Reticulation in Evolution

Inaugural-Dissertation

zur

Erlangung des Doktorgrades der

Mathematisch-Naturwissenschaftlichen Fakultät

der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Simone Linz

aus Rheinberg

März 2008

Aus dem Institut für Bioinformatik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der

Mathematisch-Naturwissenschaftlichen Fakultät der

Heinrich-Heine-Universität Düsseldorf

Referent : Prof. Dr. Arndt von Haeseler
Korreferenten : Prof. Dr. Martin Lercher und Assoc. Prof. Dr. Charles Semple

Tag der mündlichen Prüfung: 30. April 2008

Abstract

Molecular phylogenetics, the study of reconstructing evolutionary trees, is a well-established field of scientific endeavor. However, in certain circumstances evolution is not completely tree-like. For example, a comparison of gene trees representing a set of present-day species and reconstructed for different genetic loci often reveals conflicting tree topologies. These discrepancies are not always due to missampling or difficulties in the gene tree reconstruction method, but rather due to reticulation events such as horizontal gene transfer (HGT) and hybridization. During an HGT event, a DNA segment is transferred from one organism to another which is not its offspring, whereas hybridization describes the origin of a new species through a mating between two different species. Both processes yield genomes that are mixtures of DNA regions derived from different ancestors. Consequently, evolutionary relationships between species whose past includes reticulation can often be better represented by using phylogenetic networks rather than trees.

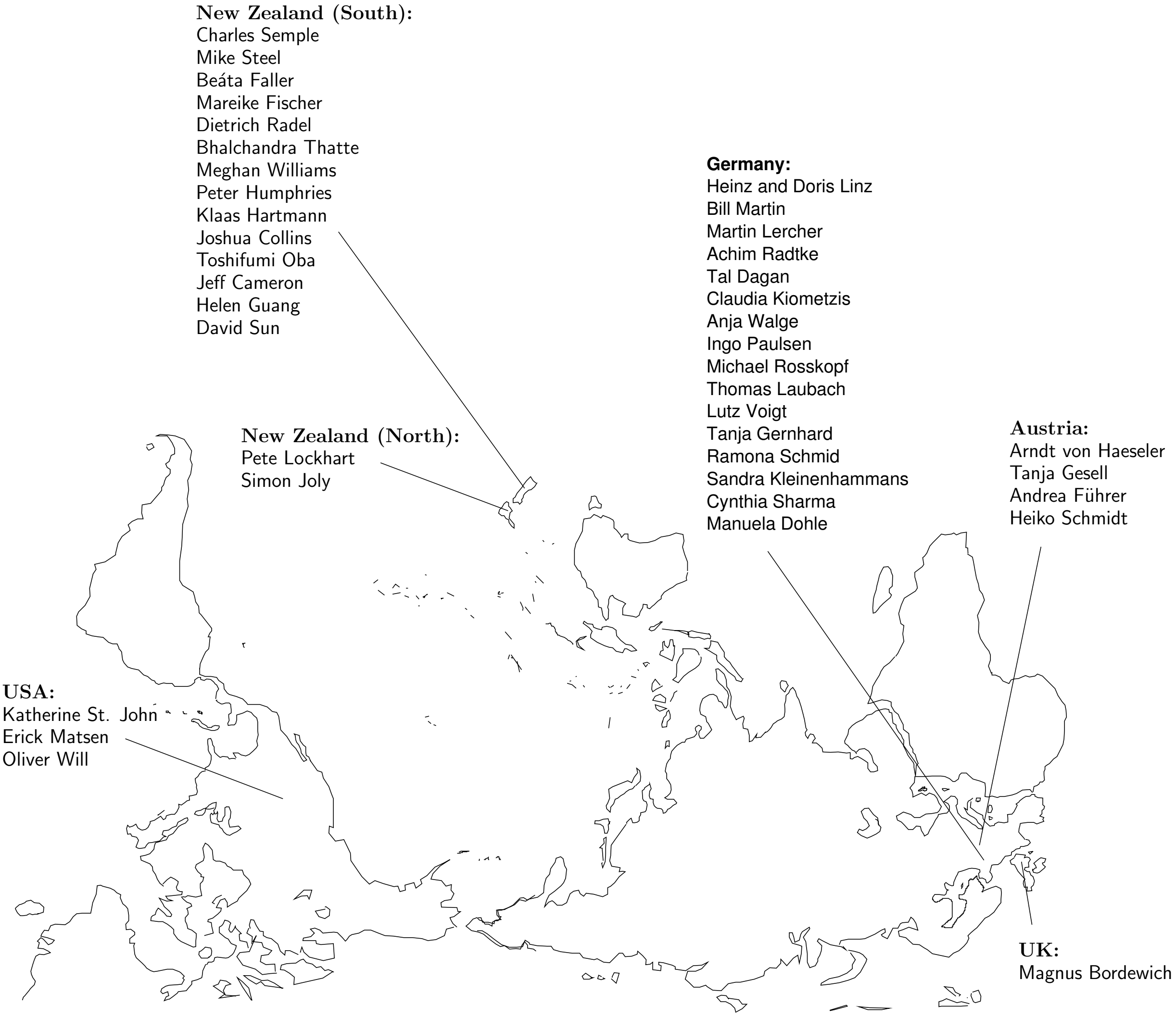
The main focus of this thesis is to develop new biologically motivated theoretical frameworks that provide insight into the extent to which reticulation events have influenced evolution. First, we have implemented the exact algorithm HybridNumber to compute the minimum number of hybridization events for two rooted binary phylogenetic trees. This approach is based on the notion of agreement forests and uses three rules that reduce the size of the problem instance, before calculating the hybridization number. We applied HybridNumber to a grass data set and analyzed the extent of hybridization. We also approached the question whether hybridization events have occurred relatively recently or in the distant past. Furthermore, since many biological data sets lead to reconstructed gene trees that are not fully resolved, we extended the above mentioned framework for rooted phylogenetic trees and showed that calculating the minimum number of hybridization events for two such trees is fixed-parameter tractable.

Second, we present a new likelihood framework to estimate a rate of HGT for a set of taxa. To this end, we simulate an increasing number of HGT events on a species tree to obtain a tree distribution that can be used to estimate an HGT rate for a set of gene trees. This framework was applied to the COG (Clusters of Orthologous Groups of Proteins) data set and inaccuracies due to the gene tree reconstruction method were considered.

Finally, we give a new result on how to speed up the exact calculation of the rooted subtree prune and regraft distance between two trees which is often used to model reticulation events and end with two interesting examples that give rise to questions for future research.

Acknowledgments

I gratefully acknowledge the advice of many people who have supported me in various kinds of ways over the last few years. Danke and thank you to:



For financial support, I thank the Computer Science Department at the University of Düsseldorf, the Department of Mathematics and Statistics at the University of Canterbury, the Allan Wilson Centre for Molecular Ecology and Evolution (AWCMEE), the Deutsche Forschungsgemeinschaft (DFG), and the Isaac Newton Institute for Mathematical Sciences in Cambridge.

Citations to Previously Published Work

Chapter 3 has been published as:

> Magnus Bordewich, Simone Linz, Katherine St. John, Charles Semple (2007). A reduction algorithm for computing the hybridization number of two trees. *Evolutionary Bioinformatics* **3**:86-98.

Chapter 5 has been submitted as:

> Simone Linz and Charles Semple. Hybridization in non-binary trees. *submitted to IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2008.

Chapter 6 has been published as:

> Simone Linz, Achim Radtke, Arndt von Haeseler (2007). A likelihood framework to measure horizontal gene transfer. *Molecular Biology and Evolution* **24**:1312-1319.

The algorithm HybridNumber (Chapter 3) and the software package to simulate and estimate horizontal gene transfer (Chapter 6) are freely available for application at:

`http://www.cs.uni-duesseldorf.de/NewMA/Personen/entry_43.`

Contents

1 Introduction

1.1 Phylogenetic Trees and Networks

Since Charles Darwin's first sketch of a phylogenetic tree in 1837, one of the main goals of evolutionary biologists is to reconstruct phylogenetic (evolutionary) trees which correctly represent the ancestral history of a set of present-day species. In such trees, each leaf represents an existing species, while the internal vertices correspond to hypothetical (extinct) ancestors, and edges, alternatively called branches, show the relationships between ancestors and their descendants.

While the reconstruction of evolutionary trees was based on morphological characters first, the vast majority of data sets that are nowadays used to infer the history of life consists of biological sequence data like nucleotide and protein sequences. This has been made possible by the progress in the field of molecular biology. Accompanied by the development of efficient DNA sequencing technologies, like the shotgun method (Venter *et al.* 1998), and a detailed computer-based analysis of the results, sequences obtained from many genome sequencing projects are freely available from publicly accessible data bases (e.g. Genbank[1] and EMBL[2]). Due to the exponentially growing amount of data[3] that is stored in such data bases, it is of utmost importance to analyze these data in a fast and efficient—but also accurate—way. In the field of phylogenetics, this means that models have to be developed that aim at analyzing the manifold and complex processes that have occurred during the evolution of the current diversity of species.

Until today, research in phylogenetics is mainly focused on developing methods to reconstruct trees that best represent the evolutionary history for different sets of taxa. Since the fossil record is incomplete, researchers mostly rely upon sequence data of contemporary species to reconstruct phylogenetic trees. Essentially, there exist three types of such methods: (1) distance-based methods like UPGMA (unweighted pair group method with arithmetic mean) (Sokal and Michener, 1958, Sneath and Sokal, 1973) and neighbor-joining (Saitou and Nei, 1987), (2) methods based on the parsimonious principle like maximum parsimony (Fitch, 1971), and (3) statistical-based methods like maximum likelihood (Felsenstein, 1981) and the closely related Bayesian method introduced by Rannala and Yang (1996). We refer the interested reader to Felsenstein (2004), where these and other tree reconstruction methods are described in detail.

[1] http://www.ncbi.nlm.nih.gov/
[2] http://www.ebi.ac.uk/embl/
[3] http://www3.ebi.ac.uk/Services/DBStats/

Under the usual assumption that species are evolving from a common ancestor by a simple branching process, the previously mentioned tree-based approaches work well and a lot of progress has been made in recent years. However, processes like hybridization, horizontal gene transfer (HGT), and recombination—collectively referred to as reticulation events—result in species whose genomes are mixtures of DNA regions derived from different ancestors. Consequently, the analysis of different genetic loci often reveals incompatibilities between gene trees (McBreen and Lockhart, 2006). Inferring phylogenies in the presence of reticulation has turned out to be more complicated, because it has become apparent that the history of life cannot be properly represented by a tree and that phylogenetic networks are more appropriate in those cases.

Phylogenetic networks are a generalization of evolutionary trees that allow for a simultaneous visualization of several conflicting or alternating histories of life. They are necessary if the evolutionary past includes reticulation. Even if the relationships between species are tree-like, phenomena like sampling error, parallel evolution, or model heterogeneity can make it difficult to represent evolution by a single tree (Gascuel, 2005). By considering analyses in which phylogenetic networks play an important role, it becomes obvious that there exist two fundamental types of such networks, namely implicit ones that aim at representing incompatible signals in a data set and explicit networks that provide a concrete scenario of reticulate evolution (Huson, 2007). Approaches that reconstruct networks of the former type are often based on split networks which represent all splits contained in a set of gene trees. Each parallelogram of the resulting network corresponds to two incompatible splits. Details of methods that describe how to obtain such a network are given by Bandelt *et al.* (1995), Bryant and Moulton (2002), Dress and Huson (2004), and others. Explicit networks model non-tree-like evolution and purpose to point out which lineages have undergone reticulation events. Examples of this type of networks are given by Gusfield and Bansal (2005) and Huson *et al.* (2005). An extended list of approaches to reconstruct phylogenetic networks can be found in Huson (2007), where detailed background information of some methods is also provided.

1.2 Processes of Reticulate Evolution

The upcoming two sections shed light on the biological processes of hybridization and HGT and point out why the extent to which reticulation events have influenced the evolutionary history of certain groups of species is still critically discussed.

1.2.1 Hybridization

Analyses that focus on the extent to which hybridization has influenced the evolutionary past of groups of present-day species have been an active and controversially discussed field of research for many years, and even several definitions of the term hybridization have been suggested (Harrison, 1993). For the purpose of this thesis, we refer to the origin of a new species through a mating between individuals of two different species as a hybridization event. This definition is commonly used by evolutionary biologists, whereas plant and animal breeders often describe hybridization as a crossing between genetically distinct individuals (Arnold, 1997). During a hybridization event, the genomes of two distinct species recombine such that the new species has either the same number of chromosomes as its parents (diploid hybridization) or the sum of all parent chromosomes (polyploid hybridization). In the latter case, the hybrid species is said to be allopolyploid. Hybrid species are sometimes adapted to habitats which are different from those of their parents (Rieseberg *et al.*, 2003). Additionally, hybridization can be seen as a source of genetic variation and functional novelty and, therefore, many researchers opine that hybridization plays an important role in evolution because of its contribution to an increased biological diversity (Seehausen, 2004, and references therein). On the other hand, hybrid events can lead to less viable or infertile offspring such that other scientists argue against a fundamental role of hybridization in evolution (e.g. Mayr, 1992).

Eukaryotes whose evolutionary history contains hybridization events include certain groups of plants (e.g. Ellstrand *et al.*, 1996, Arnold, 1997), birds (e.g. Grant and Grant, 1992), and fish (e.g. Hubbs, 1955). Besides these groups of organisms for which hybridization events are widely accepted, a number of publications exist that also report on spontaneous hybridization events in the evolutionary history of mammals (e.g. Mallet, 2005, 2007) and even primates (e.g. Arnold and Meyer, 2006, Cortes-Ortiz *et al.*, 2007). A review of hybrid species is given by Mallet (2005). This article also contains a comparison of the numbers of species that hybridize in different groups of organisms. The results indicate that a rounded average of 10 % of all species are involved in hybridization events; of course, some groups of organisms are hotspots of hybridization (e.g. vascular plants and British Duck species).

1.2.2 Horizontal Gene Transfer (HGT)

HGT is defined to be any process different from vertical inheritance in which an organism transfers a DNA segment to another organism that is not its offspring (Bushman, 2002).

HGT is known as an important mechanism to shape the genomes of bacteria (e.g. Ochman *et al.*, 2000, Boucher *et al.*, 2003) through three common mechanisms:

Conjugation, a process in which a bacterial cell transfers DNA into another bacterial cell via a cell-to-cell contact.

Transduction, a process in which a piece of DNA is transferred from a donor bacterial cell into an recipient bacterial cell by a bacteriophage.

Transformation, genetic modification of a bacterium due to an uptake of naked DNA.

For a more detailed description of these three processes, we refer the interested reader to Madigan *et al.* (2005). As a result of HGT, bacterial cells which have acquired new DNA are often better adapted to ecological niches or have an increased drug (mainly antibiotic) resistance (Maiden, 1997).

Recently, there is an accumulation of data indicating that HGT has also occurred in the evolution of eukaryotes (de la Cruz and Davies, 2000, Bergthorsson *et al.*, 2003, Andersson, 2005) and archaea (Nelson *et al.*, 1999, Diruggiero *et al.*, 2000). In 2001, Salzberg *et al.* even reported about 40 genes of the human genome that are exclusively shared by humans and bacteria and, therefore, are candidate examples for HGT. Of course, the three processes of conjugation, transduction, and transformation are most likely less common in eukaryotes than in prokaryotes (Andersson, 2005). Nevertheless, there are other possible pathways how eukaryotes can take up DNA from non-parental organisms, e.g. by phagocytosis, symbiosis, and transfection (Gogarten, 2003).

Although HGT is widely accepted as a driving force in the innovation and evolution of genomes, especially for prokaryotes, its extent and impact on the evolutionary process and the phylogeny of species remain controversial (Choi and Kim, 2007). Similar to hybridization, arguments range from the one extreme that HGT plays an important role in evolution such that phylogenetic trees may be inappropriate to represent the evolutionary history of bacteria (Doolittle, 1999, Garcia *et al.*, 2000, Gogarten *et al.*, 2002), to the other extreme that the impact of HGT in bacterial evolution is greatly overestimated (Kurland *et al.*, 2003).

1.3 Inferring Reticulate Evolution

Within the last few years, the number of publications that discuss newly developed methods to infer reticulate evolution has increased quickly. The suggested approaches use various combinations of ideas originating from the three disciplines biology, computer science, and mathematics. In the following, we give a short overview of such methods.

Focusing on the detection of HGT events, there are mainly four different types of analyses which are summarized below. Since each of these approaches has its own advantages and drawbacks, it is useful to combine several types of analyses to obtain significant results. However, it is important to note that the limitation of combined approaches is that each method is designed to detect transfers of different types and ages (Eisen, 2000).

(i) The nucleotide composition (e.g. guanine-cytosine (GC) content) which is variable among different species but relatively constant for a particular species' genome can be used to detect alien DNA by comparing the GC content of neighboring DNA regions. Lawrence and Ochman (1997) applied this approach to the *Escherichia coli* chromosome and analyzed which parts of the genome are candidates for horizontally transferred genes. They also pointed out that alien DNA inserted into an acceptor genome reflects the base composition of the donor genome at the time of introgression, and that the newly acquired DNA will ameliorate to reflect the DNA composition of the acceptor over time. Hence, this approach is particularly useful to detect recently transferred DNA.

(ii) A comparison of phylogenetic trees reconstructed for different genetic markers can indicate conflicting relationships among taxa, which might be the result of reticulation events like HGT or hybridization. Overall, phylogenetic analyses are robust indicators of reticulation, but it is important to consider that alignment and tree reconstruction methods can lead to incompatible gene trees by themselves.

(iii) Homology-based approaches are used to determine genes that are (exclusively) homologous to distantly related species by using a BLAST (Basic Local Alignment Search Tool) search (Altschul *et al.*, 1997). On the one hand, this is a rapid method to detect HGT but, on the other hand, the size of the data base can affect the results and the similarity score of two sequences does not always accurately indicate evolutionary relationships.

(iv) Gene present and absent patterns are often compared to detect candidate genes for HGT (Hao and Golding, 2006). For example, the presence of a gene in a genome that is also found in the genome of distantly related species, but not in closely related species, can indicate the occurrence of HGT. However, this type of analysis does not perform well for highly conserved universal genes, and the alternative hypothesis of gene loss should also be considered.

Recently, a number of algorithms has been developed which heuristically calculate the number of reticulation events for two phylogenetic trees $\mathcal{T}$ and $\mathcal{T}'$. For example, RIATA-HGT (Nakhleh *et al.*, 2005a) is a polynomial-time heuristic that calculates an agreement forest (see Section 2.2) by repeatedly finding a maximum agreement subtree to decompose $\mathcal{T}$ and $\mathcal{T}'$. Another popular approach uses rooted subtree prune and regraft (rSPR) operations (see Section 1.4.4) to model reticulation events. More precisely, given $\mathcal{T}$ and $\mathcal{T}'$, the minimum number of rSPR operations is calculated that transform $\mathcal{T}$ into $\mathcal{T}'$. Beiko and Hamilton (2006) developed the program EEEP (Efficient Evaluation of Edit Paths) that bounds the number of rSPR moves between two phylogenetic trees, subjected to evolutionarily reasonable constraints that reduce the overall computational burden. Hallett and Lagergren (2001) implemented the heuristic LatTrans to model HGT by calculating an rSPR distance with certain direction and time constraints. Again, using the idea of calculating the rSPR distance, MacLeod *et al.* (2005) developed the algorithm HorizStory. This heuristic detects HGT events by first eliminating identical rooted subtrees in $\mathcal{T}$ and $\mathcal{T}'$ before applying rSPR operations to transform the resulting trees. In contrast to other approaches in this field of research, HorizStory can also be applied to multifurcating trees. This is of interest because the exact order of speciation events is often unknown (see Chapter 5) due to insufficient sequence information.

In addition to these heuristics, a small number of approximation algorithms have been developed that calculate the rSPR distance between two phylogenetic trees. For example, Bonet *et al.* (2006) revised the idea of Hein *et al.* (1996) and the authors have shown that building an agreement forest locally by taking into account sibling pairs yield a 5-approximation algorithm, whereas Bordewich *et al.* (2008) approached the problem by considering so-called incompatible rooted triples and overlapping components to show that a careful analysis results in a 3-approximation.

1.4 Preliminary Notation

In this section, we introduce some basic definitions and terminology in the field of phylogenetics that is needed throughout this thesis. Unless stated otherwise, the notation follows Semple and Steel (2003).

1.4.1 Graphs

A *graph* G is an ordered pair (V, E) that consists of a non-empty set V of *vertices* and a multiset E of *edges* such that each edge is an element of $\{\{x, y\} : x, y \in V\}$. A *walk* is a sequence of at least two vertices $v_1, v_2, \ldots, v_k$ such that, for all $i \in \{1, 2, \ldots, k-1\}$, there is an edge $\{v_i, v_{i+1}\}$ in E. Additionally, a *path* is a walk in which all vertices v_i with $i \in \{1, 2, \ldots, k\}$ are distinct. A *cycle* in G is a walk whose first and last vertices are equal, whereas all other edges and vertices are pairwise distinct. Moreover, a graph is said to be *connected* if V is a singleton or if there exists a path from u to v for all $u, v \in V$ with $u \neq v$. Let v be a vertex of G. The *degree* of v, denoted by $d(v)$, is the number of edges in G that are incident with v.

A *directed graph* or *digraph* D is an ordered pair (V, A) that consists of a non-empty set V of vertices and a multiset A of *arcs* such that each arc is an element of $\{(x, y) : x, y \in V\}$. In general, the terminology for digraphs is similar to that for graphs. However, for completeness, we now give some of the basic definitions for digraphs. If $a = (u, v)$ is an element of A, then the arc a is said to be *directed from u to v*. A *directed walk* is a sequence of at least two vertices $v_1, v_2, \ldots, v_k$ such that, for all $i \in \{1, 2, \ldots, k-1\}$, there is an arc (v_i, v_{i+1}) in A. Additionally, a *directed path* is a directed walk in which all vertices v_i, for all $i \in \{1, 2, \ldots, k\}$, are distinct. We say that D contains a *directed cycle* if there exists a directed walk in D whose first and last vertices are equal and all other edges and vertices are pairwise distinct. As a consequence, D is called *acyclic* if there exists no directed cycle in D. A digraph D is *(weakly) connected* if replacing all of its arcs with undirected edges leads to a connected graph. In directed graphs, we often distinguish between the indegree of a vertex v, denoted $d^-(v)$, that is the number of arcs directed into v, and the outdegree of v, denoted $d^+(v)$, that is the number of arcs directed out of v.

1.4.2 Trees

A tree $\mathcal{T} = (V, E)$ is a connected graph with no cycles. Let v be a vertex of V. If $d(v) \leq 1$, then v is called a *leaf* or, otherwise, if $d(v) > 1$, then v is referred to as an *internal vertex*. A *rooted tree* is a tree that has exactly one distinguished vertex called the *root*. The root is an internal vertex unless it is the only vertex in the graph, in which case it is a leaf. Furthermore, a *rooted binary tree* is a rooted tree in which every internal vertex apart from the root has degree three. To represent the evolutionary history of a set of present-day species, phylogenetic trees are frequently used. A *rooted phylogenetic X-tree* is a rooted tree whose root has degree of at least two and whose leaf set is X.

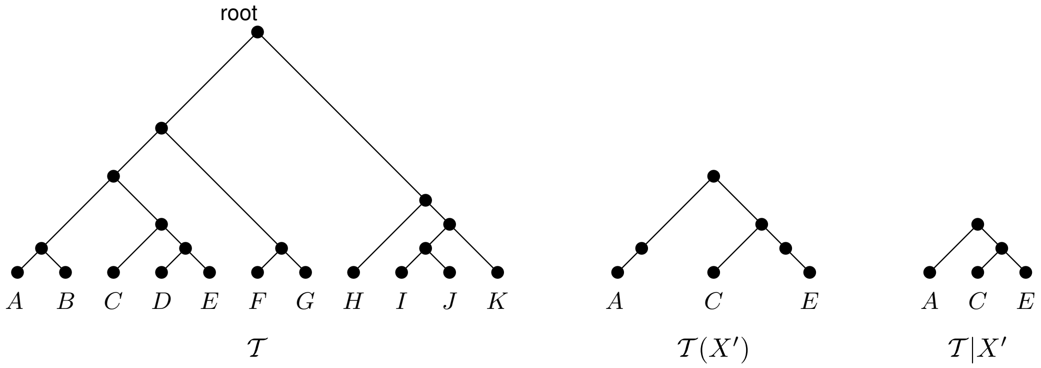


Figure 1.1: A rooted binary phylogenetic X-tree $\mathcal{T}$ with leaf set $X = \{A, B, \ldots, K\}$ and the two subtrees $\mathcal{T}(X')$ and $\mathcal{T}|X'$ with $X' = \{A, C, E\}$.

Lastly, a *rooted binary phylogenetic X-tree* is a rooted phylogenetic X-tree whose root has degree two and all other internal vertices have degree three. An example of such a tree with $X = \{A, B, \ldots, K\}$ is presented in Figure 1.1. The set X is called *label set* of $\mathcal{T}$, and we denote it with $\mathcal{L}(\mathcal{T})$.

Now consider two vertices $u, v \in V$ of a rooted phylogenetic X-tree $\mathcal{T}$ such that u is on the path from the root of $\mathcal{T}$ to v. We say that u is an *ancestor* of v and v is a *descendant* of u. Furthermore, we say that a vertex of a rooted tree is both a descendant and an ancestor of itself. If there exists an edge $e \in E$ such that u and v are incident with e, then u is said to be the *parent* of v and v is a *child* of u. Similarly to the definitions of indegree and outdegree of vertices in digraphs, the number of children of a vertex v is referred to as outdegree of v, denoted by $d^+(v)$. Trivially, the number of parent vertices of v, referred to as indegree of v, denoted by $d^-(v)$, is always one.

Let A be a subset of X. Then A is called a *cluster* of $\mathcal{T}$ if there exists a vertex v that has precisely A as its set of descendant leaves. We denote this cluster by $\mathcal{C}_{\mathcal{T}}(v)$ or $\mathcal{C}(v)$ if there is no ambiguity. The set of clusters of $\mathcal{T}$ is denoted by $\mathcal{C}(\mathcal{T})$. Additionally, the *most recent common ancestor* of A, denoted $\mathrm{mrca}_{\mathcal{T}}(A)$, is the vertex of $\mathcal{T}$ whose associated cluster is the minimal cluster of $\mathcal{T}$ containing A.

For a rooted phylogenetic X-tree $\mathcal{T}$, we next introduce two different types of rooted subtrees which will play an important role in the following chapters. To this end, let X' be a subset of X. Then $\mathcal{T}(X')$ is the *minimal rooted subtree* of $\mathcal{T}$ that connects all leaves referring to taxa in X'. Moreover, the *restriction of $\mathcal{T}$ to X'*, denoted by $\mathcal{T}|X'$, is the rooted phylogenetic tree obtained from $\mathcal{T}(X')$ by suppressing every vertex of degree two apart from the root (see Figure 1.1).

Let $\mathcal{T}$ be a rooted binary phylogenetic X-tree. A rooted subtree of $\mathcal{T}$ is *pendant* if it can be detached from $\mathcal{T}$ by deleting a single edge. For example, in Figure 1.1, the minimal rooted subtree connecting the leaves labeled with A, B, C, D, and E is pendant in $\mathcal{T}$, whereas the minimal rooted subtree connecting the three leaves labeled with E, F, and G is not pendant in $\mathcal{T}$.

1.4.3 Networks

A *hybridization network* $\mathcal{H}$ on X is a rooted acyclic digraph, in which

(i) X is the set of vertices of outdegree 0,

(ii) the root has outdegree at least 2, and

(iii) for all vertices v with outdegree 1, its indegree is at least 2.

Like for rooted phylogenetic X-trees, the set X represents a collection of taxa and is the label set of $\mathcal{H}$. Vertices with an indegree of at least two are called *hybridization vertices* and represent an exchange of genetic material between the hypothetical ancestors. Note that the above given definition also allows for hybridization vertices whose indegree is greater than two and does not require that the outdegree of such a vertex is one. A hybridization vertex v represents ambiguity in the exact order of hybridization events among all parent species of v. The bottom part of Figure 1.2 shows a hybridization network $\mathcal{H}$ with two hybridization vertices. Throughout this thesis, we adopt the convention that hybridization networks are always drawn with their arcs directed downwards (the root is the topmost vertex) and so omit the arrowheads. Note that rooted phylogenetic trees are special types of hybridization networks. As one would expect, the number of hybridization vertices of such a tree is zero.

Let $\mathcal{T}$ be a rooted phylogenetic X'-tree, and let $\mathcal{H}$ be a hybridization network on X with $X' \subseteq X$. We say that $\mathcal{H}$ *displays* $\mathcal{T}$ if all of the ancestral relationships described in $\mathcal{T}$ are covered by $\mathcal{H}$. Mathematically speaking, $\mathcal{H}$ displays $\mathcal{T}$ if $\mathcal{T}$ can be obtained from $\mathcal{H}$ by first deleting a subset of edges and vertices of $\mathcal{H}$ and suppressing any resulting degree zero and degree two vertices apart from the root, and then contracting edges. For a better understanding of this concept, Figure 1.2 shows a hybridization network $\mathcal{H}$ on $X = \{A, B, C, D, E, F\}$ that displays the two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$.

Similarly to the definition of a cluster for a rooted phylogenetic X-tree, we now introduce this term in the context of hybridization networks. To this end, let $\mathcal{H}$ be a

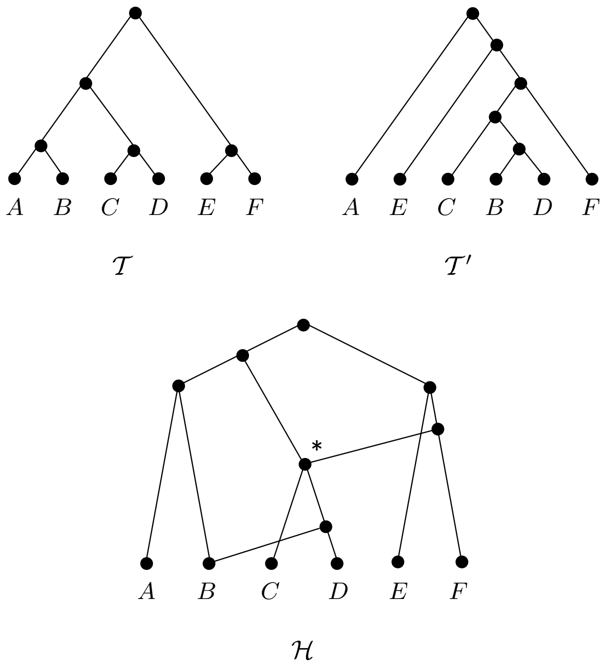


Figure 1.2: Two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ and a hybridization network $\mathcal{H}$ displaying $\mathcal{T}$ and $\mathcal{T}'$. The internal vertex $*$ as well as the leaf labeled B are hybridization vertices.

hybridization network on X with vertex set V. For all $v \in V$, we say that the *cluster* of v, denoted $\mathcal{C}_{\mathcal{H}}(v)$ or simply $\mathcal{C}(v)$ if there is no ambiguity, is the subset of X such that, if $d^+(v) = 0$, then $\mathcal{C}(v)$ is a singleton containing the label of v and, otherwise, $\mathcal{C}(v)$ contains precisely the labels of all vertices u of V for which there exists a directed path from v to u.

1.4.4 The Rooted Subtree Prune and Regraft Operation

We end this section by giving a formal definition of a single rooted subtree prune and regraft (rSPR) operation which, historically, has often been used to model reticulate evolution (e.g. see Maddison, 1997, Baroni *et al.*, 2004, Nakhleh *et al.*, 2005b). The rSPR operation also is an important tree rearrangement method and other such methods include nearest neighbor interchange (NNI) and tree bisection and reconnection (TBR) (Felsenstein, 2004). Let $\mathcal{T}$ be a rooted binary phylogenetic X-tree, and let $e = \{u, v\}$ be an edge of $\mathcal{T}$ such that u lies on the path from the root of $\mathcal{T}$ to v (see Figure 1.3). We say that $\mathcal{T}'$ can be obtained from $\mathcal{T}$ by a single *rSPR operation* if, after deleting e, a new edge f can be joined between v and the subtree $\mathcal{T}_u$ containing u in one of the following two ways (Bordewich and Semple, 2004):

(i) Create a new vertex u' that subdivides an edge in $\mathcal{T}_u$ and join u' and v via f.

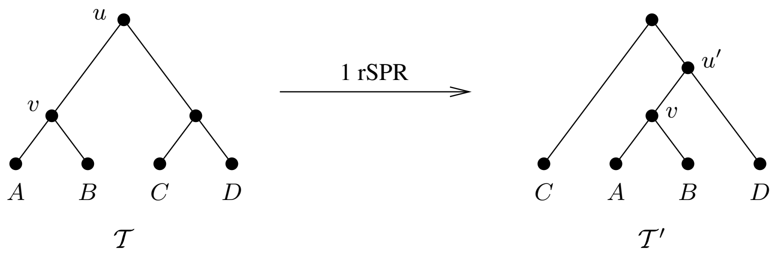


Figure 1.3: The rooted binary phylogenetic X-tree $\mathcal{T}'$ can be obtained from $\mathcal{T}$ by a single rooted subtree prune and regraft operation.

Complete the regrafting step by either suppressing the degree two vertex u or, if u is the root of $\mathcal{T}$, by deleting u and the edge incident with u and turning the other end-vertex of this edge into the root of $\mathcal{T}'$.

(ii) Create a new root vertex u' and a new edge connecting u' and the original root, join u' and v via f, and suppress the degree two vertex u.

An example of a single rSPR operation is shown in Figure 1.3. For any pair of rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$, the *rSPR distance*, denoted $d_{\text{rSPR}}(\mathcal{T}, \mathcal{T}')$, is the smallest number of rSPR operations needed to transform $\mathcal{T}$ into $\mathcal{T}'$. This distance is a metric (see Bordewich and Semple, 2004), and it has often been used to provide lower bounds on the minimum number of reticulation events (e.g. in the context of recombination Song and Hein (2003, 2005)). However, as described in Baroni *et al.* (2005) and Song and Hein (2005), the number of such events might be underestimated by using the rSPR distance (for details, see Section 2.2).

1.5 Organization of this Thesis

As the extent to which hybridization and HGT have influenced the evolutionary history of species remains largely unclear, the following chapters are devoted to gaining new insight into this fast growing field of research by developing biologically motivated mathematical models to infer reticulate evolution.

Chapter 2: This chapter describes the main ideas of a combinatorial framework to calculate the minimum number of hybridization events needed to explain the evolutionary history of two phylogenetic trees. This concept is based on a characterization of the hybridization number in terms of agreement forests which was introduced in Baroni *et al.* (2005), while Baroni *et al.* (2006), and Bordewich and Semple (2007a,b) exploited this characterization to develop three reduction rules that can be applied to reduce the size of the problem instance. All papers only considered two trees. Here, we upgrade this approach to an arbitrarily large number of trees.

Chapter 3: With the theoretical background of Chapter 2 in hand, this part describes the newly implemented exact algorithm HybridNumber that computes the minimum number of hybridization events for two rooted binary phylogenetic trees on the same taxa set. HybridNumber is based on repeated applications of three rules that reduce the size of the problem instance before calculating the hybridization number exactly. We apply this algorithm to a grass data set (Grass Phylogeny Working Group, 2001) and highlight the effectiveness of the reductions.

Chapter 4: In this chapter, we approach the question whether hybridization events have occurred relatively recently or in the distant past by showing that a combination of a modified version of the algorithm HybridNumber (see Chapter 3) with a new algorithm BuildForest is suitable to calculate all agreement forests of smallest size for a pair of (unreduced) trees. With these forests in hand, we can conclude where the hybridization events take place in an associated network to compare the number of hybridization events at the leaves of this network with those events at interior vertices.

Chapter 5: Chapter 2 and 3 approach the question of calculating the minimum number of hybridization events for rooted binary phylogenetic trees by using a combinatorial framework. However, for many biological examples, the reconstructed trees are not fully resolved. In this chapter, we show that calculating the minimum number of hybridization events for two (arbitrary) rooted phylogenetic trees is fixed-parameter tractable by upgrading the notion of agreement forests and stating three reductions that can be used to kernalize the problem instance. Moreover, a further reduction is described that breaks the problem into a number of smaller and more tractable subproblems.

Chapter 6: Here, we focus on HGT which is—beside hybridization—another process of reticulate evolution. Assuming that a species tree is given, a method is suggested which simulates HGT events on a species tree according to a Poisson process. Using the obtained tree distribution and a likelihood estimation approach, an overall rate of HGT for a set of gene trees whose taxa sets are subsets of the species tree taxa is estimated. This framework is applied to the COG (Clusters of Orthologous Groups of Proteins) data set (Tatusov *et al.*, 2001). Additionally, inaccuracies due to gene tree reconstruction methods are analyzed and results for two different species trees are compared.

Chapter 7: The last chapter focuses on the rSPR operation which is frequently used to model reticulate evolution. To calculate exactly the rSPR distance between two rooted binary phylogenetic X-trees, we present a new reduction—similar to the cluster reduction (see Chapter 2 and 3)—that can be used to break the problem into two smaller and more tractable subproblems.

2 Measuring Hybridization for a Set of Phylogenetic Trees

Hybridization and HGT events are often used to explain inconsistencies among a set of phylogenetic trees. Due to such reticulation events, the evolutionary history for certain groups of extant species is better represented by using phylogenetic networks rather than trees since the genomes of such species can be chimeras of the genomes of several distinct species. Recently, a combinatorial-based approach to compute the minimum number of hybridization events for two rooted binary phylogenetic X-trees has been developed (Baroni *et al.*, 2005, 2006, Bordewich and Semple, 2007a,b). This approach and its associated framework lays the initial groundwork for a number of results in this thesis. In the following chapter, we describe this framework and, in particular, show how it can be extended to an arbitrarily large number of trees.

2.1 Hybridization Networks

Although the extent of hybridization in evolution is still discussed controversially for many groups of organisms, its occurrence in plants is widely accepted and subject of many recent evolutionary studies (e.g. Ellstrand *et al.*, 1996, Mallet, 2005, Paun *et al.*, 2005). One question that is often asked by biologists studying the ancestral relationships of species whose past includes hybridization is the following: given a collection of rooted binary phylogenetic trees on a set of present-day species that correctly represent the tree-like evolution of different parts of their genomes, what is the minimum number of hybridization events needed to explain the evolution of the species under consideration? In the following, we formalize this optimization problem and describe a mathematical framework based on combinatorics that can be used to approach this question.

Let $\mathcal{H}$ be a hybridization network on X. To quantify the number of hybridization events, the *hybridization number* of $\mathcal{H}$ is defined as

$$h(\mathcal{H}) = \sum_{v \in (V - \{\rho\})} (d^-(v) - 1),$$

where V is the vertex set of $\mathcal{H}$ and ρ labels the root vertex of $\mathcal{H}$ (Bordewich and Semple, 2007a). Intuitively, this is the number of edges in $\mathcal{H}$ that must be deleted to turn $\mathcal{H}$ into a rooted tree with leaf set X. Hence, $h(\mathcal{H}) = 0$ if and only if $\mathcal{H}$ is a rooted tree.

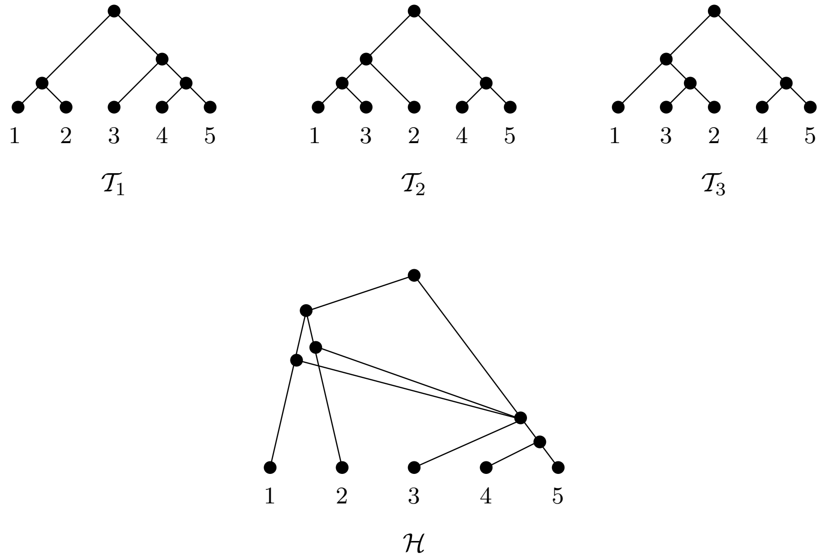


Figure 2.1: The hybridization network $\mathcal{H}$ displays the three rooted binary phylogenetic X-trees $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$. Note that there is one hybridization vertex in $\mathcal{H}$ and that $h(\mathcal{H}) = 2$.

Now let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a non-empty collection of rooted phylogenetic X-trees. Extending the corresponding definition given in Section 1.4, we say that $\mathcal{H}$ *displays* $\mathcal{P}$ if each tree in $\mathcal{P}$ is displayed by $\mathcal{H}$. Biologically speaking, all trees in $\mathcal{P}$ can be explained by an evolutionary scenario depicted in $\mathcal{H}$. Figure 2.1 shows a hybridization network $\mathcal{H}$ that displays $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\}$.

