their average and standard deviation values are shown by bar and error bar, respectively, in Figure 6.5. In particular, the x-axis represents the loss percentage values tested, which are 0.01 (1%), 0.04 (4%), 0.16 (16%), 0.24 (24%), and 0.32 (32%); the y-axis of Figure 6.5(a) plots normalized RF distances, obtained by dividing the RF distance value by the number of internal edges in the tree. Since the number of internal edges is the maximum number of bipartitions, the normalized RF puts the values between 0 and 1. Figure 6.5 (a) shows the average and the standard deviation values of the normalized RF distances between ST and ST_i s by loss percentage.

The figure from the loss simulation shows that the species tree estimation is robust to loss events, since even at 16% of loss simulation, ST_i s and ST do not differ greatly. Since a loss event does not impact the remaining part of the tree, the discordance between ST and ST_i comes solely from the information lost in the simulation. In order to check the amount of information lost, Figure 6.5(b) and (c) plot the number of the retained trees and the ratios of the remaining leaves in the trees after the

simulation for different loss percentage values. Considering that a substantial amount of information is lost both in terms of the number of trees and the leaves in the data in the simulation with 16% loss, the species tree estimation is clearly robust to loss.

While Figure 6.5(b) plots the actual number of retained trees, Figure 6.5(c) shows the percentage values of the retained leaves. As gene tree G loses l leaves in the simulation, the percentage value on the y-axis is $\frac{|L(G)|-l}{|L(G)|}$. Note that the percentage values in Figure 6.5(c) widely vary, because the simulation selects internal nodes of different heights. As the internal nodes closer to the root are selected, the set of the retained leaves decreases, because the simulation removes all leaves under the selected nodes.

From the simulation, we can see that when multiple gene trees are collected to estimate the species tree, the low coverage of a single gene tree would not deteriorate the quality of the estimation, since the estimation is robust to information loss. With this observation, we learn that in sampling the gene trees for the estimation, the focus should be on obtaining the correct branching order, rather than trying to include as many gene copies as possible.

Figure 6.6 shows the result of the transfer simulation. Figure 6.6(a) uses the normalized RF distance as the y-axis, and it plots transfer percentage values 0.01 (1%), 0.04 (4%), 0.16 (16%), 0.24 (24%), 0.32 (32%), and 0.64 (64%). In particular, Figure 6.6 (a) shows the average and the standard deviation values of RF distances between ST and ST_i by transfer percentage, and (b) shows the average and the standard deviation values of the RF distances between the original gene trees and the corresponding trees on which a different percentage of transfer is simulated. These figures show that the species tree estimation infers the original tree, even though the bipartitions of the gene trees differ by 20%. However, it is true that 1% of transfer,

which is on average 1.7 HGT events on each tree, incurs the 20% bipartition difference. The bipartition difference increases with a higher percentage value, and as the percentage value approaches 4%, the ST_i s become about 40% different from ST . The species tree estimation is clearly vulnerable to transfers, especially in comparison to the loss simulation. While it makes sense that the difference comes from the manipulated branching orders of the gene trees plotted in Figure 6.6(b), it is interesting that the changed branching orders do not proportionally transmit to the incorrect estimation, since Figure 6.6(a) and (b) do not show the same increase pattern.

The question remains; how plausible are these values in practice? It is possible that 1% transfer is too severe compared to what can actually happen. Also, it might be unreasonable to believe that such a percentage of transfer happens to all trees. However, it is not unusual that gene trees convey incorrect branching orders from noise and/or reconstruction error, and that would skew the estimation as the simulation does. What the transfer simulation shows is that the current species tree estimation is very vulnerable to the branching order changes either from noise/error or from transfer, and since the estimation is vulnerable to transfer, it is important that the species tree estimation should account for transfer.

6.3.2 The Effect of the Cost Scheme on Reconciliation

Given species tree S and a gene tree G , Algorithm 1 can identify the internal nodes in G associated with duplication, speciation, or transfer events under a cost scheme, by being coupled with Tofigh *et al.*'s algorithm. So with a particular cost scheme, Tofigh *et al.*'s algorithm and Algorithm 1 are run to detect duplication and transfer events on each gene tree. From the detection, the ratio of the duplication and the transfer events to the sum of the events is calculated. As the ratio values are retrieved

across the γ -proteobacteria gene trees, Figure 6.7 plots the average and the standard deviation of the values in bar and in error bar, respectively, by different cost schemes. In the figure, the cost schemes are represented by (Cd/Ct) , where Cd represents the cost for duplication and Ct represents the cost for transfer.

When a cost scheme bears a higher cost for transfer and penalizes transfer more heavily, it makes sense that more of the nodes are labeled as duplication, in order to yield a smaller parsimony score. The same expectation holds true for a scheme bearing a higher cost for duplication. Figure 6.7 shows that a cost scheme with a high cost for transfer yields a high red bar, indicating that many nodes in the gene trees are labeled as transfers; a cost scheme with a high cost for duplication yields a high blue bar, indicating that many nodes are labeled as duplication.

In the figure, the first thing to note is that the blue and the red bars are not symmetric across the cost schemes, where nodes tend to be labeled as duplication more than as transfer. In particular, in the cost schemes with a high penalty for transfer, there are fewer transfers retained than in cases with cost schemes with a high penalty for duplication. For example, while in the estimation with $(0.1/0.9)$, no tree has transfer labels, an estimation with $(0.9/0.1)$ yields some duplications. This result might imply that duplications have more clear signs than transfers in the data set.

Because different cost schemes yield different sets of detected events, the question becomes which cost matrix to use. Tofigh *et al.* found, through simulation, that if the true scenario is optimal under some cost scheme, $(0.5/0.5)$ cost scheme almost always catches the scenario as an optimal solution. Also note that using $(0.5/0.5)$ cost scheme makes sense in parsimony, because this cost scheme minimizes the sum of the number of the events. However, no study has verified any of the cost schemes

from a biological point of view, so it is still not clear which cost scheme should be used to accurately detect evolutionary events in the γ -proteobacteria data set.

While it is not certain which cost matrix is appropriate for the data, Figure 6.8 of the ratio values of individual trees poses another question for using the cost scheme in the detection process. As the ratio of the events is calculated on the trees, Figure 6.8 randomly selects 10 individual gene trees and shows their duplication ratio values without taking the average. While the number of nodes labeled as duplication is dropped as the estimation uses a bigger cost value for duplication, Figure 6.8 shows that the pattern is different. It implies that the gene trees have duplication/transfer signals with different strengths. It naturally follows that in order to catch the signals correctly, different cost schemes might be applied to the gene trees.

The results show some issues in using a parsimonious reconciliation algorithm for biological data analysis, related to the species tree estimation under the current assumption and the issues of the cost scheme and the uniform use of a cost scheme on the gene trees.

6.4 Discussion

In this chapter, we studied the performance of parsimonious reconciliation under the DTL model in detecting evolutionary events. For this study, we developed an algorithm to retrieve the detected events in Tofigh *et al.*'s algorithm.

6.4.1 Conclusions

In studying the performance, we investigate the two important factors of the reconciliation: the species tree estimation and the cost scheme. In the investigation of species tree estimation, we particularly focus on the effect of loss and transfer events

on the performance, to determine the effect of the unidentified rates of evolutionary events on the current species tree estimation algorithm. As long as gene trees contain accurate branching orders and collectively cover the entire species, poor coverage of a tree would not greatly deteriorate the quality of the estimation.