As described by Semple (2007), for a set $\mathcal{P}$ of rooted phylogenetic X-trees, we upgrade the definition of the hybridization number and set

$$h(\mathcal{P}) = \min\{h(\mathcal{H}) : \mathcal{H} \text{ is a hybridization network that displays } \mathcal{P}\}.$$

If $\mathcal{P} = \{\mathcal{T}, \mathcal{T}'\}$, then $h(\mathcal{P})$ is denoted by $h(\mathcal{T}, \mathcal{T}')$, and we often refer to it as the *minimum number of hybridization events* that is needed to explain the ancestor-descendant relationships of $\mathcal{T}$ and $\mathcal{T}'$ simultaneously. For this simplest case and if $\mathcal{T}$ and $\mathcal{T}'$ are binary, the calculation of $h(\mathcal{P})$ can be formalized as stated in the following optimization problem (Bordewich and Semple, 2007a):

Minimum Hybridization
Instance: Two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$.
Goal: Find a hybridization network $\mathcal{H}$ that displays $\mathcal{T}$ and $\mathcal{T}'$ with minimum hybridization number or, in other words, with minimum number of hybridization vertices.
Measure: The value $h(\mathcal{H})$.

As shown by Bordewich and Semple (2007a), Minimum Hybridization is NP-hard. However, this problem yields an attractive mathematical approach including a characterization of the minimum number of hybridization events in terms of agreement forests and several reduction rules that can be applied to reduce the size of the problem instance before calculating this minimum number exactly.

In the following, we explain the main ideas of this framework by focusing on the number of hybridization vertices in $\mathcal{H}$, denoted $h'(\mathcal{H})$, and setting

$$h'(\mathcal{P}) = \min \{h'(\mathcal{H}) : \mathcal{H} \text{ is a hybridization network that displays } \mathcal{P}\}.$$

In contrast to the definition of $h(\mathcal{P})$, we do not consider what each hybridization vertex contributes to the total hybridization number. This modification allows us to consider an arbitrarily large collection $\mathcal{P}$ of rooted binary phylogenetic X-trees. Since most data sets consist of more than two gene trees, this task is biologically well motivated and leads to the following modified optimization problem:

Minimum Hybridization Vertex
Instance: A set $\mathcal{P}$ of rooted binary phylogenetic X-trees.
Goal: Find a hybridization network $\mathcal{H}$ that displays $\mathcal{P}$ with minimum number of hybridization vertices.
Measure: The value $h'(\mathcal{H})$.

We close this section by remarking that, if $\mathcal{P} = \{\mathcal{T}, \mathcal{T}'\}$, the hybridization number is equal to the number of hybridization vertices in $\mathcal{H}$ displaying $\mathcal{P}$ because the maximal indegree of a vertex in $\mathcal{H}$ is two. Otherwise, if $|\mathcal{P}| > 2$, the number of hybridization vertices can be less than the hybridization number. As an example, see Figure 2.1, where $h(\mathcal{H}) = 2$ and $h'(\mathcal{H}) = 1$. It is deducible that Minimum Hybridization Vertex is a generalization of Minimum Hybridization, and it follows that Minimum Hybridization Vertex is NP-hard.

2.2 Agreement Forests

In the following, we characterize the number of hybridization vertices $h'(\mathcal{P})$ for a set $\mathcal{P}$ of rooted binary phylogenetic X-trees in terms of acyclic-agreement forests. To this end, the notion of agreement and acyclic-agreement forests for two rooted binary phylogenetic X-trees is extended for an arbitrary large set of such trees before proving the main result

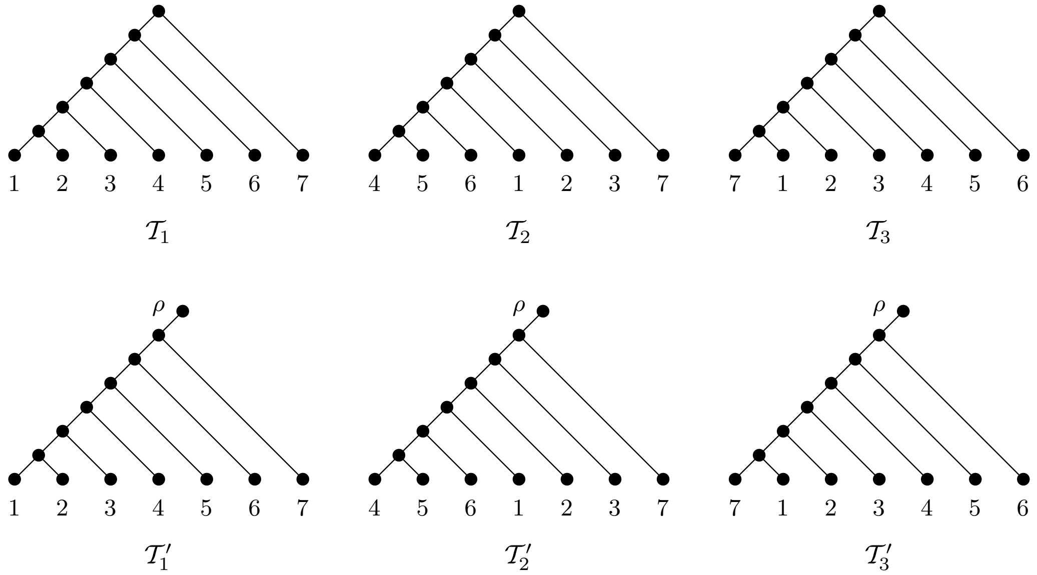


Figure 2.2: Top: Three rooted binary phylogenetic X-trees. Bottom: Each of the rooted binary tree has been obtained from its counterpart in the upper part by adding a root vertex labeled ρ at the end of a pendant edge adjoined to the original root.

(Theorem 2.1) of this section.

The definition of an agreement forest was first given by Hein *et al.* (1996) and revised by Bordewich and Semple (2004). Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees. For the purpose of the upcoming definition, we regard the root of every tree in $\mathcal{P}$ as a vertex labeled ρ at the end of a pendant edge adjoined to the original root (see Figure 2.2). Furthermore, we also view ρ as an element of $\mathcal{L}(\mathcal{T}_i)$, for all $i \in \{1, 2, \ldots, n\}$. An *agreement forest* for $\mathcal{P}$ is a collection $\{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ of leaf-labeled trees, where $\mathcal{S}_\rho$ is a rooted binary tree whose label set contains ρ, and $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k$ are rooted binary phylogenetic trees such that the following three conditions are satisfied:

(i) The label sets $\mathcal{L}(\mathcal{S}_\rho), \mathcal{L}(\mathcal{S}_1), \mathcal{L}(\mathcal{S}_2), \ldots, \mathcal{L}(\mathcal{S}_k)$ partition $X \cup \{\rho\}$.

(ii) For all $\mathcal{S}_j$ with $j \in \{\rho, 1, 2, \ldots, k\}$, the trees in $\{\mathcal{T}_i|\mathcal{L}(\mathcal{S}_j) : i \in \{1, 2, \ldots, n\}\}$ are isomorphic to $\mathcal{S}_j$.

(iii) For all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$, the trees in $\{\mathcal{T}_i(\mathcal{L}(\mathcal{S}_j)) : j \in \{\rho, 1, 2, \ldots, k\}\}$ are vertex-disjoint subtrees of $\mathcal{T}_i$.

A *maximum-agreement forest* for $\mathcal{P}$ is an agreement forest in which k is minimized over all agreement forests for $\mathcal{P}$. If $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ is a maximum-agreement forest for $\mathcal{P}$, we denote the value for k by $m(\mathcal{P})$. Examples of two agreement forests $\mathcal{F}_1$ and $\mathcal{F}_2$ for the three trees depicted in Figure 2.2 are shown in the upper part of Figure 2.3.

If $\mathcal{P} = \{\mathcal{T}, \mathcal{T}'\}$, it follows from Theorem 2.1 of Bordewich and Semple (2004) that

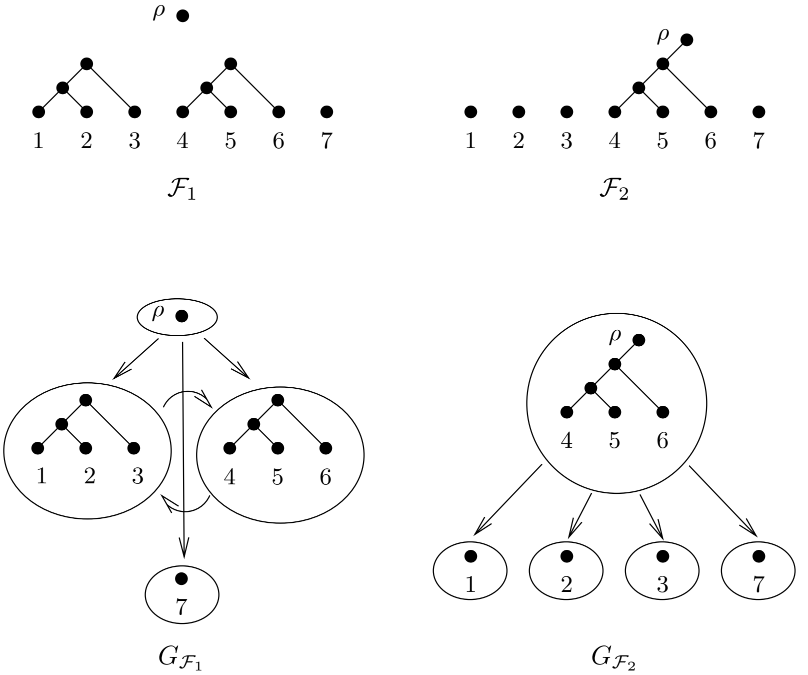


Figure 2.3: Top: Two agreement forests $\mathcal{F}_1$ and $\mathcal{F}_2$ for $\mathcal{P}$ consisting of the three rooted binary phylogenetic X-trees shown in Figure 2.2. Bottom: The digraphs $G_{\mathcal{F}_1}$ and $G_{\mathcal{F}_2}$ that correspond to $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively.

the number of trees in a maximum-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ minus one is equal to the rSPR distance between $\mathcal{T}$ and $\mathcal{T}'$ and thus $m(\mathcal{T},\mathcal{T}') = d_{\text{rSPR}}(\mathcal{T},\mathcal{T}')$. Moreover, $h(\mathcal{T},\mathcal{T}') = 1$ if and only if $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') = 1$ (Baroni *et al.*, 2004). Having this result, it is tempting to conjecture that every hybridization event can be represented by a single rSPR operation. This is not the case and, in general, $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}')$ is a lower bound for $h(\mathcal{T},\mathcal{T}')$ as one can construct pairs of trees such that $h(\mathcal{T},\mathcal{T}') > \frac{t}{2} - 1$ and $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') = 2$, where t is the number of taxa (Baroni *et al.*, 2005).

We now introduce a particular type of agreement forest that will be useful in characterizing the number of hybridization vertices $h'(\mathcal{P})$. For this purpose, an additional constraint on the properties of an agreement forest is required to exclude the existence of directed cycles in the corresponding hybridization network since, otherwise, a species can inherit genetic material from its own descendants, which is biologically not plausible. Again, let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees, and let $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ be an agreement forest for $\mathcal{P}$. Then $G_{\mathcal{F}}$ is the directed graph whose vertex set is $\mathcal{F}$ and for which $(\mathcal{S}_j, \mathcal{S}_{j'})$ with $j, j' \in \{\rho, 1, 2, \ldots, k\}$ is an arc precisely if $j \neq j'$ and if there exists a tree $\mathcal{T}_i \in \mathcal{P}$ for which the root of $\mathcal{T}_i(\mathcal{L}(\mathcal{S}_j))$ is an ancestor of the root of $\mathcal{T}_i(\mathcal{L}(\mathcal{S}_{j'}))$. Analogously to Baroni *et al.* (2005), where this concept has been introduced first, we call $\mathcal{F}$ an *acyclic-agreement forest* if $G_{\mathcal{F}}$ does not contain a directed cycle. A *maximum-acyclic-agreement forest* for $\mathcal{P}$ is an acyclic-agreement forest of smallest size. For such a forest, we denote the value for k by $m_a(\mathcal{P})$. Since every maximum-acyclic-agreement forest is an agreement forest, we have $m(\mathcal{P}) \leq m_a(\mathcal{P})$. In the lower part of Figure 2.3, the two digraphs $G_{\mathcal{F}_1}$ and $G_{\mathcal{F}_2}$ have been constructed from $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively (these are agreement-forests for the three trees that are shown in Figure 2.2). As $G_{\mathcal{F}_2}$ is acyclic, $\mathcal{F}_2$ is an acyclic-agreement forest for $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ depicted in Figure 2.2. Indeed, $\mathcal{F}_2$ is a maximum-acyclic-agreement forest for these three trees.

Analogously to Theorem 2 of Baroni *et al.* (2005), we now establish the first main result of this chapter by showing how the number of hybridization vertices for a set of rooted binary phylogenetic X-trees can be characterized in terms of acyclic-agreement forests.

Theorem 2.1. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees. Then*

$$h'(\mathcal{P}) = m_a(\mathcal{P}).$$

Proof. We first show that $h'(\mathcal{P}) \geq m_a(\mathcal{P})$. Let $\mathcal{H}$ be a hybridization network on X such that $h'(\mathcal{H}) = h'(\mathcal{P})$. Let $\mathcal{F}$ be the forest obtained from $\mathcal{H}$ by deleting, for each hybridization vertex v of $\mathcal{H}$, the arcs directed into v, and then suppressing any resulting vertex of degree two apart from the root of each tree. We show by induction on $h'(\mathcal{H})$ that $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$ with $h'(\mathcal{H})+1$ components, thus showing that $h'(\mathcal{P}) \geq m_a(\mathcal{P})$.

If $h'(\mathcal{H}) = 0$, then, up to isomorphism, all trees of $\mathcal{P}$ are identical. Thus $\mathcal{F} = \{\mathcal{T}_1\}$ and the result clearly holds. Now assume that $h'(\mathcal{H}) = m > 1$ and that the result holds for all sets of rooted binary phylogenetic X'-trees for which there exists a hybridization network that has at most $m-1$ hybridization vertices. Let v be a hybridization vertex of $\mathcal{H}$ such that the deletion of all arcs ending in v and, additionally, if $d^+(v) = 1$, the deletion of the arc leaving v and the vertex v itself, result in the following two components:

(i) a rooted binary phylogenetic tree, referred to as $\mathcal{T}_v$, whose label set is $\mathcal{C}_{\mathcal{H}}(v)$ and

(ii) a hybridization network $\mathcal{H}'$ containing the root vertex of $\mathcal{H}$.

Note that, as $\mathcal{H}$ is acyclic, there always exists such a vertex v. Furthermore, $\mathcal{T}_v$ is a pendant subtree of $\mathcal{T}_i \in \mathcal{P}$ for all $i \in \{1, 2, \ldots, n\}$. Let $\mathcal{P}' = \{\mathcal{T}'_1, \mathcal{T}'_2, \ldots, \mathcal{T}'_n\}$ be the set of rooted binary phylogenetic trees such that, for all $i \in \{1, 2, \ldots, n\}$, the tree $\mathcal{T}'_i$ has been obtained from $\mathcal{T}_i$ by pruning the pendant subtree $\mathcal{T}_v$ and suppressing any resulting vertex of degree two. Since $\mathcal{H}$ displays $\mathcal{P}$, it follows that $\mathcal{H}'$ displays $\mathcal{P}'$. Moreover, as $h'(\mathcal{H}') = m - 1$, it follows from the induction assumption that the forest $\mathcal{F}'$ obtained from $\mathcal{H}'$ by deleting the incoming arcs of each hybridization vertex and suppressing any resulting vertex of degree two is an acyclic-agreement forest for $\mathcal{P}'$ containing m trees. Let $\mathcal{F} = \mathcal{F}' \cup \{\mathcal{T}_v\}$. Since, for all $i \in \{1, 2, \ldots, n\}$, the rooted binary phylogenetic X-tree $\mathcal{T}_i$ can be obtained from $\mathcal{T}_i'$ by adjoining $\mathcal{T}_v$ via one additional new arc, $\mathcal{F}$ is an agreement forest for $\mathcal{P}$. Furthermore, as $\mathcal{T}_v$ is a pendant subtree in each $\mathcal{T}_i \in \mathcal{P}$, we also have that $\mathcal{F}$ is acyclic. Since $\mathcal{F}$ has $m + 1$ components, it follows that $h'(\mathcal{P}) \geq m_a(\mathcal{P})$.

We next show that $h'(\mathcal{P}) \leq m_a(\mathcal{P})$. To do this, it is sufficient to show that, if $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$ with $k+1$ trees, then there is a hybridization network that displays $\mathcal{P}$ with a hybridization number of at most k. The proof is by induction on k.

If $k = 0$, then, up to isomorphism, all trees of $\mathcal{P}$ are identical and so $\mathcal{T}_1$ is a hybridization network of the desired type. Now let $m_a(\mathcal{P}) = k$, and assume that the result holds for all sets of rooted binary phylogenetic X'-trees whose minimum number of components over all acyclic-agreement forests is at most k. Suppose that $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ is an acyclic-agreement forest for $\mathcal{P}$. Since $\mathcal{F}$ is acyclic, there is a vertex of $G_\mathcal{F}$ whose outdegree is zero. Clearly, as $k \geq 1$, this vertex is not $\mathcal{S}_\rho$ and so, without loss of generality, we may assume that it is the vertex $\mathcal{S}_k$. Since this vertex has outdegree zero in $G_\mathcal{F}$, it follows that $\mathcal{S}_k \in \mathcal{F}$ is a pendant subtree of all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$. Let $X' = X - \mathcal{L}(\mathcal{S}_k)$, and let $\mathcal{F}' = \mathcal{F} - \{\mathcal{S}_k\}$. As $\mathcal{S}_k$ is pendant in all trees of $\mathcal{P}$, it is easily seen that $\mathcal{F}'$ is an acyclic-agreement forest for $\mathcal{P}' = \{\mathcal{T}_1|X', \mathcal{T}_2|X', \ldots, \mathcal{T}_n|X'\}$. Therefore, since $|\mathcal{F}'| < |\mathcal{F}|$, it follows from the induction assumption that there is a hybridization network $\mathcal{H}'$ on X' with $h'(\mathcal{H}') \leq k - 1$ that displays $\mathcal{P}'$. Since $\mathcal{S}_k$ is a pendant subtree of $\mathcal{T}_1$, it is easily seen that there is a hybridization network that displays $\mathcal{T}_1$ and can be obtained from $\mathcal{H}'$ by adjoining $\mathcal{S}_k$ with a new arc a_1 that connects the root of $\mathcal{S}_k$ with a new vertex that subdivides an arc of $\mathcal{H}'$. Similarly, this process can be applied to all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{2, 3, \ldots, n\}$ using a new arc a_i. Now let $\mathcal{H}$ be a hybridization network on X obtained from $\mathcal{H}'$ by joining $\mathcal{S}_k$ to $\mathcal{H}'$ via all arcs $\{a_1, a_2, \ldots, a_n\}$. By construction, $\mathcal{H}$ displays $\mathcal{P}$. Furthermore, as $h'(\mathcal{S}_k) = 0$ and as the vertex of $\mathcal{H}$ corresponding to the root of $\mathcal{S}_k$ has indegree n, it follows that $h'(\mathcal{H}) \leq k$ and thus $h'(\mathcal{P}) \leq k = m_a(\mathcal{P})$. This completes the proof of the theorem. □

We next point out a further property of a maximum-acyclic-agreement forest $\mathcal{F}$ for a set $\mathcal{P}$ of rooted binary phylogenetic X-trees that will be of importance in Section 2.4. The proof of this lemma can be established in exactly the same way as the proof of Lemma 1 of Baroni *et al.* (2005) and is only given for reasons of completeness.

Lemma 2.2. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees, and let $\mathcal{S}_\rho$ be the tree in a maximum-acyclic-agreement forest $\mathcal{F}$ for $\mathcal{P}$ whose label set contains ρ. Then*

$$\mathcal{L}(\mathcal{S}_p) \cap X \neq \emptyset.$$

Proof. Assume that $\mathcal{L}(\mathcal{S}_p) = \{\rho\}$. Since $\mathcal{F}$ is a maximum-acyclic-agreement forest for $\mathcal{P}$, there is a vertex, $\mathcal{S}_1$ say, of $G_{\mathcal{F}} \backslash \mathcal{S}_\rho$ (That is the acyclic digraph obtained from $G_{\mathcal{F}}$ by deleting the vertex $\mathcal{S}_\rho$ and all edges incident with this vertex.) whose indegree is zero. Then it is easily checked that the forest obtained from $\mathcal{F}$ by joining the root of $\mathcal{S}_1$ with the root vertex ρ via a new edge is an acyclic-agreement forest $\mathcal{F}^*$ for $\mathcal{P}$ with $|\mathcal{F}^*| < |\mathcal{F}|$. This contradicts the maximality of $\mathcal{F}$. □

2.3 Subtree and Chain Reduction

In this section, we present the first two reduction rules initially introduced by Allen and Steel (2001) and Bordewich and Semple (2007b) for the two-tree case. Both rules can be used to reduce the size of the problem instance, before calculating the number of hybridization vertices $h'(\mathcal{P})$ for a set of rooted binary phylogenetic X-trees exactly. To make the reductions work, we first need some further definitions.

Let $\mathcal{T}$ be a rooted binary phylogenetic X-tree. For $m \geq 2$, an *m-chain* of $\mathcal{T}$ is a tuple $(a_1, a_2, \ldots, a_m)$ of leaf labels in $\mathcal{T}$ such that

(i) the parent of the vertex labeled a_1 is either the same as the parent of the vertex labeled a_2 or a child of the parent of the vertex labeled a_2 and

(ii) for all $i \in \{2, 3, \ldots, m-1\}$, the parent of the vertex labeled a_i is a child of the parent of the vertex labeled a_{i+1}.

Let P be a disjoint collection of 2-element subsets of X such that each pair $\{a, b\} \in P$ is a 2-chain common to all trees of $\mathcal{P}$. Let $w : P \to \mathbb{Z}^+$ be a weight function on the elements of P, such that each pair is assigned a positive integer weight. We refer to such a set $\mathcal{P}$ of trees with associated set P and weight function w as a set of *weighted rooted binary phylogenetic X-trees.*

Now we are in a position to state the above mentioned reduction rules for a set $\mathcal{P}$ of weighted rooted binary phylogenetic X-trees.

Subtree Reduction. Replace a maximal pendant subtree (for the definition of such a tree, see Section 1.4) that contains at least two leaves and that occurs identically in all $\mathcal{T}_i \in \mathcal{P}$ by a single leaf with a new label and delete all members of P whose elements label leaves of the pendant subtree.

Chain Reduction. For $m \geq 3$, replace a maximal m-chain $(a_1, a_2, \ldots, a_m)$ that occurs identically and with the same orientation relative to the root in all $\mathcal{T}_i \in \mathcal{P}$ by a 2-chain with new labels a and b. Furthermore, add the new 2-element set $\{a, b\}$ to P with weight

$$w(\{a,b\}) = m - 2 + \sum_{\substack{\{a_i, a_j\} \in P; \\ a_i, a_j \in \{a_1, \ldots, a_m\}}} w(\{a_i, a_j\}), \tag{2.1}$$

and delete all pairs in P whose elements are in $\{a_1, a_2, \ldots, a_m\}$.

An example, for when $\mathcal{P} = \{\mathcal{S}, \mathcal{T}\}$, is given in Figure 3.1 for the subtree reduction and in Figure 3.2 for the chain reduction.

Remark. The label set of any subtree or chain that is common to all $\mathcal{T}_i \in \mathcal{P}$ and that is reduced in the course of one of the above mentioned reductions intersects each pair in P in either both elements or neither. This is simply due to the fact that the reductions are applied to *maximal* pendant subtrees and *maximal* chains, respectively.

To reduce the trees in $\mathcal{P}$ as much as possible, it is desirable to repeat the application of the subtree and chain reduction. This implies that we need to keep track of the weighting since an m-chain may contain consecutive pairs of leaves that have previously been involved in a chain reduction. Hence, each pair $\{a, b\}$ of new leaves is assigned a weight that is the sum of the associated weights of these pairs plus $m-2$ (see Equation 2.1). Additionally, it is essential to consider the weight of each reduced 2-chain in terms of computing a maximum-acyclic-agreement forest; that is calculating $h'(\mathcal{P})$. To this end, we introduce a third notion of agreement forests (Bordewich and Semple, 2007b). For a set of weighted rooted binary phylogenetic X-trees, an agreement forest $\mathcal{F}$ is *legitimate* if it is acyclic and the following property holds:

(P) If $\{a, b\} \in P$, then a and b are either both contained in the label set of one tree of $\mathcal{F}$ or a and b label isolated vertices in $\mathcal{F}$.

Let $\mathcal{F}$ be an agreement forest for $\mathcal{P}$. We define the *weight* of $\mathcal{F}$, to be

$$w(\mathcal{F}) = |\mathcal{F}| - 1 + w_c(\mathcal{F}, P), \tag{2.2}$$

where

$$w_c(\mathcal{F}, P) = \sum_{\substack{\{a,b\} \in P;\ a \text{ and } b \\ \text{isolated in } \mathcal{F}}} w(\{a, b\}).$$

Furthermore, we set $f(\mathcal{P})$ to be the minimum weight of a legitimate-agreement forest for $\mathcal{P}$. With the term *legitimate-agreement forest of minimum weight*, we refer to a legitimate-agreement forest $\mathcal{F}$ with $w(\mathcal{F}) = f(\mathcal{P})$.

An algorithmic approach to compute such a forest for two rooted binary phylogenetic X-trees is covered in detail by Chapter 3. However, it is important to note here that the set P of 2-chains is initially empty. Hence, 2-chains are only added if they are the result of a chain reduction, which by definition means that they are the result of reducing a strictly bigger chain.

We next state a theorem showing that the subtree and chain reduction preserve the weight of a legitimate-agreement forest of minimum weight. This theorem corresponds to Proposition 3.2 of Bordewich and Semple (2007b), where it is established for when $\mathcal{P} = \{\mathcal{T}, \mathcal{T}'\}$.

Theorem 2.3. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of weighted rooted binary phylogenetic X-trees, and let $\mathcal{P}' = \{\mathcal{T}_1', \mathcal{T}_2', \ldots, \mathcal{T}_n'\}$ be a set of such trees obtained from $\mathcal{P}$ by applying either the subtree or chain reduction. Then*

$$f(\mathcal{P}) = f(\mathcal{P}').$$

Before proving this theorem, we need the following lemma pointing out some crucial properties of all legitimate-agreement forests of minimum weight for a given set of weighted rooted binary phylogenetic X-trees.

Lemma 2.4. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of weighted rooted binary phylogenetic X-trees. With $i \in \{1, 2, \ldots, n\}$, let A be the leaf set of a maximal pendant subtree common to all $\mathcal{T}_i \in \mathcal{P}$ and let $(a_1, a_2, \ldots, a_m)$ be a maximal m-chain common to all $\mathcal{T}_i \in \mathcal{P}$, where $m \geq 3$. Then every legitimate-agreement forest $\mathcal{F}$ for $\mathcal{P}$ of minimum weight fulfills the following properties:*

(a) *$\mathcal{F}$ contains a tree such that A is a subset of its label set and*

(b) *either $\mathcal{F}$ contains a tree such that $\{a_1, a_2, \ldots, a_m\}$ is a subset of its label set or each of $a_1, a_2, \ldots, a_m$ labels an isolated vertex in $\mathcal{F}$.*

Proof. To prove this lemma, we follow the approach of Lemma 3.1 in Bordewich and Semple (2007b). Let $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ be a legitimate-agreement forest for $\mathcal{P}$ of minimum weight. We start with the proof of part (a). For a contradiction, assume that A is not a subset of the label set of a single tree in $\mathcal{F}$. We construct a new legitimate-agreement forest $\mathcal{F}'$ that satisfies (a) and has a smaller weight than $\mathcal{F}$. Let J index the trees of $\mathcal{F}$ which include elements of A in their label sets. To be precise, $J = \{j \in \{\rho, 1, 2, \ldots, k\} : \mathcal{L}(\mathcal{S}_j) \cap A \neq \emptyset\}$. In the following, we denote $\bigcup_{j \in J} \mathcal{L}(\mathcal{S}_j)$ by $\mathcal{L}_A$. Since, for all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$, the trees in $\{\mathcal{T}_i(\mathcal{L}(\mathcal{S}_j)) : j \in \{\rho, 1, 2, \ldots, k\}\}$ are vertex-disjoint subtrees of $\mathcal{T}_i$ (see (iii) in the definition of an agreement forest), observe that $\mathcal{L}(\mathcal{S}_j) - A \neq \emptyset$ for at most one element of J. Let $\mathcal{F}'$ be the forest that is obtained from $\mathcal{F}$ by deleting each tree $\mathcal{S}_j$ with $j \in J$, and adding the new tree $\mathcal{S}_A = \mathcal{T}_1|\mathcal{L}_A$. Now it is easily checked that $\mathcal{F}'$ is an agreement forest for $\mathcal{P}$. Furthermore, it is legitimate since the elements of A label a pendant subtree and thus $\mathcal{F}'$ is acyclic, and since A is the label set of a maximal pendant subtree, property (P) holds. Moreover, it is easily checked that $w(\mathcal{F}) > w(\mathcal{F}')$ since $\mathcal{F}'$ has fewer components and no additional element of P labels isolated vertices in $\mathcal{F}'$. Summing up, this gives a contradiction.

We now turn to the proof of (b). For convenience in this part of the proof, we will rewrite $\mathcal{L}(\mathcal{S}_\rho), \mathcal{L}(\mathcal{S}_1), \mathcal{L}(\mathcal{S}_2), \ldots, \mathcal{L}(\mathcal{S}_k)$ as $\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k$. Assume that some a_i does not label an isolated vertex in $\mathcal{F}$. Then, without loss of generality, the label a_i is contained in the label set $\mathcal{L}_i$, where $\mathcal{L}_i - \{a_i\} \neq \emptyset$. If $\{a_1, a_2, \ldots, a_m\} \subseteq \mathcal{L}_i$, the result follows immediately. Therefore, we may assume that there exists some element of $\{a_1, a_2, \ldots, a_m\}$ which is not an element of $\mathcal{L}_i$. First, we eliminate a particular way that a_i may be related to $\mathcal{L}_i - \{a_i\}$ in elements of $\mathcal{P}$.

Let v be the vertex of $\mathcal{S}_i$ labeled a_i. Suppose that v and the root of S_i are adjacent such that the parent of v is an ancestor of mrca$(\mathcal{L}_i - \{a_i\})$, for all members of a proper subset $\mathcal{P}_1$ of $\mathcal{P}$ with $\mathcal{P}_1 \neq \emptyset$, while the parent of v is not an ancestor of mrca$(\mathcal{L}_i - \{a_i\})$ for all members of $\mathcal{P} - \mathcal{P}_1$ denoted by $\mathcal{P}_2$ (see Figure 2.4). Then each element of $\{a_1, a_2, \ldots, a_m\} - \{a_i\}$ labels an isolated vertex in $\mathcal{F}$ since, otherwise, the corresponding minimal rooted subtrees of two trees in $\mathcal{F}$ are not vertex-disjoint in all trees of $\mathcal{P}_1$ or $\mathcal{P}_2$. By deleting v and its incident edge from $\mathcal{S}_i$, suppressing any resulting degree two vertices in the resulting tree, and replacing the isolated vertices whose labels partition $\{a_1, a_2, \ldots, a_m\} - \{a_i\}$ with a single tree that is isomorphic to $\mathcal{T}_1|\{a_1, a_2, \ldots, a_m\}$, it is easily checked that the resulting

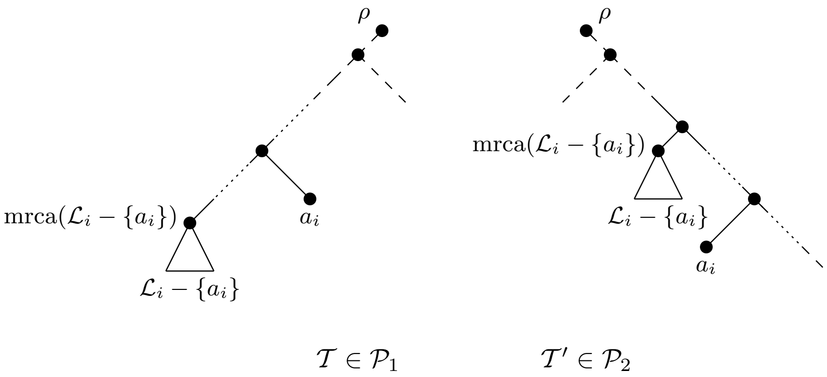

ρ
mrca($\mathcal{L}_i - \{a_i\}$)
a_i
$\mathcal{L}_i - \{a_i\}$
$\mathcal{T} \in \mathcal{P}_1$
ρ
mrca($\mathcal{L}_i - \{a_i\}$)
$\mathcal{L}_i - \{a_i\}$
a_i
$\mathcal{T}' \in \mathcal{P}_2$

Figure 2.4: Assuming that this configuration (details see text) appears in $\mathcal{F}$, the elements of $\{a_1, a_2, \ldots, a_m\} - \{a_i\}$ label isolated vertices in $\mathcal{F}$. Dotted lines indicate regions of a maximal common m-chain. (Adapted from Bordewich and Semple (2007b).)

agreement forest, denoted by $\mathcal{F}'$, is acyclic. Since $(a_1, a_2, \ldots, a_m)$ is a maximal m-chain and $\mathcal{F}$ is legitimate, it follows that $\mathcal{F}'$ also satisfies (P). But $w(\mathcal{F}') < w(\mathcal{F})$ contradicts the minimality of $\mathcal{F}$. Thus we may assume that, if v is adjoined to the root of $\mathcal{S}_i$ and the parent of v is an ancestor of mrca($\mathcal{L}_i - \{a_i\}$) for all trees of $\mathcal{P}_1$, then the parent of v is also an ancestor of mrca($\mathcal{L}_i - \{a_i\}$) for all trees of $\mathcal{P}_2$.

Now let J index the trees of $\mathcal{F}$ whose label sets contain elements of the chain. To be precise, that is $J = \{j \in \{\rho, 1, 2, \ldots, k\} : \mathcal{L}_j \cap \{a_1, a_2, \ldots, a_m\} \neq \emptyset\}$. Observe that $\mathcal{L}_j - \{a_1, a_2, \ldots, a_m\} \neq \emptyset$ for at most two elements of J since the corresponding minimal rooted subtrees are vertex-disjoint in all trees of $\mathcal{P}$. In the following, we denote $\bigcup_{j \in J} \mathcal{L}_j$ by $\mathcal{L}_C$. Let $\mathcal{F}'$ be the forest that has been obtained from $\mathcal{F}$ by deleting each tree $\mathcal{S}_j$ with $j \in J$ and inserting the new tree $\mathcal{S}_C = \mathcal{T}_1|\mathcal{L}_C$. Essentially, we have joined all trees in $\mathcal{F}$ whose label sets contain any element of $\{a_1, a_2, \ldots, a_m\}$ along the chain. It follows from the assumption at the end of the previous paragraph that $\mathcal{F}'$ is an agreement forest for $\mathcal{P}$. Furthermore, as $(a_1, a_2, \ldots, a_m)$ is a maximal chain, $\mathcal{F}'$ satisfies (P).

We next show that $\mathcal{F}'$ is acyclic. Consider the directed graphs $G_{\mathcal{F}}$ and $G_{\mathcal{F}'}$ associated with $\mathcal{F}$ and $\mathcal{F}'$, respectively. The vertex set of $G_{\mathcal{F}'}$ is obtained from $G_{\mathcal{F}}$ by deleting the vertex $\mathcal{S}_j$, for all $j \in J$, and introducing the new vertex $\mathcal{S}_C$. Furthermore, if $\mathcal{S}_u, \mathcal{S}_v \in (\mathcal{F}' - \{\mathcal{S}_C\})$ then $(\mathcal{S}_u, \mathcal{S}_v)$ is an arc in $G_{\mathcal{F}'}$ if and only if $(\mathcal{S}_u, \mathcal{S}_v)$ is an arc in $G_{\mathcal{F}}$. Regarding the arcs incident with $\mathcal{S}_C$, there are two cases to consider. First, suppose there is some $j_1 \in J$ such that the root of $\mathcal{T}_l(\mathcal{L}_{j_1})$, for some $\mathcal{T}_l \in \mathcal{P}$ with $l \in \{1, 2, \ldots, n\}$, is on the path from a_m to the root. Then the root of $\mathcal{T}_l(\mathcal{L}_C)$ is the same as the root of $\mathcal{T}_l(\mathcal{L}_{j_1})$. Due to our assumption at the end of the penultimate paragraph, these roots must coincide for every tree of $\mathcal{P}$. As a result, $(\mathcal{S}_C, \mathcal{S}_u)$ and $(\mathcal{S}_u, \mathcal{S}_C)$ are arcs in $G_{\mathcal{F}'}$ if and only if $(\mathcal{S}_{j_1}, \mathcal{S}_u)$ and $(\mathcal{S}_u, \mathcal{S}_{j_1})$ are arcs in $G_{\mathcal{F}}$, respectively. Since $G_{\mathcal{F}}$ is acyclic, $G_{\mathcal{F}'}$ must be also.

Second, suppose there is no such $j_1 \in J$. Then the root of $\mathcal{T}_l(\mathcal{L}_C)$ is the parent of a_m for all $\mathcal{T}_l \in \mathcal{P}$. Since not all of the elements labeled with $\{a_1, a_2, \ldots, a_m\}$ are isolated in $\mathcal{F}$, there is some $j_2 \in J$ such that the root of $\mathcal{T}_l(\mathcal{L}_{j_2})$ is on the path from a_1 to the root for all $\mathcal{T}_l \in \mathcal{P}$. It again follows, that $(\mathcal{S}_C, \mathcal{S}_u)$ and $(\mathcal{S}_u, \mathcal{S}_C)$ are arcs in $G_{\mathcal{F}'}$ if and only if $(\mathcal{S}_{j_2}, \mathcal{S}_u)$ and $(\mathcal{S}_u, \mathcal{S}_{j_2})$ are arcs in $G_{\mathcal{F}}$, respectively, and so $G_{\mathcal{F}'}$ is acyclic. Hence, $\mathcal{F}'$ is a legitimate-agreement forest for $\mathcal{P}$. If the vertices labeled with $a_1, a_2, \ldots, a_m$ are not all in the same component of $\mathcal{F}$ (thus $|J| > 1$), then we have reduced the number of components and so $w(\mathcal{F}') < w(\mathcal{F})$. This contradicts the minimality of $\mathcal{F}$. Under the original assumption that some a_i does not label an isolated vertex, we conclude that the chain is entirely contained in a single component of $\mathcal{F}$. This completes the proof of the lemma. □

We are now in a position to prove Theorem 2.3.

Proof. It immediately follows from Lemma 2.4(a) that this theorem holds if $\mathcal{P}'$ has been obtained from $\mathcal{P}$ by applying the subtree reduction because a maximal pendant subtree common to all trees in $\mathcal{P}$ completely stays together in one tree of a legitimate-agreement forest for $\mathcal{P}$ of minimum weight. Therefore, consider a single application of the chain reduction, where an m-chain $(a_1, a_2, \ldots, a_m)$ is common to all trees in $\mathcal{P}$ and gets reduced to a 2-chain (a, b). Furthermore, let $\mathcal{F}_{\mathcal{P}}$ be a legitimate-agreement forest for $\mathcal{P}$ of minimum weight. By Lemma 2.4(b), we either have

(i) $\{a_1, a_2, \ldots, a_m\}$ is a subset of a label set of a tree in $\mathcal{F}_{\mathcal{P}}$ or

(ii) each of $a_1, a_2, \ldots, a_m$ labels an isolated vertex in $\mathcal{F}_{\mathcal{P}}$.

Depending on whether (i) or (ii) holds, let $\mathcal{F}_{\mathcal{P}'}$ be the forest obtained from $\mathcal{F}_{\mathcal{P}}$ by either replacing the m-chain $(a_1, a_2, \ldots, a_m)$ with the 2-chain (a, b) in some element of $\mathcal{F}_{\mathcal{P}}$ or replacing the isolated vertices labeled with the elements of this m-chain with two isolated vertices labeled a and b. Since $\mathcal{F}_{\mathcal{P}}$ is a legitimate-agreement forest for $\mathcal{P}$, it is easily checked that $\mathcal{F}_{\mathcal{P}'}$ is such a forest for $\mathcal{P}'$. Moreover, in the case that (ii) holds, the contribution of the isolated vertices $a_1, a_2, \ldots, a_m$ to $w(\mathcal{F}_{\mathcal{P}})$ is exactly the same as the contribution of the isolated vertices a and b to $w(\mathcal{F}_{\mathcal{P}'})$ (see Equations 2.1 and 2.2). It now follows that $f(\mathcal{P}') \leq f(\mathcal{P})$.

Now suppose that $\mathcal{F}_{\mathcal{P}'}$ is a legitimate-agreement forest for $\mathcal{P}'$ of minimum weight. Since $\mathcal{F}_{\mathcal{P}'}$ is legitimate, either

(i) $\mathcal{F}_{\mathcal{P}'}$ contains a tree, $\mathcal{S}_i$ say, such that $\{a, b\} \subseteq \mathcal{L}(\mathcal{S}_i)$ or

(ii) a and b label isolated vertices in $\mathcal{F}_{\mathcal{P}'}$.