On the other hand, results from the transfer simulation show that transfer events affect the accuracy of the estimation by effectively disguising the bipartition information of the gene trees. This finding negates the assumption of the negligible effect of transfer on species tree estimation and calls for an algorithm that takes transfer into account.

While identifying the species tree is an important subject of study in itself, the accurately estimated species tree is used as the basis for other biological studies such as the identification of evolutionary events under the DTL model, as the identification is made in reference to the species tree. When the identification is by parsimony, another important issue is the cost scheme, which calculates the parsimony score of the event assignment.

Even though a simulation study supports the equal cost value for duplication and transfer, which makes sense in parsimony, it is not clear which combination of cost values is appropriate for biological data. Additionally, the simulation study shows that using a cost scheme uniformly for all gene trees might lead to inaccurate detection.

While the study brings up some issues of the parsimonious reconciliation and the events it detects, the results give an idea of the significance of estimation. In Figure 6.7, it is easy to suppose that the set of nodes of a gene tree estimated as a particular event under a cost scheme are estimated as the same event under a cost scheme that penalizes the event less. For example, in Figure 6.7, the nodes estimated

as duplication under 0.9/0.1 ($Cd=0.9$) are estimated as duplication under 0.8/0.2 ($Cd=0.8$) and 0.7/0.3 ($Cd=0.7$). As 0.1/0.9 clearly is a severe cost scheme to detect transfer, and 0.9/0.1 for duplication, 0.7/0.3 might be a good cost scheme that reflects the signal strength of the significant duplications, in the sense that it retains most of the nodes estimated as duplication in the most severe condition. On the other hand, the figure shows that the most severe condition for transfer does not detect any transfer, unlike duplication. In this case, 0.3/0.7 might be a cost scheme for significant transfer, since γ -proteobacteria is known to have many transfers.

6.4.2 Future Work

In this chapter, we investigate some issues of parsimonious reconciliation in detecting evolutionary events and discuss how to obtain significant estimates of these events. Even though one can infer significant estimates and the corresponding cost scheme for the γ -proteobacteria data using the rationale described as above, it is true that the significance estimation is very *ad hoc* and data-specific. It is imperative to develop a systematic method to obtain significant estimates under the DTL model.

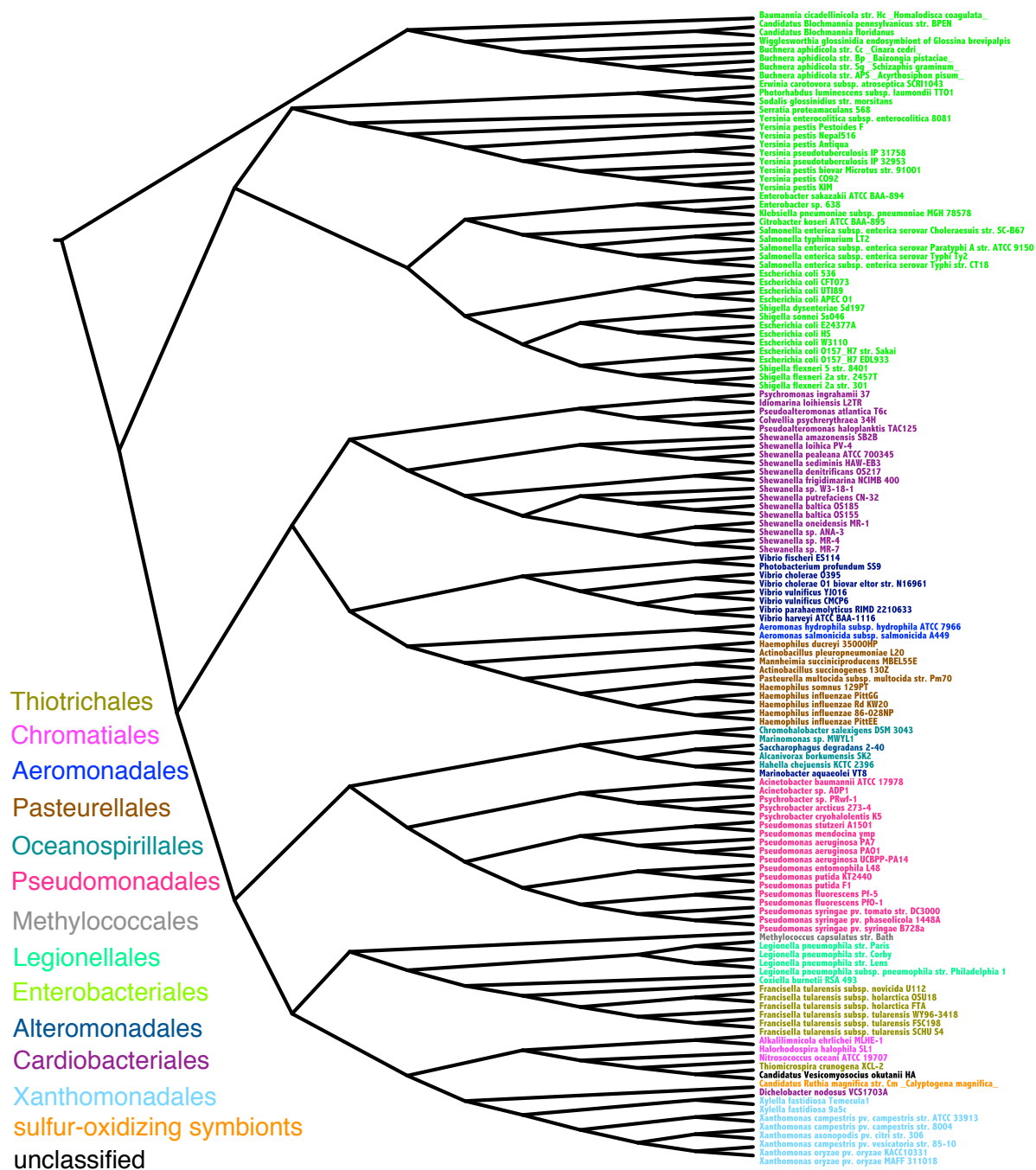


Figure 6.4 : Species tree of the γ -proteobacterial strains estimated by DupTree [6] based on the γ -proteobacteria gene trees. Strains are colored by taxonomic order.

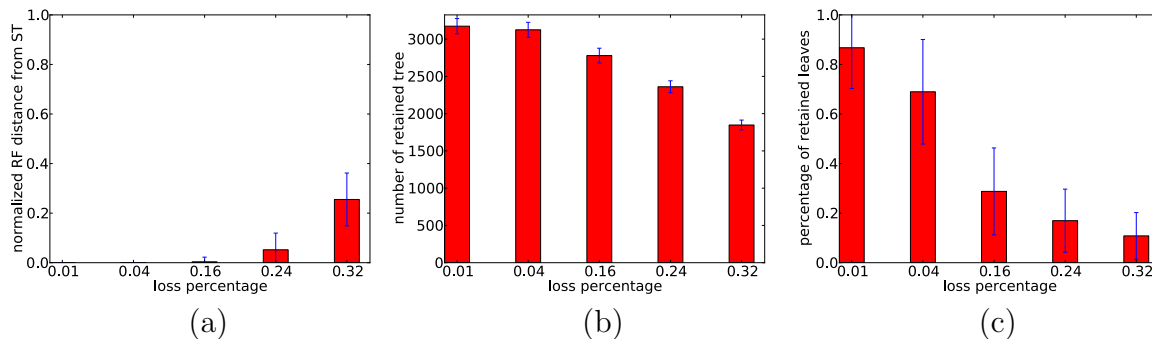


Figure 6.5 : The RF distance values between ST and ST_i from the loss simulations. The percentage value i is given on the x-axis and the normalized RF distance value between ST_i and ST is plotted on the y-axis. (a) The average (in bars) and the standard deviation (in error bars) of the RF distance values. While the loss simulation decreases the number of leaves in a tree and sometimes removes all nodes of a tree, (b) shows the number of the trees that the simulation retains to use for the estimation, and (c) shows the percentage of retained leaves in the trees.

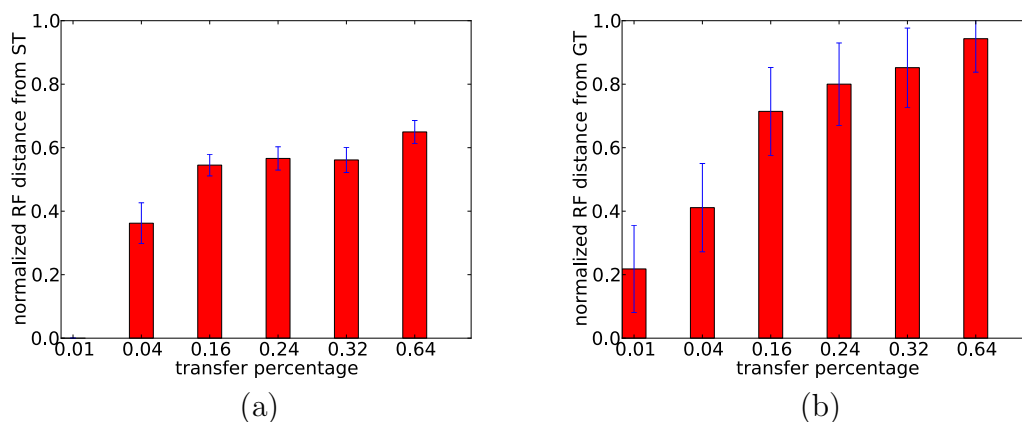


Figure 6.6 : Results from the transfer simulations. The values on the x- and y-axis are similar as in Figure 6.5. (a) shows the normalized RF distance values between ST and ST_i , and (b) plots the average of the normalized RF distance values between gene trees and their manipulated trees with the transfers in bars and the standard deviation values in error bars.

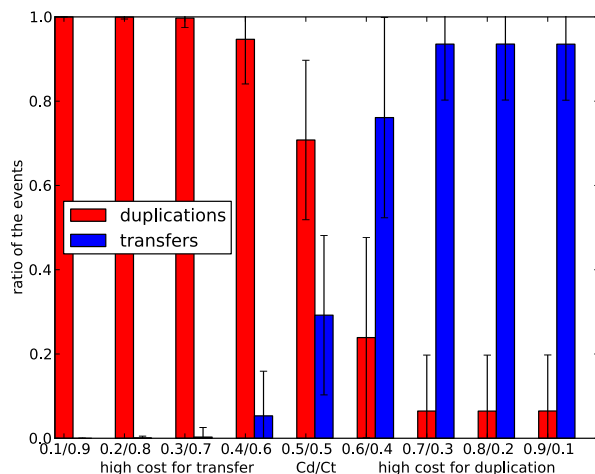


Figure 6.7 : The ratios of the duplication and transfer events across γ -proteobacteria gene trees in average (bars) and standard deviation values (error bars) based on a specific cost scheme. The sets of the values as cost schemes are denoted on the x-axis. Basically, the value before the slash (Cd) refers to the cost value for duplication, and that after the slash (Ct) refers to the cost for transfer. Schemes on the left side of the x-axis bear high costs for transfer, and those on the right bear high costs for duplication. With the cost schemes, the detections are conducted by Tofigh *et al.*'s algorithm and Algorithm 1.

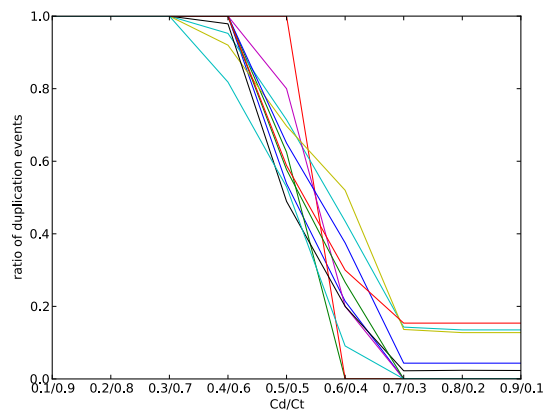


Figure 6.8 : The ratios of duplication of 10 gene trees of the γ -proteobacteria data by different cost schemes. The ratios of the trees of different cost values is plotted by color. Note that the ratios of the transfer of the gene trees show a complementary pattern.

Chapter 7

Conclusions

7.1 Discussion

The current scale of biological data clearly shows several evolutionary events as the cause of gene tree/species tree incongruence. Because these events are rampant in species evolution, a species tree loses its resolution to model genomic evolution, and a better model is needed to represent it. Among several models that represent the genomic evolutionary history, phylogenetic networks are widely used for their direct interpretability and efficiency.

In this dissertation, we first focus on developing computational methods that reconstruct phylogenetic networks under an optimization criteria as a first step for the large-scale data analysis. As both criteria, MP and ML, have a tendency to overestimate HGT events, we developed methods to suppress this tendency by estimating the significance of the estimates and allowing only significant ones. Since MP and ML measure optimality in a different way, we estimate the significance differently in them. Under MP, because the parsimony score does not indicate significance of an estimate, we developed a bootstrap-based algorithm. This algorithm not only performs well in identifying significant HGT events, but also allows analysis to use the estimated significance to make a sophisticated and useful interpretation of the result. On the other hand, even though the likelihood score can directly be used to estimate the significance, the problem is that it inherently prefers a complex model,

usually resulting in overestimation. For ML, we study the effect of some properties of an HGT on its identifiability and investigate the performance of two information criteria for addressing the issue of overestimation. Our study establishes ML as a good criterion and allows the analysis to make a better use of this criterion.

Another advantage of the methods proposed in this dissertation is their computational structure; our methods set up the problems in multiple layers such that the problem in a lower layer is connected to a simpler computational task. Given progress made in phylogenetic study, making use of the efficient algorithms for the pre-existing problems can be an efficient and reasonable solution for problems at a higher layer. In that regard, both the reticulation detection algorithms under MP and ML place an external tree-scoring program in the lower layer, not bound by an internal module. As a result, it is easy to switch algorithms for the module and advances in tree-scoring programs can directly be incorporated. Further, as M1, M2, and MURPAR in Chapter 5 make use of an external program for pairwise SPR calculation, the result shows that they perform better than their competing tools, while staying open for improvement in pairwise SPR computation algorithms. In particular, while the current pairwise SPR computation algorithms concern only the minimum distance, if there is a method that returns all or most of the possible SPR solutions, no matter the distance, then MURPAR would generate a close estimate to the global optimum, since a part of the global minimal set of SPR moves for a set of trees should address any pair in the set, even if it is not always the minimum for that pair. In the last chapter, we study the performance of a parsimonious reconciliation under the DTL model and discuss issues of reconciliation for analysis. Since significance estimation is critical for analysis, we suggest *ad hoc* ways to infer the significant estimates.