Depending on which holds, let $\mathcal{F}_{\mathcal{P}}$ be the forest obtained from $\mathcal{F}_{\mathcal{P}'}$ by either replacing $\mathcal{S}_i$ with $\mathcal{T}_1|((\mathcal{L}(\mathcal{S}_i) - \{a, b\}) \cup \{a_1, a_2, \ldots, a_m\})$ or replacing the isolated vertices labeled a and b with m isolated vertices labeled $a_1, a_2, \ldots, a_m$, respectively. Since $\mathcal{F}_{\mathcal{P}'}$ is a legitimate-agreement forest for $\mathcal{P}'$, it is easily checked that $\mathcal{F}_{\mathcal{P}}$ is such a forest for $\mathcal{P}$. Furthermore, in case (ii), the contribution of the isolated vertices labeled a and b to $w(\mathcal{F}_{\mathcal{P}'})$ is the same as the contribution of the isolated vertices labeled $a_1, a_2, \ldots, a_m$ to $w(\mathcal{F}_{\mathcal{P}})$ (see Equations 2.1 and 2.2 and) in case (ii). Thus we can deduce that $f(\mathcal{P}) \leq f(\mathcal{P}')$. Combining both inequalities gives the desired result. □

To conclude, let $\mathcal{P}$ be a set of weighted rooted binary phylogenetic X-trees with an associated set P that is empty, and let $\mathcal{P}'$ be a set of such trees obtained from $\mathcal{P}$ by applying a sequence of subtree and chain reductions. Consider that we always have $f(\mathcal{P}) \geq h'(\mathcal{P})$ since the weight function is non-negative and $f(\mathcal{P}) = h'(\mathcal{P})$ whenever $P = \emptyset$. Then, by Theorem 2.3, we can deduce that $h'(\mathcal{P}) = f(\mathcal{P}) = f(\mathcal{P}')$.

2.4 Cluster Reduction

Beside applying the subtree and chain reduction to a set $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ of rooted binary phylogenetic X-trees, the calculation of $h'(\mathcal{P})$ can additionally be sped up by using an efficient divide-and-conquer approach (Baroni *et al.*, 2006) if one or more clusters are common to all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$. Loosely speaking, while the subtree and chain reduction reduce the size of the problem instance, the cluster reduction breaks the problem into a number of smaller and more tractable subproblems. We next state the cluster reduction and prove two results in terms of (ordinary) acyclic-agreement forests before closing this chapter by showing how this reduction can easily be fitted into the framework of legitimate-agreement forests.

Cluster reduction. Suppose that A is a cluster with $|A| \geq 2$ that is common to all trees of $\mathcal{P}$. Then replace $\mathcal{P}$ with two new sets of trees. The first set $\mathcal{P}_a$ is obtained from $\mathcal{P}$ by replacing the minimal pendant subtree of $\mathcal{T}_i \in \mathcal{P}$ whose leaf set is A by a leaf with a new label for all $i \in \{1, 2, \ldots, n\}$, while the second set $\mathcal{P}_A$ contains the trees $\mathcal{T}_1|A, \mathcal{T}_2|A, \ldots, \mathcal{T}_n|A$ (see Figure 3.3).

Analogously to Theorem 1 of Baroni *et al.* (2006), we establish the first result of this section.

Theorem 2.5. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees. Suppose that $A \subset X$ is a cluster common to all trees of $\mathcal{P}$. Let $\mathcal{P}_a$ and $\mathcal{P}_A$ be the two sets of rooted binary phylogenetic trees that have been obtained from $\mathcal{P}$ in the course of a cluster reduction. Then*

$$h'(\mathcal{P}) = h'(\mathcal{P}_A) + h'(\mathcal{P}_a).$$

Proof. If $A = X$, then the result clearly holds. Therefore, we may assume that $A \subset X$. We first show that

$$h'(\mathcal{P}) \leq h'(\mathcal{P}_A) + h'(\mathcal{P}_a).$$

To this end, let $\mathcal{F}_A$ be a maximum-acyclic-agreement forest for $\mathcal{P}_A$, and let $\mathcal{F}_a$ be such a forest for $\mathcal{P}_a$. Suppose that $\mathcal{P}_a$ has been obtained from $\mathcal{P}$ by replacing the minimal rooted subtree induced by A with a single leaf labeled a for all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$. Let $\mathcal{S}_{\rho,A}$ be the unique tree in $\mathcal{F}_A$ with a vertex labeled ρ, and let $\mathcal{S}_{j,a}$ be the unique tree in $\mathcal{F}_a$ containing a vertex labeled a. Then obtain the rooted binary tree $\mathcal{S}_{A,a}$ in one of the following two ways:

(a) If $\mathcal{S}_{j,a}$ is an isolated vertex, then obtain $\mathcal{S}_{A,a}$ from $\mathcal{S}_{\rho,A}$ by deleting the edge which is incident with the vertex labeled ρ and the vertex labeled ρ itself.

(b) Otherwise, obtain $\mathcal{S}_{A,a}$ by adjoining $\mathcal{S}_{\rho,A}$ to $\mathcal{S}_{j,a}$ via a new edge joining the vertices labeled ρ and a, removing the labels a and ρ, and suppressing any vertex of degree two apart from the root.

By Lemma 2.2, note that $\mathcal{L}(\mathcal{S}_{\rho,A}) - \{\rho\} \neq \emptyset$ and thus $\mathcal{L}(\mathcal{S}_{A,a}) \neq \emptyset$. Since $\mathcal{F}_A$ and $\mathcal{F}_a$ are acyclic-agreement forests,

$$\mathcal{F} = ((\mathcal{F}_a \cup \mathcal{F}_A) - \{\mathcal{S}_{j,a}, \mathcal{S}_{\rho,A}\}) \cup \{\mathcal{S}_{A,a}\}$$

is an acyclic-agreement forest for $\mathcal{P}$ with $|\mathcal{F}| = |\mathcal{F}_A| + |\mathcal{F}_a| - 1$. It now follows from Theorem 2.1 that

$$h'(\mathcal{P}_A) + h'(\mathcal{P}_a) = m_a(\mathcal{P}_A) + m_a(\mathcal{P}_a) = |\mathcal{F}_A| - 1 + |\mathcal{F}_a| - 1 = |\mathcal{F}| - 1 \geq h'(\mathcal{P}).$$

We next show that

$$h'(\mathcal{P}) \geq h'(\mathcal{P}_A) + h'(\mathcal{P}_a).$$

Let $\mathcal{F}$ be a maximum-acyclic-agreement forest for $\mathcal{P}$, then there are two cases to consider:

(i) there exists $\mathcal{S}_j \in \mathcal{F}$ such that $\mathcal{L}(\mathcal{S}_j) \cap A \neq \emptyset$ and $\mathcal{L}(\mathcal{S}_j) \cap ((X - A) \cup \{\rho\}) \neq \emptyset$, and
(ii) for all $\mathcal{S}_l \in \mathcal{F}$, either $\mathcal{L}(\mathcal{S}_l) \subseteq A$ or $\mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})$.

For case (i), we assume that $\mathcal{S}_j$ is such a tree in $\mathcal{F}$. Then, for all $i \in \{1, 2, \ldots, n\}$, the minimal rooted subtree of $\mathcal{T}_i \in \mathcal{P}$ that contains the label set of $\mathcal{S}_j$ includes the root of $\mathcal{T}_i|A$. Since $\mathcal{F}$ is an agreement forest, this implies that $\mathcal{S}_j$ is the unique such tree in $\mathcal{F}$. Then obtain $\mathcal{S}_{j,A}$ from $\mathcal{S}_j$ by adding a vertex labeled ρ at the end of a pendant edge adjoined to the root of $\mathcal{S}_j|(A \cap \mathcal{L}(\mathcal{S}_j))$. Additionally, obtain $\mathcal{S}_{j,a}$ from $\mathcal{S}_j$ by replacing the pendant subtree having leaf set $A \cap \mathcal{L}(\mathcal{S}_j)$ with a single leaf labeled a. As $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$,

$$\mathcal{F}_A = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\} \cup \{\mathcal{S}_{j,A}\}$$

is an acyclic-agreement forest for $\mathcal{P}_A$ and

$$\mathcal{F}_a = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})\} \cup \{\mathcal{S}_{j,a}\}$$

is such a forest for $\mathcal{P}_a$. With $|\mathcal{F}| = |\mathcal{F}_A| + |\mathcal{F}_a| - 1$, we can deduce that

$$h'(\mathcal{P}) = m_a(\mathcal{P}) = |\mathcal{F}| - 1 = |\mathcal{F}_A| + |\mathcal{F}_a| - 1 - 1 \geq h'(\mathcal{P}_A) + h'(\mathcal{P}_a).$$

Next, we show that the inequality also holds for case (ii). As $G_{\mathcal{F}}$ of $\mathcal{P}$ is acyclic, it follows that the subdigraph induced by the set $\{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\}$ does not contain any directed cycle. Hence, this subdigraph has a vertex, $\mathcal{S}_0$ say, with indegree zero. Let $\mathcal{S}_{0,\rho}$ be the tree obtained from $\mathcal{S}_0$ by adding a vertex labeled ρ at the end of a pendant edge adjoined to the original root of $\mathcal{S}_0$. Since $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$, it is easily seen that

$$\mathcal{F}_A = (\{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\} - \{\mathcal{S}_0\}) \cup \{\mathcal{S}_{0,\rho}\}$$

is an acyclic-agreement forest for $\mathcal{P}_A$. Furthermore,

$$\mathcal{F}_a = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})\} \cup \{a\}$$

is such a forest for $\mathcal{P}_a$ in which a is used to denote an isolated vertex labeled a. Thus

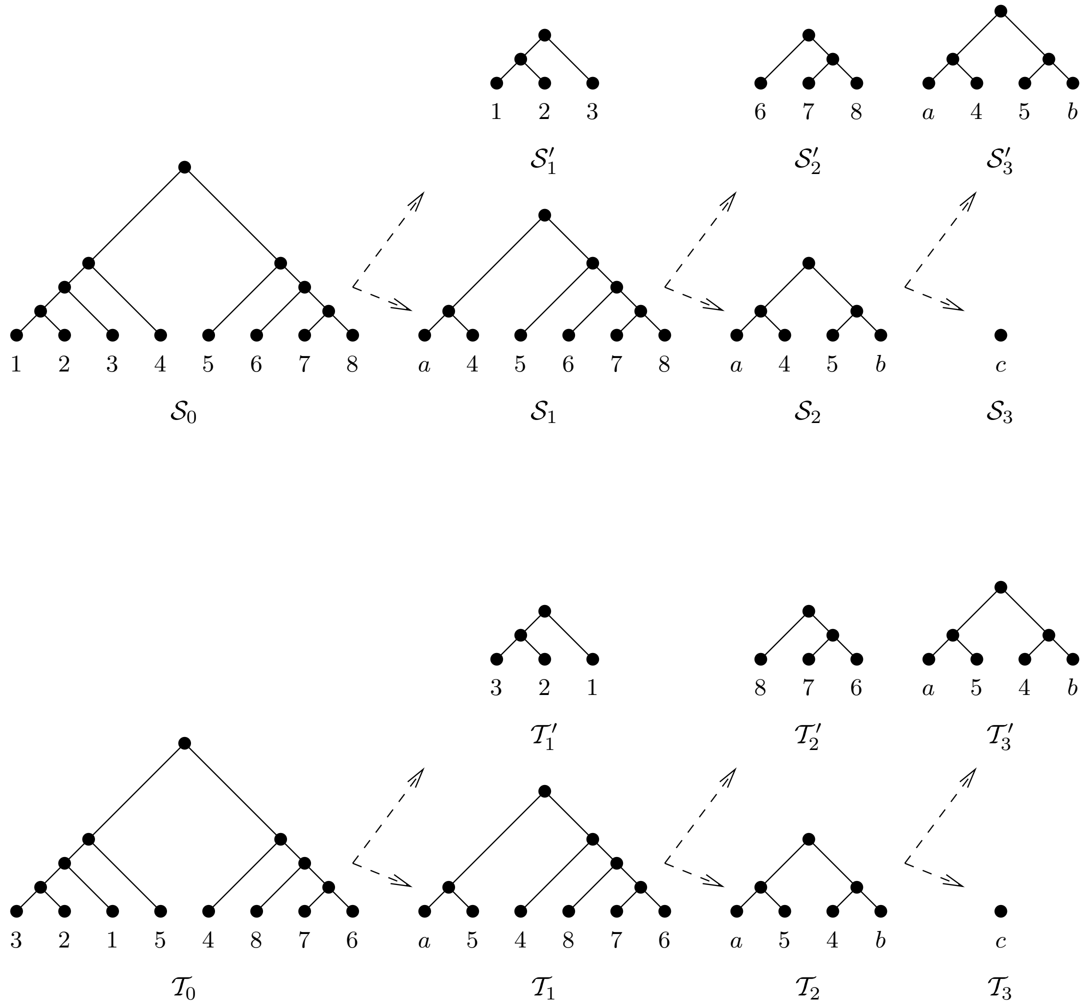


Figure 2.5: Repeated applications of the cluster reduction on $\mathcal{S}_0$ and $\mathcal{T}_0$. Note that $h(\mathcal{S}_0, \mathcal{T}_0) = h(\mathcal{S}_1', \mathcal{T}_1') + h(\mathcal{S}_2', \mathcal{T}_2') + h(\mathcal{S}_3', \mathcal{T}_3') + h(\mathcal{S}_3, \mathcal{T}_3)$.

$|\mathcal{F}| = |\mathcal{F}_A| + |\mathcal{F}_a| - 1$ and so

$$h'(\mathcal{P}) = m_a(\mathcal{P}) = |\mathcal{F}| - 1 = |\mathcal{F}_A| + |\mathcal{F}_a| - 1 - 1 \geq h'(\mathcal{P}_A) + h'(\mathcal{P}_a).$$

This completes the proof of the theorem. □

Figure 2.5 shows an example of three repeated applications of the cluster reduction when $\mathcal{P} = \{\mathcal{S}_0, \mathcal{T}_0\}$. For all $j \in \{1, 2, 3\}$, the trees $\mathcal{S}_j$ and $\mathcal{S}_j'$, and $\mathcal{T}_j$ and $\mathcal{T}_j'$, respectively, are obtained by applying the cluster reduction to $\mathcal{S}_{j-1}$ and $\mathcal{T}_{j-1}$. In general, let $\mathcal{P}_0 = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees. We say that the cluster reduction has been applied t times if, for all $j \in \{1, 2, \ldots, t\}$, the j^{th} cluster reduction replaces $\mathcal{P}_{j-1}$ with two new sets of rooted binary phylogenetic trees:

(i) the *cluster-reduced tree set* $\mathcal{P}_j$ which has been obtained from $\mathcal{P}_{j-1}$ by replacing the subtree whose label set A_j (with $|A_j| \geq 2$) is a common cluster of $\mathcal{T}_i \in \mathcal{P}_{j-1}$, for all $i \in \{1, 2, \ldots, n\}$, with a single vertex labeled $l_j \notin (X \cup \{\rho\} \cup \{l_1, l_2, \ldots, l_{j-1}\})$ for all $\mathcal{T}_i \in \mathcal{P}_{j-1}$ and

(ii) the *cluster-tree set* $\mathcal{P}'_j$ obtained from $\mathcal{P}_{j-1}$ by replacing each tree $\mathcal{T}_i \in \mathcal{P}_{j-1}$ with $\mathcal{T}_i|A_j$.

In the following, we refer to the $(t+1)$-tuple $\mathcal{R} = (\mathcal{P}'_1, \mathcal{P}'_2, \ldots, \mathcal{P}'_t, \mathcal{P}_t)$ as a *cluster-tree collection* of $\mathcal{P}_0$.

We next state a corollary that shows how the cluster reduction can repeatedly be applied to calculate the number of hybridization vertices $h'(\mathcal{P})$ for a set $\mathcal{P}$ of rooted binary phylogenetic X-trees.

Corollary 2.6. *Let $\mathcal{P}_0 = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of rooted binary phylogenetic X-trees, and let $\mathcal{R} = (\mathcal{P}'_1, \mathcal{P}'_2, \ldots, \mathcal{P}'_t, \mathcal{P}_t)$ be a cluster-tree collection resulting from applying the cluster reduction t times. Then*

$$h'(\mathcal{P}_0) = h'(\mathcal{P}_t) + \sum_{j=1}^{t} h'(\mathcal{P}'_j).$$

Proof. The proof is by induction on $t = |\mathcal{R}| - 1$. If $t = 1$, the result clearly follows from Theorem 2.5. Now let $t > 1$ and assume that the result holds for all cluster-tree collections $\mathcal{R}'$ with $|\mathcal{R}'| \leq t$. Let A_1 be the leaf set of all trees $\mathcal{T}_i \in \mathcal{P}'_1$ with $i \in \{1, 2, \ldots, n\}$. Then $\mathcal{R}_1 = (\mathcal{P}'_2, \mathcal{P}'_3, \ldots, \mathcal{P}'_t, \mathcal{P}_t)$ is a cluster-tree collection of the set $\mathcal{P}_1$ that has been obtained from $\mathcal{P}_0$ by replacing the pendant subtree having leaf set A_1 with a single leaf labeled l_1 for all $\mathcal{T}_i \in \mathcal{P}_0$. Since $|\mathcal{R}_1| < |\mathcal{R}|$, it follows from the induction assumption that

$$h'(\mathcal{P}_1) = h'(\mathcal{P}_t) + \sum_{j=2}^{t} h'(\mathcal{P}'_j)$$

and by Theorem 2.5 that

$$h'(\mathcal{P}_0) = h'(\mathcal{P}_1) + h'(\mathcal{P}'_1).$$

Combining both equations establishes the proof. □

Note that the result of Theorem 2.5 and Corollary 2.6 holds when the cluster reduction is applied to an arbitrarily large common cluster. For algorithmic purposes, it is of interest to combine the application of the cluster reduction with the subtree and chain reduction in a way that each tree pair of a cluster-tree collection has been reduced as much as possible by the other two reductions. Therefore, it is necessary to fit the cluster reduction into the framework of legitimate-agreement forests of minimum weight. This is easily achievable by considering *minimal* clusters only. Hence, each cluster intersects a weighted 2-chain in either both elements or neither.

We close this section by establishing two analogous corollaries for Theorem 2.5 and Corollary 2.6 that allow for applications of the cluster reduction in the context of legitimate-agreement forests.

Corollary 2.7. *Let $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n\}$ be a set of weighted rooted binary phylogenetic X-trees. Suppose that $A \subset X$ is a minimal cluster common to all trees of $\mathcal{P}$. Let $\mathcal{P}_A$ and $\mathcal{P}_a$ be the cluster-tree set and the cluster-reduced tree set of weighted rooted binary phylogenetic trees obtained from $\mathcal{P}$ by applying the cluster reduction once. Then*

$$f(\mathcal{P}) = f(\mathcal{P}_A) + f(\mathcal{P}_a).$$

Proof. The following proof can be established in exactly the same way as the proof of Theorem 2.5 by considering the weight of a forest instead of its cardinality. We first show that

$$f(\mathcal{P}) \leq f(\mathcal{P}_A) + f(\mathcal{P}_a).$$

To this end, let $\mathcal{F}_A$ be a legitimate-agreement forest for $\mathcal{P}_A$ of minimum weight with an associated set P_A of weighted 2-chains, and let $\mathcal{F}_a$ be such a forest for $\mathcal{P}_a$ with an associated set P_a. Suppose that $\mathcal{P}_a$ has been obtained from $\mathcal{P}$ by replacing the minimal rooted subtree induced by A with a single leaf labeled a for all $\mathcal{T}_i \in \mathcal{P}$ with $i \in \{1, 2, \ldots, n\}$. Let $\mathcal{S}_{j,a}$ be the unique tree in $\mathcal{F}_a$ with a vertex labeled a, and let $\mathcal{S}_{\rho,A}$ be the unique tree in $\mathcal{F}_A$ with a vertex labeled ρ. Then obtain the rooted binary tree $\mathcal{S}_{A,a}$ in one of the following two ways:

(a) If $\mathcal{S}_{j,a}$ is an isolated vertex, then obtain $\mathcal{S}_{A,a}$ from $\mathcal{S}_{\rho,A}$ by deleting the edge which is incident with the vertex labeled ρ and the vertex labeled ρ itself.

(b) Otherwise, obtain $\mathcal{S}_{A,a}$ by adjoining $\mathcal{S}_{\rho,A}$ to $\mathcal{S}_{j,a}$ via a new edge joining the vertices labeled ρ and a, removing the labels a and ρ, and suppressing any vertex of degree two apart from the root.

Consider that A is minimal. Then, since $\mathcal{F}_A$ and $\mathcal{F}_a$ are acyclic-agreement forests,

$$\mathcal{F} = ((\mathcal{F}_a \cup \mathcal{F}_A) - \{\mathcal{S}_{j,a}, \mathcal{S}_{\rho,A}\}) \cup \{\mathcal{S}_{A,a}\}$$

is such a forest for $\mathcal{P}$. Furthermore, every pair of isolated vertices that corresponds to a weighted 2-chain exists in $\mathcal{F}$ if and only if it exists in $\mathcal{F}_A$ or $\mathcal{F}_a$ and thus $\mathcal{F}$ is legitimate and $w_c(\mathcal{F}, P_A \cup P_a) = w_c(\mathcal{F}_A, P_A) + w_c(\mathcal{F}_a, P_a)$. With $|\mathcal{F}| = |\mathcal{F}_A| + |\mathcal{F}_a| - 1$, we have

$$f(\mathcal{P}_A)+f(\mathcal{P}_a) = |\mathcal{F}_A|-1+w_c(\mathcal{F}_A, P_A)+|\mathcal{F}_a|-1+w_c(\mathcal{F}_a, P_a) = |\mathcal{F}|-1+w_c(\mathcal{F}, P_A \cup P_a) \geq f(\mathcal{P}).$$

We next show that

$$f(\mathcal{P}) \geq f(\mathcal{P}_A) + f(\mathcal{P}_a).$$

Let $\mathcal{F}$ be a legitimate-agreement forest for $\mathcal{P}$ of minimum weight, and let P be a set of weighted 2-chains associated with $\mathcal{P}$. There are two cases to consider:

(i) there exists $\mathcal{S}_j \in \mathcal{F}$ such that $\mathcal{L}(\mathcal{S}_j) \cap A \neq \emptyset$ and $\mathcal{L}(\mathcal{S}_j) \cap ((X - A) \cup \{\rho\}) \neq \emptyset$, and
(ii) for all $\mathcal{S}_l \in \mathcal{F}$, either $\mathcal{L}(\mathcal{S}_l) \subseteq A$ or $\mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})$.

For case (i), we assume that $\mathcal{S}_j$ is such a tree in $\mathcal{F}$. Then, for all $i \in \{1, 2, \ldots, n\}$, the minimal rooted subtree of $\mathcal{T}_i \in \mathcal{P}$ that contains the label set of $\mathcal{S}_j$ includes the root of $\mathcal{T}_i|A$. Since $\mathcal{F}$ is an agreement forest, this implies that $\mathcal{S}_j$ is the unique such tree in $\mathcal{F}$. Then obtain $\mathcal{S}_{j,A}$ from $\mathcal{S}_j$ by adding a vertex labeled ρ at the end of a pendant edge adjoined to the root of $\mathcal{S}_j|(A \cap \mathcal{L}(\mathcal{S}_j))$. Additionally, obtain $\mathcal{S}_{j,a}$ from $\mathcal{S}_j$ by replacing the pendant subtree having label set $A \cap \mathcal{L}(\mathcal{S}_j)$ with a single leaf labeled a. As $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$,

$$\mathcal{F}_A = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\} \cup \{\mathcal{S}_{j,A}\}$$

is an acyclic-agreement forest for $\mathcal{P}_A$ and

$$\mathcal{F}_a = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})\} \cup \{\mathcal{S}_{j,a}\}$$

is such a forest for $\mathcal{P}_a$.

We now turn to case (ii). As $G_{\mathcal{F}}$ of $\mathcal{P}$ is acyclic, it follows that the subdigraph induced by the set $\{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\}$ does not contain any directed cycle. Hence, this subdigraph has a vertex, $\mathcal{S}_0$ say, with indegree zero. Let $\mathcal{S}_{0,\rho}$ be the tree obtained from $\mathcal{S}_0$ by adding a vertex labeled ρ at the end of a pendant edge adjoined to the original root of $\mathcal{S}_0$. Since $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{P}$, it is easily seen that

$$\mathcal{F}_A = (\{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq A\} - \{\mathcal{S}_0\}) \cup \{\mathcal{S}_{0,\rho}\}$$

is an acyclic-agreement forest for $\mathcal{P}_A$ and

$$\mathcal{F}_a = \{\mathcal{S}_l \in \mathcal{F} : \mathcal{L}(\mathcal{S}_l) \subseteq ((X - A) \cup \{\rho\})\} \cup \{a\}$$

is such a forest for $\mathcal{P}_a$, where the singleton $\{a\}$ is used to denote an isolated vertex labeled a.

In both cases (i) and (ii), let P_A be the set of weighted 2-chains associated with $\mathcal{P}_A$ such that P_A contains exactly each element of P that is a subset of A. Similarly, let P_a be the set of weighted 2-chains associated with $\mathcal{P}_a$ such that P_a contains exactly each element of P that is a subset of $(X - A) \cup \{\rho\}$. Since A is minimal, note that P_A and P_a are disjoint sets with $P = P_A \cup P_a$. Furthermore, every pair of isolated vertices that corresponds to a weighted 2-chain exists in $\mathcal{F}_A$ or $\mathcal{F}_a$ if and only if it exists in $\mathcal{F}$, and thus it follows that $\mathcal{F}_A$ and $\mathcal{F}_a$ are legitimate and $w_c(\mathcal{F}, P) = w_c(\mathcal{F}_A, P_A) + w_c(\mathcal{F}_a, P_a)$. Then with $|\mathcal{F}| = |\mathcal{F}_A| + |\mathcal{F}_a| - 1$, we can finally deduce that

$$f(\mathcal{P}) = |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) = |\mathcal{F}_A| - 1 + w_c(\mathcal{F}_A, P_A) + |\mathcal{F}_a| - 1 + w_c(\mathcal{F}_a, P_a) \geq f(\mathcal{P}_A) + f(\mathcal{P}_a).$$

Combining both parts of the proof gives the desired result. □

Corollary 2.8. *Let $\mathcal{P}_0$ be a set of weighted rooted binary phylogenetic X-trees with an associated set P of weighted 2-chains, and let $\mathcal{R} = (\mathcal{P}'_1, \mathcal{P}'_2, \ldots, \mathcal{P}'_t, \mathcal{P}_t)$ be a cluster-tree collection resulting from applying the cluster reduction t times to a minimal cluster. Then*

$$f(\mathcal{P}_0) = f(\mathcal{P}_t) + \sum_{j=1}^{t} f(\mathcal{P}'_j).$$

Proof. This corollary is an immediate consequence of Corollary 2.6 and Corollary 2.7. □

3 HybridNumber: A Reduction Algorithm for Hybridization

In this chapter, we describe a new reduction-based algorithm for computing the minimum number of hybridization events for two rooted binary phylogenetic trees on the same set of taxa. The algorithm, called HybridNumber, is based on the combinatorial framework, which is described in Chapter 2. HybridNumber is outlined in Section 3.2 and pseudocode is given in Appendix A.1. Although the two-tree problem is NP-hard (Bordewich and Semple, 2007a), HybridNumber always gives the exact solution and runs efficiently on many biological problems. In terms of their running time, a full range of instances is highlighted in Section 3.3, where we apply HybridNumber to a grass (*Poaceae*) data set. We end this chapter by giving some conclusions in Section 3.4.

3.1 Introduction

In the following, we restrict our attention to hybridization whose effect in evolution has been recognized for quite some time. For example, since the 1930's, botanists have suggested that the morphological variation in the New Zealand flora is due to hybridization (Allan, 1961). However, the computational task of determining how much hybridization has occurred has been a much more recent consideration. Assuming that we are given a collection of rooted phylogenetic trees on a set of present-day species that correctly represent the tree-like evolution of different genetic loci, an important step in the study of hybridization is to analyze the minimum number of hybridization events needed to explain the evolution of the species under consideration. As well as providing a lower bound on the number of such events, this smallest number also acts as an indicator for the extent to which hybridization has influenced the evolutionary history of the considered collection of present-day species.

Formalized mathematically, this fundamental problem is NP-hard even when the initial collection consists of two rooted binary phylogenetic trees (Bordewich and Semple, 2007a). Consequently, as a result of this computational difficulty, most current research considers the two-tree problem. There are now several algorithms focusing on this problem. However, all of these algorithms are either algorithms solving a restricted version of the problem (e.g. Hallett and Lagergren, 2001, Huson *et al.*, 2005, Nakhleh *et al.*, 2005b) or polynomial-time heuristics with no guarantee of the closeness to the exact solution (e.g. Nakhleh *et al.*, 2005a).

Here, we describe a newly implemented and exact algorithm for solving the two-tree problem (without any restrictions) based on the subtree, chain, and cluster reduction (see Sections 2.3 and 2.4) that makes use of similarities between the original two trees. It has recently been shown that the subtree and chain reductions are enough to 'kernalize' the problem and give a fixed-parameter tractable algorithm, where the parameter is the smallest number of hybridization events needed to explain the initial two trees (Bordewich and Semple, 2007b). This means that the algorithm runs efficiently when this smallest number is bounded. In other words, while the general problem is NP-hard, many biologically interesting instances of the problem may be solvable in a reasonable time even for a very large number of taxa as long as the number of hybridization events is relatively small. For further information on fixed-parameter tractability, we refer the interested reader to Downey and Fellows (1998). Additionally, the cluster reduction provides an extremely useful tool for breaking the problem into a number of smaller subproblems; all that is required is that the subtrees should have identical leaf sets, the topologies of the two subtrees can be completely different. However, there are going to be some instances for which HybridNumber will not return an answer in a reasonable time—in particular, instances that have a high level of hybridization and few similarities.

The algorithm HybridNumber has been implemented in Perl. The program expects two rooted binary phylogenetic trees on the same set of taxa as input, where each taxa needs to be represented as an integer value, and outputs the simplified trees after each application of the three reductions and the minimum number of hybridization events to explain the two initial trees. Full details of the algorithm described in this chapter can be found in Appendix A.1, where a pseudocode version is given. As HybridNumber is restricted to the two-tree problem, note that the hybridization number is equal to the number of hybridization vertices in a hybridization network that displays the two input trees. This is simply due to the fact, that the indegree of a vertex in such a network is at most two.

3.2 The Algorithm HybridNumber

Next, we briefly describe a combinatorial characterization of computing the minimum number of hybridization events $h(\mathcal{S}, \mathcal{T})$ for two rooted binary phylogenetic trees $\mathcal{S}$ and $\mathcal{T}$. This characterization underlies HybridNumber and finds the exact solution to Minimum Hybridization (see page 15). Loosely speaking, a *forest* of $\mathcal{S}$ (or $\mathcal{T}$) is a collection of non-overlapping rooted subtrees of $\mathcal{S}$ (or $\mathcal{T}$) whose disjoint union of leaf sets is X. An *agreement* forest $\mathcal{F}$ of $\mathcal{S}$ and $\mathcal{T}$ is a forest of both $\mathcal{S}$ and $\mathcal{T}$. Beginning with a hybridization network that displays $\mathcal{S}$ and $\mathcal{T}$, one way to obtain an agreement forest for $\mathcal{S}$ and $\mathcal{T}$ is by deleting each of the edges coming into every hybridization vertex (for a more detailed description, see Section 2.2). Biologically, the deleted edges correspond to different paths of genetic inheritance. Thus the fewer the number of hybridization vertices of such a network, the smaller the size of the resulting agreement forest for $\mathcal{S}$ and $\mathcal{T}$, where the size of a forest is the number of trees in the forest. On the other hand, if we are given an agreement forest for $\mathcal{S}$ and $\mathcal{T}$, then one can reverse this process to construct a hybridization network $\mathcal{H}$ that displays $\mathcal{S}$ and $\mathcal{T}$ provided the forest has a particular acyclicity property. This property excludes the possibility of circular inheritance which means that a vertex in $\mathcal{H}$ does not inherit genetic information from its own descendants, in which case $\mathcal{H}$ contains no directed cycles. An agreement forest with the acyclicity property is called *acyclic* (see Section 2.2). Theorem 2 of Baroni *et al.* (2005) shows that $h(\mathcal{S},\mathcal{T})$ is one less than the minimum size of an acyclic-agreement forest for $\mathcal{S}$ and $\mathcal{T}$.

The algorithm HybridNumber is based on the repeated use of three polynomial-time reductions (see Section 3.2.1). Essentially, each of these reductions preserves the hybridization number in some way. The subtree and chain reduction reduce the size of the problem instance, while the cluster reduction breaks the problem into a number of smaller and more tractable subproblems. An exhaustive search part on each of the smaller problems completes the algorithm. While it is likely that the general problem Minimum Hybridization has no polynomial-time solution, it would be interesting to see how one could speed up the exhaustive search part of HybridNumber. Improvements that are already implemented in this part of the algorithm are described in Section 3.2.2.

3.2.1 Reductions

In this subsection, we give a brief description of the three reductions and their effect on computing $h(\mathcal{S},\mathcal{T})$ for two rooted binary phylogenetic X-trees $\mathcal{S}$ and $\mathcal{T}$. Detailed explanations of these reductions can be found in Section 2.3 and 2.4 and they are illustrated in Figures 3.1, 3.2, and 3.3, respectively. Pseudocode for each of the three reductions is given in Appendix A.1.

Subtree reduction. Replace a maximal pendant subtree with at least two leaves that occurs identically in $\mathcal{S}$ and $\mathcal{T}$ by a single leaf with a new label. If $\mathcal{S}'$ and $\mathcal{T}'$ denote the resulting trees, then

$$h(\mathcal{S},\mathcal{T}) = h(\mathcal{S}',\mathcal{T}').$$

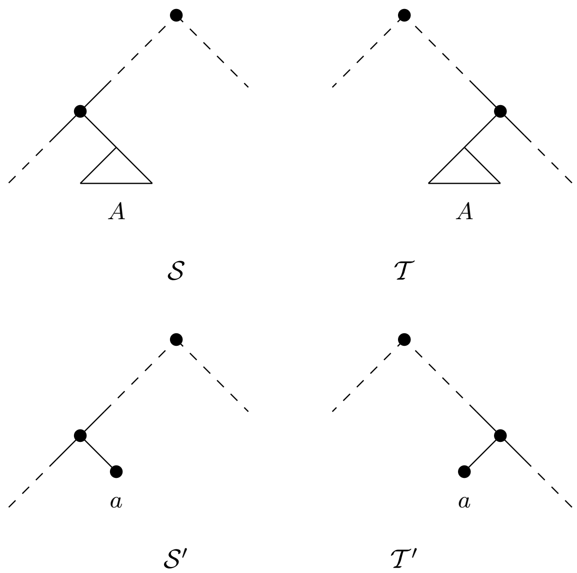


Figure 3.1: Two rooted binary phylogenetic trees $\mathcal{S}$ and $\mathcal{T}$ reduced under the subtree reduction. The triangle A indicates a maximal pendant subtree which is common to both trees and replaced by a new leaf labeled a in $\mathcal{S}'$ and $\mathcal{T}'$.

Chain reduction. Replace a maximal chain $(a_1, a_2, \ldots, a_m)$ with $m \geq 3$ that occurs identically and with the same orientation relative to the root in $\mathcal{S}$ and $\mathcal{T}$ by a 2-chain with new labels, a and b say, correctly orientated to preserve the direction of the chain. If the chain consists of m leaves, then assign the pair $\{a, b\}$ of new leaves weight $m - 2$. If $\mathcal{S}'$ and $\mathcal{T}'$ denote the resulting trees, then either

$$h(\mathcal{S}, \mathcal{T}) = h(\mathcal{S}', \mathcal{T}')$$

or

$$h(\mathcal{S}, \mathcal{T}) = h(\mathcal{S}', \mathcal{T}') + m - 2,$$

depending on whether a minimum-sized acyclic-agreement forest for $\mathcal{S}'$ and $\mathcal{T}'$ has the property that a and b are contained in the label set of one tree or not, respectively. In the case that a and b are not contained in the same label set, a and b label isolated vertices in the minimum-sized acyclic-agreement forest (Bordewich and Semple, 2007b). Due to Lemma 2.4, note that these are the only two cases need to be considered. The purpose of the weighting is to keep track of the number of leaves in the original chain when a and b label isolated vertices because then each of $\{a_1, a_2, \ldots, a_m\}$ labels an isolated vertex in a minimum-sized acyclic-agreement forest for $\mathcal{S}$ and $\mathcal{T}$. There is a slight complication here in that the reducing chain may contain consecutive pairs of leaves that have previously been involved in a chain reduction. In such cases, the pair $\{a, b\}$ of new leaves is assigned

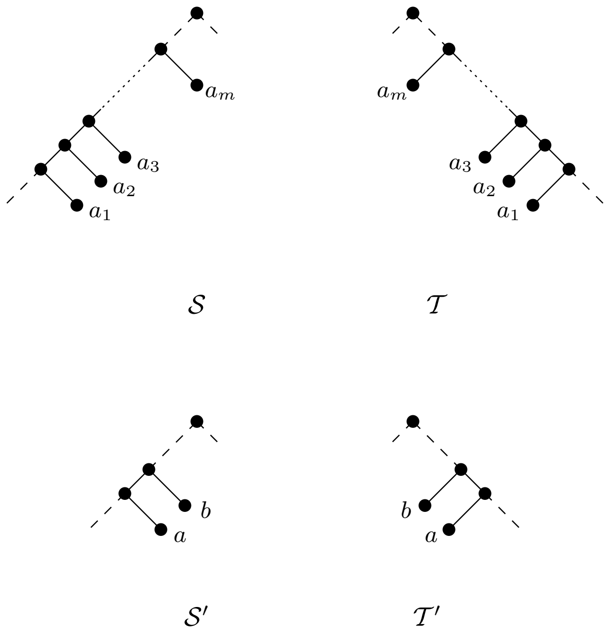


Figure 3.2: Two rooted binary phylogenetic trees $\mathcal{S}$ and $\mathcal{T}$ reduced under the chain reduction. Dotted lines indicate regions of the m-chain.

a weight that is the sum of the associated weights of these pairs and $m-2$. The effect on $h(\mathcal{S}, \mathcal{T})$ is a generalization of the previous outcome.

Cluster reduction. If A is a minimal cluster common to $\mathcal{S}$ and $\mathcal{T}$ and with at least two leaves, then replace $\mathcal{S}$ and $\mathcal{T}$ with two pairs of new trees. The *cluster-reduced tree pair*, $\mathcal{S}_1$ and $\mathcal{T}_1$ say, is obtained from $\mathcal{S}$ and $\mathcal{T}$ by replacing the subtree whose leaf set is A with a new label, while the *cluster-tree pair*, $\mathcal{S}_2$ and $\mathcal{T}_2$ say, contains the subtrees $\mathcal{S}|A$ and $\mathcal{T}|A$. The point of this is that

$$h(\mathcal{S}, \mathcal{T}) = h(\mathcal{S}_1, \mathcal{T}_1) + h(\mathcal{S}_2, \mathcal{T}_2).$$

Remark. Without going into details, the cluster reduction has a similar flavor to the "Decomposition Theorem" in Huson *et al.* (2005). This theorem describes a one-to-one correspondence between the overlapping cycles of an (unrooted) network $\mathcal{H}$, the connected components of the incompatibility graph of the splits generated by $\mathcal{H}$, and the netted components of the splits graph of the splits generated by $\mathcal{H}$. However, while this theorem yields an algorithm for minimizing the number of hybridization vertices amongst a restricted class of networks, it is important to note that it does not give a general strategy for minimizing this number amongst all hybridization networks as there is no guarantee that such a reduction leads to an optimal solution. In contrast, Baroni *et al.*

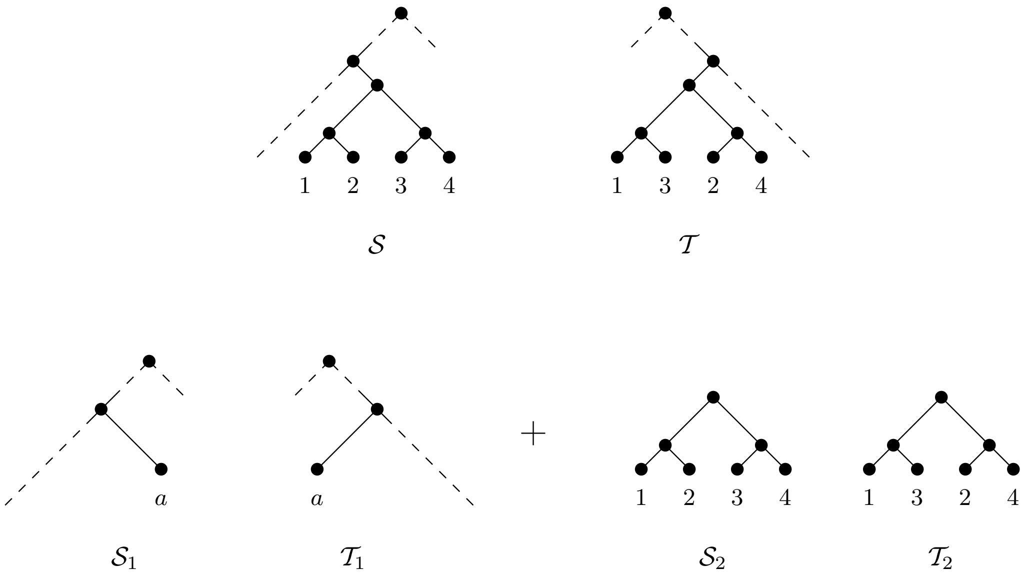


Figure 3.3: Two rooted binary phylogenetic trees $\mathcal{S}$ and $\mathcal{T}$ divided under the cluster reduction applied to $A = \{1, 2, 3, 4\}$. The hybridization number of $\mathcal{S}$ and $\mathcal{T}$ is the sum of the hybridization numbers of $\mathcal{S}_1$ and $\mathcal{T}_1$, and $\mathcal{S}_2$ and $\mathcal{T}_2$.

(2006) showed that such a strategy, in particular the cluster reduction, works for two trees. An analogous problem has also been posed by Gusfield and Bansal (2005) within the framework of population genetics.