In sum, we propose computational methods to estimate the significance of the

HGT events using an efficient computational structure. Also, we demonstrate how a computational method would contribute to biological data analysis. While this dissertation introduces efficient computational methods for the phylogenetic network reconstruction problem, we believe that it also bridges the gap between computational methods for biology and the actual problems they intend to solve.

7.2 Future Directions

We present several methods to detect particular evolutionary events, and they show good performance in both simulation studies and biological data. However, they still need extensive validation with biological data. When a computational method is proposed for a biological problem, it usually comes with many assumptions. For example, besides the assumption of our methods that reticulation is the only cause for incongruence, we also assume the independence between genetic evolution and the minimum amount of signs a reticulate event would leave on the sequence. However, as we found in Chapter 4, the signal strength of an event would be affected by many factors, and a combination of these factors might break some of the computational assumptions. A computational method should be extensively validated with various biological data before using it for biological data analysis.

A more serious challenge lies in the assumption that all methods proposed in this dissertation are based on the cause of the incongruence, because it is clear that this assumption can hardly hold in practice, especially as data cover many loci on a large set of species. As the data is collected at the species level, duplication and loss would be dominant together with transfer over other evolutionary events. There are a number of algorithms that attempt to detect these events under the model allowing for both events. However, their performances are not well-studied, as to how well they

detect the ancient transfer events or those transfer events that span a far distance. It is important to study their performance before conducting analysis with them in order to make accurate interpretations from that analysis. In particular, as parsimony-based approaches are more suited for large-scale data analysis, it is important to study their performance in regard to the cost schemes, with which parsimony scores are calculated. In order to obtain significant estimates without prior knowledge in the parsimony-based approaches, one can use the estimates based on a conservative matrix for the analysis, as we suggested in the last chapter. However, it is critical to develop a method to determine the significance of the estimates in a more systematic way.

The advancement of technology produces an extensive scale of biological data, in which multiple kinds of events can occur on a large set of species; the phylogenetic community has developed several algorithms that either detect multiple kinds of events or conduct a fine-grained search with constraints and significance estimation. However, obtaining significant estimates for the former approaches has yet to be determined. Because it is critical to conduct an extensive biological data analysis, our efforts will help develop such methods in the future.

Bibliography

- [1] C. Delwiche and J. Palmer, “Rampant horizontal transfer and duplication of rubisco genes in eubacteria and plastids,” *Mol. Biol. Evol.*, vol. 13, pp. 873–882, 1996.
- [2] A. Rokas, B. Williams, N. King, and S. Carroll, “Genome-scale approaches to resolving incongruence in molecular phylogenies,” *Nature*, vol. 425, pp. 798–804, 2003.
- [3] H. Park, G. Jin, and L. Nakhleh, “Algorithmic strategies for estimating the amount of reticulation from a collection of gene trees,” in *Proceedings of the 9th Annual International Conference on Computational Systems Biology*, pp. 114–123, 2010.
- [4] Y. Wu, “Close lower and upper bounds for the minimum reticulate network of multiple phylogenetic trees,” *Bioinformatics [ISMB]*, vol. 26, no. 12, pp. 140–148, 2010.
- [5] L. van Iersel, S. Kelk, R. Rupp, and D. Huson, “Phylogenetic networks do not need to be complex: using fewer reticulations to represent conflicting clusters,” *Bioinformatics [ISMB]*, vol. 26, pp. i124–i131, June 2010.
- [6] A. Wehe, M. Bansal, J. Burleigh, and O. Eulenstein, “DupTree a program for large-scale phylogenetic analyses using gene tree parsimony,” *Bioinformatics*, vol. 24, pp. 1540–1541, 2008.

- [7] W. Maddison, "Gene trees in species trees," *Syst. Biol.*, vol. 46, no. 3, pp. 523–536, 1997.
- [8] W. Doolittle, "Phylogenetic classification and the universal tree," *Science*, vol. 284, pp. 2124–2129, 1999.
- [9] C. Kurland, B. Canback, and O. Berg, "Horizontal gene transfer: A critical view," *Proc. Nat'l Acad. Sci., USA*, vol. 100, no. 17, pp. 9658–9662, 2003.
- [10] M. McClelland, K. E. Sanderson, S. W. Clifton, P. Latreille, S. Porwollik, A. Sabo, R. Meyer, T. Bieri, P. Ozersky, M. McLellan, C. R. Harkins, C. Wang, C. Nguyen, A. Berghoff, G. Elliott, S. Kohlberg, C. Strong, F. Du, J. Carter, C. Kremizki, D. Layman, S. Leonard, H. Sun, L. Fulton, W. Nash, T. Miner, P. Minx, K. Delehaunty, C. Fronick, V. Magrini, M. Nhan, W. Warren, L. Florea, J. Spieth, and R. K. Wilson, "Comparison of genome degradation in paratyphi a and typhi, human-restricted serovars of salmonella enterica that cause typhoid," *Nature Genetics*, vol. 36, pp. 1268–1274, 2004.
- [11] Y. Nakamura, T. Itoh, H. Matsuda, and T. Gojobor, "Biased biological functions of horizontally transferred genes in prokaryotic genomes," *Nature Genetics*, vol. 36, pp. 760–766, 2004.
- [12] H. Ochman and I. Jones, "Evolutionary dynamics of full genome content in *Escherichia coli*," *Embo J.*, vol. 19, no. 24, pp. 6637–6643, 2000.
- [13] R. Welch, V. Burland, G. Plunkett, P. Redford, P. Roesch, D. Rasko, E. Buckles, S. Liou, A. Boutin, and J. Hackett, "Extensive mosaic structure revealed by the complete genome sequence of uropathogenic *Escherichia coli*," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 99, pp. 17020–17024, 2002.

- [14] U. Bergthorsson, K. Adams, B. Thomason, and J. Palmer, “Widespread horizontal transfer of mitochondrial genes in flowering plants,” *Nature*, vol. 424, pp. 197–201, 2003.
- [15] J. Mower, S. Stefanovic, G. Young, and J. Palmer, “Gene transfer from parasitic to host plants,” *Nature*, vol. 432, pp. 165–166, 2004.
- [16] M. Lynch and J. S. Conery, “The origins of genome complexity,” *Science*, vol. 302, pp. 1401–1404, Nov 2003.
- [17] M. Lynch and J. S. Conery, “The evolutionary fate and consequences of duplicate genes,” *Science*, vol. 290, p. 11511155, 2000.
- [18] A. Force and M. Lynch and F. B. Pickett and A. Amores and Y. N. Yan and J. Postlethwait, “Preservation of duplicate genes by complementary, degenerative mutations,” *Genetics*, vol. 151, pp. 1531–1545, Apr 1999.
- [19] F. Ohno, *Evolution by gene duplication*. E Allen and Unwin, 1970.
- [20] H. J. Bandelt, V. Macaulay, and M. Richards, “Median networks: speedy construction and greedy reduction, one simulation, and two case studies from human mtdna,” *Mol Phylogenet Evol*, vol. 16, pp. 8–28, Jul 2000.
- [21] B. Holland and V. Moulton, “Consensus networks: a method for visualizing incompatibilities in collections of trees,” in *Proceedings of the 2006 Workshop on Algorithms in BioInformatics (WABI2006)*, pp. 165–176, 2006.
- [22] D. Bryant and V. Moulton, “Neighbor-net: an agglomerative method for the construction of phylogenetic networks,” *Molecular Biology and Evolution*, vol. 21, pp. 255–65, Feb 2004.