Using the three reductions, the algorithm HybridNumber initially attempts to reduce the size of the problem instance as much as possible. It begins by repeatedly applying the subtree reduction where possible before applying the chain reduction in the same way. Once this is done, it finds the smallest common cluster of size at least two of the resulting trees and uses this cluster to perform a cluster reduction, thus replacing the pair of subtree-and-chain-reduced trees with two smaller pairs of trees. Putting aside the cluster-tree pair, the algorithm now repeats this process for the cluster-reduced tree pair. Eventually, no more reductions are possible and we are left with pairs of trees for which we exhaustively find each of their hybridization numbers. Because of the combinatorial characterization mentioned earlier, up to the weightings resulting from a chain reduction, this exhaustive process finds an acyclic-agreement forest of smallest size for each pair of trees. The sum of these sizes minus one for each such pair gives the hybridization number of the initial two trees.

3.2.2 Exhaustive Search Strategy

In this subsection, we describe some improvements that have been implemented to speed up the exhaustive search of HybridNumber. This is the computationally most intensive

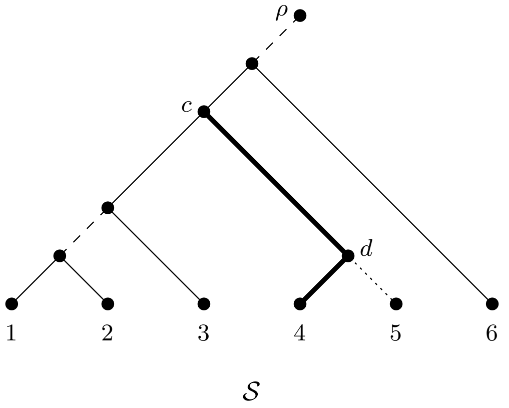


Figure 3.4: A rooted binary tree $\mathcal{S}$. Suppose that the 2-chain $(2, 3)$ reduces a strictly bigger chain. To calculate an appropriate acyclic-agreement forest for $\mathcal{S}$ and a second tree on the same label set that is omitted here, the dashed lines indicate edges that never get deleted. Furthermore, given that the two thicker edges $\{c, d\}$ and $\{d, 4\}$ are deleted, the dotted line indicates an edge that does not get deleted when it comes to deleting more than these two edges.

part of the algorithm and calculates the minimum number of hybridization events needed to explain the evolutionary history of a pair of rooted binary phylogenetic trees $\mathcal{S}$ and $\mathcal{T}$ that result from applying the three reductions as much as possible.

A first-up approach would be to exhaustively delete an increasing number of edges from $\mathcal{S}$ and $\mathcal{T}$, and then see if (i) the resulting forests, $\mathcal{F}_\mathcal{S}$ and $\mathcal{F}_\mathcal{T}$ say, are the same acyclic-agreement forests for $\mathcal{S}$ and $\mathcal{T}$ and (ii) for every 2-chain (a, b) reducing a strictly bigger chain, the two leaf labels a and b label isolated vertices or both are contained in a label set of one tree of $\mathcal{F}_\mathcal{S}$ and $\mathcal{F}_\mathcal{T}$, respectively. However, a much faster approach is to delete edges from just one of the trees, $\mathcal{S}$ say, to obtain a forest $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_k\}$ and then see if for all $i \in \{\rho, 1, 2, \dots, k\}$ the tree $\mathcal{S}_i$ is isomorphic to $\mathcal{T}|\mathcal{L}(\mathcal{S}_i)$ and if the collection

$$\{\mathcal{T}(\mathcal{L}(\mathcal{S}_i)) : i \in \{\rho, 1, 2, \dots, k\}\}$$

of trees is vertex-disjoint in $\mathcal{T}$. If no, then $\mathcal{F}$ is not an agreement forest for $\mathcal{S}$ and $\mathcal{T}$. On the other hand, if yes, then $\mathcal{F}$ is such a forest. Of course, one also needs to check if $\mathcal{F}$ is acyclic and if, for every 2-chain (a, b) reducing a strictly bigger chain, the two leaf labels a and b label isolated vertices or both are contained in a label set of one tree of $\mathcal{F}$.

Additionally, there are edges in $\mathcal{S}$ that never get deleted since, otherwise, the resulting forest does not have all desired properties. There are mainly two types of such edges:

(i) the edge incident with the root vertex labeled ρ and

(ii) for each 2-chain (a, b) reducing a strictly bigger chain, the edge separating the

Table 3.1: The *Poaceae* data set.

Locus	Sequence Origin	# Sequences	Alignment Length [nc]
ITS	nucleus	47	322
ndhF	chloroplast	65	2210
phyB	nucleus	40	1182
rbcL	chloroplast	37	1344
rpoC2	chloroplast	34	777
waxy	nucleus	19	773

parents of the leaves labeled a and b, respectively.

Item (i) is due to Lemma 2.2, whereas (ii) follows from the fact that a and b are both contained in the label set of one tree of $\mathcal{F}$ or a and b label isolated vertices in $\mathcal{F}$. Such edges, which never get deleted, are indicated by dashed lines for the example depicted in Figure 3.4. Moreover, no agreement forest for $\mathcal{S}$ and $\mathcal{T}$ contains an isolated internal vertex and, hence, we do not need to consider sets of edges to delete which contain three edges incident with one vertex. In general, after deleting an edge of $\mathcal{S}$, we always exclude those edges in $\mathcal{S}$ from getting deleted in a subsequent step whose deletion would result in a forest containing a tree with an empty label set. To see this point, consider Figure 3.4 and suppose that the two edges $\{c, d\}$ and $\{d, 4\}$ have been deleted in the rooted binary tree $\mathcal{S}$. Then each combination of edges to delete that additionally includes the edge $\{d, 5\}$ results in a forest containing a subtree that consists of the (unlabeled) single vertex d. Lastly, noting that a forest $\mathcal{F}$ can result from deleting different combinations of edges, we perform a well-ordered iteration through all edges of $\mathcal{S}$ to avoid that it is checked more than once if $\mathcal{F}$ is an appropriate forest for $\mathcal{S}$ and $\mathcal{T}$.

3.3 The Grass (*Poaceae*) Data Set

In this section, we describe an application of HybridNumber to a grass (*Poaceae*) data set. This data set was provided by the Grass Phylogeny Working Group (2001). Although the extent of hybridization is still discussed controversially (Arnold, 1997, Rieseberg *et al.*, 2003), the occurrence of such events in certain groups of plants is generally accepted. In 1996, Ellstrand *et al.* examined the frequency of spontaneous hybridization in five biosystematic floras and found that, in four of these floras, the *Poaceae* family is among the six families with the highest number of natural hybrids. Therefore, it is more likely that the conflicting signals in the data are due to hybridization rather than other factors and so it is an appropriate data set for our purposes.

Table 3.2: Results for the *Poaceae* data set.

Pairwise Combination		**# Taxa**	**Hybridization Number**	**Run Time**[a]
ndhF	*phyB*	40	14	11 h
ndhF	*rbcL*	36	13	11.8 h
ndhF	*rpoC2*	34	12	26.3 h
ndhF	*waxy*	19	9	320 s
ndhF	*ITS*	46	at least 15	2 d
phyB	*rbcL*	21	4	1 s
phyB	*rpoC2*	21	7	180 s
phyB	*waxy*	14	3	1 s
phyB	*ITS*	30	8	19 s
rbcL	*rpoC2*	26	13	29.5 h
rbcL	*waxy*	12	7	230 s
rbcL	*ITS*	29	at least 9	2 d
rpoC2	*waxy*	10	1	1 s
rpoC2	*ITS*	31	at least 10	2 d
waxy	*ITS*	15	8	620 s

[a] Run time on a 2000 MHz CPU, 2 GB RAM machine measured in seconds (s), hours (h), and days (d), respectively.

The *Poaceae* data set consists of sequence data for six different genetic loci: internal transcribed spacer of ribosomal DNA (*ITS*); NADH dehydrogenase, subunit F (*ndhF*); phytochrome B (*phyB*); ribulose 1,5-biphosphate carboxylase/oxygenase, large subunit (*rbcL*); RNA polymerase II, subunit β'' (*rpoC2*); and granule bound starch synthase I (*waxy*). A summary describing the sequence origin, the number of sequences for each locus, and the alignment length for each gene in the data set is given in Table 3.1.

For each loci, a rooted binary phylogenetic tree was reconstructed using the fastDNAmL program (Olsen *et al.*, 1994). These gene trees were supplied by Heiko Schmidt who has previously analyzed this data set (Schmidt, 2003). We applied HybridNumber to each of the 15 different pairwise combinations of gene trees, where, for each combination, we restricted the gene trees to taxa common to both. The size of the overlapping taxa set for each combination is given in the second column of Table 3.2.

Before detailing the contents of Table 3.2, we describe one particular application of HybridNumber that highlights the extent to which the reductions incorporated in HybridNumber can reduce the size of the problem instance. This application involves the two phylogenetic trees of the chloroplast sequence phytochrome B (*phyB*) and the nuclear sequence of the internal transcribed spacer of ribosomal DNA (*ITS*) which have an overlapping taxa set of 30 present-day species (see the row indicated by the gray background in Table 3.2). These two trees with the restricted taxa set are shown in Figure 3.5. To enable a reader-friendly presentation of both trees, we have replaced the correct species

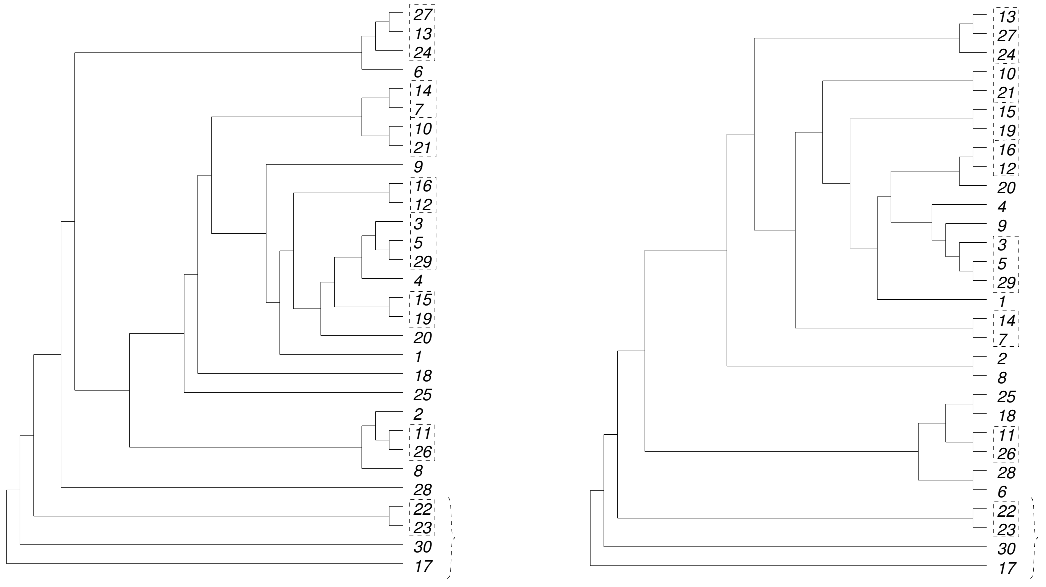


Figure 3.5: The input to HybridNumber for the combination *phyB* and *ITS*. Restricting to overlapping taxa, the tree resulting from the nuclear sequence *ITS* is on the left, while the tree resulting from the chloroplast sequence *phyB* is on the right. Labels in boxes denote the eight maximal pendant subtrees that are common to both trees, and the brace denotes a maximal chain once we have applied the subtree reductions.

names by numbers.

Taking the two trees in Figure 3.5 as input to HybridNumber, the algorithm initially finds all maximal pendant subtrees that are common to both trees (indicated by small boxes in Figure 3.5) and replaces each such subtree with a single leaf whose label is a concatenation of the subtree labels. Here, there are eight such subtrees. Next, HybridNumber checks for any identical chains of leaves in the two resulting trees. There is one such maximal chain of leaves and this is denoted by the brace in Figure 3.5. Applying the chain reduction, the labeling of the species which has evolved first is kept, while the labels of all other chain leaves are concatenated. The two trees resulting from the subtree and chain reductions are shown in Figure 3.6.

In the next step, the cluster reduction divides the problem into two smaller problems by searching for a minimal cluster of size at least two that is common to both trees in Figure 3.6. The first such cluster, indicated by square bracket A in Figure 3.6, is $\{(9), (12, 16), (3, 5, 29), (4), (15, 19), (20), (1)\}$ and the corresponding subtrees are shown at the top of Figure 3.7. At this point, HybridNumber has completed one iteration. Beginning with the two trees that result from replacing the cluster A with a single new leaf (a concatenation of the leave labels of the cluster induced subtree), the algorithm performs two further iterations. At the end of these two iterations, we obtain two more pairs of trees as indicated by the square brackets B and C in Figure 3.6. These two pairs

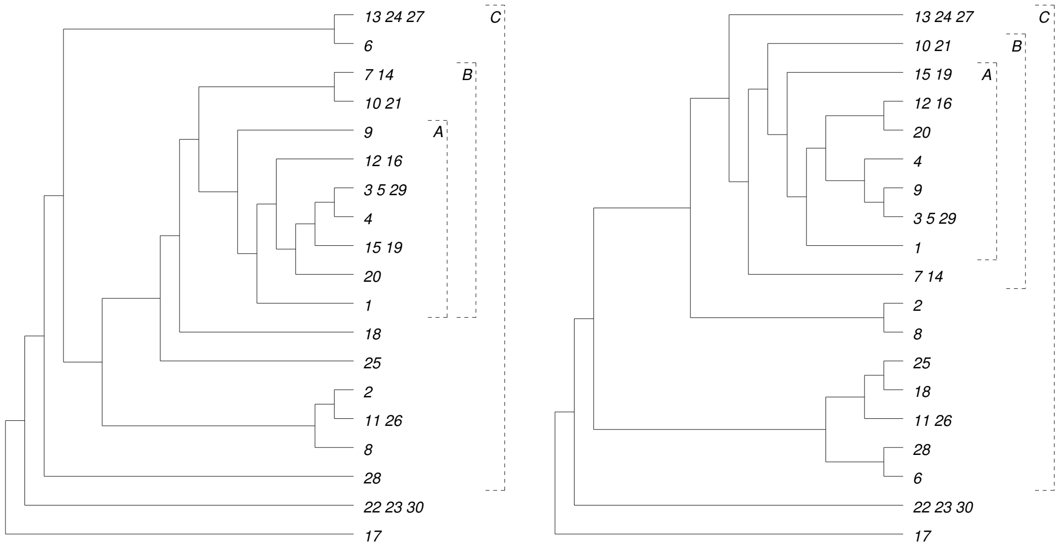


Figure 3.6: The two resulting phylogenetic trees (left: *ITS*, right: *phyB*) after repeated applications of the subtree reduction and then the chain reduction to the two trees in Figure 3.5. The three brackets A, B, and C indicate common minimal clusters.

are shown in Figure 3.7. At this stage, the original inputted trees have been reduced to two identical trees.

The final step in the algorithm is to exhaustively find the hybridization number of the three pairs of non-identical trees in Figure 3.7. The first pair has hybridization number 3, while the second and third pairs have hybridization numbers of 1 and 4, respectively. Adding the three numbers together gives the hybridization number of 8 for the two trees shown in Figure 3.5. The running time of this particular application is about 19 seconds (see Table 3.2). This is remarkably quick given that the two initial trees contain 30 taxa and the hybridization number is 8. As a comparison, we tried finding the hybridization number of these two trees without the three reductions. After one week, the algorithm was still running!

In Table 3.2, the results for all 15 pairs of trees are summarized. The running times are given in days, hours, or seconds. For eight pairs, HybridNumber calculates the hybridization number within a couple of minutes. Furthermore, the hybridization numbers of all but three pairs are found within a time span of two days. The successfully completed pairs contained up to 40 taxa and have hybridization numbers as high as 14. Those three pairs of trees for which the running time is given as 2 days in Table 3.2 are instances of the described NP-hard problem for which the algorithm will not return an answer in reasonable time. Nevertheless, we still have a lower bound on their respective hybridization numbers depending upon the intermediate result of the algorithm after two days at which time we stopped the algorithm. Lastly, the difference in running times

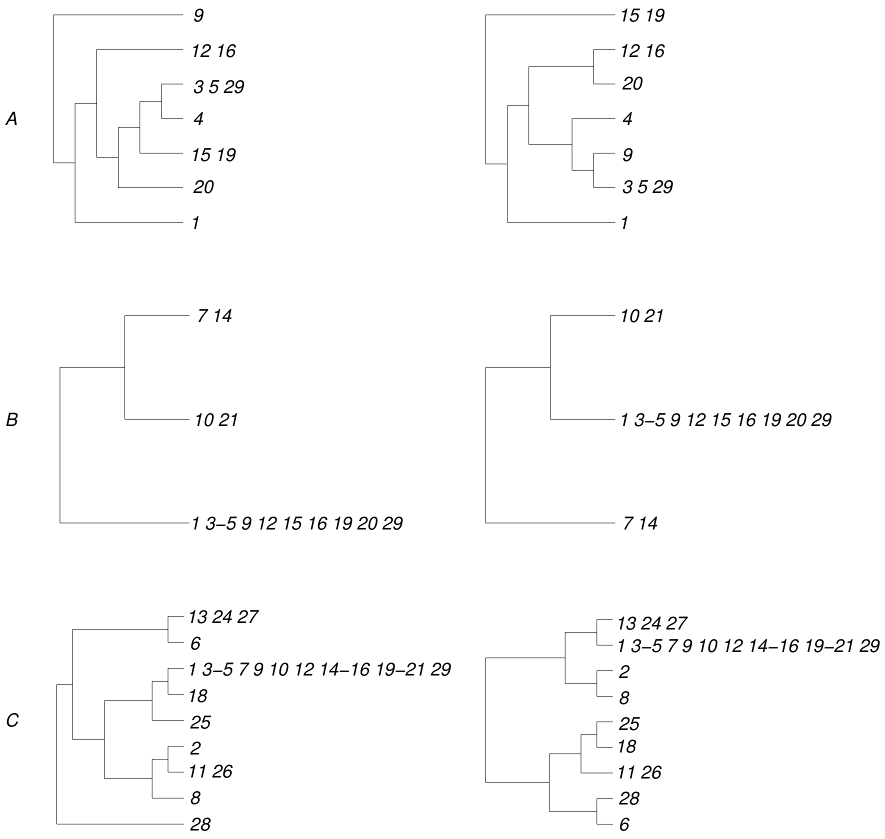


Figure 3.7: The three cluster-tree pairs corresponding to the clusters A, B, and C of Figure 3.6 for which HybridNumber (separately) calculates the minimum number of hybridization events (left: *ITS*, right: *phyB*).

of the various pairs is due to the extent of the reductions that we were able to use to reduce the problem instance and their hybridization number if the reductions have little effect. (The running time is dependent on the exhaustive search part of the algorithm as the reductions take a matter of seconds.) However, it is worth noting that it is always possible to reduce the number of leaves in a pair of trees to a linear function of its hybridization number (Bordewich and Semple, 2007b)—again highlighting the effectiveness of the reductions.

From a more biological point of view, it remains to remark that the hybridization numbers for the three gene tree pairs consisting of two trees which have both been reconstructed for a gene coded in the chloroplasts should be interpreted carefully since these organelles are inherited maternally. Hence, all genes that are coded in the chloroplasts have the same evolutionary history and, therefore, gene tree incongruence is more likely due to problems in the tree reconstruction method for example.

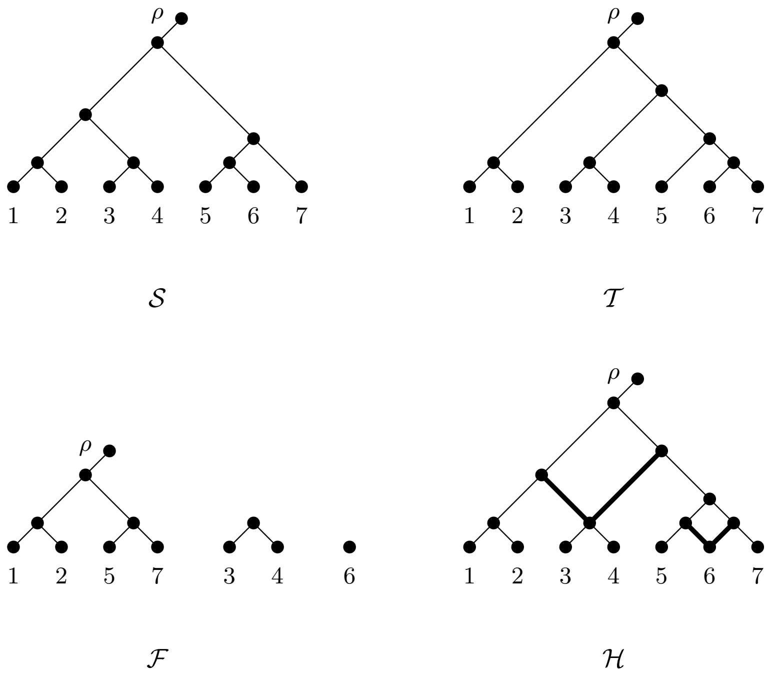


Figure 3.8: Top: Two rooted binary phylogenetic X-trees $\mathcal{S}$ and $\mathcal{T}$. Bottom: A minimum-sized acyclic-agreement forest $\mathcal{F}$ for $\mathcal{S}$ and $\mathcal{T}$ and a hybridization network $\mathcal{H}$ constructed from $\mathcal{F}$ by successively adjoining the trees in $\mathcal{F}$ via new (thicker) edges (for more details, see text).

3.4 Conclusions

Due to reticulate evolution, phylogenetic gene trees reconstructed for different genetic loci often reveal conflicting tree topologies, because processes like hybridization, HGT, and recombination are not tree-like. The extent to which such events occur is of increasing interest for many evolutionary studies.

In this chapter, we have described a newly implemented algorithm to calculate exactly the minimum number of hybridization events that explains two phylogenetic gene trees. Unlike previous algorithms, HybridNumber is not a heuristic, and its solution is not restricted in any way. Calculating this minimum number is a computationally hard problem, and so if the initial two gene trees only share a few similarities, then in many cases the exact calculation of the hybridization number is computationally infeasible. However, if the two gene trees share a number of common features—pendant subtrees, chains, or clusters—which is likely for many biological examples, the new algorithm performs remarkably well and the hybridization number can be found in reasonable time.

Note that HybridNumber calculates a lower bound for the number of hybridization events to explain the differences between two phylogenetic gene trees (assuming that hybridization is the only cause of incongruence between the two trees). It is possible that the real number of hybridization events that happened during the evolution of the collection of present-day species under consideration is underestimated. Indeed, it is possible that some hybridization events are never recognized. Nevertheless, the algorithm provides an important first step towards an understanding of the extent to which hybridization has influenced evolution.

Of course, in addition to computing the hybridization number of two rooted phylogenetic X-trees $\mathcal{S}$ and $\mathcal{T}$, one is also interested in constructing hybridization networks that realize this number. This can be efficiently done from a minimum-sized acyclic-agreement forest $\mathcal{F}$ for $\mathcal{S}$ and $\mathcal{T}$. Intuitively, one takes the tree in $\mathcal{F}$ containing the root of $\mathcal{S}$ and $\mathcal{T}$, and then systematically adjoins the rest of the trees in $\mathcal{F}$ as follows. At each step, adjoin a tree from $\mathcal{F}$ whose root is not the descendant (relative to either $\mathcal{S}$ or $\mathcal{T}$) of any tree not already adjoined. Each tree in $\mathcal{F}$ is adjoined with two new edges to the current hybridization network so that the resulting hybridization network $\mathcal{H}$ displays the appropriate restrictions of $\mathcal{S}$ and $\mathcal{T}$. Note that $\mathcal{H}$ is not necessarily uniquely defined. An example of such a construction is shown in Figure 3.8, where thicker edges indicate how trees of $\mathcal{F}$ have been joined together. An explicit algorithm called HybridNetwork that builds a network from an acyclic-agreement forest is given by Semple (2007).

4 How Deep is a Hybridization Event?

4.1 Introduction

The following extension of the HybridNumber algorithm was motivated by Peter Lockhart who posed the question whether hybridization events have occurred relatively recently or in the distant past. If the hybridization events are uniformly distributed over a hybridization network, these can possibly be interpreted as artifacts due to difficulties in gene tree reconstruction methods (Lockhart, 2007). Otherwise, if hybrid species are concentrated in some parts of the hybridization network, whereas other parts are completely tree-like, there is an increased probability that those events indicate true processes of hybridization. Most of all, this is the case for hybridization events between closely related recent species since such species are more likely to hybridize successfully and produce viable offspring (Mallet, 2007).

To approach the above mentioned question, we need to localize hybridization vertices in a hybridization network. Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are two weighted rooted binary phylogenetic X-trees. The construction of a hybridization network $\mathcal{H}$ displaying $\mathcal{T}$ and $\mathcal{T}'$ (see page 48) shows that each hybridization vertex in $\mathcal{H}$ corresponds to the root of a tree in a legitimate-agreement forest $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Given a hybridization network $\mathcal{H}$, we refer to a hybridization event as *non-deep* if the resulting hybrid species is a leaf in $\mathcal{H}$ and, otherwise, as a *deep* event. As an example, see Figure 1.2, where the leaf labeled B indicates a non-deep hybridization event, whereas the vertex $*$ corresponds to a species that has originated through a deep hybridization event. Note that the number of non-deep events is equal to the number of isolated vertices in $\mathcal{F}$. To compare the number of deep with the number of non-deep events, it is necessary to calculate all legitimate-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Here, we only consider legitimate-agreement forests of minimum weight because hybridization events are supposed to have occurred rarely during evolution and, hence, we focus on the minimum number of such events that is necessary to explain $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, since we have approached the question of inferring hybridization with a combinatorial framework, there can exist several legitimate-agreement forests of minimum weight for a given pair of trees. For example, all legitimate-agreement forests of minimum weight for the four-taxa trees $\mathcal{T}$ and $\mathcal{T}'$ are depicted in Figure 4.1.

In the remainder of this chapter, we describe four ways of how to obtain a forest $\mathcal{F}$ for a pair of weighted rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ by considering a legitimate-

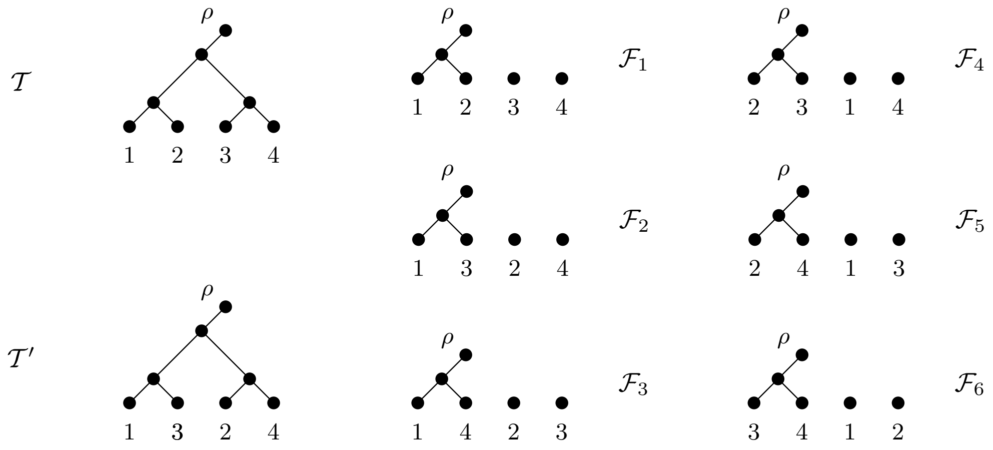


Figure 4.1: All six legitimate-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight.

agreement forest of minimum weight for two weighted rooted binary phylogenetic trees $\mathcal{T}_1$ and $\mathcal{T}_1'$ that have been obtained from $\mathcal{T}$ and $\mathcal{T}'$ by a single application of the subtree, chain, or cluster reduction. Depending on the reduction, we show that $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight (Section 4.2). Having these results, we then present a new algorithm BuildForest that calculates a legitimate-agreement forest $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight by considering such a forest for each cluster-tree pair (for the definition, see page 39) into which $\mathcal{T}$ and $\mathcal{T}'$ can be decomposed by repeatedly applying the subtree, chain, and cluster reduction (Section 4.3). Loosely speaking, given all legitimate-agreement forests of minimum weight for each cluster-tree pair, we show that all such forests for $\mathcal{T}$ and $\mathcal{T}'$ can be obtained by reversing the three reductions in an appropriate way. We close this chapter by applying this framework to a grass data set (Grass Phylogeny Working Group, 2001) and analyzing the ratio between deep and non-deep hybridization events (Section 4.4).

First, we obtain the following intuitive lemma.

Lemma 4.1. *A pair of rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ with $|X| \geq 2$ has a cluster A with $2 \leq |A| \leq |X|$ taxa in common.*

Proof. Since $\mathcal{T}$ and $\mathcal{T}'$ have leaf set X, the cluster $A = X$ is always a common cluster of $\mathcal{T}$ and $\mathcal{T}'$ such that $\mathcal{T} \cong \mathcal{T}|A$ and $\mathcal{T}' \cong \mathcal{T}'|A$, respectively. □

Let $\mathcal{T}_0$ and $\mathcal{T}_0'$ be two rooted binary phylogenetic X-trees with $|X| \geq 2$. We say that the two rooted binary phylogenetic trees $\mathcal{T}_1$ and $\mathcal{T}_1'$ have been obtained by a *reduction operation* from $\mathcal{T}_0$ and $\mathcal{T}_0'$ if they have been derived in one of the following ways:

(i) If $\mathcal{T}_0$ and $\mathcal{T}_0'$ have a subtree with label set A in common with $|A| \geq 2$, then $\mathcal{T}_1$ and

$\mathcal{T}_1'$ are obtained by applying the subtree reduction to a maximal subtree of $\mathcal{T}_0$ and $\mathcal{T}_0'$.

(ii) If $\mathcal{T}_0$ and $\mathcal{T}_0'$ have no such common subtree, but a chain with at least three leaves in common, then $\mathcal{T}_1$ and $\mathcal{T}_1'$ are obtained by applying the chain reduction to a maximal chain of $\mathcal{T}_0$ and $\mathcal{T}_0'$.

(iii) If $\mathcal{T}_0$ and $\mathcal{T}_0'$ have no such common chain, then $\mathcal{T}_1$ and $\mathcal{T}_1'$ is the cluster-reduced tree pair (for the definition, see page 39) after applying the cluster reduction to a minimal common cluster of $\mathcal{T}_0$ and $\mathcal{T}_0'$.

Due to Lemma 4.1, note that it is always possible to apply the cluster reduction.

4.2 Reduced Forests

Let $\mathcal{T}$ and $\mathcal{T}'$ be two weighted rooted binary phylogenetic X-trees, and let $\mathcal{F}$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight with an associated set P of weighted 2-chains. Furthermore, let $\mathcal{T}_1$ and $\mathcal{T}_1'$ be two trees obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying a single reduction operation. Depending on the reduction, we next describe how to obtain a forest $\mathcal{F}_1$ for $\mathcal{T}_1$ and $\mathcal{T}_1'$ from $\mathcal{F} = \{\mathcal{S}_\rho, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$. In particular, the upcoming four lemmas show that $\mathcal{F}_1$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight. Considering these results, we then prove the main theorem of this chapter in Section 4.3.

4.2.1 A Subtree-Reduced Forest

Let A be the label set of a maximal common subtree with $|A| \geq 2$. Since $\mathcal{T}|A$ and $\mathcal{T}'|A$ are isomorphic, there exists precisely one tree $\mathcal{S}_i \in \mathcal{F}$ with $i \in \{\rho, 1, 2, \ldots, k\}$ whose label set contains A and thus $A \subseteq \mathcal{L}(\mathcal{S}_i)$. We refer to a forest $\mathcal{F}_1$ that has been obtained from $\mathcal{F}$ by replacing the pendant subtree $\mathcal{S}_i|A$ with a single vertex labeled s as a *subtree-reduced forest*. Note that s labels an isolated vertex in $\mathcal{F}_1$ if $A = \mathcal{L}(\mathcal{S}_i)$. This construction is depicted in Figure 4.2.

Lemma 4.2. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two weighted rooted binary phylogenetic X-trees with a maximal common subtree whose label set is A, and let $\mathcal{F}$ be a legitimate-agreement forest of minimum weight for both trees. Furthermore, let $\mathcal{T}_1$ and $\mathcal{T}_1'$ be obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the subtree reduction to A. Then the subtree-reduced forest $\mathcal{F}_1$ obtained from $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight.*

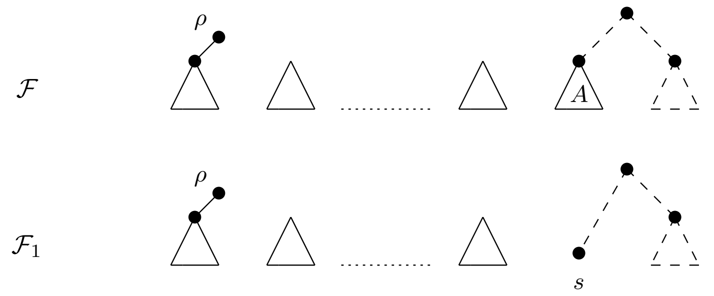


Figure 4.2: A subtree-reduced forest $\mathcal{F}_1$ obtained from a legitimate-agreement forest $\mathcal{F}$ of minimum weight. The triangle labeled A indicates a maximal common subtree.

Proof. Note first that A is maximal. Then, since $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that $\mathcal{F}_1$ is such a forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$. This implies that $w_c(\mathcal{F}, P) = w_c(\mathcal{F}_1, P)$. Furthermore, by construction of $\mathcal{F}_1$, we have $|\mathcal{F}| = |\mathcal{F}_1|$. Hence,

$$f(\mathcal{T}, \mathcal{T}') = w(\mathcal{F}) = |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) = |\mathcal{F}_1| - 1 + w_c(\mathcal{F}_1, P) = w(\mathcal{F}_1).$$

Due to Theorem 2.3, stating that $f(\mathcal{T}, \mathcal{T}') = f(\mathcal{T}_1, \mathcal{T}_1')$, we can now deduce that $\mathcal{F}_1$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight. □

4.2.2 A Chain-Reduced Forest

Let $(a_1, a_2, \ldots, a_m)$ be a maximal common m-chain with $m \geq 3$ that has been replaced with a 2-chain with leaves labeled a and b and with an associated weight of

$$w(\{a, b\}) = m - 2 + \sum_{\substack{\{a_i, a_j\} \in P; \\ a_i, a_j \in \{a_1, \ldots, a_m\}}} w(\{a_i, a_j\}).$$

By Lemma 2.4, two cases need to be considered to obtain $\mathcal{F}_1$ from $\mathcal{F}$:

(i) If there exists a tree $\mathcal{S}_i \in \mathcal{F}$ with $\{a_1, a_2, \ldots, a_m\} \subseteq \mathcal{L}(\mathcal{S}_i)$ and $i \in \{\rho, 1, 2, \ldots, k\}$, then $\mathcal{F}_1$ is obtained from $\mathcal{F}$ by replacing the m-chain in $\mathcal{S}_i$ by a 2-chain with new labels a and b (see Figure 4.3).

(ii) Otherwise, $\mathcal{F}_1$ is obtained from $\mathcal{F}$ by replacing precisely m isolated vertices whose labels partition $\{a_1, a_2, \ldots, a_m\}$ with two new such vertices labeled a and b, respectively (see Figure 4.4).

We refer to $\mathcal{F}_1$ as a *chain-reduced forest.*

Lemma 4.3. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two weighted rooted binary phylogenetic X-trees that have a*

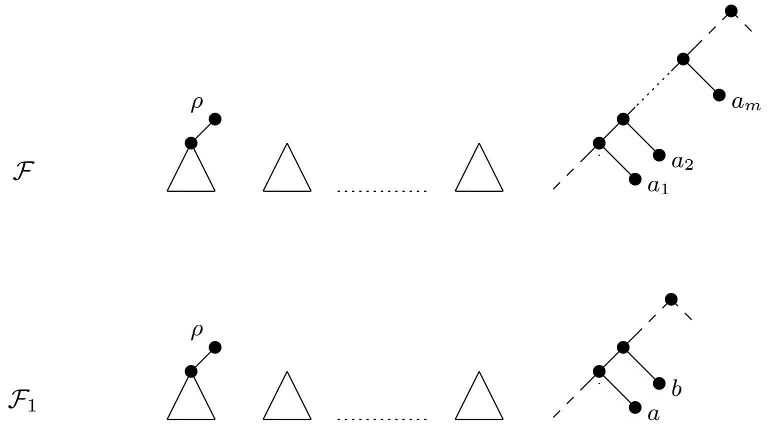


Figure 4.3: A chain-reduced forest $\mathcal{F}_1$ obtained from a legitimate-agreement forest $\mathcal{F}$ of minimum weight for which there exists one tree whose label set contains all labels $a_1, a_2, \ldots, a_m$ of a maximal common m-chain.

maximal m-chain $A = (a_1, a_2, \ldots, a_m)$ with $m \geq 3$ in common, and let $\mathcal{F}$ be a legitimate-agreement forest of minimum weight for both trees. Furthermore, let $\mathcal{T}_1$ and $\mathcal{T}_1'$ be obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the chain reduction to A. Then the chain-reduced forest $\mathcal{F}_1$ obtained from $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight.

Proof. Note first that A is maximal. Then, since $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that $\mathcal{F}_1$ is such a forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$. We first consider the case that $\mathcal{F}_1$ has been constructed according to (i) in the definition of a chain-reduced forest. By construction, we have $|\mathcal{F}| = |\mathcal{F}_1|$. Thus, with $w_c(\mathcal{F}, P) = w_c(\mathcal{F}_1, P)$, we can deduce that

$$f(\mathcal{T}, \mathcal{T}') = w(\mathcal{F}, P) = |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) = |\mathcal{F}_1| - 1 + w_c(\mathcal{F}_1, P) = w(\mathcal{F}_1).$$

We now turn to the construction of $\mathcal{F}_1$ according to (ii). Let $w(\{a, b\})$ be the weight of the 2-chain with leaves labeled a and b, and let $w_{c'}(\mathcal{F}, \mathcal{F}_1)$ be the weight of all pairs of isolated vertices labeled c and d that are contained in $\mathcal{F}$ and $\mathcal{F}_1$. To be precise,

$$w_{c'}(\mathcal{F}, \mathcal{F}_1) = \sum_{\substack{\{c,d\} \in P;\ c,d \text{ are} \\ \text{isolated in } \mathcal{F} \text{ and } \mathcal{F}_1}} w(\{c, d\}).$$

Note that $w_c(\mathcal{F}, P)$ and $w_{c'}(\mathcal{F}, \mathcal{F}_1)$ both do not include $w(\{a, b\})$ and thus $w(\{a, b\}) = m - 2 + w_c(\mathcal{F}, P) - w_{c'}(\mathcal{F}, \mathcal{F}_1)$. Since a and b label isolated vertices in $\mathcal{F}_1$, this implies that m isolated vertices in $\mathcal{F}$ have been replaced with two such vertices in $\mathcal{F}_1$ and thus

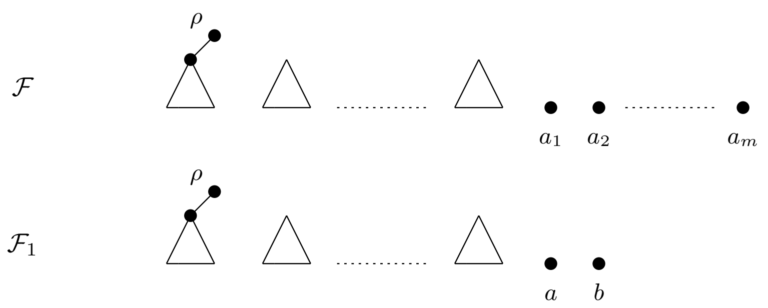


Figure 4.4: A chain-reduced forest $\mathcal{F}_1$ obtained from a legitimate-agreement forest $\mathcal{F}$ of minimum weight for which all elements of a maximal m-chain $(a_1, a_2, \ldots, a_m)$ label isolated vertices.

$|\mathcal{F}| = |\mathcal{F}_1| + m - 2$. Hence,

$$\begin{aligned}
w(\mathcal{F}_1) &= |\mathcal{F}_1| - 1 + w(\{a,b\}) + w_{c'}(\mathcal{F}, \mathcal{F}_1)\\
&= |\mathcal{F}_1| - 1 + m - 2 + w_c(\mathcal{F}, P) - w_{c'}(\mathcal{F}, \mathcal{F}_1) + w_{c'}(\mathcal{F}, \mathcal{F}_1)\\
&= |\mathcal{F}_1| - 1 + m - 2 + w_c(\mathcal{F}, P)\\
&= |\mathcal{F}| - 1 + w_c(\mathcal{F}, P)\\
&= w(\mathcal{F})\\
&= f(\mathcal{T}, \mathcal{T}').
\end{aligned}$$

Due to the fact that $f(\mathcal{T}, \mathcal{T}') = f(\mathcal{T}_1, \mathcal{T}_1')$ (see Theorem 2.3), we deduce that $\mathcal{F}_1$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight in both cases (i) and (ii). This completes the proof of the lemma. □

4.2.3 A Cluster-Reduced Forest and a Cluster-Pair Forest

Let A be the label set of a minimal common cluster with $|A| \geq 2$. There can be at most one tree $S_m \in \mathcal{F}$ such that $\mathcal{L}(\mathcal{S}_m) \cap A \neq \emptyset$ and $\mathcal{L}(\mathcal{S}_m) \cap ((X - A) \cup \{\rho\}) \neq \emptyset$. In the following, we refer to $\mathcal{S}_m$ as a *mixed tree.* For all other trees $\mathcal{S}_i$ in the forest (or all trees if there does not exist a mixed tree) either $\mathcal{L}(\mathcal{S}_i) \subseteq A$ or $\mathcal{L}(\mathcal{S}_i) \subseteq ((X - A) \cup \{\rho\})$.