- [23] C. Semple, “Hybridization networks,” in *Reconstructing Evolution: New Mathematical and Computational Advances*, pp. 277–314, Oxford University Press, Oxford, 2007.
- [24] D. H. Huson and T. H. Klopper, “Computing recombination networks from binary characters,” *Bioinformatics*, vol. 21, pp. ii159–ii165, 2005.
- [25] B. M. E. Moret, L. Nakhleh, T. Warnow, C. R. Linder, Tholse, A. Padolina, J. Sun, and R. Timme, “Phylogenetic networks: modeling, reconstructibility, and accuracy,” *The IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 1, pp. 13–23, 2004.
- [26] D. H. Huson, R. Rupp, and C. Scornavacca, *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press, 2011.
- [27] C. Semple and M. Steel, *Phylogenetics*. Oxford Lecture Series in Mathematics and its Applications 24, Oxford University Press, 2003.
- [28] J. Felsenstein, “The newick tree format,” 1986.
<http://evolution.genetics.washington.edu/phylip/newicktree.html>.
- [29] D. Robinson and L. Foulds, “Comparison of phylogenetic trees,” *Math. Biosciences*, vol. 53, pp. 131–147, 1981.
- [30] W. H. E. Day, “Optimal algorithms for comparing trees with labeled leaves,” *Journal Of Classification*, vol. 2, pp. 7–28, 1985.
- [31] D. F. Robinson, “Comparison of labeled trees with valency three,” *Journal of Combinatorial Theory*, vol. 11, pp. 105–119, 1971.

- [32] B. Allen and M. Steel, “Subtree transfer operations and their induced metrics on evolutionary trees,” *Annals of Combinatorics*, vol. 5, pp. 1–13, 2001.
- [33] G. Hickey, F. Dehne, A. Rau-Chaplin, and C. Blouin, “SPR distance computation for unrooted trees,” *Evolutionary Bioinformatics Online*, vol. 4, pp. 17–27, 2008.
- [34] M. Bordewich and C. Semple, “On the computational complexity of the rooted subtree prune and regraft distance,” *Annals of Combinatorics*, vol. 8, pp. 409–423, 2004.
- [35] Y. Wu, “A practical method for exact computation of subtree prune and regraft distance,” *Bioinformatics*, vol. 25, no. 2, pp. 190–196, 2009.
- [36] M. Hallett and J. Lagergren, “Efficient algorithms for lateral gene transfer problems,” in *Proc. 5th Ann. Int’l Conf. Comput. Mol. Biol. (RECOMB01)*, (New York), pp. 149–156, ACM Press, 2001.
- [37] R. Beiko and N. Hamilton, “Phylogenetic identification of lateral genetic transfer events,” *BMC Evolutionary Biology*, vol. 6, pp. 15+, 2006.
- [38] D. MacLeod, R. Charlebois, F. Doolittle, and E. Baptiste, “Deduction of probable events of lateral gene transfer through comparison of phylogenetic trees by recursive consolidation and rearrangement,” *BMC Evolutionary Biology*, vol. 5, 2005.
- [39] L. Nakhleh, D. Ruths, and L. Wang, “RIATA-HGT: A fast and accurate heuristic for reconstructing horizontal gene transfer,” in *Proceedings of the Eleventh International Computing and Combinatorics Conference (COCOON 05)* (L. Wang, ed.), pp. 84–93, 2005. LNCS #3595.

- [40] C. Than and L. Nakhleh, “SPR-based tree reconciliation: Non-binary trees and multiple solutions,” in *Proceedings of the Sixth Asia Pacific Bioinformatics Conference (APBC)*, pp. 251–260, 2008.
- [41] Y. Song and J. Hein, “Parsimonious reconstruction of sequence evolution and haplotyde blocks: finding the minimum number of recombination events,” in *Proceedings of the 2008 Workshop on Algorithms in Bioinformatics (WABI2008)* (G. Benson and R. Page, eds.), vol. 2812 of *Lecture Notes in Bioinformatics*, pp. 287–302, 2003.
- [42] Y. Song and J. Hein, “Constructing minimal ancestral recombination graphs,” *J. Comput. Biol.*, vol. 12, pp. 147–169, 2005.
- [43] M. Baroni, S. Grunewald, V. Moulton, and C. Semple, “Bounding the number of hybridisation events for a consistent evolutionary history,” *J. Math. Biol.*, vol. 51, pp. 171–182, 2005.
- [44] L. Nakhleh, “Evolutionary phylogenetic networks: models and issues,” in *The Problem Solving Handbook for Computational Biology and Bioinformatics* (L. Heath and N. Ramakrishnan, eds.), pp. 125–158, New York: Springer, 2010.
- [45] G. Jin, L. Nakhleh, S. Snir, and T. Tuller, “Maximum likelihood of phylogenetic networks,” *Bioinformatics*, vol. 22, no. 21, pp. 2604–2611, 2006.
- [46] 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.*, vol. 75, no. 1, pp. 35–45, 2009.

- [47] L. S. Kubatko, “Identifying hybridization events in the presence of coalescence via model selection,” *Systematic Biology*, vol. 58, pp. 478–488, 2009.
- [48] Y. Yu, C. Than, J. Degnan, and L. Nakhleh, “Coalescent histories on phylogenetic networks and detection of hybridization despite incomplete lineage sorting,” *Systematic Biology*, vol. 60, no. 2, pp. 138–149, 2011.
- [49] F. de la Cruz and J. Davies, “Horizontal gene transfer and the origin of species: lessons from bacteria,” *Trends Microbiol.*, vol. 8, pp. 128–133, 2000.
- [50] P. Planet, “Reexamining microbial evolution through the lens of horizontal transfer,” in *Molecular Systematics and Evolution: Theory and Practice* (R. DeSalle, G. Giribet, and W. Wheeler, eds.), pp. 247–270, Birkhauser Verlag, 2002.
- [51] J. Mallet, “Hybridization as an invasion of the genome,” *TREE*, vol. 20, no. 5, pp. 229–237, 2005.
- [52] M. Arnold, “Natural hybridization as an evolutionary process,” *Ann. Rev. Ecol. Syst.*, vol. 23, pp. 237–261, 1992.
- [53] N. Sueoka, “Directional mutation pressure, mutator mutations, and dynamics of molecular evolution,” *Journal of Molecular Evolution*, pp. 137–153, 1993.
- [54] P. Deschavanne, A. Giron, J. Vilain, G. Fagot, and B. Fertil, “Genomic signature: characterization and classification of species assessed by chaos game representation of sequences,” *Molecular Biology and Evolution*, vol. 16, pp. 1391–1399, 1999.
- [55] S. Karlin and C. Burge, “Dinucleotide relative abundance extremes: a genomic signature,” *Trends In Genetics*, vol. 11, pp. 283–290, 1995.