Furthermore, let $\mathcal{N}_c \subset \mathcal{F}$ be the set of trees whose label sets are all subsets of A, and let $\mathcal{N}_d \subset \mathcal{F}$ be the set of trees whose label sets are all subsets of $(X - A) \cup \{\rho\}$. Note that neither $\mathcal{N}_c$ nor $\mathcal{N}_d$ contains the mixed tree $\mathcal{S}_m$ if such a tree exists in $\mathcal{F}$.

We first consider the cluster-reduced tree pair $\mathcal{T}_1$ and $\mathcal{T}_1'$ and obtain $\mathcal{F}_1$ from $\mathcal{F}$ depending on whether a mixed tree exists in $\mathcal{F}$:

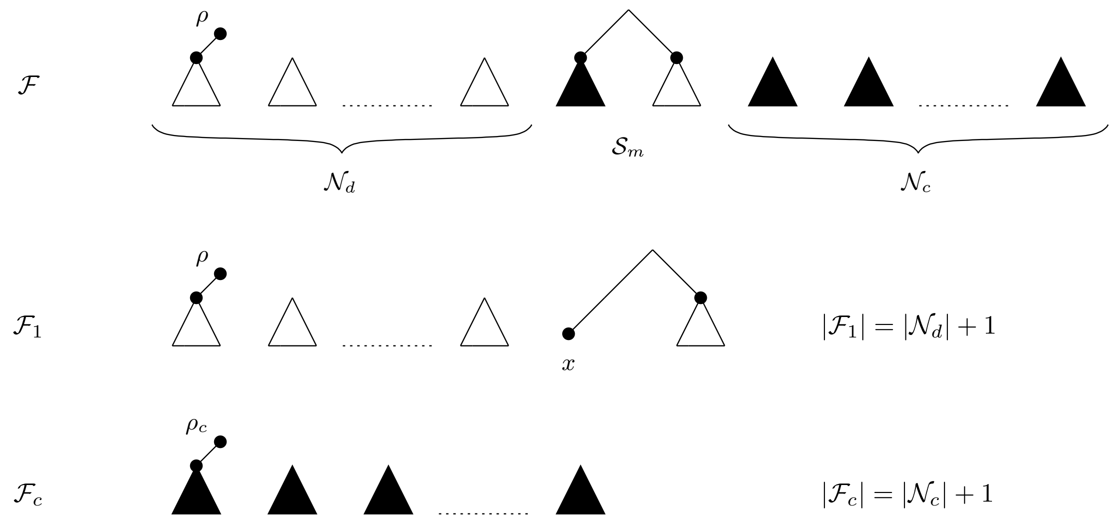


Figure 4.5: A cluster-reduced forest $\mathcal{F}_1$ and a cluster-pair forest $\mathcal{F}_c$ both obtained from a legitimate-agreement forest $\mathcal{F}$ of minimum weight that contains a mixed tree $\mathcal{S}_m$.

(i) If there exists a mixed tree $\mathcal{S}_m \in \mathcal{F}$, the minimal pendant subtree $\mathcal{S}_m|(\mathcal{L}(\mathcal{S}_m) \cap A)$ is replaced with a single vertex labeled x and all trees $\mathcal{S}_i \in \mathcal{N}_c$ are deleted to obtain $\mathcal{F}_1$ (see Figure 4.5).

(ii) Otherwise, the forest $\mathcal{F}_1$ is obtained by replacing all trees $\mathcal{S}_i \in \mathcal{N}_c$ with an isolated vertex labeled x (see Figure 4.6).

We refer to $\mathcal{F}_1$ as a *cluster-reduced forest.*

Depending on whether a mixed tree exists in $\mathcal{F}$, we next obtain a forest $\mathcal{F}_c$ for the cluster-tree pair $\mathcal{T}|A$ and $\mathcal{T}'|A$:

(i) If there exists a mixed tree $\mathcal{S}_m \in \mathcal{F}$, the forest $\mathcal{F}_c$ is obtained from $\mathcal{F}$ by replacing $\mathcal{S}_m$ with $\mathcal{S}_m|(\mathcal{L}(\mathcal{S}_m) \cap A)$, adding a vertex labeled ρ_c at the end of a pendant edge adjoined to the root of the resulting tree, and deleting all trees $\mathcal{S}_i \in \mathcal{N}_d$ (see Figure 4.5).

(ii) Otherwise, $\mathcal{F}_c$ is obtained from $\mathcal{F}$ by deleting all trees $\mathcal{S}_i \in \mathcal{N}_d$ and adding a vertex labeled ρ_c at the end of a pendant edge adjoined to the root of the tree that corresponds to a vertex whose indegree is zero in the subdigraph of $G_{\mathcal{F}}$ induced by the set $\mathcal{N}_c$ (see Figure 4.6).

We refer to $\mathcal{F}_c$ as a *cluster-pair forest.*

Lemma 4.4. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two weighted rooted binary phylogenetic X-trees with a minimal common cluster A, where $|A| \geq 2$, and let $\mathcal{F}$ be a legitimate-agreement forest of minimum weight for both trees. Furthermore, let $\mathcal{T}_1$ and $\mathcal{T}_1'$ be the cluster-reduced tree pair* *obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the cluster reduction to A. Then the cluster-reduced forest $\mathcal{F}_1$ obtained from $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight.*

Proof. Let P be the set of weighted 2-chains associated with $\mathcal{T}$ and $\mathcal{T}'$, and let P_1 be such a set associated with $\mathcal{T}_1$ and $\mathcal{T}_1'$ obtained from P by deleting those elements that contain an element of A. Since A is a minimal cluster, note that A intersects a weighted 2-chain in either both elements or neither. Thus we have $w_c(\mathcal{F}, P) = w_c(\mathcal{F}_1, P_1) + w_c(\mathcal{N}_c, P - P_1)$. If a mixed tree exists in $\mathcal{F}$, we have $|\mathcal{F}| = |\mathcal{N}_c| + |\mathcal{N}_d| + 1$ and otherwise $|\mathcal{F}| = |\mathcal{N}_c| + |\mathcal{N}_d|$. Since $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that $\mathcal{F}_1$ is such a forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ with $|\mathcal{F}_1| = |\mathcal{N}_d| + 1$ (see Figures 4.5 and 4.6). To show that this forest is also of minimum weight, suppose that there exists a legitimate-agreement forest $\mathcal{F}_1^*$ for $\mathcal{T}_1$ and $\mathcal{T}_1'$ such that $w(\mathcal{F}_1^*) < w(\mathcal{F}_1)$ or, equivalently, $|\mathcal{F}_1^*| - 1 + w_c(\mathcal{F}_1^*, P_1) < |\mathcal{F}_1| - 1 + w_c(\mathcal{F}_1, P_1)$. Depending on whether x labels an isolated vertex in $\mathcal{F}_1^*$ and whether a mixed tree exists in $\mathcal{F}$, four cases need to be considered. For each case, we obtain a forest $\mathcal{F}^*$ from $\mathcal{F}_1^*$ by reversing the process with which $\mathcal{F}_1$ has been obtained from $\mathcal{F}$ and show that $\mathcal{F}^*$ contradicts the optimality of $\mathcal{F}$.

(i) $\mathcal{F}_1^*$ contains an isolated vertex labeled x and $\mathcal{F}$ contains a mixed tree $\mathcal{S}_m$.
To obtain a forest $\mathcal{F}^*$ from $\mathcal{F}_1^*$, the isolated vertex labeled x is replaced with $\mathcal{S}_m|(\mathcal{L}(\mathcal{S}_m) \cap A)$ and all trees of $\mathcal{N}_c$ are added. Thus

$$\begin{aligned} w(\mathcal{F}^*) &= |\mathcal{F}_1^*| - 1 + w_c(\mathcal{F}_1^*, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &< |\mathcal{F}_1| - 1 + w_c(\mathcal{F}_1, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &= |\mathcal{N}_d| - 1 + 1 + w_c(\mathcal{F}_1, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &= |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) \\ &= w(\mathcal{F}). \end{aligned}$$

(ii) $\mathcal{F}_1^*$ contains an isolated vertex labeled x and $\mathcal{F}$ does not contain a mixed tree $\mathcal{S}_m$.
We can derive $\mathcal{F}^*$ from $\mathcal{F}_1^*$ by replacing the isolated vertex labeled x with all trees of $\mathcal{N}_c$.

As a result, we have

$$\begin{aligned} w(\mathcal{F}^*) &= |\mathcal{F}_1^*| - 2 + w_c(\mathcal{F}_1^*, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &< |\mathcal{F}_1| - 2 + w_c(\mathcal{F}_1, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &= |\mathcal{N}_d| - 1 + w_c(\mathcal{F}_1, P_1) + |\mathcal{N}_c| + w_c(\mathcal{N}_c, P - P_1) \\ &= |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) \\ &= w(\mathcal{F}). \end{aligned}$$

(iii) $\mathcal{F}_1^*$ does not contain an isolated vertex labeled x and $\mathcal{F}$ contains a mixed tree $\mathcal{S}_m$.
Since there does not exist an isolated vertex labeled x in $\mathcal{F}_1^*$, there is some $\mathcal{S}_i \in \mathcal{F}_1^*$ whose label set contains x and at least one other label l with $l \in ((X - A)\,\{\rho\})$. Then the forest $\mathcal{F}^*$ can be obtained from $\mathcal{F}_1^*$ by joining the vertex labeled x and the root of $\mathcal{S}_m|(\mathcal{L}(\mathcal{S}_m) \cap A)$ via a new edge, removing x, suppressing any vertices of degree two apart from the root, and adding all trees of $\mathcal{N}_c$. Summing up, we get the same inequality as in case (i).

(iv) $\mathcal{F}_1^*$ does not contain an isolated vertex labeled x and $\mathcal{F}$ does not contain a mixed tree $\mathcal{S}_m$.
Since $G_{\mathcal{F}}$ is acyclic, there exists a tree $\mathcal{S}_i \in \mathcal{N}_c$ whose corresponding vertex in the subdigraph of $G_{\mathcal{F}}$ induced by the set $\mathcal{N}_c$ has indegree zero. Then the forest $\mathcal{F}^*$ can be obtained from $\mathcal{F}_1^*$ by joining the vertex labeled x with the root of $\mathcal{S}_i$ via a new edge, removing x, suppressing any vertices of degree two apart from the root, and adding all trees of $\mathcal{N}_c - \{\mathcal{S}_i\}$. Hence, we get the same inequality as in case (ii).

In all four cases, we have $w(\mathcal{F}^*) < w(\mathcal{F})$. Since $f(\mathcal{T}, \mathcal{T}') = w(\mathcal{F})$, this contradicts the optimality of $\mathcal{F}$. As a result, we can conclude that $\mathcal{F}_1$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight. □

Lemma 4.5. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two weighted rooted binary phylogenetic X-trees with a minimal common cluster A, where $|A| \geq 2$, and let $\mathcal{F}$ be a legitimate-agreement forest of minimum weight for both trees. Then, applying the cluster reduction to A, the cluster-pair forest $\mathcal{F}_c$ obtained from $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ of minimum weight.*

Proof. Let P be the set of weighted 2-chains associated with $\mathcal{T}$ and $\mathcal{T}'$, and let P_c be such a set associated with $\mathcal{T}|A$ and $\mathcal{T}'|A$ obtained from P by deleting those elements

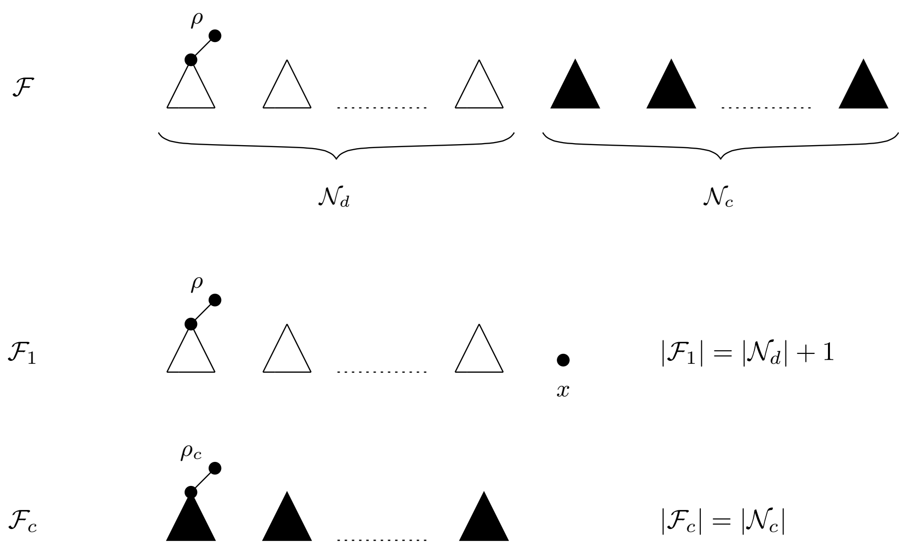


Figure 4.6: A cluster-reduced forest $\mathcal{F}_1$ and a cluster-pair forest $\mathcal{F}_c$ both obtained from a legitimate-agreement forest $\mathcal{F}$ of minimum weight that contains no mixed tree.

that contain an element of $(X - A) \cup \{\rho\}$. Since A is a minimal cluster, note that A intersects a weighted 2-chain in either both elements or neither. Thus we have $w_c(\mathcal{F}, P) = w_c(\mathcal{F}_c, P_c) + w_c(\mathcal{N}_d, P - P_c)$. Since $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that $\mathcal{F}_c$ is such a forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$. To show that this forest is also of minimum weight, suppose that there is a legitimate-agreement forest $\mathcal{F}_c^*$ for $\mathcal{T}|A$ and $\mathcal{T}'|A$ such that $w(\mathcal{F}_c^*) < w(\mathcal{F}_c)$ or, equivalently, $|\mathcal{F}_c^*| - 1 + w_c(\mathcal{F}_c^*, P_c) < |\mathcal{F}_c| - 1 + w_c(\mathcal{F}_c, P_c)$. Let $\mathcal{S}_{p_c}$ be the tree in $\mathcal{F}_c^*$ whose label set contains ρ_c. Depending on whether a mixed tree exists in $\mathcal{F}$, two cases need to be considered. For each case, we obtain a forest $\mathcal{F}^*$ from $\mathcal{F}_c^*$ by reversing the process with which $\mathcal{F}_c$ has been obtained from $\mathcal{F}$. Afterwards, we show that $\mathcal{F}^*$ contradicts the optimality of $\mathcal{F}$.

(i) $\mathcal{F}$ contains a mixed tree $\mathcal{S}_m$.

Since $\mathcal{F}$ contains a mixed tree $\mathcal{S}_m$, we have $|\mathcal{F}| = |\mathcal{N}_c| + |\mathcal{N}_d| + 1$ and $|\mathcal{F}_c| = |\mathcal{N}_c| + 1$ (see Figure 4.5). Let $\mathcal{S}'_m$ be obtained from $\mathcal{S}_m$ by replacing $\mathcal{S}_m|(\mathcal{L}(\mathcal{S}_m) \cap A)$ with a new leaf labeled x. Then obtain $\mathcal{F}^*$ from $\mathcal{F}_c^*$ by joining $\mathcal{S}'_m$ and $\mathcal{S}_{p_c}$ via a new edge connecting the vertices labeled x and ρ_c, removing both labels, suppressing any vertices of degree two apart from the root, and adding all trees of $\mathcal{N}_d$. Thus

$$\begin{aligned}
w(\mathcal{F}^*) &= |\mathcal{F}_c^*| - 1 + w_c(\mathcal{F}_c^*, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\
&< |\mathcal{F}_c| - 1 + w_c(\mathcal{F}_c, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\
&= |\mathcal{N}_c| + w_c(\mathcal{F}_c, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\
&= |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) \\
&= w(\mathcal{F}).
\end{aligned}$$

(ii) $\mathcal{F}$ does not contain a mixed tree $\mathcal{S}_m$.
In such a case, we $|\mathcal{F}| = |\mathcal{N}_c| + |\mathcal{N}_d|$ and $|\mathcal{F}_c| = |\mathcal{N}_c|$ (see Figure 4.6). Then obtain $\mathcal{F}^*$ from $\mathcal{F}_c^*$, by deleting the vertex labeled ρ_c and the edge incident with this vertex from $\mathcal{S}_{p_c}$ and adding all trees of $\mathcal{N}_d$. Hence, we have

$$\begin{aligned} w(\mathcal{F}^*) &= |\mathcal{F}_c^*| - 1 + w_c(\mathcal{F}_c^*, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\ &< |\mathcal{F}_c| - 1 + w_c(\mathcal{F}_c, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\ &= |\mathcal{N}_c| - 1 + w_c(\mathcal{F}_c, P_c) + |\mathcal{N}_d| + w_c(\mathcal{N}_d, P - P_c) \\ &= |\mathcal{F}| - 1 + w_c(\mathcal{F}, P) \\ &= w(\mathcal{F}). \end{aligned}$$

In both cases, we have $w(\mathcal{F}^*) < w(\mathcal{F})$. Since $f(\mathcal{T}, \mathcal{T}') = w(\mathcal{F})$, this contradicts the optimality of $\mathcal{F}$. As a result, we can conclude that $\mathcal{F}_c$ is a legitimate-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ of minimum weight. □

4.3 The Algorithm BuildForest

In this section, we present the algorithm BuildForest that, basically, describes how one can obtain a legitimate-agreement forest of minimum weight for a pair of trees by considering such a forest for each cluster-tree pair into which the initial tree pair can be broken down by repeatedly applying the reduction operation. A proof of correctness for BuildForest is given in Theorem 4.6.

First, we need some further definitions. Let $\mathcal{T}_0$ and $\mathcal{T}_0'$ be two weighted rooted binary phylogenetic X-trees. Furthermore, let

$$\mathcal{R} = (\{\mathcal{T}_1, \mathcal{T}_1'\}, \{\mathcal{T}_2, \mathcal{T}_2'\}, \ldots, \{\mathcal{T}_l, \mathcal{T}_l'\})$$

be a tuple of tree pairs such that the following two properties are satisfied:

(i) for all $i \in \{1, 2, \ldots, l\}$, the trees $\mathcal{T}_i$ and $\mathcal{T}_i'$ are obtained from $\mathcal{T}_{i-1}$ and $\mathcal{T}_{i-1}'$, respectively, by applying a single reduction operation and

(ii) for all $i \in \{0, 1, \ldots, l-1\}$, the trees $\mathcal{T}_i$ and $\mathcal{T}_i'$ are not isomorphic.

We refer to $\mathcal{R}$ as a *tuple of reduced tree pairs* for $\mathcal{T}_0$ and $\mathcal{T}_0'$. Suppose that $l' \leq l$ tree pairs of $\mathcal{R}$ have been obtained by applying the cluster reduction to a minimal common cluster. We

$$\begin{tikzcd}
\mathcal{T}_0, \mathcal{T}_0' \arrow[r, "\mathsf{SR}"] & \mathcal{T}_1, \mathcal{T}_1' \arrow[r, "\mathsf{ClR}"] \arrow[dr] & \mathcal{T}_2, \mathcal{T}_2' \arrow[r, "\mathsf{ChR}"] & \mathcal{T}_3, \mathcal{T}_3' \arrow[r, "\mathsf{ClR}"] \arrow[dr] & \mathcal{T}_4, \mathcal{T}_4' \arrow[r, "\mathsf{ClR}"] \arrow[dr] & \mathcal{T}_5, \mathcal{T}_5' \arrow[r, "\mathsf{SR}"] & \mathcal{T}_6, \mathcal{T}_6' \arrow[dd] \\
 & & \mathcal{C}_1, \mathcal{C}_1' \arrow[d] & & \mathcal{C}_2, \mathcal{C}_2' \arrow[d] & \mathcal{C}_3, \mathcal{C}_3' \arrow[d] & \\
 & & \mathcal{F}_1 & & \mathcal{F}_2 & \mathcal{F}_3 & \mathcal{F}_4
\end{tikzcd}$$

Figure 4.7: An example of how to obtain the tuple $\mathcal{R} = (\{\mathcal{T}_1, \mathcal{T}_1'\}, \{\mathcal{T}_2, \mathcal{T}_2'\}, \ldots, \{\mathcal{T}_6, \mathcal{T}_6'\})$ of reduced tree pairs for two weighted rooted binary phylogenetic X-trees $\mathcal{T}_0$ and $\mathcal{T}_0'$ by applying six reduction operations. Since the cluster reduction has been applied three times (resulting in the cluster-reduced tree pairs $\mathcal{T}_i$ and $\mathcal{T}_i'$ with $i \in \{2, 4, 5\}$, and the cluster-tree pairs $\mathcal{C}_j$ and $\mathcal{C}_j'$ with $j \in \{1, 2, 3\}$), note that we have $l' = 3$ and that $(\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \mathcal{F}_4)$ is a collection of forests for $\mathcal{R}$. The first three forests are legitimate-agreement forests for $\mathcal{C}_j$ and $\mathcal{C}_j'$ of minimum weight, while $\mathcal{F}_4$ is such a forest for the last tree pair of $\mathcal{R}$. (SR: subtree reduction, ChR: chain reduction, ClR: cluster reduction.)

call $(\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_{l'+1})$ a *collection of forests* for $\mathcal{R}$, where $\mathcal{F}_{l'+1}$ is a legitimate-agreement forest for $\mathcal{T}_l$ and $\mathcal{T}_l'$ of minimum weight and each $\mathcal{F}_i$ with $i \in \{1, 2, \ldots, l'\}$ is such a forest for the cluster-tree pair resulting from the i^{th} cluster reduction that has been applied to a minimal cluster in the course of all l reduction operations. An explicit example of how to obtain a tuple $\mathcal{R}$ of reduced tree pairs and an associated collection of forests is given in Figure 4.7, where two weighted rooted binary phylogenetic X-trees $\mathcal{T}_0$ and $\mathcal{T}_0'$ are broken down by a repeated applications of the reduction operation.

Next, we describe the algorithm BuildForest that calculates a legitimate-agreement forest of minimum weight for two weighted rooted binary phylogenetic X-trees if a corresponding tuple of reduced tree pairs and a collection of forests are given.

Algorithm: BuildForest
Input: Two weighted rooted binary phylogenetic X-trees $\mathcal{T}_0$ and $\mathcal{T}_0'$, a tuple of reduced tree pairs $\mathcal{R} = (\{\mathcal{T}_1, \mathcal{T}_1'\}, \{\mathcal{T}_2, \mathcal{T}_2'\}, \ldots, \{\mathcal{T}_l, \mathcal{T}_l'\})$ for $\mathcal{T}_0$ and $\mathcal{T}_0'$, and a collection of forests $(\mathcal{F}_1', \mathcal{F}_2', \ldots, \mathcal{F}_{l'+1}')$ for $\mathcal{R}$.
Output: A legitimate-agreement forest $\mathcal{F}_0$ for $\mathcal{T}_0$ and $\mathcal{T}_0'$ of minimum weight.

1. If $\mathcal{T}_0 \cong \mathcal{T}_0'$, return $\mathcal{T}_0$ and halt. Otherwise, set $i = l$, set $i' = l'$, and set $\mathcal{F}_i = \mathcal{F}_{l'+1}'$.

2. If the trees $\mathcal{T}_i$ and $\mathcal{T}_i'$ have been obtained from $\mathcal{T}_{i-1}$ and $\mathcal{T}_{i-1}'$ by applying the subtree reduction to a maximal common subtree with label set A, set $\mathcal{F}_{i-1}$ to be the forest obtained from $\mathcal{F}_i$ by joining the vertex labeled s with the root of $\mathcal{T}_{i-1}|A$ via a new edge, removing the label s, and suppressing any vertices of degree two apart from the root.

3. If the trees $\mathcal{T}_i$ and $\mathcal{T}_i'$ have been obtained from $\mathcal{T}_{i-1}$ and $\mathcal{T}_{i-1}'$ by applying the chain reduction to a maximal common m-chain, the forest $\mathcal{F}_i$ either contains two isolated vertices labeled a and b, or $\mathcal{F}_i$ contains a tree $\mathcal{S}$ such that $a, b \in \mathcal{L}(\mathcal{S})$. In the former case, set $\mathcal{F}_{i-1}$ to be the forest obtained from $\mathcal{F}_i$ by replacing the two isolated vertices labeled a and b with m isolated vertices labeled according to the leaf labels $(a_1, a_2, \ldots, a_m)$ of the original m-chain. In the latter case, let $\mathcal{F}_{i-1}$ be the forest obtained from $\mathcal{F}_i$ by replacing the 2-chain of $\mathcal{S}$ with the corresponding m-chain of $\mathcal{T}_{i-1}$ and $\mathcal{T}'_{i-1}$, respectively.

4. If $\mathcal{T}_i$ and $\mathcal{T}'_i$ is the cluster-reduced tree pair that has been obtained from $\mathcal{T}_{i-1}$ and $\mathcal{T}'_{i-1}$ by the cluster reduction applied to a minimal common cluster, identify the unique subtree $\mathcal{S}_{\rho_c} \in \mathcal{F}'_{i'}$ containing a vertex labeled ρ_c and the unique subtree $\mathcal{S}_x \in \mathcal{F}_i$ with a vertex labeled x. Then obtain the tree $\mathcal{S}$ in one of the following two ways:

(a) if $\mathcal{S}_x$ is an isolated vertex, then set $\mathcal{S}$ to be the tree obtained from $\mathcal{S}_{\rho_c}$ by deleting the edge of $\mathcal{S}_{\rho_c}$ which is incident with the vertex labeled ρ_c and the vertex labeled ρ_c itself or,

(b) otherwise, obtain $\mathcal{S}$ from $\mathcal{S}_x$ and $\mathcal{S}_{\rho_c}$ by adjoining $\mathcal{S}_{\rho_c}$ to $\mathcal{S}_x$ via a new edge joining the vertices labeled ρ_c and x, removing the labels ρ_c and x, and suppressing any vertices of degree two apart from the root.

Set $\mathcal{F}_{i-1} = ((\mathcal{F}_i - \{\mathcal{S}_x\}) \cup (\mathcal{F}'_{i'} - \{\mathcal{S}_{\rho_c}\}) \cup \{\mathcal{S}\})$ and decrement i' by 1.

5. Decrement i by 1. Return to step **2** if $i \geq 1$; otherwise, return $\mathcal{F}_0$ and halt.

Remarks.

(1) The construction of a legitimate-agreement forest for two (unreduced) weighted rooted binary phylogenetic X-trees $\mathcal{T}_0$ and $\mathcal{T}'_0$ is of importance since we want to recognize whether hybridization events are deep or not which cannot easily be achieved by considering collections of forests only. Additionally, such a construction is also important if one wants to reconstruct a hybridization network that displays $\mathcal{T}$ and $\mathcal{T}'$ with the minimum number of hybridization events.

(2) In each iteration, the algorithm only executes one of the steps 2, 3, or 4 and computes a legitimate-agreement forest for $\mathcal{T}_{i-1}$ and $\mathcal{T}'_{i-1}$ of minimum weight.

(3) Since the algorithm iterates backwards through $\mathcal{R}$, it is ensured that any subtree, chain, and cluster that has been reduced in the course of a reduction operation gets expanded in the correct order. This is of importance because any such motif can consist of previously reduced subtrees, chains, or clusters.

Theorem 4.6. *Let $\mathcal{T}_0$ and $\mathcal{T}_0'$ be two weighted rooted binary phylogenetic X-trees, and let*

$$\mathcal{R} = (\{\mathcal{T}_1, \mathcal{T}_1'\}, \{\mathcal{T}_2, \mathcal{T}_2'\}, \ldots, \{\mathcal{T}_l, \mathcal{T}_l'\})$$

be a tuple of reduced tree pairs for $\mathcal{T}_0$ and $\mathcal{T}_0'$. Applying the algorithm BuildForest *to $\mathcal{R}$ and an associated collection of forests returns a legitimate-agreement forest for $\mathcal{T}_0$ and $\mathcal{T}_0'$ of minimum weight. In particular, by considering all such collections of forests for $\mathcal{R}$,* BuildForest *calculates all legitimate-agreement forests of minimum weight for $\mathcal{T}_0$ and $\mathcal{T}_0'$.*

Proof. If $\mathcal{T}_0 \cong \mathcal{T}_0'$, then the result follows immediately because the algorithm BuildForest directly returns $\mathcal{T}_0$. Therefore, we may assume that this is not the case. Let $\mathcal{F}$ be a legitimate-agreement forest for $\mathcal{T}_0$ and $\mathcal{T}_0'$ of minimum weight. The proof is by induction on $|\mathcal{R}|$. First, assume that $|\mathcal{R}| = 1$. Depending on the reduction which has been used to obtain $\mathcal{T}_1$ and $\mathcal{T}_1'$ from $\mathcal{T}_0$ and $\mathcal{T}_0'$, respectively, let the forest $\mathcal{F}_1^*$ (obtained from $\mathcal{F}$) be

(i) a subtree-reduced forest if $\mathcal{T}_0$ and $\mathcal{T}_0'$ have a maximal subtree with label set A and $|A| \geq 2$ in common,

(ii) a chain-reduced forest if $\mathcal{T}_0$ and $\mathcal{T}_0'$ have no such subtree but a maximal m-chain $(a_1, a_2, \ldots, a_m)$ with $m > 2$ in common, or

(iii) a cluster-reduced forest if $\mathcal{T}_0$ and $\mathcal{T}_0'$ have no such chain but a minimal cluster A with $|A| \geq 2$ in common.

Due to one of the Lemmas 4.2, 4.3, and 4.4, note that $\mathcal{F}_1^*$ is a legitimate-agreement forest for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight. In particular, since all legitimate-agreement forests of minimum weight for $\mathcal{T}_1$ and $\mathcal{T}_1'$ have been calculated, there exists a forest, $\mathcal{F}_1$ say, with $\mathcal{F}_1 = \mathcal{F}_1^*$. Hence, it is possible to reverse the process with which $\mathcal{F}_1^*$ has been constructed from $\mathcal{F}$ to obtain $\mathcal{F}$ from $\mathcal{F}_1$, thus applying the algorithm BuildForest for one iteration. There are three different cases to consider:

Case (i): If $\mathcal{T}_1$ and $\mathcal{T}_1'$ are the resulting trees after an application of the subtree reduction, there exists a tree $\mathcal{S} \in \mathcal{F}_1$ with a vertex labeled s such that $\mathcal{F}$ can be obtained from $\mathcal{F}_1$ by joining the vertex labeled s with the root of $\mathcal{T}_0|A$ via a new edge, removing the label s, and suppressing any resulting degree two vertices apart from the root.

Case (ii): If $\mathcal{T}_1$ and $\mathcal{T}_1'$ are the resulting trees after an application of the chain reduction, then $\mathcal{F}_1$ either contains a tree $\mathcal{S} \in \mathcal{F}_1$ with $a, b \in \mathcal{L}(\mathcal{S})$, or $\mathcal{F}_1$ contains two isolated vertices labeled a and b, respectively. In the former case, $\mathcal{F}$ can be obtained from

$\mathcal{F}_1$ by replacing the 2-chain (a, b) of $\mathcal{S}$ with the corresponding m-chain $(a_1, a_2, \ldots, a_m)$ of $\mathcal{T}_0$ and, in the latter case, $\mathcal{F}$ can be obtained from $\mathcal{F}_1$ by replacing the vertices labeled a and b with m isolated vertices labeled according to the leaf labels $(a_1, a_2, \ldots, a_m)$.

Case (iii): If $\mathcal{T}_1$ and $\mathcal{T}_1'$ is the cluster-reduced tree pair after an application of the cluster reduction, let $\mathcal{N}$ be the set of all legitimate-agreement forests of minimum weight for $\mathcal{T}|A$ and $\mathcal{T}'|A$ (such forests can be calculated directly without any further reduction). Furthermore, let $\mathcal{F}_c^*$ be a cluster-pair forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ obtained from $\mathcal{F}$. Due to Lemma 4.5, $\mathcal{F}_c^*$ is a legitimate-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ of minimum weight. Hence, there exists a forest in $\mathcal{N}$, $\mathcal{F}_c$ say, with $\mathcal{F}_c^* = \mathcal{F}_c$. Let $\mathcal{S}_{\rho_c} \in \mathcal{F}_c$ be the unique tree whose label set contains ρ_c, and let $\mathcal{S}_x \in \mathcal{F}_1$ be the unique tree whose label set contains x. If $\mathcal{S}_x$ is an isolated vertex, then set $\mathcal{S}$ to be the tree obtained from $\mathcal{S}_{\rho_c}$ by deleting the edge of $\mathcal{S}_{\rho_c}$ which is incident with the vertex labeled ρ_c and the vertex labeled ρ_c itself. Otherwise, obtain $\mathcal{S}$ from $\mathcal{S}_x$ and $\mathcal{S}_{\rho_c}$ by adjoining $\mathcal{S}_{\rho_c}$ to $\mathcal{S}_x$ via a new edge joining the vertices labeled ρ_c and x, removing the labels ρ_c and x, and suppressing any vertices of degree two apart from the root. Then we can deduce that $\mathcal{F} = ((\mathcal{F}_c \cup \mathcal{F}_1 - \{\mathcal{S}_{\rho_c}, \mathcal{S}_x\}) \cup \{\mathcal{S}\})$.

Now suppose that $|\mathcal{R}| > 1$ and that the theorem holds for all pairs of weighted rooted binary phylogenetic X'-trees whose corresponding tuple of reduced tree pairs $\mathcal{R}'$ has the property that $|\mathcal{R}'| < |\mathcal{R}|$. Let $\mathcal{R}_1 = (\{\mathcal{T}_2, \mathcal{T}_2'\}, \{\mathcal{T}_3, \mathcal{T}_3'\}, \ldots, \{\mathcal{T}_l, \mathcal{T}_l'\})$ be a tuple of reduced tree pairs for $\mathcal{T}_1$ and $\mathcal{T}_1'$. Since $\mathcal{R}_1$ is an $(l-1)$-tuple, it follows from the induction assumption that all legitimate-agreement forests for $\mathcal{T}_1$ and $\mathcal{T}_1'$ of minimum weight can be calculated by applying the algorithm BuildForest to all collections of forests for $\mathcal{R}_1$ and, as shown for the base case, given all legitimate-agreement forests for $\mathcal{T}_1$ and $\mathcal{T}_1'$, we can finally obtain $\mathcal{F}$ by applying BuildForest for one further iteration. This completes the proof of the theorem. □

4.4 Application

The result of Theorem 4.6 gives us a good strategy how to compute all legitimate-agreement forests of minimum weight for a pair of (unreduced) weighted rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ and still allows for an application of the algorithm HybridNumber that makes use of the subtree, chain, and cluster reduction and efficiently calculates the minimum number of hybridization events for many biological examples (see Chapter 3). More precisely, we can first apply an extended version of HybridNumber to $\mathcal{T}$ and $\mathcal{T}'$ such that all legitimate-agreement forests of minimum weight are calculated for each cluster-tree pair and the very last tree pair that have been obtained in the course

Table 4.1: Analysis of deep and non-deep hybridization events for the *Poaceae* data set.

Pairwise Combination		**Hybridization Number**	**# Forests**	**Deep Ratio**
ndhF	*phyB*	14	2268	0.62
ndhF	*rbcL*	13	48	0.52
ndhF	*rpoC2*	12	27	0.63
ndhF	*waxy*	9	396	0.73
ndhF	*ITS*	least 15		
phyB	*rbcL*	4	4	0.31
phyB	*rpoC2*	7	1	0.71
phyB	*waxy*	3	6	1
phyB	*ITS*	8	9	0.29
rbcL	*rpoC2*	13	9	0.79
rbcL	*waxy*	7	35	0.82
rbcL	*ITS*	at least 9		
rpoC2	*waxy*	1	1	0
rpoC2	*ITS*	at least 10		
waxy	*ITS*	8	18	0.77

of the reductions. Second, having all such forests it is possible to compute quickly all legitimate-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight by applying the algorithm BuildForest.

We have implemented an extension of HybridNumber (see above) as well as the algorithm BuildForest and used the *Poaceae* data set which has been published by the Grass Phylogeny Working Group (2001) as an example application to analyze the ratio of deep and non-deep hybridization events. A detailed description of this data set is given in Section 3.3. Additionally, we note here that all 66 analyzed taxa belong to different genera and that about 50 % are composite taxa representing either sequences from several genera or sequences of different species (Table 2 of Grass Phylogeny Working Group (2001)). For each pair of gene trees, we applied the extended version of HybridNumber and reconstructed afterwards all legitimate-agreement forests of minimum weight for the initial two trees by using BuildForest.

We next define the *deep ratio* r of two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ to compare the number of deep with the number of non-deep hybridization events. To this end, let $\{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_n\}$ be the set of all legitimate-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, and let d_i be the number of isolated vertices in $\mathcal{F}_i$ with $i \in \{1, 2, \ldots, n\}$. Then

$$r = \frac{\sum\limits_{i=1}^{n} d_i}{n \cdot h(\mathcal{T}, \mathcal{T}')}.$$

For each pair of gene trees of the *Poaceae* data set restricted to common taxa, the number of legitimate-agreement forests of minimum weight and the deep ratio r are presented in Table 4.1. Those three tree pairs for which no deep ratio r is given are instances for which the number of hybridization events cannot be calculated in a reasonable time (see Section 3.3). With exception of the three gene tree pairs *phyB/rbcL*, *phyB/ITS*, and *rpoC2/waxy*, more than 50 % of all hybridization events are non-deep events and, hence, the majority of hybrid species are present-day species. These results are in line with our expectations since hybrid species are often less fit than their parents and, therefore, have a reduced probability to survive and produce viable offspring. This is particularly true for successful intergeneric hybridization which is in general less likely than intrageneric hybridization.

However, due to a high level of variation among the deep ratios

$$r_i = \frac{d_i}{h(\mathcal{T}, \mathcal{T}')}$$

per forest, for all $i \in \{1, 2, \ldots, n\}$ of a given gene tree pair, these results have a limited significance. Nevertheless, it is possible that a combination of this analysis with other methods—e.g. increased taxon sampling that is frequently used to increase the accuracy of phylogenetic estimates (Zwickl and Hillis, 2002)—will aid to gain more insight into non-tree-like evolution; in particular, into ways to distinguish between hybridization and other causes of gene tree incongruence.

5 Hybridization in Non-Binary Trees

5.1 Introduction

Chapter 2 and 3 describe an exact algorithm that is based on a combinatorial framework and three reduction rules to calculate the minimum number of hybridization events for two rooted binary phylogenetic trees. Bordewich and Semple (2007a) showed that this problem is NP-hard even when the initial collection consists of two rooted binary phylogenetic trees. However, the same authors showed that in the case of two binary trees the problem is fixed-parameter tractable (Bordewich and Semple, 2007b). In particular, they showed that the minimum number of hybridization events can be computed in time $O(f(k)+p(|X|))$, where k is the actual minimum number, f is some computable function, $|X|$ is the number of species, and p is a fixed polynomial. Due to the NP-hardness of the problem, such a result is of importance, since for many practical instances, the minimum number of hybridization events is small and, therefore, the problem may be tractable, even for a large number of taxa. This can be seen by considering the separation of the variables k and $|X|$. For more details about fixed-parameter tractability, we refer the interested reader to Downey and Fellows (1998).

Despite the above fixed-parameter tractable algorithm, for many biological data sets in practice (e.g. Paun *et al.*, 2005, Fehrer *et al.*, 2007), the reconstructed phylogenetic trees are not fully resolved; that is, they contain *polytomies*. For example, this may be due to either the tree reconstruction method or the use of consensus trees for a certain analysis. Polytomies—alternatively called *multifurcations*—refer to vertices which have more than two direct descendants. A polytomy is *hard* if it refers to an event during which an ancestral species gave rise to more than two offspring species at the same time, whereas a *soft* polytomy represents ambiguous evolutionary relationships as a result of insufficient information (Maddison, 1989).

Since simultaneous speciation events only occur rarely, we typically assume that all polytomies in a phylogenetic tree are soft. The reconstruction of a strictly bifurcating tree may consequently force refinements that are not necessarily optimal in terms of the hybridization number. An example for that is depicted in Figure 5.1, where two binary refinements $\mathcal{S}_1$ and $\mathcal{S}_2$ of the tree $\mathcal{T}'$ are shown. While the hybridization number for $\mathcal{S}_1$ and $\mathcal{T}$ is 0, this number for $\mathcal{S}_2$ and $\mathcal{T}$ is 1.

In this chapter, we show that the decision problem of asking whether the minimum

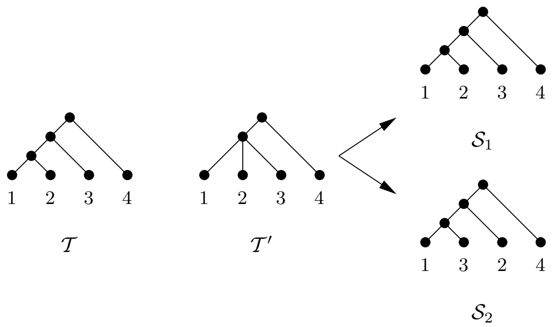


Figure 5.1: Two rooted phylogenetic tree $\mathcal{T}$ and $\mathcal{T}'$ and two binary refinements $\mathcal{S}_1$ and $\mathcal{S}_2$ of $\mathcal{T}'$. The hybridization number for $\mathcal{S}_1$ and $\mathcal{T}$ is 0, while this number for $\mathcal{S}_2$ and $\mathcal{T}$ is 1.

number of hybridization events to explain two (arbitrary) rooted phylogenetic trees is at most k is fixed-parameter tractable. We now describe the above-mentioned problem formally beginning with several definitions.

Given the definition of a *cluster* (see Section 1.4.2) for a rooted phylogenetic X-tree, we first note that we sometimes refer to such a cluster as *edge cluster* in this chapter. Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. We say that $\mathcal{T}'$ *refines* $\mathcal{T}$, or equivalently $\mathcal{T}'$ is a *refinement* of $\mathcal{T}$ if $\mathcal{C}(\mathcal{T}) \subseteq \mathcal{C}(\mathcal{T}')$. In addition, $\mathcal{T}'$ is a *binary refinement* if $\mathcal{T}'$ is binary. Note that $\mathcal{T}$ is a refinement of itself. Graphically speaking, it is straightforward to see that if $\mathcal{T}'$ refines $\mathcal{T}$, then $\mathcal{T}$ can be obtained from $\mathcal{T}'$ by contracting interior edges.

Recalling the definition of the hybridization number of a network (see Section 2.1) and extending it to a collection $\mathcal{P}$ of rooted phylogenetic trees, we set

$$h(\mathcal{P}) = \min\{h(\mathcal{H}) : \mathcal{H} \text{ is a hybridization network that displays } \mathcal{P}\}.$$

If $\mathcal{P}$ contains precisely two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$, then we denote the hybridization number $h(\mathcal{P})$ by $h(\mathcal{T}, \mathcal{T}')$. Next, we show that the beforehand given definition of the hybridization number for two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ is equivalent to

$$h(\mathcal{T}, \mathcal{T}') = \min\{h(\mathcal{S}, \mathcal{S}') : \mathcal{S} \text{ and } \mathcal{S}' \text{ are binary refinements of } \mathcal{T} \text{ and } \mathcal{T}', \text{ respectively}\}$$

and use both definitions interchangeably in the rest of this chapter.

Lemma 5.1. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. Then*

$$\min\{h(\mathcal{S}, \mathcal{S}') : \mathcal{S} \textit{ and } \mathcal{S}' \textit{ are binary refinements of } \mathcal{T} \textit{ and } \mathcal{T}', \textit{ respectively}\}$$
$$=$$
$$\min\{h(\mathcal{H}) : \mathcal{H} \textit{ is a hybridization network that displays } \mathcal{T} \textit{ and } \mathcal{T}'\}.$$

Proof. First, suppose that $h(\mathcal{S},\mathcal{S}')=k$. Then there exists a hybridization network $\mathcal{H}$ that displays $\mathcal{S}$ and $\mathcal{S}'$ and whose hybridization number (the number of hybridization vertices) is k. Since $\mathcal{S}$ and $\mathcal{S}'$ are binary refinements of $\mathcal{T}$ and $\mathcal{T}'$, respectively, $\mathcal{H}$ also displays $\mathcal{T}$ and $\mathcal{T}'$. This implies that

$$\min\{h(\mathcal{S},\mathcal{S}') : \mathcal{S} \text{ and } \mathcal{S}' \text{ are binary refinements of } \mathcal{T} \text{ and } \mathcal{T}', \text{ respectively}\} \geq \\ \min\{h(\mathcal{H}) : \mathcal{H} \text{ is a hybridization network that displays } \mathcal{T} \text{ and } \mathcal{T}'\}.$$

Second, suppose that $h(\mathcal{H})=k$. We show that there exists a hybridization network $\mathcal{H}'$ with $h(\mathcal{H}')=k$ that displays binary refinements $\mathcal{S}$ and $\mathcal{S}'$ of $\mathcal{T}$ and $\mathcal{T}'$, respectively. To this end, obtain $\mathcal{H}'$ from $\mathcal{H}$ such that $\mathcal{C}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}')$, all trees embedded in $\mathcal{H}'$ are binary, and $h(\mathcal{H}')=k$. With $\mathcal{C}(\mathcal{H})$ and $\mathcal{C}(\mathcal{H}')$, we denote the cluster set of $\mathcal{H}$ and $\mathcal{H}'$, respectively. Note that $\mathcal{H}'$ again displays $\mathcal{T}$ and $\mathcal{T}'$. Now it is easily checked that $\mathcal{H}'$ displays a rooted binary tree $\mathcal{S}$ with $\mathcal{C}(\mathcal{T}) \subseteq \mathcal{C}(\mathcal{S})$, Similarly, the same holds for $\mathcal{S}'$. Hence, we can deduce that

$$\min\{h(\mathcal{S},\mathcal{S}') : \mathcal{S} \text{ and } \mathcal{S}' \text{ are binary refinements of } \mathcal{T} \text{ and } \mathcal{T}', \text{ respectively}\} \leq \\ \min\{h(\mathcal{H}) : \mathcal{H} \text{ is a hybridization network that displays } \mathcal{T} \text{ and } \mathcal{T}'\}$$

holds. This establishes the lemma. □

We can now formally state the decision problem Minimum Hybridization for when $\mathcal{P}=\{\mathcal{T},\mathcal{T}'\}$:

Minimum Hybridization
Instance: Two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$, and an integer k.
Question: Is $h(\mathcal{T},\mathcal{T}') \leq k$?

Since computing $h(\mathcal{T},\mathcal{T}')$ is NP-hard when $\mathcal{T}$ and $\mathcal{T}'$ are binary (Bordewich and Semple, 2007a), calculating this value for when $\mathcal{T}$ and $\mathcal{T}'$ are arbitrary rooted phylogenetic X-trees is also NP-hard.

The main result of this chapter is the following theorem.

Theorem 5.2. *The decision problem* Minimum Hybridization *is fixed-parameter tractable with $h(\mathcal{T},\mathcal{T}')$ being the parameter.*

The overall approach in proving Theorem 5.2 is similar to that used to show that Minimum Hybridization is fixed-parameter tractable when the initial two trees are binary. Basically, we use three reductions to kernalize the problem instance in a regulated way before calculating exactly the minimum number of hybridization events using an exhaustive search. The reason that this is sufficient to prove Theorem 5.2 is that the size of the label set of the trees $\mathcal{S}$ and $\mathcal{S}'$ obtained from $\mathcal{T}$ and $\mathcal{T}'$ by repeatedly applying the three reductions is linear in $h(\mathcal{T}, \mathcal{T}')$.

The chapter is organized as follows. The next section contains some additional preliminaries that are used throughout this chapter. In Sections 5.3 and 5.4, we characterize Minimum Hybridization in terms of a particular type of agreement forest. This characterization is essential to getting the main result of this chapter. Section 5.5 describes the three reductions that are used to kernalize the problem instance and also includes three key lemmas that are needed for the proof of Theorem 5.2. We also show how a fourth reduction can be used to break the problem into a number of smaller and more tractable subproblems. The proof of Theorem 5.2 is given in Section 5.6.

We end the introduction by remarking that despite the similarities between the approaches used to prove Theorem 5.2 and the analogous result for binary trees, we see no obvious way that this latter result can be used to directly establish Theorem 5.2. Part of the reason for this is that a number of additional and non-trivial complications arise in the non-binary case.

5.2 Preliminaries

In this section, we give some preliminary definitions that are used throughout this chapter. For a rooted phylogenetic X-tree $\mathcal{T}$, a subset Y of X is called a *vertex cluster* of $\mathcal{T}$ if there is a refinement of $\mathcal{T}$ in which Y is an edge cluster. For example, considering Figure 5.1, the taxa set $\{1, 2\}$ is an edge cluster in $\mathcal{T}$, but a vertex cluster (and not an edge cluster) in $\mathcal{T}'$. Note that edge clusters are special types of vertex clusters. Furthermore, a subtree of $\mathcal{T}$ is *pendant* if it can be obtained from a refinement of $\mathcal{T}$ by deleting a single edge. Note that this definition is different from the definition of *pendant* throughout the rest of this thesis. Lastly, a subtree is *non-trivial* if it contains at least two leaves.

5.3 Agreement Forests

Various types of agreement forests have recently been used to analyze reticulate evolution for a set of gene trees and its impact on evolution (Song and Hein, 2003, Bordewich and

Semple, 2004, Baroni *et al.*, 2005, Song and Hein, 2005, Bordewich and Semple, 2007b). All of these approaches are restricted to the case when the trees under consideration are binary. Here, we extend the definition of agreement forests to arbitrary rooted phylogenetic trees. For the reader familiar with agreement forests, we note that the following definitions coincide with those previously given for rooted binary phylogenetic trees.

Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. For the purposes of the upcoming definitions, we regard the root of both $\mathcal{T}$ and $\mathcal{T}'$ as a vertex labeled ρ at the end of a pendant edge adjoined to the original root. Furthermore, we also regard ρ as part of the label set of $\mathcal{T}$ and $\mathcal{T}'$, thus we view their label sets as $X \cup \{\rho\}$.

A *forest* of $\mathcal{T}$ is a partition $\{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ of its label set $X \cup \{\rho\}$, where $\mathcal{L}_\rho$ contains ρ, no part is empty, and the trees in $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are edge-disjoint rooted subtrees of $\mathcal{T}$. An *agreement forest* $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ is a forest $\{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ of $\mathcal{T}$ and $\mathcal{T}'$ such that, for all $i \in \{\rho, 1, 2, \ldots, k\}$, the trees $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement. To illustrate these concepts, two examples of agreement forests $\mathcal{F}_1$ and $\mathcal{F}_2$ are shown in Figure 5.2 for the two rooted phylogenetic trees $\mathcal{T}$ and $\mathcal{T}'$ also shown in that figure. Considering $\mathcal{F}_1$, it is easily checked that, for each label set $\mathcal{L}_i$, the restrictions of $\mathcal{T}$ and $\mathcal{T}'$, respectively, to $\mathcal{L}_i$ have a common binary refinement.

A *maximum-agreement forest* for $\mathcal{T}$ and $\mathcal{T}'$ is an agreement forest $\{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ in which k (the number of parts minus one) is minimized. The minimum possible value for k is denoted by $m(\mathcal{T}, \mathcal{T}')$. Bordewich and Semple (2004) established the following result:

Theorem 5.3. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted binary phylogenetic X-trees. Then*

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') = m(\mathcal{T}, \mathcal{T}').$$

The corresponding characterization for the minimum number of hybridization events for the same pair of trees requires an additional condition. This condition excludes the possibility that species inherit genetic material from their own descendants. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be an agreement forest for two arbitrary rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$. Let $G_{\mathcal{F}}$ be the directed graph that has vertex set $\mathcal{F}$ and an arc $(\mathcal{L}_i, \mathcal{L}_j)$ from $\mathcal{L}_i$ to $\mathcal{L}_j$ precisely if $i \neq j$ and either

(I) the path from the root of $\mathcal{T}(\mathcal{L}_i)$ to the root of $\mathcal{T}(\mathcal{L}_j)$ contains an edge of $\mathcal{T}(\mathcal{L}_i)$, or

(II) the path from the root of $\mathcal{T}'(\mathcal{L}_i)$ to the root of $\mathcal{T}'(\mathcal{L}_j)$ contains an edge of $\mathcal{T}'(\mathcal{L}_i)$.

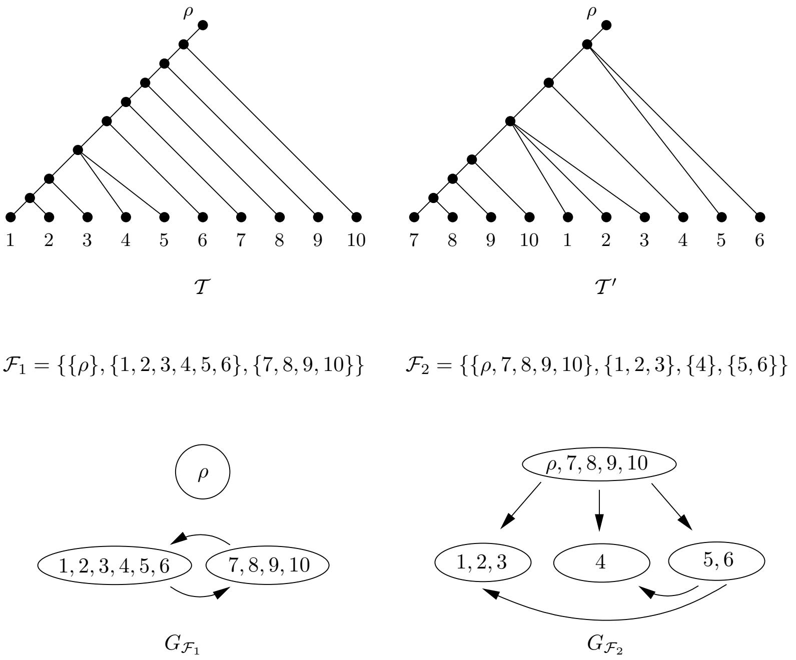


Figure 5.2: Two agreement forests $\mathcal{F}_1$ and $\mathcal{F}_2$ for the two rooted trees $\mathcal{T}$ and $\mathcal{T}'$ and their associated digraphs $G_{\mathcal{F}_1}$ and $G_{\mathcal{F}_2}$.

We say that $\mathcal{F}$ is an *acyclic-agreement forest* for $\mathcal{T}$ and $\mathcal{T}'$ if $G_{\mathcal{F}}$ contains no directed cycles, that is, $G_{\mathcal{F}}$ is acyclic. For the example depicted in Figure 2.3, $\mathcal{F}_2$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ since $G_{\mathcal{F}_2}$ is acyclic, whereas $\mathcal{F}_1$ is not an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. If $\mathcal{F}$ contains the smallest number of parts over all acyclic-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$, we say that $\mathcal{F}$ is a *maximum-acyclic-agreement forest* for $\mathcal{T}$ and $\mathcal{T}'$, in which case, we denote this value of k by $m_a(\mathcal{T}, \mathcal{T}')$. In the case that both $\mathcal{T}$ and $\mathcal{T}'$ are binary, these definitions again extend those typically given for two rooted binary phylogenetic trees. Baroni *et al.* (2005) established the following characterization for binary trees.

Theorem 5.4. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted binary phylogenetic X-trees. Then*

$$h(\mathcal{T}, \mathcal{T}') = m_a(\mathcal{T}, \mathcal{T}').$$

5.4 Characterizing $h(\mathcal{T}, \mathcal{T}')$ in Terms of Agreement Forests

In this section, we prove the analogs of Theorems 5.3 and 5.4 for arbitrary rooted phylogenetic trees. The second analog is crucial in proving the main result of the chapter.

Theorem 5.5. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. Then*

$$h(\mathcal{T}, \mathcal{T}') = m_a(\mathcal{T}, \mathcal{T}').$$

Essentially, all of the work in establishing this theorem is done in proving the next two lemmas.

Lemma 5.6. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let $\mathcal{F}$ be an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Then there exist binary refinements $\mathcal{S}$ and $\mathcal{S}'$ of $\mathcal{T}$ and $\mathcal{T}'$, respectively, such that $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$.*

Proof. Suppose that $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, and let $\mathcal{B}_i$ be a common binary refinement of $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ for all i. The proof of the lemma is by induction on k. Clearly, the result holds if $k = 0$. Now suppose that the result holds for all acyclic-agreement forests of $\mathcal{T}$ and $\mathcal{T}'$ of size at most k. Since $\mathcal{F}$ is acyclic, $G_{\mathcal{F}}$ contains a vertex, $\mathcal{L}_m$ say, with outdegree zero. Since $\mathcal{L}_m$ has outdegree zero, $\mathcal{T}(\mathcal{L}_m)$ is a pendant subtree of $\mathcal{T}$ and $\mathcal{T}'(\mathcal{L}_m)$ is a pendant subtree of $\mathcal{T}'$.

Let $\mathcal{T}_m$ and $\mathcal{T}_m'$ be the rooted phylogenetic trees $\mathcal{T}|((X \cup \{\rho\}) - \mathcal{L}_m)$ and $\mathcal{T}'|((X \cup \{\rho\}) - \mathcal{L}_m)$, respectively, and let $\mathcal{F}_m = \mathcal{F} - \{\mathcal{L}_m\}$. Since $\mathcal{F}$ is an acyclic-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that, as $\mathcal{T}(\mathcal{L}_m)$ is a pendant subtree of $\mathcal{T}$ and $\mathcal{T}'(\mathcal{L}_m)$ is a pendant subtree of $\mathcal{T}'$, the collection $\mathcal{F}_m$ is an acyclic-agreement forest of $\mathcal{T}_m$ and $\mathcal{T}_m'$. Therefore, by the induction assumption, there are binary refinements $\mathcal{S}_m$ and $\mathcal{S}_m'$ of $\mathcal{T}_m$ and $\mathcal{T}_m'$, respectively, such that $\mathcal{F}_m$ is an acyclic-agreement forest for $\mathcal{S}_m$ and $\mathcal{S}_m'$.

We now construct a binary refinement of $\mathcal{T}$ from $\mathcal{S}_m$. Let u be the vertex of $\mathcal{T}$ with the property that $\mathcal{C}(u)$ is the minimal cluster of $\mathcal{T}$ that properly contains $\mathcal{L}_m$. By construction, $\mathcal{C}(u) - \mathcal{L}_m$ is a cluster of $\mathcal{T}_m$. Furthermore, as $\mathcal{S}_m$ is a binary refinement of $\mathcal{T}_m$, the set $\mathcal{C}(u) - \mathcal{L}_m$ is a cluster of $\mathcal{S}_m$. Let u_m be the vertex of $\mathcal{S}_m$ such that $\mathcal{C}(u_m) = \mathcal{C}(u) - \mathcal{L}_m$. Let $\mathcal{S}$ be the rooted binary phylogenetic tree obtained from $\mathcal{S}_m$ by subdividing the edge coming into u_m with a new vertex v and adjoining the root of $\mathcal{B}_m$ to this new vertex v via a new edge. Observing that $\mathcal{C}(v) = \mathcal{C}(u)$, it is easily checked that $\mathcal{S}$ is a binary refinement of $\mathcal{T}$. Furthermore, by construction and because of the induction assumption, it follows that $\mathcal{F}$ is a forest of $\mathcal{S}$ and, for all i, we have $\mathcal{S}|\mathcal{L}_i = \mathcal{B}_i$.

By the same construction and argument, there is a binary refinement $\mathcal{S}'$ of $\mathcal{T}'$ such that $\mathcal{F}$ is a forest of $\mathcal{S}'$ and, for all i, we have $\mathcal{S}'|\mathcal{L}_i = \mathcal{B}_i$. It now follows that $\mathcal{F}$ is an agreement forest for $\mathcal{S}$ and $\mathcal{S}'$. Moreover, as $\mathcal{F}_m$ is an acyclic-agreement forest for $\mathcal{S}_m$ and $\mathcal{S}_m'$, it is easily seen that $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$. This completes the proof of the lemma. □

Lemma 5.7. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$, respectively. If $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$,* *then $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$.*

Proof. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be an acyclic-agreement forest of $\mathcal{S}$ and $\mathcal{S}'$. Since $\mathcal{S}$ and $\mathcal{S}'$ are both binary, it is easily seen, for all i, that $\mathcal{S}|\mathcal{L}_i$ and $\mathcal{S}'|\mathcal{L}_i$ are binary. Therefore, as $\mathcal{S}$ and $\mathcal{S}'$ are binary refinements of $\mathcal{T}$ and $\mathcal{T}'$, respectively, $\mathcal{S}|\mathcal{L}_i$ is a common binary refinement of $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ for all i. To see that the trees in $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are edge-disjoint rooted subtrees of $\mathcal{T}$, suppose that this is not the case. Then, for some $r \neq s$, the subtrees $\mathcal{T}(\mathcal{L}_r)$ and $\mathcal{T}(\mathcal{L}_s)$ are not edge-disjoint. That is, $\mathcal{T}(\mathcal{L}_r)$ and $\mathcal{T}(\mathcal{L}_s)$ have an edge $e = \{u, v\}$ in common. Let u be the end vertex of e closest to ρ. Since $\mathcal{S}$ is a binary refinement of $\mathcal{T}$, there are vertices u' and v' of $\mathcal{S}$ with $\mathcal{C}_\mathcal{S}(u') = \mathcal{C}_\mathcal{T}(u)$ and $\mathcal{C}_\mathcal{S}(v') = \mathcal{C}_\mathcal{T}(v)$. Now it is easily seen that $\mathcal{S}(\mathcal{L}_r)$ contains u' and v', and $\mathcal{S}(\mathcal{L}_s)$ contains u' and v'. In other words, $\mathcal{S}(\mathcal{L}_r)$ and $\mathcal{S}(\mathcal{L}_s)$ are not edge-disjoint in $\mathcal{S}$, contradicting that $\mathcal{F}$ is an agreement forest of $\mathcal{S}$ and $\mathcal{S}'$. Thus the trees in $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are edge-disjoint rooted subtrees of $\mathcal{T}$ and, similarly, the trees in $\{\mathcal{T}'(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are edge-disjoint rooted subtrees of $\mathcal{T}'$. Hence, $\mathcal{F}$ is an agreement forest of $\mathcal{T}$ and $\mathcal{T}'$.

Now relative to $\mathcal{S}$ and $\mathcal{S}'$, the graph $G_\mathcal{F}$ is acyclic. With respect to $\mathcal{F}$, consider the analogous graph, $G'_\mathcal{F}$ say, for $\mathcal{T}$ and $\mathcal{T}'$. Noting that both graphs have the same vertex set, it is clear that if $(\mathcal{L}_r, \mathcal{L}_s)$ is an arc in $G'_\mathcal{F}$, then $(\mathcal{L}_r, \mathcal{L}_s)$ is an arc in $G_\mathcal{F}$. Thus the arc set of $G'_\mathcal{F}$ is a subset of the arc set of $G_\mathcal{F}$. Since $G_\mathcal{F}$ is acyclic, it follows that $G'_\mathcal{F}$ is acyclic. This completes the proof of the lemma. □

Proof of Theorem 5.5. Let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$ that satisfy the hypothesis of Lemma 5.6. Then, by that lemma, $m_a(\mathcal{T}, \mathcal{T}') \geq m_a(\mathcal{S}, \mathcal{S}')$. But, by Theorem 5.4, $m_a(\mathcal{S}, \mathcal{S}') = h(\mathcal{S}, \mathcal{S}')$. It now follows that, as $h(\mathcal{S}, \mathcal{S}') \geq h(\mathcal{T}, \mathcal{T}')$, we have $m_a(\mathcal{T}, \mathcal{T}') \geq h(\mathcal{T}, \mathcal{T}')$.

To establish the converse, now let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$ such that $h(\mathcal{S}, \mathcal{S}') = h(\mathcal{T}, \mathcal{T}')$. Then, by Theorem 5.4, there is an acyclic-agreement forest $\mathcal{F}$ of $\mathcal{S}$ and $\mathcal{S}'$ such that

$$|\mathcal{F}| - 1 = h(\mathcal{S}, \mathcal{S}') = h(\mathcal{T}, \mathcal{T}').$$

By Lemma 5.7, $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, so

$$m_a(\mathcal{T}, \mathcal{T}') \leq |\mathcal{F}| - 1 = h(\mathcal{T}, \mathcal{T}').$$

It now follows that $h(\mathcal{T}, \mathcal{T}') = m_a(\mathcal{T}, \mathcal{T}')$. This completes the proof of the theorem. □

For the reader interested in calculating the rSPR distance $d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}')$ between two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ which we define to be

$$\min\{d_{\mathrm{rSPR}}(\mathcal{S},\mathcal{S}') : \mathcal{S} \text{ and } \mathcal{S}' \text{ are binary refinements of } \mathcal{T} \text{ and } \mathcal{T}', \text{ respectively}\},$$

we end this section by establishing a further result. Again, we start with the proofs of two lemmas.

Lemma 5.8. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let $\mathcal{F}$ be an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Then there exist binary refinements $\mathcal{S}$ and $\mathcal{S}'$ of $\mathcal{T}$ and $\mathcal{T}'$, respectively, such that $\mathcal{F}$ is an agreement forest for $\mathcal{S}$ and $\mathcal{S}'$.*

Proof. The proof of this lemma can be established in the same way as the proof of Lemma 5.6, without considering the acyclic condition. □

Lemma 5.9. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$, respectively. If $\mathcal{F}$ is an agreement forest for $\mathcal{S}$ and $\mathcal{S}'$, then $\mathcal{F}$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$.*

Proof. The proof of this lemma can be established in the same way as the proof of Lemma 5.7, without considering the acyclic condition. □

Theorem 5.10. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. Then*

$$d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') = m(\mathcal{T},\mathcal{T}').$$

Proof. Let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$ that satisfy the hypothesis of Lemma 5.8. Then, by that lemma, $m(\mathcal{T},\mathcal{T}') \geq m(\mathcal{S},\mathcal{S}')$. But, by Theorem 5.3, $m(\mathcal{S},\mathcal{S}') = d_{\mathrm{rSPR}}(\mathcal{S},\mathcal{S}')$. It now follows that, as $d_{\mathrm{rSPR}}(\mathcal{S},\mathcal{S}') \geq d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}')$, we have $m(\mathcal{T},\mathcal{T}') \geq d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}')$.

To establish the converse, now let $\mathcal{S}$ and $\mathcal{S}'$ be binary refinements of $\mathcal{T}$ and $\mathcal{T}'$ such that $d_{\mathrm{rSPR}}(\mathcal{S},\mathcal{S}') = d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}')$. Then, by Theorem 5.3, there is an agreement forest $\mathcal{F}$ of $\mathcal{S}$ and $\mathcal{S}'$ such that

$$|\mathcal{F}| - 1 = d_{\mathrm{rSPR}}(\mathcal{S},\mathcal{S}') = d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}').$$

By Lemma 5.9, $\mathcal{F}$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, so

$$m(\mathcal{T},\mathcal{T}') \leq |\mathcal{F}| - 1 = d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}').$$

It now follows that $d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') = m(\mathcal{T}, \mathcal{T}')$. This completes the proof of the theorem. □

5.5 Reducing the Size of the Problem Instance

In this section, we introduce three reductions which kernalize Minimum Hybridization and a fourth reduction which breaks the problem instance into a number of smaller and more tractable subproblems. The *subtree* and *long-chain reductions* extend the subtree and chain reductions described in Bordewich and Semple (2007b). Additionally, we introduce the *short-chain reduction* which—in combination with the other two reductions—guarantees that all problem instances can be kernalized. The *cluster reduction* extends the result of Theorem 1 described in Baroni *et al.* (2006). Although this reduction is not necessary to prove Theorem 5.2 it has proven to be useful in practice (Bordewich *et al.*, 2007c). We begin with some preliminaries.

Let $\mathcal{T}$ be a rooted phylogenetic X-tree, and let x be an element of X. Viewing $\mathcal{T}$ as a directed graph with edges directed away from its root, the unique vertex, u say, of $\mathcal{T}$ such that (u, x) is an arc of $\mathcal{T}$ is called the *parent* of x and is denoted by $p_{\mathcal{T}}(x)$.

For all $n \geq 2$, an *n-chain* of $\mathcal{T}$ is an ordered tuple $(a_1, a_2, \ldots, a_n)$ of distinct elements of X that satisfies the following properties:

(i) for all $i \in \{1, 2, \ldots, n-1\}$, either $p_{\mathcal{T}}(a_i) = p_{\mathcal{T}}(a_{i+1})$ or $p_{\mathcal{T}}(a_i)$ is a child of $p_{\mathcal{T}}(a_{i+1})$, and

(ii) there is an ordering, $p_1, p_2, \ldots, p_m$ say, of the parents of $a_1, a_2, \ldots, a_n$ such that, for all $i \in \{1, 2, \ldots, m-1\}$, the vertex p_i is a child of p_{i+1} and, apart from p_1 and p_m, each of the vertices $p_2, p_3, \ldots, p_{m-1}$ has at exactly one child not in $\{a_1, a_2, \ldots, a_n\}$.

If p is a parent of an element in $A = \{a_1, a_2, \ldots, a_n\}$, then p is called *internal* if it has at most one child not in A; otherwise p is said to be *external.* An element of A is *internal* if its parent is internal, otherwise it is *external.* Note that $p_2, \ldots, p_{m-1}$ are always internal, but that p_1 and p_m can be internal or external. Thus if a_i is external, then it is a child of p_1 or p_m. Furthermore, if $\mathcal{T}$ is binary, then all elements of A are internal. Throughout this chapter, we will assume that if $(a_1, a_2, \ldots, a_n)$ is an n-chain of both $\mathcal{T}$ and $\mathcal{T}'$, where $\mathcal{T}$ and $\mathcal{T}'$ are rooted phylogenetic X-trees, then $\mathcal{T}$ and $\mathcal{T}'$ have no common non-trivial pendant subtree whose label set is a subset of $\{a_1, a_2, \ldots, a_n\}$. As we will soon see, this assumption does not restrict the results in this chapter; it is simply for convenience and to avoid repetition in the statements. As an illustration, $(a_1, a_2, \ldots, a_n)$ is an n-chain of the two rooted phylogenetic trees $\mathcal{T}$ and $\mathcal{T}'$ shown in Figure 5.3, where triangles represent subtrees outside of the chain.

Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. Let P be a disjoint collection of subsets $\{a_1, a_2, \ldots, a_n\}$ of X each being the set of elements of a chain $(a_1, a_2, \ldots, a_n)$ common to both $\mathcal{T}$ and $\mathcal{T}'$ such that either

(i) $(a_1, a_2, \ldots, a_n)$ has exactly three elements that are internal in both $\mathcal{T}$ and $\mathcal{T}'$, or

(ii) for one of the trees, $(a_1, a_2, \ldots, a_n)$ has exactly two internal elements while, in the other tree, $(a_1, a_2, \ldots, a_n)$ has exactly one parent.

Depending on whether the chain satisfies (i) or (ii), we assign a triple of weights or a single weight from $\mathbb{Z}^+ \times \mathbb{Z}^+ \times \mathbb{Z}^+$ and $\mathbb{Z}^+$, respectively. We call such a pair of trees with associated weighted set P a *pair of weighted rooted phylogenetic X-trees.*

We now describe the four reductions. Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees with an associated set P, and let A be a subset of X. We say that A does not *cross* P if, for each member S in P, the intersection $S \cap A$ is empty.

Subtree Reduction: For $|A| \geq 2$, if A is the label set of a maximal pendant subtree in $\mathcal{T}$ and $\mathcal{T}'$ with the properties that $\mathcal{T}|A$ and $\mathcal{T}'|A$ have a common binary refinement and A does not cross P, then replace these subtrees with either a single new leaf labeled a or a pendant edge ending in a new leaf labeled a depending on whether the subtree can be obtained without or with refinement, respectively. In all cases, the new label is the same in both resulting trees.

Long-Chain Reduction: For $n \geq 4$, let $(a_1, a_2, \ldots, a_n)$ be a maximal n-chain of $\mathcal{T}$ and $\mathcal{T}'$ that does not cross P with the following properties:

(i) The chain has at least three internal parents in both $\mathcal{T}$ and $\mathcal{T}'$, and at least three elements that are internal in both $\mathcal{T}$ and $\mathcal{T}'$.

(ii) If a_1 is external in one of the trees, then a_2 is internal in the same tree and a_1 is internal in the other tree.

(iii) If a_n is external in one of the trees, then a_{n-1} is internal in the same tree while, in the other tree, a_n is internal and there are not exactly three internal parents of which one has a_n as its only child in $\{a_1, a_2, \ldots, a_n\}$.

Depending upon whether $\emptyset$, $\{a_1\}$, $\{a_n\}$, or $\{a_1, a_n\}$ is the subset of elements of $\{a_1, a_2, \ldots, a_n\}$

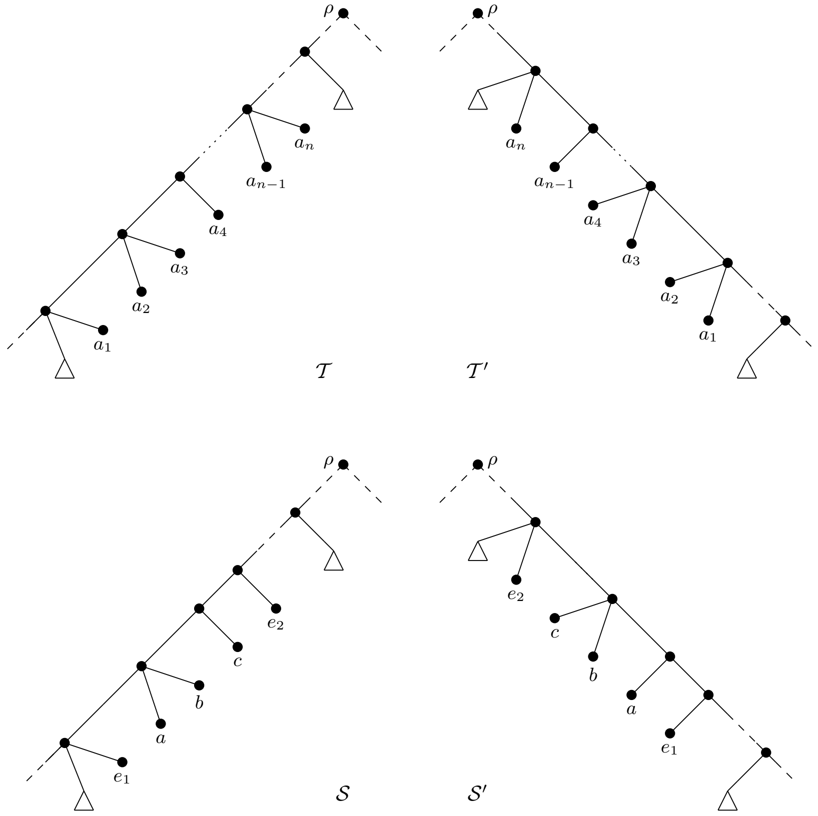


Figure 5.3: Two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ reduced under the long-chain reduction, where $\mathcal{S}$ and $\mathcal{S}'$ are the resulting trees. Dotted lines indicate regions of the chain $(a_1, a_2, \ldots, a_n)$. In $\mathcal{T}$, a_1 is external while a_n is internal and, in $\mathcal{T}'$, a_1 is internal while a_n is external.

that are external in either $\mathcal{T}$ or $\mathcal{T}'$, respectively replace this chain in $\mathcal{T}$ and $\mathcal{T}'$ with the chain (a, b, c), (e_1, a, b, c), (a, b, c, e_2), or (e_1, a, b, c, e_2) as follows:

(i) In $\mathcal{T}$,

$$p_{\mathcal{T}}(e_1) \neq p_{\mathcal{T}}(a) = p_{\mathcal{T}}(b) \neq p_{\mathcal{T}}(c) \neq p_{\mathcal{T}}(e_2),$$

where e_1 is external if a_1 is external in $\mathcal{T}$, otherwise e_1 is internal; and where e_2 is external if a_n is external in $\mathcal{T}$, otherwise e_2 is internal.

(ii) In $\mathcal{T}'$,

$$p_{\mathcal{T}}(e_1) \neq p_{\mathcal{T}}(a) \neq p_{\mathcal{T}}(b) = p_{\mathcal{T}}(c) \neq p_{\mathcal{T}}(e_2),$$

where e_1 is external if a_1 is external in $\mathcal{T}'$, otherwise e_1 is internal; and where e_2 is external if a_n is external in $\mathcal{T}'$, otherwise e_2 is internal.

Relative to $(a_1, a_2, \ldots, a_n)$, if m denotes the number of internal parents in $\mathcal{T}$ and m' denotes the number of internal parents in $\mathcal{T}'$, then respectively add the new set $\{a, b, c\}$, $\{e_1, a, b, c\}$, $\{a, b, c, e_2\}$, or $\{e_1, a, b, c, e_2\}$ to P and, calling this set S, assign it a tuple of weights in which the first coordinate w_1 is $n - |S|$, the second coordinate w_2 is m minus the number of internal parents of the resulting chain in $\mathcal{T}$, and the third coordinate w_3 is m' minus the number of internal parents of the resulting chain in $\mathcal{T}'$. Intuitively, the reduction results in replacing a_1 and a_n with e_1 and e_2, respectively, if a_1 or a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$, and replacing the elements of the chain that are internal in both trees with a, b, and c. Figure 5.3 depicts an example of the long-chain reduction, where $\mathcal{T}$ and $\mathcal{T}'$ are the trees before, and $\mathcal{S}$ and $\mathcal{S}'$ are the trees after applying the long-chain reduction. In this example, a_1 is external in $\mathcal{T}$, while a_n is external in $\mathcal{T}'$, and so the chain $(a_1, a_2, \ldots, a_n)$ is replaced with the chain (e_1, a, b, c, e_2).

Short-Chain Reduction: For $n \geq 3$, let $(a_1, a_2, \ldots, a_n)$ be a maximal n-chain of $\mathcal{T}$ and $\mathcal{T}'$ that does not cross P with the property that in one of the trees, say $\mathcal{T}$, this chain has exactly one parent, while in the other tree $\mathcal{T}'$ this chain has at least three internal parents. (Due to the assumption that no element of an n-chain is part of a common non-trivial pendant subtree of $\mathcal{T}$ and $\mathcal{T}'$, note that $p_{\mathcal{T}'}(a_1), \ldots, p_{\mathcal{T}'}(a_n)$ are pairwise distinct vertices in $\mathcal{T}'$ and so only a_1 or a_n may be external in $\mathcal{T}'$.) Depending upon whether $\emptyset$, $\{a_1\}$, $\{a_n\}$, or $\{a_1, a_n\}$ is the subset of external elements of this chain in $\mathcal{T}'$, respectively replace this chain in $\mathcal{T}$ and $\mathcal{T}'$ with the chain (a, b), (e_1, a, b), (a, b, e_2), or (e_1, a, b, e_2) as follows:

(i) In $\mathcal{T}$,

$$p_{\mathcal{T}}(e_1) = p_{\mathcal{T}}(a) = p_{\mathcal{T}}(b) = p_{\mathcal{T}}(e_2).$$

(ii) In $\mathcal{T}'$,

$$p_{\mathcal{T}'}(e_1) \neq p_{\mathcal{T}'}(a) \neq p_{\mathcal{T}'}(b) \neq p_{\mathcal{T}'}(e_2),$$

where e_1 is external if a_1 is external in $\mathcal{T}'$ and e_2 is external if a_n is external in $\mathcal{T}'$.

Furthermore, add the new set $\{a, b\}$, $\{e_1, a, b\}$, $\{a, b, e_2\}$, or $\{e_1, a, b, e_2\}$ to P and, calling this set S, assign it weight $w = n - |S|$. Intuitively, the reduction results in replacing a_1 and a_n with e_1 and e_2, respectively, if either a_1 or a_n is external in $\mathcal{T}'$ and, relative to $\mathcal{T}'$, replacing the internal elements with a and b. Figure 5.4 depicts an example of the short-chain reduction, where $\mathcal{T}$ and $\mathcal{T}'$ are the trees before, and $\mathcal{S}$ and $\mathcal{S}'$ are the trees after applying the short-chain reduction. Here a_1 is external in $\mathcal{T}'$, but a_n is internal in $\mathcal{T}'$, and so the chain $(a_1, a_2, \ldots, a_n)$ is replaced with the chain (e_1, a, b).

Cluster Reduction: For $|A| \geq 2$, if A is a minimal vertex cluster in both $\mathcal{T}$ and $\mathcal{T}'$ with no common non-trivial subtree and that does not cross P, then replace $\mathcal{T}$ and $\mathcal{T}'$ with two new pairs of weighted trees. The first pair of weighted trees is the *cluster-tree pair* $\mathcal{T}|A$ and $\mathcal{T}'|A$ whose associated weighted set P_A is precisely the subset of P whose members are subsets of A. The second pair of weighted trees is the *cluster-reduced tree*

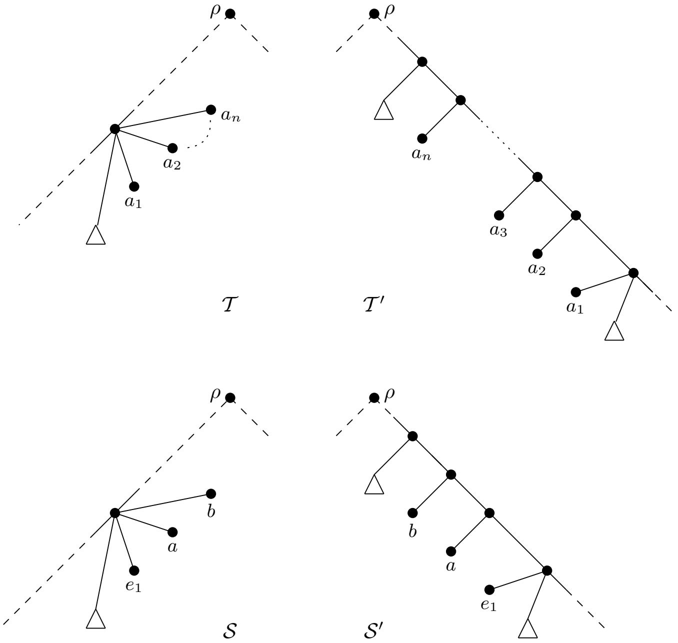


Figure 5.4: Two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ reduced under the short-chain reduction, where $\mathcal{S}$ and $\mathcal{S}'$ are the resulting trees. Dotted lines indicate regions of the chain $(a_1, a_2, \ldots, a_n)$. Note that a_1 is external in $\mathcal{T}'$ while a_n is internal in $\mathcal{T}'$

pair $\mathcal{T}_a$ and $\mathcal{T}'_a$ that is obtained from $\mathcal{T}$ and $\mathcal{T}'$, respectively, by replacing $\mathcal{T}(A)$ and $\mathcal{T}'(A)$ with a pendant edge ending in a new leaf labeled a or a new leaf labeled a depending on whether the subtree can be obtained with or without refinement. The weighted set P_a associated with this pair is the subset of P whose members are subsets of $((X - A) \cup \{\rho\})$. An example of the cluster reduction is illustrated in Figure 5.5.