- [56] S. Karlin, “Global dinucleotide signatures and analysis of genomic heterogeneity,” *Curr. Opin. Microbiol.*, vol. 1, p. 598610, 1998.
- [57] R. Rolfe and M. Meselson, “The relative homogeneity of microbial dna,” *Proceedings of the National Academy of Sciences*, pp. 1039–1043, 1959.
- [58] M. Ragan, “On surrogate methods for detecting lateral gene transfer,” *FEMS Microbiology letters*, vol. 201, pp. 187–191, 2001.
- [59] C. Dufraigne, B. Fertil, S. Lespinats, A. Giron, and P. Deschavanne, “Detection and characterization of horizontal transfers in prokaryotes using genomic signature,” *Nucleic Acids Res*, vol. 33, p. e6, 2005.
- [60] L. Nakhleh, G. Jin, F. Zhao, and J. Mellor-Crummey, “Reconstructing phylogenetic networks using maximum parsimony,” in *Proceedings of the 2005 IEEE Computational Systems Bioinformatics Conference (CSB2005)*, pp. 93–102, 2005.
- [61] G. Jin, L. Nakhleh, S. Snir, and T. Tuller, “Inferring phylogenetic networks by the maximum parsimony criterion: a case study,” *Molecular Biology and Evolution*, vol. 24, no. 1, pp. 324–337, 2007.
- [62] G. Jin, L. Nakhleh, S. Snir, and T. Tuller, “Parsimony score of phylogenetic networks: Hardness results and a linear-time heuristic,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 6, no. 3, pp. 495–505, 2009.
- [63] G. Jin, L. Nakhleh, S. Snir, and T. Tuller, “A new linear-time heuristic algorithm for computing the parsimony score of phylogenetic networks: Theoretical

- bounds and empirical performance,” in *Proceedings of the International Symposium on Bioinformatics Research and Applications* (I. Mandoiu and A. Zelikovsky, eds.), vol. 4463 of *Lecture Notes in Bioinformatics*, pp. 61–72, 2007.
- [64] J. Zhang, “Evolution by gene duplication: an update,” *Trends in Ecology and Evolution*, vol. 18, pp. 292–298, 2003.
- [65] M. Lynch and A. Force, “The probability of duplicate gene preservation by subfunctionalization,” *Genetics*, vol. 154, pp. 459–473, Jan 2000.
- [66] B. Vogelstein and K. W. Kinzler, *The genetic basis of human cancer*. McGraw-Hill, 2002.
- [67] P. Górecki and J. Tiuryn, “Dls-trees: a model of evolutionary scenarios,” *Theoretical Computer Science*, vol. 359, pp. 378–399, 2006.
- [68] L. Zhang, “On a mirkin-muchnik-smith conjecture for comparing molecular phylogenies,” *Journal of Computational Biology*, vol. 4, no. 2, pp. 177–187, 1997.
- [69] B. Ma, M. Li, and L. Zhang, “From gene trees to species trees,” *SIAM Journal on Computing*, vol. 30, no. 3, pp. 729–752, 2000.
- [70] O. Akerborg, B. Sennblad, L. Arvestad, and J. Lagergren, “Simultaneous bayesian gene tree reconstruction and reconciliation analysis,” *Proc Natl Acad Sci U S A*, vol. 106, no. 14, pp. 5714–5719, 2009.
- [71] S. H. J.P. Doyon JP, C. Chauve, “Space of gene/species tree reconciliations and parsimonious models,” *J Comput Biol*, vol. 16, pp. 1399–1418, 2009.

- [72] J. Degnan and L. Salter, “Gene tree distributions under the coalescent process,” *Evolution*, vol. 59, pp. 24–37, 2005.
- [73] L. Liu and D. Pearl, “Species trees from gene trees: reconstructing bayesian posterior distributions of a species phylogeny using estimated gene tree distributions,” *Syst. Biol.*, vol. 56, no. 3, pp. 504–14, 2007.
- [74] L. Arvestad, A. Berglund, J. Lagergren, and B. Sennblad, “Bayesian gene/species tree reconciliation and orthology analysis using mcmc,” *BIOINFORMATICS*, vol. 19, pp. i7–i15, 2003.
- [75] L. Arvestad, A. Berglund, J. Lagergren, and B. Sennblad, “Gene tree reconstruction and orthology analysis based on an integrated model for duplications and sequence evolution,” in *In Eighth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, pp. 326–335, Springer, 2004.
- [76] L. Arvestad, J. Lagergren, and B. Sennblad, “The gene evolution model and computing its associated probabilities,” *J. ACM*, vol. 56, no. 2, 2009.
- [77] J. Doyon, S. Hamel, and C. Chauve, “An efficient method for exploring the space of gene tree/species tree reconciliations in a probabilistic framework,” *LIRMM technical report*, pp. RR–10002, 2010.
- [78] P. Górecki, G. Burleigh, and O. Eulenstein, “Maximum likelihood models and algorithms for gene tree evolution with duplications and losses,” *BMC Bioinformatics*, vol. 12, p. S15, 2011.
- [79] W. Fitch, “Toward defining the course of evolution: Minimum change for a specified tree topology,” *Syst. Zool.*, vol. 20, pp. 406–416, 1971.

- [80] W. Day, “Computationally difficult parsimony problems in phylogenetic systematics,” *Journal of Theoretical Biology*, vol. 103, pp. 429–438, 1983.
- [81] L. Foulds and R. Graham, “The Steiner problem in phylogeny is NP-complete,” *Adv. Appl. Math.*, vol. 3, pp. 43–49, 1982.
- [82] D. Swofford, “PAUP*: Phylogenetic analysis using parsimony (and other methods),” 1996. Sinauer Associates, Underland, Massachusetts, Version 4.0.
- [83] J. Hein, “Reconstructing evolution of sequences subject to recombination using parsimony,” *Math. Biosci.*, vol. 98, pp. 185–200, 1990.
- [84] J. Hein, “A heuristic method to reconstruct the history of sequences subject to recombination,” *J. Mol. Evol.*, vol. 98, no. 2, pp. 396–405, 1993.
- [85] G. Jin, L. Nakhleh, S. Snir, and T. Tuller, “Efficient parsimony-based methods for phylogenetic network reconstruction,” *Bioinformatics*, vol. 23, pp. e123–e128, 2006. Proceedings of the European Conference on Computational Biology (ECCB 06).
- [86] H. Park, G. Jin, and L. Nakhleh, “Bootstrap-based support of hgt inferred by maximum parsimony,” *BMC Evolutionary Biology*, vol. 10, p. 131, 2010.
- [87] B. Moret, L. Nakhleh, T. Warnow, C. Linder, A. Tholse, A. Padolina, J. Sun, and R. Timme, “Phylogenetic networks: Modeling, reconstructibility, and accuracy,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 13–23, 2004.
- [88] U. Bergthorsson, A. Richardson, G. Young, L. Goertzen, and J. Palmer, “Massive horizontal transfer of mitochondrial genes from diverse land plant donors to

- basal angiosperm Amborella,” *Proc. Nat’l Acad. Sci., USA*, vol. 101, pp. 17747–17752, 2004.
- [89] D. Hillis and J. Bull, “An empirical test of bootstrapping as a method for assessing confidence in phylogenetic,” *Systematic Biology*, vol. 42, pp. 182–192, 1993.
- [90] P. Soltis and D. Soltis, “Applying the bootstrap in phylogeny reconstruction,” *Statistical Science*, vol. 18, pp. 256–267, 2003.
- [91] A. Rambaut, “Phylogen: Phylogenetic tree simulator package,” 2002. Available from <http://evolve.zoo.ox.ac.uk/software/PhyloGen/main.html>.
- [92] N. Galtier, “A model of horizontal gene transfer and the bacterial phylogeny problem,” *Systematic Biology*, vol. 56, no. 4, pp. 633–642, 2007.
- [93] H. Shimodaira and M. Hasegawa, “Multiple comparisons of log-likelihoods with applications to phylogenetic inference,” *Molecular Biology and Evolution*, vol. 16, pp. 1114–1116, 1999.
- [94] H. Park and L. Nakhleh, “Inference of reticulate evolutionary histories by maximum likelihood,” 2012. under review.
- [95] A. Rokas, B. L. Williams, N. King, and S. B. Carroll, “Genome-scale approaches to resolving incongruence in molecular phylogenies,” *Nature*, vol. 425, pp. 798–804, 2003.
- [96] 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*, vol. 92, pp. 2086–2100, 2005.

- [97] 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.*, vol. 2, pp. 1634–1647, 2006.
- [98] C. Than, R. Sugino, H. Innan, and L. Nakhleh, “Efficient inference of bacterial strain trees from genome-scale multi-locus data,” *Bioinformatics*, vol. 24, pp. i123–i131, 2008.
- [99] C. Kuo, J. P. Wares, and J. C. Kissinger, “The Apicomplexan whole-genome phylogeny: An analysis of incongruence among gene trees,” *Mol. Biol. Evol.*, vol. 25, no. 12, pp. 2689–2698, 2008.
- [100] J. H. Degnan and N. A. Rosenberg, “Gene tree discordance, phylogenetic inference and the multispecies coalescent,” *Trends Ecol. Evol.*, vol. 24, pp. 332–340, 2009.
- [101] W. Doolittle, “Lateral genomics,” *Trends in Biochemical Sciences*, vol. 24, no. 12, pp. M5–M8, 1999.
- [102] W. Doolittle, “Phylogenetic classification and the universal tree,” *Science*, vol. 284, pp. 2124–2129, 1999.
- [103] H. Ochman, J. Lawrence, and E. Groisman, “Lateral gene transfer and the nature of bacterial innovation,” *Nature*, vol. 405, no. 6784, pp. 299–304, 2000.
- [104] W. Hao and G. Golding, “Patterns of bacterial gene movement,” *Mol. Biol. Evol.*, vol. 21, no. 7, pp. 1294–1307, 2004.
- [105] M. McCllland, K. Sanderson, S. Clifton, and P. Latreille, “Comparison of genome degradation in Paratyphi A and Typhi, human-restricted serovars of

- Salmonella enterica that cause typhoid,” *Nature Genetics*, vol. 36, no. 12, pp. 1268–1274, 2004.
- [106] Y. Nakamura, T. Itoh, H. Matsuda, and T. Gojobori, “Biased biological functions of horizontally transferred genes in prokaryotic genomes,” *Nature Genetics*, vol. 36, no. 7, pp. 760–766, 2004.
- [107] D. Posada and K. Crandall, “The effect of recombination on the accuracy of phylogeny estimation,” *J. Mol. Evol.*, vol. 54, no. 3, pp. 396–402, 2002.
- [108] D. Posada, K. Crandall, and E. Holmes, “Recombination in evolutionary genomics,” *Annu. Rev. Genet.*, vol. 36, pp. 75–97, 2002.
- [109] N. Ellstrand, R. Whitkus, and L. Rieseberg, “Distribution of spontaneous plant hybrids,” *Proc. Nat’l Acad. Sci., USA*, vol. 93, no. 10, pp. 5090–5093, 1996.
- [110] L. Rieseberg and S. Carney, “Plant hybridization,” *New Phytologist*, vol. 140, no. 4, pp. 599–624, 1998.
- [111] C. Linder and L. Rieseberg, “Reconstructing patterns of reticulate evolution in plants,” *American Journal of Botany*, vol. 91, pp. 1700–1708, 2004.
- [112] J. Mallet, “Hybridization as an invasion of the genome,” *TREE*, vol. 20, no. 5, pp. 229–237, 2005.
- [113] M. Noor and J. Feder, “Speciation genetics: Evolving approaches,” *Nature Review Genetics*, vol. 7, pp. 851–861, 2006.
- [114] L. Rieseberg, S. Baird, and K. Gardner, “Hybridization, introgression, and linkage evolution,” *Plant Molecular Biology*, vol. 42, no. 1, pp. 205–224, 2000.

- [115] M. Arnold, *Natural Hybridization and Evolution*. Oxford: Oxford University Press, 1997.
- [116] J. Mallet, "Hybrid speciation," *Nature*, vol. 446, pp. 279–283, 2007.
- [117] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, pp. 716–723, 1974.
- [118] G. E. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461–464, 1978.
- [119] D. Posada and T. R. Buckley, "Model selection and model averaging in phylogenetics: advantages of akaike information criterion and bayesian approaches over likelihood ratio tests.," *Systematic Biology*, vol. 53, pp. 793–808, 2004.
- [120] A. Luo, H. Qiao, Y. Zhang, W. Shi, S. Ho, W. Xu, A. Zhang, and C. Zhu, "Performance of criteria for selecting evolutionary models in phylogenetics: a comprehensive study based on simulated datasets," *BMC Evolutionary Biology*, vol. 10, p. 242, 2010.
- [121] Y. Yu, J. Degnan, and L. Nakhleh, "The probability of a gene tree topology within a phylogenetic network with applications to hybridization detection," *PLoS Genetics*, 2012. In press.
- [122] J. Felsenstein, "Phylip - phylogeny inference package," *Cladistics*, vol. 5, pp. 164–166, 1989.
- [123] A. Rambaut and N. C. Grassly, "Seq-gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees," *Comp. Appl. Biosci.*, vol. 13, pp. 235–238, 1997.

- [124] C. Andam and J. Gogarten, “Biased gene transfer in microbial evolution,” *Nature Reviews Microbiology*, vol. 9, pp. 543–555, 2011.
- [125] J. Guohua, L. Nakhleh, S. Snir, and T. Tuller, “Inferring phylogenetic networks by the maximum parsimony criterion: A case study,” *Molecular Biology and Evolution*, vol. 24, no. 1, pp. 324–337, 2007.
- [126] E. W. Bloomquist and M. A. Suchard, “Unifying vertical and nonvertical evolution: A stochastic ARG-based framework,” *Syst. Biol.*, vol. 59, no. 1, pp. 27–41, 2010.
- [127] G. Olsen, H. Matsuda, R. Hagstrom, and R. Overbeek, “FastDNAm1: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood,” *Computations in Applied Biosciences*, vol. 10, no. 1, pp. 41–48, 1994.
- [128] P. Lewis, “A genetic algorithm for maximum likelihood phylogeny inference using nucleotide sequence data,” *Molecular Biology and Evolution*, vol. 15, pp. 277–283, 98.
- [129] F. Ronquist and J. P. Huelsenbeck, “Mrbayes 3: Bayesian phylogenetic inference under mixed models,” *Bioinformatics*, vol. 19, pp. 1572–1574, 2003.
- [130] D. Huson and R. Rupp, “Summarizing multiple gene trees using cluster networks,” in *Proceedings of the 2008 Workshop on Algorithms in Bioinformatics (WABI2008)* (K. Crandall and J. Lagergren, eds.), vol. 5251 of *Lecture Notes in Bioinformatics*, pp. 296–305, 2008.
- [131] R. Beiko and M. Ragan, “Untangling hybrid phylogenetic signals: Horizontal gene transfer and artifacts of phylogenetic reconstruction,” *Methods Mol Biol.*, vol. 532, pp. 241–256, 2009.