An agreement forest $\mathcal{F}$ for a pair of weighted rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ is *legitimate* if $\mathcal{F}$ is acyclic and satisfies the following property (P), where, depending on the set in P, the elements e_1 and e_2 may or may not exist:

(P): If $\{e_1, a, b, c, e_2\} \in P$, then exactly one of the following holds:

(i) $\{e_1, a, b, c, e_2\}$ is a subset of a label set in $\mathcal{F}$,

(ii) $\{a\}$, $\{b\}$, and $\{c\}$ are label sets in $\mathcal{F}$, and e_1 and e_2 are in separate label sets in $\mathcal{F}$,

(iii) $\{a, b\}$ and $\{c\}$ are label sets in $\mathcal{F}$, e_1 and e_2 are in separate label sets in $\mathcal{F}$ and, relative to (e_1, a, b, c, e_2), if e_1 or e_2 is internal in $\mathcal{T}$, then $\{e_1\}$ or $\{e_2\}$ is a label set in $\mathcal{F}$, respectively,

(iv) $\{a\}$ and $\{b, c\}$ are label sets in $\mathcal{F}$, e_1 and e_2 are in separate label sets in $\mathcal{F}$ and, relative to (e_1, a, b, c, e_2), if e_1 or e_2 is internal in $\mathcal{T}'$, then $\{e_1\}$ or $\{e_2\}$ is a label set in $\mathcal{F}$, respectively,

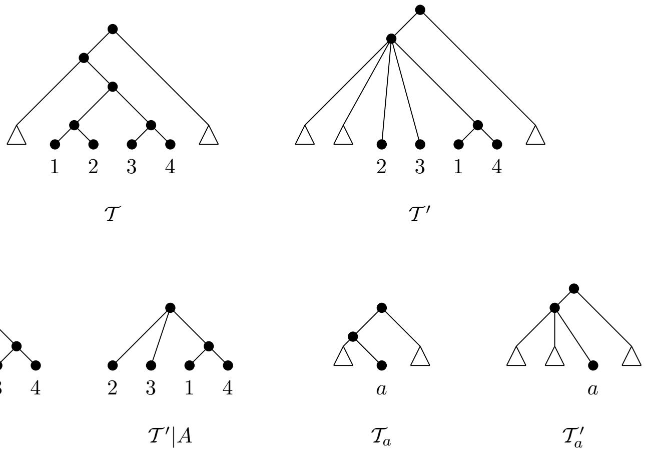


Figure 5.5: Two rooted phylogenetic trees $\mathcal{T}$ and $\mathcal{T}'$ divided under the cluster reduction applied to $A = \{1, 2, 3, 4\}$. The hybridization number of $\mathcal{T}$ and $\mathcal{T}'$ is the sum of the hybridization numbers of $\mathcal{T}|A$ and $\mathcal{T}'|A$, and $\mathcal{T}_a$ and $\mathcal{T}'_a$. Note that $\mathcal{T}|A$ has been replaced with a single vertex labeled a to obtain $\mathcal{T}_a$ from $\mathcal{T}$, whereas $\mathcal{T}'|A$ has been replaced with a pendant edge ending in a new leaf labeled a to obtain $\mathcal{T}'_a$ from $\mathcal{T}'$.

while if $\{e_1, a, b, e_2\} \in P$, then exactly one of the following holds:

(I) $\{e_1, a, b, e_2\}$ is a subset of a label set in $\mathcal{F}$,

(II) $\{a\}$ and $\{b\}$ are label sets in $\mathcal{F}$, and e_1 and e_2 are in separate label sets in $\mathcal{F}$.

Furthermore, referring to property (P), for an arbitrary agreement forest of $\mathcal{T}$ and $\mathcal{T}'$, we define the *weight* of $\mathcal{F}$, denoted by $w(\mathcal{F})$, to be

$$\begin{aligned} w(\mathcal{F}) = |\mathcal{F}| - 1 \quad &+ \sum_{S=\{e_1,a,b,c,e_2\}\in P; S \text{ satisfies (ii) in } \mathcal{F}} w_1(S) \\ &+ \sum_{S=\{e_1,a,b,c,e_2\}\in P; S \text{ satisfies (iii) in } \mathcal{F}} w_2(S) \\ &+ \sum_{S=\{e_1,a,b,c,e_2\}\in P; S \text{ satisfies (iv) in } \mathcal{F}} w_3(S) \\ &+ \sum_{S=\{e_1,a,b,e_2\}\in P; S \text{ satisfies (II) in } \mathcal{F}} w(S). \end{aligned}$$

We denote the minimum weight of a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ by $f(\mathcal{T}, \mathcal{T}')$. Observe that $f(\mathcal{T}, \mathcal{T}') \geq h(\mathcal{T}, \mathcal{T}')$ as the weightings are non-negative, and $f(\mathcal{T}, \mathcal{T}') = h(\mathcal{T}, \mathcal{T}')$ whenever P is empty.

Lemmas 5.13, 5.14, and 5.15 are key lemmas in proving that Minimum Hybridization is fixed-parameter tractable. Each lemma describes how particular common configurations in $\mathcal{T}$ and $\mathcal{T}'$ behave in a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. For convenience in the proofs of these lemmas, we will frequently refer to the property of a forest $\mathcal{F}$ that the trees in $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are edge-disjoint rooted subtrees of $\mathcal{T}$ as *no two label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$*.

Much of the proofs in the rest of this section involve taking a given legitimate-agreement forest $\mathcal{F}$, modifying it slightly, and showing that the resulting partition $\mathcal{F}'$ is also a legitimate-agreement forest. Two of the repetitive tasks is to show that $\mathcal{F}'$ is an agreement forest and acyclic. To avoid some of the repetition and to provide some intuition, let $\mathcal{L}_i \in \mathcal{F}$ and $\mathcal{L}'_i \in \mathcal{F}'$ with $\mathcal{L}'_i \subseteq \mathcal{L}_i$. First observe that if $\mathcal{L}'_i$ is the label set of a pendant subtree of $\mathcal{T}|\mathcal{L}_i$, then $\mathcal{L}'_i$ is the label set of a pendant subtree of $\mathcal{T}'|\mathcal{L}_i$. Analogously, if $\mathcal{L}'_i$ is the label set of a pendant subtree of $\mathcal{T}'|\mathcal{L}_i$, then $\mathcal{L}'_i$ is the label set of a pendant subtree of $\mathcal{T}|\mathcal{L}_i$. Second, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $\mathcal{T}|\mathcal{L}'_i$ and $\mathcal{T}'|\mathcal{L}'_i$ have a common binary refinement. Third, if $\mathcal{L}_r, \mathcal{L}_s \in \mathcal{F} \cap \mathcal{F}'$, then $(\mathcal{L}_r, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}}$ if and only if it is an arc in $G_{\mathcal{F}'}$. Since $\mathcal{F}$ is acyclic, it follows that if $G_{\mathcal{F}'}$ contains a directed cycle, then this cycle must use a vertex in $\mathcal{F}' - \mathcal{F}$. Furthermore, if $\mathcal{L}_i \neq \mathcal{L}_r$, then, as $\mathcal{L}_r$ and $\mathcal{L}_i$ are edge-disjoint in $\mathcal{T}$ and $\mathcal{T}'$, we have $(\mathcal{L}_r, \mathcal{L}'_i)$ is an arc in $G_{\mathcal{F}'}$ if and only if $(\mathcal{L}_r, \mathcal{L}_i)$ is an arc in $G_{\mathcal{F}}$. Also, if $(\mathcal{L}'_i, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}'}$, then $(\mathcal{L}_i, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}}$. Specializing these observations to when $\mathcal{F}'$ is a *refinement* of $\mathcal{F}$, that is, for each $\mathcal{L}'_i \in \mathcal{F}'$, we have $\mathcal{L}'_i \subseteq \mathcal{L}_i$ for some $\mathcal{L}_i \in \mathcal{F}$, it is straightforward to prove the following lemma.

Lemma 5.11. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees, and let $\mathcal{F}$ be an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Let $\mathcal{F}'$ be an agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ that is a refinement of $\mathcal{F}$. Then $\mathcal{F}'$ is acyclic.*

The above observations will be freely used in the rest of this section. The next lemma is repeatedly used in the key lemmas to show that our modified agreement forest satisfies (P).

Lemma 5.12. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees, and let $\mathcal{F}$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Let S be an element of P such that S contains elements of the form e_1 and e_2, and let A be the label set of either a pendant subtree of $\mathcal{T}$ and $\mathcal{T}'$ that could be used for a subtree reduction or a chain of $\mathcal{T}$ and $\mathcal{T}'$ that could be used for a long-chain or short-chain reduction. Then there are no distinct label sets $\mathcal{L}_1, \mathcal{L}_2 \in \mathcal{F}$ such that $e_1 \in \mathcal{L}_1$, $e_2 \in \mathcal{L}_2$, and $\mathcal{L}_1 \cap A$ and $\mathcal{L}_2 \cap A$ both non-empty.*

Proof. Suppose that there exist such label sets $\mathcal{L}_1$ and $\mathcal{L}_2$. Clearly, S does not satisfy either (i) or (I) in the definition of (P). Assume S satisfies (ii). Most of the work in

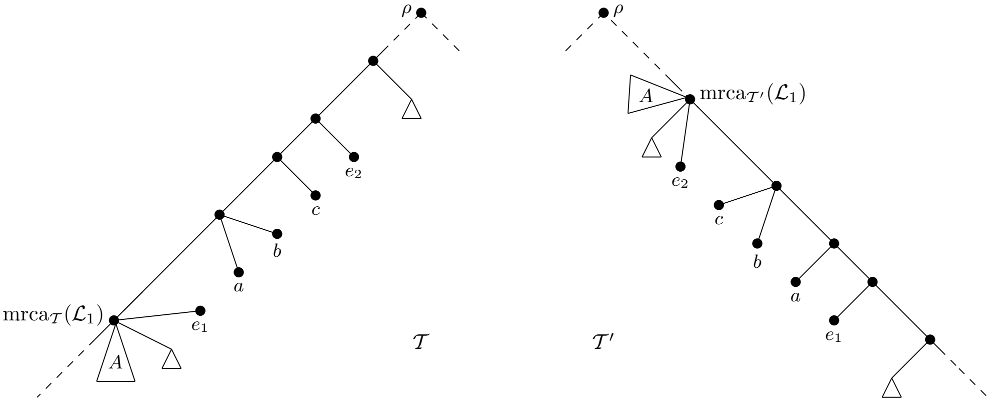

ρ
e_2
c
b
a
$\mathrm{mrca}_{\mathcal{T}}(\mathcal{L}_1)$
e_1
A
$\mathcal{T}$
$\mathcal{T}'$
ρ
A
$\mathrm{mrca}_{\mathcal{T}'}(\mathcal{L}_1)$
e_2
c
b
a
e_1

Figure 5.6: Set-up in the proof of Lemma 5.12 for when S satisfies (ii) in the definition of (P) and A is the label set of a pendant subtree of $\mathcal{T}$ and $\mathcal{T}'$. The roots of $\mathcal{T}(\mathcal{L}_1)$ and $\mathcal{T}'(\mathcal{L}_1)$ are indicated by $\mathrm{mrca}_{\mathcal{T}}(\mathcal{L}_1)$ and $\mathrm{mrca}_{\mathcal{T}'}(\mathcal{L}_1)$, respectively.

the proof is involved in eliminating this particular case. Since there exist such label sets $\mathcal{L}_1$ and $\mathcal{L}_2$, and $\mathcal{L}_1$ and $\mathcal{L}_2$ are edge-disjoint in $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that e_1 is external in one of the trees, while e_2 is external in the other tree. The upcoming argument is independent of whether or not a and b have the same parent or b and c have the same parent, thus, without loss of generality, we may assume e_1 is external in $\mathcal{T}$, while e_2 is external in $\mathcal{T}'$. Thus e_1 is internal in $\mathcal{T}'$ and e_2 is internal in $\mathcal{T}$. Furthermore, $\mathcal{T}(\mathcal{L}_2)$ contains the parents of a, b, and c in $\mathcal{T}$, and $\mathcal{T}'(\mathcal{L}_1)$ contains the parents of a, b, and c in $\mathcal{T}'$. As $\mathcal{F}$ is acyclic, it follows that either the roots of $\mathcal{T}'(\mathcal{L}_1)$ and $\mathcal{T}'(\mathcal{L}_2)$ coincide in $\mathcal{T}'$, in particular, both roots are $p_{\mathcal{T}'}(e_2)$, or the root of $\mathcal{T}'(\mathcal{L}_2)$ is an ancestor of the root of $\mathcal{T}'(\mathcal{L}_1)$.

If A is the label set of a pendant subtree, then, as $\mathcal{L}_1 \cap A$ and $\mathcal{L}_2 \cap A$ are both non-empty, the paths in $\mathcal{T}$ from any element in $\mathcal{L}_1 \cap A$ to ρ and from any element in $\mathcal{L}_2 \cap A$ to ρ meet at $p_{\mathcal{T}}(e_1)$, while the paths in $\mathcal{T}'$ from any element in $\mathcal{L}_1 \cap A$ to ρ and from any element in $\mathcal{L}_2 \cap A$ to ρ meet at $p_{\mathcal{T}'}(e_2)$. This set-up is depicted in Figure 5.6. Let $\mathcal{L}_2'$ denote the subset of elements of $\mathcal{L}_2$ for which $p_{\mathcal{T}}(e_1)$ is an ancestor. Since $\mathcal{T}|\mathcal{L}_2$ and $\mathcal{T}'|\mathcal{L}_2$ have a common binary refinement and $\mathcal{L}_2 \cap A$ is non-empty, each of the elements in $\mathcal{L}_2'$ is a descendant of $p_{\mathcal{T}'}(e_2)$ in $\mathcal{T}'$. Let $\mathcal{L}_1'$ denote the subset of elements of $\mathcal{L}_1$ for which $p_{\mathcal{T}'}(e_1)$ is not an ancestor in $\mathcal{T}'$. Let $\mathcal{F}'$ be the partition obtained from $\mathcal{F}$ by replacing $\mathcal{L}_1$, $\mathcal{L}_2$, $\{a\}$, $\{b\}$, and $\{c\}$ with $(\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\}$, $(\mathcal{L}_1 - \mathcal{L}_1') - \{e_1\}$, $\mathcal{L}_2'$, and $\mathcal{L}_1'$. Since $\mathcal{F}$ is an agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ and since $\mathcal{F}$ satisfies (P), it is easily checked that $\mathcal{F}'$ is an agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ that satisfies (P). To see that $\mathcal{F}'$ is acyclic, note that, up to $(\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\}$, $\mathcal{F}'$ is a refinement of $\mathcal{F}$. Moreover, for $\mathcal{L}_r, \mathcal{L}_s \in \mathcal{F} \cap \mathcal{F}'$, $(\mathcal{L}_r, (\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\})$ is an arc in $G_{\mathcal{F}'}$ if and only if $(\mathcal{L}_r, \mathcal{L}_2)$ is an arc in $G_{\mathcal{F}}$, and if $((\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\}, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}'}$, then $(\mathcal{L}_2, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}}$ unless the root of $\mathcal{T}'(\mathcal{L}_2)$ is not a strict ancestor of $p_{\mathcal{T}'}(e_2)$. In this exceptional instance, $(\mathcal{L}_1, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}}$ and, whenever $(\mathcal{L}_r, (\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\})$ is an arc in $G_{\mathcal{F}'}$, $(\mathcal{L}_r, \mathcal{L}_1)$ is an arc in $G_{\mathcal{F}}$. Using the observations prior to Lemma 5.11, a routine check shows that if there is a directed cycle in $G_{\mathcal{F}'}$, then there is a directed cycle in $G_{\mathcal{F}}$. It follows that $\mathcal{F}'$ is a legitimate agreement forest of $\mathcal{T}$ and $\mathcal{T}'$. But $w(\mathcal{F}') < w(\mathcal{F})$, contradicting the minimality of $\mathcal{F}$, and so A is not the label set of a pendant subtree.

Now assume that A is the set of elements of a chain $(a_1, a_2, \ldots, a_n)$ that could be used for a long-chain reduction. Since $\mathcal{L}_1 \cap A$ and $\mathcal{L}_2 \cap A$ are both non-empty, $p_{\mathcal{T}}(a_n) = p_{\mathcal{T}}(e_1)$, $p_{\mathcal{T}'}(a_1) = p_{\mathcal{T}'}(e_2)$, $a_1 \in \mathcal{L}_1$, and $a_n \in \mathcal{L}_2$. Thus a_1 is external in $\mathcal{T}'$ and a_n is external in $\mathcal{T}$. Also, $\mathcal{L}_1 \cap A = \{a_1\}$ and $\mathcal{L}_2 \cap A = \{a_n\}$. Furthermore, as $\mathcal{T}|\mathcal{L}_2$ and $\mathcal{T}'|\mathcal{L}_2$ have a common binary refinement, except for a_n, no element in $\mathcal{L}_2$ is a descendant of $p_{\mathcal{T}}(e_1)$ in $\mathcal{T}$ and, except for e_2, no element in $\mathcal{L}_2$ is a descendant of $p_{\mathcal{T}'}(e_2)$ in $\mathcal{T}'$. Let $\mathcal{L}_1'$ denote the subset of elements of $\mathcal{L}_1$ for which $p_{\mathcal{T}'}(e_1)$ is not an ancestor. Let $\mathcal{F}'$ be the partition obtained from $\mathcal{F}$ by replacing $\mathcal{L}_1$, $\mathcal{L}_2$, $\{a\}$, $\{b\}$, and $\{c\}$ with $(\mathcal{L}_2 - \{a_n\}) \cup \{e_1, a, b, c\}$, $(\mathcal{L}_1 - \mathcal{L}_1') - \{e_1\}$, $\{a_n\}$, and $\mathcal{L}_1'$. The set-up is similar to that of the last paragraph where we assumed A was a pendant subtree. Indeed, a similar argument now leads to the desired contradiction.

Next assume that A is the set of elements of a chain $(a_1, a_2, \ldots, a_n)$ that could be used for a short-chain reduction. Since $\mathcal{L}_1 \cap A$ and $\mathcal{L}_2 \cap A$ are both non-empty, $p_{\mathcal{T}}(a_n) = p_{\mathcal{T}}(e_1)$ and $p_{\mathcal{T}'}(a_1) = p_{\mathcal{T}'}(e_2)$ regardless in which tree the chain has a single parent. If the chain has a single parent in $\mathcal{T}$, then $\mathcal{L}_1 \cap A = \{a_1\}$ and $\mathcal{T}'(\mathcal{L}_2)$ contains a parent of one of the elements in $\{a_2, \ldots, a_n\}$. Now $|\mathcal{L}_2 \cap A| = 1$ otherwise $\mathcal{T}|\mathcal{L}_2$ and $\mathcal{T}'|\mathcal{L}_2$ do not have a common binary refinement, and so each of the at least two internal elements of the chain in $\mathcal{T}'$ that is not the element in $\mathcal{L}_2 \cap A$ is a singleton in $\mathcal{F}$. It is now easily checked that the partition

$$\mathcal{F}' = \{\mathcal{L}_i - A : \mathcal{L}_i \in \mathcal{F} - \{\mathcal{L}_2\}\} \cup \{\mathcal{L}_2 \cup A\}$$

is a legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$. But $w(\mathcal{F}') < w(\mathcal{F})$, contradicting the minimality of $\mathcal{F}$. Therefore assume that the chain has a single parent in $\mathcal{T}'$. Let $\mathcal{L}_2'$ denote the subset of elements of $\mathcal{L}_2$ for which $p_{\mathcal{T}}(e_1)$ is an ancestor. As $\mathcal{T}|\mathcal{L}_2$ and $\mathcal{T}'|\mathcal{L}_2$ have a common binary refinement and $\mathcal{L}_2 \cap A$ is non-empty, each of the elements in $\mathcal{L}_2'$ is a descendant of $p_{\mathcal{T}'}(e_2)$ in $\mathcal{T}'$. Let $\mathcal{L}_1'$ denote the subset of elements of $\mathcal{L}_1$ for which $p_{\mathcal{T}'}(e_1)$ is not an ancestor. Let $\mathcal{F}'$ be the partition obtained from $\mathcal{F}$ by replacing $\mathcal{L}_1$, $\mathcal{L}_2$, $\{a\}$, $\{b\}$, and $\{c\}$ with $(\mathcal{L}_2 - \mathcal{L}_2') \cup \{e_1, a, b, c\}$, $(\mathcal{L}_1 - \mathcal{L}_1') - \{e_1\}$, $\mathcal{L}_2'$, and $\mathcal{L}_1'$. This set-up is again similar to that when we assumed A was a pendant subtree and, as above, a similar argument leads to the desired contradiction. It now follows that S does not satisfy (ii).

If S satisfies (iii), then either $\mathcal{L}_1$ or $\mathcal{L}_2$ edge-overlap with $\{a, b\}$ in $\mathcal{T}'$; a contradiction. Therefore S does not satisfy (iii) and, similarly, S does not satisfy (iv). Lastly, assume S satisfies (II). Then, using the fact that e_1, a, b, and e_2 have the same parent in $\mathcal{T}$, a routine check shows that the partition $\mathcal{F}'$ obtained from $\mathcal{F}$ by replacing $\mathcal{L}_1$, $\mathcal{L}_2$, $\{a\}$, and $\{b\}$ with $\mathcal{L}_1 \cup \{a, b, e_2\}$ and $\mathcal{L}_2 - \{e_2\}$ or $\mathcal{L}_1 - \{e_1\}$ and $\mathcal{L}_2 \cup \{a, b, e_1\}$ depending on whether $\mathcal{T}'(\mathcal{L}_1)$ or $\mathcal{T}'(\mathcal{L}_2)$ includes the parents of a and b in $\mathcal{T}'$, respectively, is a legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$, But $w(\mathcal{F}') < w(\mathcal{F})$, contradicting the minimality of $\mathcal{F}$. Thus there are no such distinct label sets $\mathcal{L}_1$ and $\mathcal{L}_2$. □

Lemma 5.13. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees. Let A be the label set of a maximal pendant subtree in $\mathcal{T}$ and $\mathcal{T}'$ with the properties that $\mathcal{T}|A$ and $\mathcal{T}'|A$ have a common binary refinement and A does not cross P. Then, for every legitimate-agreement forest $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, A is a subset of a label set in $\mathcal{F}$.*

Proof. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Suppose that two subsets, $\mathcal{L}_i$ and $\mathcal{L}_j$ say, have the property that $\mathcal{L}_i \cap A$ and $\mathcal{L}_j \cap A$ are both non-empty. If there are no such subsets $\mathcal{L}_i$ and $\mathcal{L}_j$ so that $\mathcal{L}_i \cap ((X \cup \{\rho\}) - A)$ and $\mathcal{L}_j \cap ((X \cup \{\rho\}) - A)$ are both non-empty, then it is easily checked that the partition

$$\{\mathcal{L}_i : \mathcal{L}_i \cap A = \emptyset, \mathcal{L}_i \in \mathcal{F}\} \cup \{\mathcal{L}_A\},$$

where $\mathcal{L}_A = \bigcup_{\mathcal{L}_i \in \mathcal{F} : \mathcal{L}_i \cap A \neq \emptyset} \mathcal{L}_i$, is legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ but with smaller weight than $\mathcal{F}$; a contradiction. Therefore, we may assume that we can choose $\mathcal{L}_i$ and $\mathcal{L}_j$ such that $\mathcal{L}_i \cap ((X \cup \{\rho\}) - A)$ and $\mathcal{L}_j \cap ((X \cup \{\rho\}) - A)$ are both non-empty. Because of this assumption, the pendant subtree with label set A cannot be obtained from $\mathcal{T}$ or $\mathcal{T}'$ by deleting a single edge. Let e denote the edge of $\mathcal{T}$ that is directed into the root of $\mathcal{T}(A)$ and let e' denote the edge of $\mathcal{T}'$ that is directed into the root of $\mathcal{T}'(A)$. Since no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$ or $\mathcal{T}'$, at most one of $\mathcal{T}(\mathcal{L}_i)$ and $\mathcal{T}(\mathcal{L}_j)$ includes e and at most one of $\mathcal{T}'(\mathcal{L}_i)$ and $\mathcal{T}'(\mathcal{L}_j)$ includes e'. Also, since $G_\mathcal{F}$ is acyclic, if $\mathcal{T}(\mathcal{L}_i)$ includes e, then $\mathcal{T}'(\mathcal{L}_j)$ does not include e'. Similar conclusions hold for the other combinations including e or e'. Let $\mathcal{F}'$ be the partition of $X \cup \{\rho\}$ obtained from $\mathcal{F}$ by replacing $\mathcal{L}_i$ and $\mathcal{L}_j$ with $\mathcal{L}_i \cup \mathcal{L}_j$. It follows from the above conclusions and the observations prior to Lemma 5.11 that $\mathcal{F}'$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, as $\mathcal{F}$ satisfies (P), it follows by Lemma 5.12 that $\mathcal{F}'$ satisfies (P), and so $\mathcal{F}'$ is a legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$. But, as $w(\mathcal{F}') < w(\mathcal{F})$, we obtain a contradiction to the minimality of $\mathcal{F}$. This contradiction completes the proof of the lemma. □

Lemma 5.14. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees. Let*

$(a_1, a_2, \ldots, a_n)$ be a maximal chain of both $\mathcal{T}$ and $\mathcal{T}'$ that does not cross P with properties (i)-(iii) in the definition of the long-chain reduction. Then, for every legitimate-agreement forest $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, one of the following holds:

(i) *$\{a_1, a_2, \ldots, a_n\}$ is a subset of a label set in $\mathcal{F}$,*

(ii) *no label set in $\mathcal{F}$ contains at least two elements of the chain and, if a_i is an internal element of both $\mathcal{T}$ and $\mathcal{T}'$, then $\{a_i\}$ is a singleton in $\mathcal{F}$, or*

(iii) *for either $\mathcal{T}$ or $\mathcal{T}'$, say $\mathcal{T}$, two elements of the chain are in the same label set precisely if they have the same parent and, moreover, if that parent is internal in $\mathcal{T}$, then the corresponding set contains no other elements of $X \cup \{\rho\}$.*

Proof. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Let $A = \{a_1, a_2, \ldots, a_n\}$. The proof is partitioned into two cases depending on which of the following properties, up to interchanging the roles of $\mathcal{T}$ and $\mathcal{T}'$, is satisfied by $\mathcal{F}$:

(A) For all $\mathcal{L}_i \in \mathcal{F}$ with $\mathcal{L}_i \cap A$ non-empty and $p_{\mathcal{T}}(a_1)$ an ancestor of all elements in $\mathcal{L}_i - A$ in $\mathcal{T}$, the vertex $p_{\mathcal{T}'}(a_1)$ is an ancestor of all elements in $\mathcal{L}_i - A$ in $\mathcal{T}'$.

(B) There is a label set, $\mathcal{L}_i$ say, in $\mathcal{F}$ with both $\mathcal{L}_i \cap A$ and $\mathcal{L}_i - A$ non-empty and such that, in $\mathcal{T}$, the vertex $p_{\mathcal{T}}(a_1)$ is an ancestor of all elements in $\mathcal{L}_i - A$, but, in $\mathcal{T}'$, the vertex $p_{\mathcal{T}'}(a_1)$ is not an ancestor of all elements in $\mathcal{L}_i - A$.

Case (A). Let J index the label sets of $\mathcal{F}$ that contain elements of the chain. More precisely,

$$J = \{j \in \{\rho, 1, 2, \ldots, k\} : \mathcal{L}_j \cap \{a_1, a_2, \ldots, a_n\} \neq \emptyset\}.$$

Relative to the chain $(a_1, a_2, \ldots, a_n)$, we will call an edge of $\mathcal{T}$ or $\mathcal{T}'$ a *non-pendant chain edge* if the edge is not incident with an element in A, but it is incident with an internal parent in $\mathcal{T}$ or $\mathcal{T}'$, respectively. The analysis of (A) is partitioned into two subcases:

(I) There exists (not necessarily distinct) label sets $\mathcal{L}_i$ and $\mathcal{L}_{i'}$ in $\mathcal{F}$ such that $\mathcal{T}(\mathcal{L}_i)$ and $\mathcal{T}'(\mathcal{L}_{i'})$ contain a non-pendant edge of the chain $(a_1, a_2, \ldots, a_n)$ in $\mathcal{T}$ and $\mathcal{T}'$, respectively.

(II) $\mathcal{F}$ contains no such label sets $\mathcal{L}_i$ and $\mathcal{L}_{i'}$.

Subcase (I). Without loss of generality, we may assume that $\mathcal{L}_i$ and $\mathcal{L}_{i'}$ are chosen so that the roots of $\mathcal{T}(\mathcal{L}_i)$ and $\mathcal{T}'(\mathcal{L}_{i'})$ are as close to ρ as possible in $\mathcal{T}$ and $\mathcal{T}'$. If neither

$\mathcal{L}_i$ nor $\mathcal{L}_{i'}$ contains an element of A, then, as A contains no common non-trivial pendant subtree, it is easily seen that $\mathcal{F}$ satisfies (ii) in the statement of the lemma. Thus, we may assume that either $\mathcal{L}_i$ or $\mathcal{L}_{i'}$, say $\mathcal{L}_i$, contains an element of A. If $\mathcal{L}_{i'}$ does not contain an element of A, then one of the following holds: (a) for some $a_j, a_{j'} \in (\mathcal{L}_i \cap A)$, we have $p_{\mathcal{T}}(a_j) \neq p_{\mathcal{T}}(a_{j'})$ but $p_{\mathcal{T}'}(a_j) = p_{\mathcal{T}'}(a_{j'})$; (b) $a_1 \in \mathcal{L}_i$, $a_n \notin \mathcal{L}_i$, and a_1 is an external element of the chain in $\mathcal{T}'$; or (c) $a_n \in \mathcal{L}_i$, $a_1 \notin \mathcal{L}_i$, and a_n is an external element of the chain in $\mathcal{T}'$. Since $\mathcal{L}_{i'}$ does not contain an element of A, it follows that if a label set in $\mathcal{F}$ contains an element in A and an element in $(X \cup \{\rho\}) - A$, then that label set contains either a_1 or a_n, in which case a_1 or a_n are external in $\mathcal{T}'$, respectively, but no other elements from A. Furthermore, no label set in $\mathcal{F}$ contains two elements of A that have different parents in $\mathcal{T}'$. It is now easily checked that, as $\mathcal{F}$ is a legitimate-agreement forest of minimum weight, $\mathcal{F}$ satisfies (iii) if (a) holds and $\mathcal{F}$ satisfies either (ii) or (iii) if (b) or (c) holds. In all cases, if (iii) holds, then $\mathcal{T}'$ is the distinguished tree.

Now assume that $\mathcal{L}_i$ and $\mathcal{L}_{i'}$ contain an element of A. The rest of the analysis for (I) is partitioned into two parts. Let $\mathcal{L}'_i$ denote the subset of elements in $\mathcal{L}_i - A$ that are descendants of $p_{\mathcal{T}}(a_1)$, and let X'_1 denote the subset of elements in $\mathcal{L}_i - A$ that are descendants of $p_{\mathcal{T}'}(a_1)$ in $\mathcal{T}'$. Analogously, let $\mathcal{L}'_{i'}$ denote the subset of elements in $\mathcal{L}_{i'} - A$ that are descendants of $p_{\mathcal{T}'}(a_1)$, and let X_1 denote the subset of elements in $\mathcal{L}_{i'} - A$ that are descendants of $p_{\mathcal{T}}(a_1)$ in $\mathcal{T}$.

For the first part, suppose that $\mathcal{L}'_i = X'_1$ and $\mathcal{L}'_{i'} = X_1$. Let $\mathcal{F}'$ be the forest obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label set $\mathcal{L}_a = \bigcup_{j\in J} \mathcal{L}_j$. Since we are in case (A), $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. To see that $\mathcal{F}'$ is acyclic, consider the directed graphs $G_{\mathcal{F}}$ and $G_{\mathcal{F}'}$. The vertex set of $G_{\mathcal{F}'}$ is obtained from $G_{\mathcal{F}}$ by deleting the vertices $\mathcal{L}_j$ for all $j \in J$, and adding the new vertex $\mathcal{L}_a$. Also, if $\mathcal{L}_r, \mathcal{L}_s \in \mathcal{F}' - \{\mathcal{L}_a\}$, then $(\mathcal{L}_r, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}'}$ if and only if $(\mathcal{L}_r, \mathcal{L}_s)$ is an arc in $G_{\mathcal{F}}$. Regarding the arcs in $G_{\mathcal{F}'}$ incident with $\mathcal{L}_a$, there are two instances to consider. First assume that $\mathcal{L}_i - A$ is non-empty and contains an element that is not a descendant of $p_{\mathcal{T}}(a_1)$ in $\mathcal{T}$. Then $\mathcal{L}_i - A$ contains an element that is not a descendant of $p_{\mathcal{T}'}(a_1)$ in $\mathcal{T}'$. Since $G_{\mathcal{F}}$ is acyclic, there is no arc from $\mathcal{L}_{i'}$ to $\mathcal{L}_i$ in $G_{\mathcal{F}}$; otherwise, $G_{\mathcal{F}}$ contains a directed 2-cycle. Therefore, either the roots of $\mathcal{T}'(\mathcal{L}_i)$ and $\mathcal{T}'(\mathcal{L}_{i'})$ coincide in $\mathcal{T}'$ or the root of $\mathcal{T}'(\mathcal{L}_{i'})$ is a descendant of the root of $\mathcal{T}'(\mathcal{L}_i)$. Since the root of $\mathcal{T}(\mathcal{L}_a)$ is the same as the root of $\mathcal{T}(\mathcal{L}_i)$ in $\mathcal{T}$, it follows that if $(\mathcal{L}_r, \mathcal{L}_a)$ is an arc in $G_{\mathcal{F}'}$, then $(\mathcal{L}_r, \mathcal{L}_i)$ and $(\mathcal{L}_r, \mathcal{L}_{i'})$ are arcs in $G_{\mathcal{F}}$. Moreover, if $(\mathcal{L}_a, \mathcal{L}_r)$ is an arc in $G_{\mathcal{F}'}$, then either $(\mathcal{L}_a, \mathcal{L}_i)$ or $(\mathcal{L}_a, \mathcal{L}_{i'})$ is an arc in $G_{\mathcal{F}}$. Thus, as $G_{\mathcal{F}}$ is acyclic, $G_{\mathcal{F}'}$ is also acyclic.

Second assume that either $\mathcal{L}_i - A$ is empty or if $\mathcal{L}_i - A$ is non-empty, then it only contains elements that are descendants of $p_{\mathcal{T}}(a_1)$. Because of the first instance, we may assume that the analogous property holds for $\mathcal{L}_{i'}$ and $\mathcal{T}'$. Then the root of $\mathcal{T}(\mathcal{L}_a)$ is $p_{\mathcal{T}}(a_n)$ in $\mathcal{T}$ and the root of $\mathcal{T}'(\mathcal{L}_a)$ is $p_{\mathcal{T}'}(a_n)$ in $\mathcal{T}'$. Suppose that $G_{\mathcal{F}'}$ contains the directed cycle C. Then, as $G_{\mathcal{F}}$ is acyclic, C must contain $\mathcal{L}_a$. Let $\mathcal{L}_l$ and $\mathcal{L}_m$ denote the vertices in C that immediately precede and succeed $\mathcal{L}_a$, respectively, in this directed cycle. Except for $\mathcal{L}_a$, all other vertices in C are also vertices in $G_{\mathcal{F}}$. Thus either $(\mathcal{L}_i, \mathcal{L}_m)$ or $(\mathcal{L}_{i'}, \mathcal{L}_m)$ is an arc in $G_{\mathcal{F}}$. But $(\mathcal{L}_l, \mathcal{L}_i)$ and $(\mathcal{L}_l, \mathcal{L}_{i'})$ are also arcs in $G_{\mathcal{F}}$, implying that $G_{\mathcal{F}}$ contains a directed cycle; a contradiction. Thus $G_{\mathcal{F}'}$ is acyclic. Hence $\mathcal{F}'$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, as $\mathcal{F}$ satisfies (P), it follows by Lemma 5.12 that $\mathcal{F}'$ satisfies (P). Thus if $|J| \geq 2$, then $w(\mathcal{F}') < w(\mathcal{F})$, contradicting the minimality of $\mathcal{F}$. Therefore, A is a subset of a label set in $\mathcal{F}$ and so $\mathcal{F}$ satisfies (i) in the statement of the lemma.

For the second part, suppose that either $\mathcal{L}'_i \neq X'_1$ or $\mathcal{L}'_{i'} \neq X_1$. Without loss of generality, we may assume that $\mathcal{L}'_i \neq X'_1$ and $a_i \in \mathcal{L}_i \cap A$. Since we are in case (A), this implies that $p_{\mathcal{T}}(a_1)$ is not an ancestor of all elements in $\mathcal{L}_i - A$. Let $\mathcal{L}''_i$ denote the subset of elements in $\mathcal{L}_i - A$ that are not descendants of $p_{\mathcal{T}}(a_1)$. Because we are in case (A), $X'_1 \neq \mathcal{L}_i - A$, and so there is an element in $\mathcal{L}_i - A$ that is not a descendant of $p_{\mathcal{T}'}(a_1)$ in $\mathcal{T}'$. Furthermore, we may assume that $\mathcal{L}'_i$ is non-empty. To see this, observe that if $\mathcal{L}_i$ and $\mathcal{L}_{i'}$ are distinct, then X'_1 is empty, and so $\mathcal{L}'_i$ is non-empty. Also, if $\mathcal{L}_i$ and $\mathcal{L}_{i'}$ are the same, then, without loss of generality, we may assume that $\mathcal{L}'_i$ is non-empty.

First assume that either a_i is internal in both $\mathcal{T}$ and $\mathcal{T}'$, or $a_i = a_1$. If $(\mathcal{L}_i - A) \cap X'_1$ is non-empty, then, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $\mathcal{L}'_i \subseteq X'_1$. Furthermore, if $a_i \neq a_1$ or $a_i = a_1$ and a_1 is internal in $\mathcal{T}'$, then the same reasoning implies that $X'_1 \cap \mathcal{L}''_i$ is empty. But then $X'_1 = \mathcal{L}'_i$; a contradiction. Therefore, assume that $a_i = a_1$ and a_1 is external in $\mathcal{T}'$. If $a_n \notin \mathcal{L}_i$, then, as $\mathcal{F}$ is a legitimate-agreement forest of minimum weight, $\mathcal{F}$ satisfies (ii) in the statement of the lemma. So assume that $a_n \in \mathcal{L}_i$. If a_n is internal in $\mathcal{T}$, then, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, another check shows that $X'_1 \cap \mathcal{L}''_i$ is empty and so $X'_1 = \mathcal{L}'_i$. So now assume that a_n is external in $\mathcal{T}$, and therefore internal in $\mathcal{T}'$. Again as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, it is straightforward to check that, for any two elements in $\mathcal{L}''_i \cap X'_1$ the path in $\mathcal{T}$ from each of these elements to ρ meets the path from a_n to ρ in exactly one place. With this in hand, let $\mathcal{F}'$ be the partition of $X \cup \{\rho\}$ obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label sets $\bigcup_{j \in J} \mathcal{L}_j - (\mathcal{L}''_i \cap X'_1)$ and $\mathcal{L}''_i \cap X'_1$. Clearly, $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, and it is easily checked that, as $\mathcal{F}$ is acyclic, $\mathcal{F}'$ is acyclic. Furthermore, by Lemma 5.12, $\mathcal{F}'$ satisfies (P). Thus $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. But, in $\mathcal{F}$, each of the elements of the chain that are internal in both $\mathcal{T}$ and $\mathcal{T}'$ are singletons. Since there are at least three such elements, $w(\mathcal{F}') < w(\mathcal{F})$; a contradiction.

Now say that $(\mathcal{L}_i - A) \cap X'_1$ is empty. As $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, for any two elements in $\mathcal{L}'_i$, the path in $\mathcal{T}'$ from each of these elements to ρ meets the path from a_n to ρ in exactly one place. If a_1 is external in $\mathcal{T}$, not in $\mathcal{L}_i$, and the label set containing a_1 contains elements in $(X \cup \{\rho\}) - A$, then, as we are in case (A), $p_{\mathcal{T}}(a_1)$ and $p_{\mathcal{T}'}(a_1)$ are ancestors of each of the elements in this label set. The same reasoning also shows that if a_n is external in $\mathcal{T}$ and not in $\mathcal{L}_i$, then its label set contains no elements in $(X \cup \{\rho\}) - A$. Furthermore, if a_j and a_k are internal elements of both $\mathcal{T}$ and $\mathcal{T}'$, then, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, the label set containing a_j is a subset of A if $p_{\mathcal{T}}(a_j) \neq p_{\mathcal{T}}(a_i)$. Also, as no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$, the elements a_j and a_k are in separate label sets in $\mathcal{F}$ if $p_{\mathcal{T}}(a_j) \neq p_{\mathcal{T}}(a_k)$. Thus there are two such subsets of A in $\mathcal{F}$. Now let $\mathcal{F}'$ be the partition of $X \cup \{\rho\}$ obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label sets $\bigcup_{j \in J} \mathcal{L}_j - \mathcal{L}'_i$ and $\mathcal{L}'_i$. It is clear that $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, and, by Lemma 5.12, that $\mathcal{F}'$ satisfies (P). Moreover, it is easily checked that, as $\mathcal{F}$ is acyclic, $\mathcal{F}'$ is acyclic. But $w(\mathcal{F}') < w(\mathcal{F})$; a contradiction.