- [132] H. Park and L. Nakhleh, “MURPAR: A fast heuristic for inferring parsimonious phylogenetic networks from multiple gene trees,” in *International Symposium on Bioinformatics Research and Applications (ISBRA 12)*, 2012. To appear.
- [133] S. Kullback, “The Kullback-Leibler distance,” *The American Statistician*, vol. 41, pp. 340–341, 1987.
- [134] M. Suchard, “Stochastic models for horizontal gene transfer: taking a random walk through tree space,” *Genetics*, vol. 170, pp. 419–431, 2005.
- [135] C. Than, D. Ruths, H. Innan, and L. Nakhleh, “Confounding factors in HGT detection: Statistical error, coalescent effects, and multiple solutions,” *Journal of Computational Biology*, vol. 14, no. 4, pp. 517–535, 2007.
- [136] C. Than, D. Ruths, and L. Nakhleh, “PhyloNet: A software package for analyzing and reconstructing reticulate evolutionary relationships,” *BMC Bioinformatics*, vol. 9, p. 322, 2008.
- [137] L. Addario-Berry, M. Hallett, and J. Lagergren, “Towards identifying lateral gene transfer events,” *Proc. Eighth Pacific Symp. Biocomputing (PSB 03)*, pp. 279–290, 2003.
- [138] A. Tofigh, M. Hallett, and J. Lagergren, “Simultaneous identification of duplications and lateral gene transfers,” *IEEE/ACM transactions on computational biology and bioinformatics*, pp. 1–19, Jan 2011.
- [139] H. Schmidt and W. Martin, *Phylogenetic Trees from Large Datasets Inaugural-Dissertation zur*. PhD thesis, Heinrich-Heine-Universitt, Dsseldorf, 2003.

- [140] T. Hill, K. Nordstrom, M. Thollesson, T. Safstrom, A. Vernersson, R. Fredriksson, and H. Schioth, "Sprit: Identifying horizontal gene transfer in rooted phylogenetic trees," *BMC Evolutionary Biology*, vol. 10, pp. 42+, February 2010.
- [141] R. Beiko, T. Harlow, and M. Ragan, "Highways of gene sharing in prokaryotes," *Proc Natl Acad Sci*, vol. 102, no. 40, pp. 14332–7, 2005.
- [142] T. Dagan and W. Martin, "The tree of one percent," *Genome Biology*, vol. 7, no. 10, p. 118, 2006.
- [143] W. Matthew, T. D. Bie, J. Stajich, C. Nguyen, and N. Cristianini, "Estimating the tempo and mode of gene family evolution from comparative genomic data," *Genome Res.*, vol. 15, pp. 1153–1160, 2005.
- [144] M. Csűrös and I. Miklós, "A probabilistic model for gene content evolution with duplication, loss, and horizontal transfer," in *In Tenth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, pp. 206–220, Springer, 2006.
- [145] A. Tofigh and J. Lagergren, "Inferring duplications and lateral gene transfers an algorithm for parametric tree reconciliation," 2010. manuscript.
- [146] S. Powell, D. Szklarczyk, K. Trachana, A. Roth, M. Kuhn, J. Muller, R. Arnold, T. Rattei, I. Letunic, T. Doerks, L. Jensen, C. von Mering, and P. Bork, "eggno3.0: orthologous groups covering 1133 organisms at 41 different taxonomic ranges," *Nucleic Acids Res*, 2011.
- [147] J. Muller, C. J. Creevey, J. D. Thompson, D. Arendt, and P. Bork, "Aqua: automated quality improvement for multiple sequence alignments," *Bioinformatics*, vol. 26, no. 2, pp. 263–265, 2010.

- [148] J. Muller, D. Szklarczyk, P. Julien, I. Letunic, A. Roth, M. Kuhn, S. Powell, C. von Mering, T. Doerks, L. J. Jensen, and P. Bork, “eggnoG v2.0: extending the evolutionary genealogy of genes with enhanced non-supervised orthologous groups, species and functional annotations,” *Nucleic Acids Research*, vol. 38, pp. D190–D195, 2010.
- [149] J. Coombs and T. Barkay, “Molecular evidence for the evolution of metal homeostasis genes by lateral gene transfer in bacteria from the deep terrestrial subsurface,” *Appl. Environ. Microbiol.*, vol. 70, pp. 1698–1707, 2004.
- [150] W. Doolittle and E. Bapteste, “Pattern pluralism and the tree of life hypothesis,” *Proc Natl Acad Sci*, vol. 13, pp. 2043–9, 2007.
- [151] E. Koonin and Y. Wolf, “Genomics of bacteria and archaea: the emerging dynamic view of the prokaryotic world,” *Nucleic Acids Research*, vol. 36, no. 21, pp. 6688–6719, 2008.
- [152] V. Kunin, L. Goldovsky, N. Darzentas, and C. Ouzounis, “The net of life: reconstructing the microbial phylogenetic network,” *Genome Res.*, vol. 15, no. 7, pp. 954–9, 2005.
- [153] B. Mirkin, T. Fenner, M. Galperin, and E. Koonin, “Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes,” *BMC Evol Biol*, vol. 6, 2003.
- [154] B. Snel, P. Bork, and M. Huynen, “Genomes in flux: the evolution of archaeal and proteobacterial gene content,” *Genome Res.*, vol. 12, no. 1, pp. 17–25, 2002.

- [155] D. Merkle and M. Middendorf, “Reconstruction of the cophylogenetic history of related phylogenetic trees with divergence timing information,” *Theory Biosci*, vol. 123, no. 4, pp. 277–299, 2005.
- [156] D. Merkle, M. Middendorf, and N. Wieseke, “A parameter-adaptive dynamic programming approach for inferring cophylogenies,” *BMC Bioinformatics*, vol. 11, p. S60, 2010.
- [157] R. Libeskind-Hadas and M. Charleston, “On the computational complexity of the reticulate cophylogeny reconstruction problem,” *J. Comput. Biol.*, vol. 16, no. 1, pp. 105–17, 2009.
- [158] A. Tofigh, *Using trees to capture reticulate evolution, lateral gene transfers and cancer progression*. PhD thesis, Sweden: KTH Royal Institute of Technology, 2009.
- [159] L. David and E. Alm, “Rapid evolutionary innovation during an archaean genetic expansion,” *Nature*, vol. 469, no. 7328, p. 936, 2011.
- [160] J. Doyon, C. Scornavacca, and G. Szöllacúteosi, “An efficient algorithm for gene/species trees parsimonious reconciliation with losses, duplications and transfers,” *Proc 14th Int Conf Res Comput Mol Biol (RECOMB-CG)*, vol. 6398, p. 93108, 2011.
- [161] J. P. Doyon, V. Ranwez, V. Daubin, and V. Berry, “Models, algorithms and programs for phylogeny reconciliation,” *BRIEFINGS IN BIOINFORMATICS*, vol. 12, no. 5, pp. 392–400, 2012.
- [162] K. Williams, J. Gillespie, B. Sobral, E. Nordberg, E. Snyder, J. Shallom, and

A. Dickerman, "Phylogeny of gammaproteobacteria," *J. Bacteriol.*, vol. 192, no. 9, pp. 2305–14, 2010.