It now follows that we may assume that $\mathcal{L}_i \cap A = \{a_n\}$, where a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$. By considering $\mathcal{T}$, it is easily seen that if a_j and a_k are internal elements in $\mathcal{T}$ and $a_n \notin \{a_j, a_k\}$, then the label set in $\mathcal{F}$ containing a_j is a subset of A, and a_j and a_k can only be in the same label set in $\mathcal{F}$ if they have the same parent in $\mathcal{T}$. Now consider $\mathcal{T}'$. If $p_{\mathcal{T}'}(a_1)$ is an ancestor of an element in $\mathcal{L}_i$, then $\mathcal{F}$ satisfies (ii) in the statement of the lemma. Therefore, assume that $p_{\mathcal{T}'}(a_1)$ is not an ancestor of any element in $\mathcal{L}_i$, that is X'_1 is empty. Now $\mathcal{L}_{i'}$ contains an element of A and $\mathcal{T}'(\mathcal{L}_{i'})$ contains a non-pendant edge of $(a_1, a_2, \ldots, a_n)$. If $a_1 \in \mathcal{L}_{i'}$ and $\mathcal{L}_{i'}$ contains an element in $(X \cup \{\rho\}) - A$ that is not a descendant of $p_{\mathcal{T}'}(a_1)$ in $\mathcal{T}'$, then again $\mathcal{F}$ satisfies (ii) in the lemma. Noting that the label set containing a_1 can only contain another element of A if a_1 is internal in $\mathcal{T}$, it is now easily checked that, as $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, then $\mathcal{F}$ satisfies (iii) in the statement of the lemma with $\mathcal{T}$ as the distinguished tree unless a_n is internal in $\mathcal{T}$. But then a similar argument to that in the previous paragraph shows that the partition $\mathcal{F}'$ of $X \cup \{\rho\}$ obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label sets $\bigcup_{j \in J} \mathcal{L}_j - \mathcal{L}'_i$ and $\mathcal{L}'_i$ is a legitimate-agreement forest of smaller weight than $\mathcal{F}$; a contradiction. This completes the analysis of the second part, and therefore (I).

Subcase (II). We may assume that for one of the trees, say $\mathcal{T}$, whenever a label set $\mathcal{L}_r$ in $\mathcal{F}$ contains an element in A, then, unless this element is external, $\mathcal{L}_r \subseteq A$ and all elements in $\mathcal{L}_r$ have the same parent in $\mathcal{T}$. If $\mathcal{F}$ satisfies (ii) in the statement of the lemma, then we are done; so assume that this is not the case. Then there is a label set, $\mathcal{L}_i$ say, in $\mathcal{F}$ that contains at least two elements in A. In $\mathcal{T}'$, these elements have different parents. Since $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, it is now easily checked that $\mathcal{F}$ satisfies (iii) in the statement of the lemma. This completes the analysis of (II) and, therefore, (A).

Case (B). First note that, since $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $p_{\mathcal{T}'}(a_1)$ is not an ancestor of any element in $\mathcal{L}_i - A$ in $\mathcal{T}'$ unless $\mathcal{L}_i \cap A = \{a_1\}$ and a_1 is external in $\mathcal{T}$ or $\mathcal{L}_i \cap A = \{a_n\}$ and a_n is external in $\mathcal{T}'$. The analysis of this case is separated into two subcases:

(I) $\mathcal{L}_i \cap A$ contains an element that is internal in both $\mathcal{T}$ and $\mathcal{T}'$.
(II) $\mathcal{L}_i \cap A$ contains no element that is internal in both $\mathcal{T}$ and $\mathcal{T}'$.

Subcase (I). Let a_i be an element of $\mathcal{L}_i \cap A$ that is internal in both $\mathcal{T}$ and $\mathcal{T}'$. Let a_j be an element of A that is internal in both $\mathcal{T}$ and $\mathcal{T}'$. If $p_{\mathcal{T}}(a_j) \neq p_{\mathcal{T}}(a_i)$, then using the facts that no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$ or $\mathcal{T}'$, that $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, and that $\mathcal{F}$ is acyclic, it is easily checked that a_j is in a label set of $\mathcal{F}$ containing only elements of A and all of the elements in this set have the same parent in $\mathcal{T}$. Because of the requirement on internal parents in (iii) in the definition of the long-chain reduction, there are at least two such label sets. Also, if $p_{\mathcal{T}}(a_j) = p_{\mathcal{T}}(a_i)$ for some $j \neq i$ and $a_j \notin \mathcal{L}_i$, then, because $\mathcal{F}$ is acyclic and no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$, a_j is in a label set of $\mathcal{F}$ containing only elements of A and all of the elements in this set have the same parent. Furthermore, since $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, any two distinct elements in $\mathcal{L}_i - A$ intersect the path from a_n to ρ in $\mathcal{T}'$ in exactly one place.

We next consider a_1 if a_1 is external in either $\mathcal{T}$ or $\mathcal{T}'$, and a_n if a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$. If a_1 is external in $\mathcal{T}$, then, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $a_1 \notin \mathcal{L}_i$. Furthermore, a_1 is in a label set of $\mathcal{F}$ that contains no other elements of A and, moreover, both $p_{\mathcal{T}}(a_1)$ and $p_{\mathcal{T}'}(a_1)$ are ancestors of all elements in this label set. If a_1 is external in $\mathcal{T}'$, then it easily checked that a_1 behaves in the same way as elements in A that are internal in both $\mathcal{T}$ and $\mathcal{T}'$. Now consider a_n. If a_n is external in $\mathcal{T}$, then, as $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $a_n \notin \mathcal{L}_i$. Also, as $\mathcal{F}$ is acyclic, a_n is in a label set of $\mathcal{F}$ that contains no other elements of A and, moreover, $p_{\mathcal{T}}(a_n)$ is an ancestor of all elements in this label set, but $p_{\mathcal{T}}(a_1)$ is an ancestor of none. Furthermore, except for a_n, the vertex $p_{\mathcal{T}'}(a_1)$ is an ancestor of all elements in this set. Now assume that a_n is external in $\mathcal{T}'$. If $a_n \notin \mathcal{L}_i$, then, as no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}'$, the element a_n is the only element of A in its label set and, if this label set contains elements in $(X \cup \{\rho\}) - A$, then $p_{\mathcal{T}'}(a_1)$ is not an ancestor of any of these elements and all elements in $\mathcal{L}_i$ are descendants of $p_{\mathcal{T}'}(a_n)$.

With the above conclusions in hand and noting that it is possible for a_n to be external in $\mathcal{T}'$ and $a_n \in \mathcal{L}_i$, let J index the label sets of $\mathcal{F}$ that contain elements of the chain. Let $\mathcal{F}'$ be the forest obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label sets

$$\bigcup_{j \in J} \mathcal{L}_j - (\mathcal{L}_i - A) - (\mathcal{L}_n - \{a_n\}),$$

$\mathcal{L}'_i = \mathcal{L}_i - A$, and $\mathcal{L}'_n = \mathcal{L}_n - A$ if a_n is external in $\mathcal{T}$, where $\mathcal{L}_n$ is the label set in $\mathcal{F}$ containing a_n, and

$$\bigcup_{j \in J} \mathcal{L}_j - (\mathcal{L}_i - A)$$

and $\mathcal{L}'_i = \mathcal{L}_i - A$ if a_n is external in $\mathcal{T}'$. Note that $\mathcal{F}'$ is a partition of $X \cup \{\rho\}$. By considering the possibilities for a_1 and a_n, and noting that $p_{\mathcal{T}'}(a_1)$ is not an ancestor of any element in $\mathcal{L}_i - A$, it is clear that $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Using arguments similar to that used in (A), a straightforward check shows that, as $\mathcal{F}$ is acyclic, $\mathcal{F}'$ is acyclic. Since $\mathcal{F}$ satisfies (P), it follows by Lemma 5.12 that $\mathcal{F}'$ satisfies (P). Therefore, $\mathcal{F}'$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. But, as there are at least two label sets in $\mathcal{F}$ containing just elements of A, we have $w(\mathcal{F}') < w(\mathcal{F})$; contradicting the minimality of $\mathcal{F}$. Thus subcase (I) does not arise.

Subcase (II). First observe that $\mathcal{L}_i \cap A$ is a non-empty subset of $\{a_1, a_n\}$ and each of the elements in $\mathcal{L}_i \cap A$ is external in either $\mathcal{T}$ or $\mathcal{T}'$. Let $a_j, a_k \in A$ such that neither a_j nor a_k is a_1 if a_1 is external in either $\mathcal{T}$ or $\mathcal{T}'$ and neither a_j nor a_k is a_n if a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$. Assume first that $a_1 \in \mathcal{L}_i$. Since $\mathcal{F}$ is acyclic and no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$ or $\mathcal{T}'$, it is easily checked that a_j and a_k are in separate label sets in $\mathcal{F}$ and none of these label sets contain elements in $(X \cup \{\rho\}) - A$. Arguing similarly, if a_n is external in $\mathcal{T}$, and therefore internal in $\mathcal{T}'$, then $\{a_n\}$ is a label set in $\mathcal{F}$. It now follows that if a_n is not external in $\mathcal{T}'$, then $\mathcal{F}$ satisfies (ii) in the statement of the lemma. Therefore, assume that a_n is external in $\mathcal{T}'$. If $a_n \notin \mathcal{L}_i$, then, as no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}'$, the elements a_j and a_n are not in the same label set in $\mathcal{F}$ for all j. Thus $\mathcal{F}$ again satisfies (ii) in the statement of the lemma, so assume that $a_n \in \mathcal{L}_i$. Since $\mathcal{T}|\mathcal{L}_i$ and $\mathcal{T}'|\mathcal{L}_i$ have a common binary refinement, $p_{\mathcal{T}'}(a_n)$ is an ancestor of all elements in $\mathcal{L}_i$. Let $\mathcal{F}'$ be the partition of $X \cup \{\rho\}$ that is obtained from $\mathcal{F}$ by replacing $\mathcal{L}_i$ and all other label sets containing elements of A with the three sets $\mathcal{L}'_i$, $\mathcal{L}''_i$, and A, where $\mathcal{L}''_i$ contains precisely the elements in $\mathcal{L}_i - A$ that are descendants of $p_{\mathcal{T}'}(a_1)$ in $\mathcal{T}'$ and $\mathcal{L}'_i = \mathcal{L}_i - (A \cup \mathcal{L}''_i)$. Clearly, $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, using arguments similar to that used in (A), it is easily checked that, as $\mathcal{F}$ is acyclic, $\mathcal{F}'$ is acyclic. By Lemma 5.12, $\mathcal{F}'$ satisfies (P) as $\mathcal{F}$ satisfies (P), and so $\mathcal{F}'$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. But $\mathcal{F}$ has the property that $\{a_j\} \in \mathcal{F}$ for all $a_j \in A - \{a_1, a_n\}$. Since $|A| \geq 5$, this implies that $w(\mathcal{F}) < w(\mathcal{F}')$; a contradiction.

We may now assume that $a_n \in \mathcal{L}_i$ and $a_1 \notin \mathcal{L}_i$. First note that if $p_{\mathcal{T}'}(a_1)$ is an ancestor of an element in $\mathcal{L}_i$, then, as the label sets in $\mathcal{F}$ are edge-disjoint in $\mathcal{T}'$, $\mathcal{F}$ satisfies (ii) in the statement of the lemma. Thus we may also assume that $p_{\mathcal{T}'}(a_1)$ is not an ancestor of any element in $\mathcal{L}_i$. Since no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$, it follows that if $p_{\mathcal{T}}(a_j) \neq p_{\mathcal{T}}(a_k)$ or $p_{\mathcal{T}}(a_1) \neq p_{\mathcal{T}}(a_j)$, then a_j and a_k, and a_1 and a_j are in separate label sets in $\mathcal{F}$, respectively. Furthermore, unless $p_{\mathcal{T}}(a_j) = p_{\mathcal{T}}(a_n)$ and a_n is external in $\mathcal{T}'$, the label set containing a_j does not contain an element of $(X \cup \{\rho\}) - A$. Also, if a_1 is internal in $\mathcal{T}$, then its label set does not contain an element of $(X \cup \{\rho\}) - A$. It is now easily checked that if a_n is external in $\mathcal{T}$, then, as a_n is internal in $\mathcal{T}'$ and $\mathcal{F}$ is a legitimate-agreement forest of minimum weight, $\mathcal{F}$ satisfies (iii) in the statement of the lemma with $\mathcal{T}$ as the distinguished tree. Therefore, assume that a_n is external in $\mathcal{T}'$.

If a_1 is external in $\mathcal{T}$ and its label set contains an element in $(X \cup \{\rho\}) - A$ that is not an ancestor of $p_{\mathcal{T}'}(a_1)$, then $\mathcal{F}$ satisfies (ii) in the lemma. Thus if the label set containing a_1 contains an element in $(X \cup \{\rho\}) - A$, we may assume that it is a descendant of $p_{\mathcal{T}'}(a_1)$.

Now, apart from $\mathcal{L}_i$ and the label set containing a_1, if a_1 is external in $\mathcal{T}$, the only other possible label set, $\mathcal{L}_k$ say, in $\mathcal{F}$ that has a non-empty intersection with A and $(X \cup \{\rho\}) - A$ has the property that if $a_k \in \mathcal{L}_k \cap A$, then $p_{\mathcal{T}}(a_k) = p_{\mathcal{T}}(a_n)$. If no label set in $\mathcal{F}$ contains at least two elements of A each having a different parent in $\mathcal{T}'$ and there exists no such label set $\mathcal{L}_k$, then $\mathcal{F}$ satisfies (ii) in the statement of the lemma. Therefore, suppose that one of these two possibilities occur. Let $\mathcal{F}'$ be the partition of $X \cup \{\rho\}$ obtained from $\mathcal{F}$ by replacing $\mathcal{L}_i$, $\mathcal{L}_k$ if such a label set exists, and all other label sets containing elements in A with the sets $\mathcal{L}'_i$, $A \cup \mathcal{L}_1 \cup \mathcal{L}'_k$ and $\mathcal{L}''_k$, where $\mathcal{L}'_i = \mathcal{L}_i - \{a_n\}$, $\mathcal{L}_1$ is the label set of $\mathcal{F}$ containing a_1 if a_1 is external in $\mathcal{T}$, $\mathcal{L}''_k$ contains precisely the elements in $\mathcal{L}_k - A$ that are descendants of $p_{\mathcal{T}'}(a_1)$, and $\mathcal{L}'_k = \mathcal{L}_k - \mathcal{L}''_k$. Note that, as no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}$ or $\mathcal{T}'$, either $\mathcal{L}_1 - \{a_1\}$ or $\mathcal{L}''_k$ is empty. Clearly, $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, using the fact that one of the two above possibilities occur, it is easily checked that, as $\mathcal{F}$ is acyclic, $\mathcal{F}'$ is acyclic. Moreover, as $\mathcal{F}$ satisfies (P), it follows by

Lemma 5.12 that $\mathcal{F}'$ satisfies (P), and so $\mathcal{F}'$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. But $w(\mathcal{F}') < w(\mathcal{F})$ as $\mathcal{T}$ has at least three internal parents. This contradiction completes the proof of (B) and hence the lemma. □

Lemma 5.15. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees. Let $(a_1, a_2, \ldots, a_n)$ be a maximal chain of both $\mathcal{T}$ and $\mathcal{T}'$ that does not cross P with the property that in one of the trees, say $\mathcal{T}$, this chain has exactly one parent, while in the other tree $\mathcal{T}'$ this chain has at least three internal parents. Then, for every legitimate-agreement forest $\mathcal{F}$ for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, exactly one of the following holds:*

(i) *$\{a_1, a_2, \ldots, a_n\}$ is a subset of a label set in $\mathcal{F}$, or*

(ii) *no label set in $\mathcal{F}$ contains at least two elements of the chain and, if a_i is an internal element of $(a_1, a_2, \ldots, a_n)$ in $\mathcal{T}'$, then $\{a_i\}$ is a singleton in $\mathcal{F}$.*

Proof. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight, and let $A = \{a_1, a_2, \ldots, a_n\}$. Let J index the label sets of $\mathcal{F}$ that contain elements of A, and let $\mathcal{L}_a = \bigcup_{j \in J} \mathcal{L}_j$. Suppose that neither (i) nor (ii) holds for $\mathcal{F}$. If no label set in $\mathcal{F}$ contains at least two elements of A, then, relative to $\mathcal{T}'$, there is a label set in $\mathcal{F}$ that contains an internal element of the chain as well as an element of $(X \cup \{\rho\}) - A$. By considering the structure of $(a_1, a_2, \ldots, a_n)$ in $\mathcal{T}'$, it is easily seen that, as $(a_1, a_2, \ldots, a_n)$ has at least three internal elements relative to $\mathcal{T}'$, at least one of these internal elements is a singleton in $\mathcal{F}$. A routine check shows that, apart from one exceptional case, we can replace such a singleton and a label set in $\mathcal{F}$ that contains an internal element of the chain in $\mathcal{T}'$ as well as an element of $(X \cup \{\rho\}) - A$ with the union of these two sets to obtain a legitimate-agreement forest of $\mathcal{T}$ and $\mathcal{T}'$ that has smaller weight then $\mathcal{F}$; a contradiction. In the exceptional case, there is exactly one label set, $\mathcal{L}_i$ say, in $\mathcal{F}$ that contains an internal element of the chain in $\mathcal{T}'$ and an element in $(X \cup \{\rho\}) - A$, and this set has the properties that $|\mathcal{L}_i \cap A| = 1$, and $p_{\mathcal{T}'}(a_1)$ is an ancestor of all the elements in $\mathcal{L}_i - A$, but $p_{\mathcal{T}}(a_1)$ is not an ancestor of all the elements in $\mathcal{L}_i$. Since $\mathcal{F}$ is acyclic, it follows that each of the remaining internal elements of the chain in $\mathcal{T}'$ are singletons in $\mathcal{F}$. A straightforward check now shows that

$$\{\mathcal{L} - A : \mathcal{L} \in \mathcal{F}\} \cup \{A\}$$

is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, but with smaller weight than $\mathcal{F}$. This contradiction implies that there is a label set in $\mathcal{F}$ containing at least two elements of A. Without loss of generality, we may assume that this set is $\mathcal{L}_i$ and that $a_i \in \mathcal{L}_i \cap A$, where $i > i'$ for all $a_{i'} \in \mathcal{L}_i \cap A$.

Suppose that there exists an $\mathcal{L}_h \in \mathcal{F} - \{\mathcal{L}_i\}$ such that $|\mathcal{L}_h \cap A| \geq 1$, $|\mathcal{L}_h \cap ((X \cup \{\rho\}) - A)| \geq 1$, and let $a_h \in (\mathcal{L}_h \cap A)$. If $p_{\mathcal{T}'}(a_h)$ is a descendant of $p_{\mathcal{T}'}(a_i)$, then, as $|\mathcal{L}_i| \geq 2$ and no label sets in $\mathcal{F}$ edge-overlap in $\mathcal{T}'$, the vertex $p_{\mathcal{T}'}(a_h)$ in $\mathcal{T}'$ is an ancestor of all elements in $\mathcal{L}_h \cap ((X \cup \{\rho\}) - A)$. Because $\mathcal{F}$ is acyclic, it follows that the vertex $p_{\mathcal{T}}(a_h)$ in $\mathcal{T}$ is an ancestor of all elements in $\mathcal{L}_h \cap ((X \cup \{\rho\}) - A)$; otherwise $G_{\mathcal{F}}$ contains a directed 2-cycle. Now assume that $p_{\mathcal{T}'}(a_h)$ is an ancestor of $p_{\mathcal{T}'}(a_i)$. If $\mathcal{L}_i$ contains an element z that is not a descendant of $p_{\mathcal{T}'}(a_n)$ in $\mathcal{T}'$, then, as $G_{\mathcal{F}}$ is acyclic, $p_{\mathcal{T}}(a_n)$ is an ancestor of all elements in $\mathcal{L}_h$ in $\mathcal{T}$. Similarly, if $\mathcal{L}_h$ contains an element z that is not a descendant of $p_{\mathcal{T}'}(a_n)$ in $\mathcal{T}'$, then, as $G_{\mathcal{F}}$ is acyclic, $p_{\mathcal{T}}(a_n)$ is an ancestor of all elements in $\mathcal{L}_i$ in $\mathcal{T}$. Now let $\mathcal{F}'$ be the forest obtained from $\mathcal{F}$ by removing each label set $\mathcal{L}_j$ with $j \in J$ and inserting the new label set $\mathcal{L}_a$. Using the outcomes of the above two possibilities, it is easily seen that $\mathcal{F}'$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, as $\mathcal{F}$ satisfies (P), it follows by Lemma 5.12 that $\mathcal{F}'$ satisfies (P). Using the facts that $\mathcal{F}$ is acyclic and at least one of the label sets in $\mathcal{F}$ contains at least two elements of A, it is straightforward to show that $\mathcal{F}'$ is acyclic. But then $w(\mathcal{F}') < w(\mathcal{F})$; a contradiction to the minimality of $\mathcal{F}$. Thus $\mathcal{F}$ satisfies either (i) or (ii). □

We end this section by showing how the number of hybridization events for two rooted phylogenetic X-trees corresponds to this number for a cluster-tree pair and a cluster-reduced tree pair that have been obtained from the original tree pair by a cluster reduction.

Proposition 5.16. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees with an associated weighted set P. Let A be a vertex cluster of both $\mathcal{T}$ and $\mathcal{T}'$ that does not cross an element of P. Let $\mathcal{T}|A$ and $\mathcal{T}'|A$ and $\mathcal{T}_a$ and $\mathcal{T}'_a$ be the two pairs of weighted rooted phylogenetic trees obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the cluster reduction to A. Then*

$$f(\mathcal{T}, \mathcal{T}') = f(\mathcal{T}|A, \mathcal{T}'|A) + f(\mathcal{T}_a, \mathcal{T}'_a).$$

Proof. If $A = X$ the proposition clearly follows. Therefore, we may assume that $A \subset X$. Let ρ_A denote the root of $\mathcal{T}|A$ and $\mathcal{T}'|A$. First, we show that

$$f(\mathcal{T}, \mathcal{T}') \leq f(\mathcal{T}|A, \mathcal{T}'|A) + f(\mathcal{T}_a, \mathcal{T}'_a).$$

Let $\mathcal{F}_A$ be a legitimate-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ of minimum weight, and let $\mathcal{F}_a$ be a legitimate-agreement forest for $\mathcal{T}_a$ and $\mathcal{T}'_a$ of minimum weight. Let $\mathcal{L}_a$ denote the label set in $\mathcal{F}_a$ that contains a, and let $\mathcal{L}_{\rho_A}$ denote the label set in $\mathcal{F}_A$ that contains the root label ρ_A of $\mathcal{T}|A$ and $\mathcal{T}'|A$. Furthermore, let

$$\mathcal{F} = (\mathcal{F}_a \cup \mathcal{F}_A - \{\mathcal{L}_a, \mathcal{L}_{\rho_A}\}) \cup \{(\mathcal{L}_a \cup \mathcal{L}_{\rho_A}) - \{a, \rho_A\})\}.$$

Using the fact that $\mathcal{F}_a$ and $\mathcal{F}_A$ are acyclic-agreement forests for $\mathcal{T}_a$ and $\mathcal{T}'_a$, and $\mathcal{T}|A$ and $\mathcal{T}'|A$, respectively, it is easily checked that $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, as $\mathcal{F}_a$ and $\mathcal{F}_A$ satisfy (P) for P_a and P_A, respectively, it is clear that $\mathcal{F}$ satisfies (P) unless a is an element of an element in P_a. But by construction, a is a new label and so not in any element in P. Thus $\mathcal{F}$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. As $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 1$, we have

$$f(\mathcal{T}, \mathcal{T}') \leq w(\mathcal{F}) = f(\mathcal{T}|A, \mathcal{T}'|A) + f(\mathcal{T}_a, \mathcal{T}'_a).$$

Next, we show that

$$f(\mathcal{T}, \mathcal{T}') \geq f(\mathcal{T}|A, \mathcal{T}'|A) + f(\mathcal{T}_a, \mathcal{T}'_a).$$

Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. There are two cases to consider here:

(i) $\mathcal{F}$ contains mixed label sets $\mathcal{L}_{m_1}, \mathcal{L}_{m_2}, \ldots, \mathcal{L}_{m_r}$, that is, label sets that have a non-empty intersection with both A and $((X - A) \cup \{\rho\})$.

(ii) $\mathcal{F}$ contains no mixed label set.

Case (i). For all i, let $\{\mathcal{L}'_{m_i}, \mathcal{L}''_{m_i}\}$ be the partition of $\mathcal{L}_{m_i}$ such that $\mathcal{L}'_{m_i} \subseteq ((X - A) \cup \{\rho\})$ and $\mathcal{L}''_{m_i} \subseteq A$. Let

$$\mathcal{F}_a = \{(\mathcal{L}'_{m_1} \cup \{a\}), \mathcal{L}'_{m_2}, \ldots, \mathcal{L}'_{m_r}\} \cup \{\mathcal{L}_i : \mathcal{L}_i \subseteq ((X - A) \cup \{\rho\}) \text{ and } \mathcal{L}_i \in \mathcal{F}\}$$

and let

$$\mathcal{F}_A = (\mathcal{L}''_{m_1} \cup \mathcal{L}''_{m_2} \cup \ldots \cup \mathcal{L}''_{m_r} \cup \{\rho_A\}) \cup \{\mathcal{L}_i : \mathcal{L}_i \subseteq A \text{ and } \mathcal{L}_i \in \mathcal{F}\}.$$

Since A does not cross P and $\mathcal{F}$ satisfies (P), it is easily checked that $\mathcal{F}_a$ and $\mathcal{F}_A$ satisfy (P) for P_a and P_A, respectively. Moreover, as $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily seen that $\mathcal{F}_a$ and $\mathcal{F}_A$ are acyclic-agreement forests for $\mathcal{T}_a$ and $\mathcal{T}'_a$, and for

$\mathcal{T}|A$ and $\mathcal{T}'|A$, respectively. Therefore, as $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 1$,

$$f(\mathcal{T}_a, \mathcal{T}'_a) + f(\mathcal{T}|A, \mathcal{T}'|A) \leq w(\mathcal{F}_a) + w(\mathcal{F}_A) = f(\mathcal{T}, \mathcal{T}').$$

Case (ii). Since $G_{\mathcal{F}}$ is acyclic, the directed subdigraph of $G_{\mathcal{F}}$ induced by the label sets that are subsets of A is also acyclic. Thus this subdigraph has a vertex, $\mathcal{L}_0$ say, with indegree zero. Now let

$$\mathcal{F}_a = \{\mathcal{L}_i : \mathcal{L}_i \in \mathcal{F} \text{ and } \mathcal{L}_i \subseteq ((X - A) \cup \{\rho\})\} \cup \{a\},$$

and let

$$\mathcal{F}_A = \{\mathcal{L}_i : \mathcal{L}_i \in \mathcal{F} - \{\mathcal{L}_0\} \text{ and } \mathcal{L}_i \subseteq A\}) \cup \{\mathcal{L}_0 \cup \{\rho_A\}\}.$$

Since $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is clear that $\mathcal{F}_a$ and $\mathcal{F}_A$ are acyclic-agreement forests for $\mathcal{T}_a$ and $\mathcal{T}'_a$, and for $\mathcal{T}|A$ and $\mathcal{T}'|A$, respectively. Moreover, as $\mathcal{F}$ satisfies (P), $\mathcal{F}_a$ and $\mathcal{F}_A$ satisfy (P) for P_a and P_A, respectively. Thus, as $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 1$, we have

$$f(\mathcal{T}_a, \mathcal{T}'_a) + f(\mathcal{T}|A, \mathcal{T}'|A) \leq w(\mathcal{F}_a) + w(\mathcal{F}_A) = f(\mathcal{T}, \mathcal{T}').$$

This completes the proof of the proposition. □

5.6 Minimum Hybridization is Fixed-Parameter Tractable

In this section, we prove Theorem 5.2. We begin by showing that each of the subtree, long-chain, and short-chain reductions described in the last section preserves the minimum weight of a legitimate-agreement forest. For a chain $(a_1, a_2, \ldots, a_n)$ of $\mathcal{T}$, the partition of $\{a_1, a_2, \ldots, a_n\}$ defined by putting a_i and a_j in the same part precisely if $p_{\mathcal{T}}(a_i) = p_{\mathcal{T}}(a_j)$ is called the *parent partition* of $(a_1, a_2, \ldots, a_n)$ induced by $\mathcal{T}$.

Proposition 5.17. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees. Let $\mathcal{S}$ and $\mathcal{S}'$ be the pair of weighted rooted phylogenetic X'-trees obtained from $\mathcal{T}$ and $\mathcal{T}'$, respectively, by applying the subtree, long-chain, or short-chain reduction. Then $f(\mathcal{T}, \mathcal{T}') = f(\mathcal{S}, \mathcal{S}')$.*

Proof. It is an immediate consequence of Lemma 5.13 that if $\mathcal{S}$ and $\mathcal{S}'$ have been obtained from $\mathcal{T}$ and $\mathcal{T}'$ by an application of the subtree reduction, then the proposition holds. We next prove the result for when $\mathcal{S}$ and $\mathcal{S}'$ have been obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the long-chain reduction. The proof of the result for the short-chain reduction is similar and omitted.

Suppose that $(a_1, a_2, \ldots, a_n)$ is the common chain of $\mathcal{T}$ and $\mathcal{T}'$ used in this application of the long-chain reduction. Now let $\mathcal{F}_T$ be a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of minimum weight. Then, by Lemma 5.14 one of the following holds:

(i) $\{a_1, a_2, \ldots, a_n\}$ is a subset of a label set of $\mathcal{F}_T$,

(ii) no label set in $\mathcal{F}_T$ contains at least two elements of the chain and, if a_i is an internal element of both $\mathcal{T}$ and $\mathcal{T}'$, then $\{a_i\}$ is a singleton in $\mathcal{F}_T$, or

(iii) for either $\mathcal{T}$ or $\mathcal{T}'$, say $\mathcal{T}$, two elements of the chain are in the same label set precisely if they have the same parent and, moreover, if that parent is internal in $\mathcal{T}$, then the corresponding set contains no other elements of $X \cup \{\rho\}$.

Let $\mathcal{F}_S$ be the forest obtained from $\mathcal{F}_T$ by replacing a_1 and a_n with e_1 and e_2, respectively, if a_1 or a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$ and, then, depending on which of (i), (ii), or (iii) holds, respectively replace the remaining elements of A as follows: replace $a_1, a_2, \ldots, a_n$ with a, b, and c; collectively replace the label sets of the form $\{a_i\}$ with $\{a\}$, $\{b\}$, and $\{c\}$; or collectively replace the label sets of the form $\{a_i, a_{i+1}, \ldots, a_j\}$ with $\{a, b\}$ and $\{c\}$ and, if there is a label set of the form $\{e_1, a_2, \ldots, a_{i'}\}$ or $\{a_{j'}, a_{j'+1}, \ldots, e_2\}$, replace it with $\{e_1\}$ or $\{e_2\}$, respectively. Since $\mathcal{F}_T$ is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily checked that $\mathcal{F}_S$ is a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$. In the case that (ii) holds, the contribution of the singletons containing elements that are internal in both $\mathcal{T}$ and $\mathcal{T}'$ to $w(\mathcal{F}_T)$ is exactly the same as the contribution of $\{a\}$, $\{b\}$, and $\{c\}$ to $w(\mathcal{F}_S)$. Furthermore, in the case that (iii) holds, the contribution of the label sets containing just internal elements of A in $\mathcal{T}$ to $w(\mathcal{F}_T)$ is equal to the contribution of $\{a, b\}$, $\{c\}$, and $\{e_1\}$ and $\{e_2\}$ if either e_1 or e_2 are internal elements of the reduced chain in $\mathcal{S}$ respectively, to $w(\mathcal{F}_S)$. Thus $w(\mathcal{F}_S) = w(\mathcal{F}_T)$, and so $f(\mathcal{S}, \mathcal{S}') \leq f(\mathcal{T}, \mathcal{T}')$.

Now suppose that $\mathcal{F}_S$ is a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$ of minimum weight. As $\mathcal{F}_S$ is legitimate, one of the following holds, where e_1 and e_2 may or may not exist depending on whether a_1 or a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$:

(i) $\{e_1, a, b, c, e_2\}$ is contained in a label set, $\mathcal{L}$ say, in $\mathcal{F}_S$,

(ii) $\{a\}$, $\{b\}$, and $\{c\}$ are label sets in $\mathcal{F}_S$, and e_1 and e_2 are in separate label sets in $\mathcal{F}_S$,

(iii) $\{a, b\}$ and $\{c\}$ are label sets in $\mathcal{F}_S$, and e_1 and e_2 are in separate label sets in $\mathcal{F}_S$ and, relative to (e_1, a, b, c, e_2), if e_1 or e_2 is internal in $\mathcal{T}$, then $\{e_1\}$ or $\{e_2\}$ is a label set in $\mathcal{F}_S$, respectively, or

(iv) $\{a\}$ and $\{b,c\}$ are label sets in $\mathcal{F}_S$, and e_1 and e_2 are in separate label sets in $\mathcal{F}_S$ and, relative to (e_1,a,b,c,e_2), if e_1 or e_2 is internal in $\mathcal{T}'$, then $\{e_1\}$ or $\{e_2\}$ is a label set in $\mathcal{F}_S$, respectively.

Let $\mathcal{F}_T$ be the forest obtained from $\mathcal{F}_S$ by replacing e_1 and e_2 with a_1 and a_n, respectively, if a_1 or a_n is external in either $\mathcal{T}$ or $\mathcal{T}'$ and, then, depending on which of (i) to (iv) holds, make one of the following replacements for a, b, and c:

(i) $\mathcal{L}$ with $(\mathcal{L}-\{a,b,c\})\cup A$,

(ii) $\{a\}$, $\{b\}$, and $\{c\}$ with the sets $\{a_i\}$, where a_i is an internal element in both $\mathcal{T}$ and $\mathcal{T}'$,

(iii) $\{a,b\}$ and $\{c\}$ with the parts of the parent partition of $(a_1,a_2,\ldots,a_n)$ induced by $\mathcal{T}$ whose corresponding parents are internal in $\mathcal{T}$, and deleting $\{a_1\}$ or $\{a_n\}$ if e_1 or e_2 is internal in $\mathcal{S}$, or

(iv) $\{a\}$ and $\{b,c\}$ with the parts of the parent partition of $(a_1,a_2,\ldots,a_n)$ induced by $\mathcal{T}'$ whose corresponding parents are internal in $\mathcal{T}'$, and deleting $\{a_1\}$ or $\{a_n\}$ if e_1 or e_2 is internal in $\mathcal{S}'$.

A routine check shows that, as $\mathcal{F}_S$ is a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$, the collection $\mathcal{F}_T$ of sets is a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. In (ii), the contribution of the singletons $\{a\}$, $\{b\}$, and $\{c\}$ to $w(\mathcal{F}_S)$ is the same as the contribution of the sets $\{a_i\}$ to $w(\mathcal{F}_T)$, where a_i is an internal element of both $\mathcal{T}$ and $\mathcal{T}'$. Furthermore, in (iii) and analogously in (iv), the contribution of $\{a,b\}$ and $\{c\}$, and $\{e_1\}$ and $\{e_2\}$ if e_1 or e_2, respectively, are internal in $\mathcal{S}$ to $w(\mathcal{F}_S)$ is equal to the contribution of the label sets in $\mathcal{F}_T$ which exclusively contain internal elements of A in $\mathcal{T}$ to $w(\mathcal{F}_T)$. Thus $w(\mathcal{F}_T)=w(\mathcal{F}_S)$, and so $f(\mathcal{T},\mathcal{T}')\le f(\mathcal{S},\mathcal{S}')$. Hence, $f(\mathcal{T},\mathcal{T}')=f(\mathcal{S},\mathcal{S}')$, completing the proof of the proposition. □

Lemma 5.18. *Let $\mathcal{T}$ and $\mathcal{T}'$ be a pair of weighted rooted phylogenetic X-trees, and let $(a_1,a_2,\ldots,a_n)$ be a maximal chain of $\mathcal{T}$ and $\mathcal{T}'$ that does not cross P. Then, by a sequence of long- and short-chain reductions applied to this chain, the length of the resulting chain is at most* 17.

Proof. Suppose first that there is an element of the chain that is internal in both $\mathcal{T}$ and $\mathcal{T}'$. With $i\le j$, choose a_i and a_j as follows:

(a) If a_1 is internal in both $\mathcal{T}$ and $\mathcal{T}'$, choose a_i to be a_1. If a_1 is external in both $\mathcal{T}$ and $\mathcal{T}'$, but a_2 is internal in both $\mathcal{T}$ and $\mathcal{T}'$, choose a_i to be a_2. Otherwise, for some $\mathcal{R} \in \{\mathcal{T}, \mathcal{T}'\}$, a_1 and a_2 are external in $\mathcal{R}$. In this case, choose a_i to be the element of the chain that is external in $\mathcal{R}$ and has maximum index with $a_1, a_2, \ldots, a_i$ all external in $\mathcal{R}$.

(b) If a_n is internal in both $\mathcal{T}$ and $\mathcal{T}'$, choose a_j to be a_n. If a_n is external in both $\mathcal{T}$ and $\mathcal{T}'$, but a_{n-1} is internal in both $\mathcal{T}$ and $\mathcal{T}'$, choose a_j to be a_{n-1}. Otherwise, for some $\mathcal{S} \in \{\mathcal{T}, \mathcal{T}'\}$, a_n and a_{n-1} are external in $\mathcal{S}$. In this case, choose a_j to be the element of the chain that is external in $\mathcal{S}$ and has minimum index with $a_j, a_{j+1}, \ldots, a_n$ all external in $\mathcal{S}$.

Having picked a_i and a_j, consider the chain $(a_i, a_{i+1}, \ldots, a_j)$. If this chain satisfies (i) and the condition on internal parents at the end of (iii) in the description of the long-chain reduction, then we can apply this reduction to get a chain with at most 5 elements. Furthermore, if $(a_1, a_2, \ldots, a_{i-1})$ is a chain with at least three internal elements in the tree in $\{\mathcal{T}, \mathcal{T}'\}$ that is not $\mathcal{R}$, then we can apply the short-chain reduction to get a chain with at most 3 elements. Lastly, if $(a_{j+1}, a_{j+2}, \ldots, a_n)$ is a chain with at least three internal elements in the tree in $\{\mathcal{T}, \mathcal{T}'\}$ that is not $\mathcal{S}$, then we can again apply the short-chain reduction to get a chain with at most 3 elements. Note that if we cannot apply the first or the second of these short-chain reductions, then $i - 1 \leq 3$ and $n - j \leq 3$, respectively. It now follows that after these three reductions, the resulting chain has length at most 11.

Now assume that $(a_i, a_{i+1}, \ldots, a_j)$ does not satisfy (i) or the condition on internal parents at the end of (iii) in the description of the long-chain reduction. Then, up to the possibility of an additional internal parent which only has a_j as its only child in $\{a_i, a_{i+1}, \ldots, a_j\}$, this chain has at most two internal parents in either $\mathcal{T}$ or $\mathcal{T}'$. Except for the children of these two parents, all of the remaining elements of $\{a_1, \ldots, a_n\}$ are external in either $\mathcal{T}$ or $\mathcal{T}'$. In particular, $a_1, \ldots, a_{i-1}$ share the same parent in $\mathcal{R}$, and $a_{j+1}, \ldots, a_n$ share the same parent in $\mathcal{S}$. As $(a_1, a_2, \ldots, a_n)$ has an internal element in both $\mathcal{T}$ and $\mathcal{T}'$, these two shared parents are distinct. Applying at most four short-chain reductions, it is easily checked that the resulting chain has length at most 17.

Now suppose that no element of the chain is internal in both $\mathcal{T}$ and $\mathcal{T}'$, then each element of the chain is external in either $\mathcal{T}$ or $\mathcal{T}'$. In this case, either we apply a single application of the short-chain reduction to get a chain of length at most 4 or we apply two applications of the short-chain reduction to get a chain of length at most 8. This completes the proof of the lemma. □

Proposition 5.17 showed that the weight function is preserved under the subtree, long-chain, and short-chain reductions. Part (iii) of the next lemma shows that these reductions can be applied so that the size of the label set of the resulting rooted phylogenetic trees is bounded by a linear function in the minimum hybridization number.

Lemma 5.19. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let P initially be an empty collection of subsets of X. Let $\mathcal{S}$ and $\mathcal{S}'$ be two weighted rooted phylogenetic X'-trees obtained from $\mathcal{T}$ and $\mathcal{T}'$, respectively, by repeatedly applying the subtree reduction until no further reduction is possible and, then, for each maximal chain common to both resulting trees, repeatedly applying the long-chain and short-chain reductions. Then*

(i) *$\mathcal{S}$ and $\mathcal{S}'$ have no pendant subtrees with common label set A such that $\mathcal{S}|A$ and $\mathcal{S}'|A$ have a common binary refinement and $|A| \geq 2$,*

(ii) *the length of any chain common to both $\mathcal{S}$ and $\mathcal{S}'$ is at most 17, and*

(iii) *$|X'| < 89h(\mathcal{T}, \mathcal{T}')$.*

Proof. For the proof of (i) and (ii), let $\mathcal{T}_1$ and $\mathcal{T}_1'$ be the rooted phylogenetic trees obtained from $\mathcal{T}$ and $\mathcal{T}'$ after repeatedly applying the subtree reduction until no further reduction is possible. Furthermore, observe that if $P_1, P_2 \in P$, then $\mathcal{S}(P_1)$ and $\mathcal{S}(P_2)$ are edge-disjoint, and $\mathcal{S}'(P_1)$ and $\mathcal{S}'(P_2)$ are edge-disjoint. Suppose that (i) does not hold, and let A be such a label set. Without loss of generality, we may assume that A is maximal. Then, because of maximality, if A intersects a set in P, then that set is a subset of A. Now let A' be the set obtained from A by replacing the elements belonging to a set in P with their original counterparts. Using the above observation, it is easily seen that A' is a pendant subtree of $\mathcal{T}_1$ and $\mathcal{T}_1'$. But, as $\mathcal{S}|A$ and $\mathcal{S}'|A$ have a common binary refinement, $\mathcal{T}_1|A'$ and $\mathcal{T}_1'|A'$ have a common binary refinement; a contradiction. Thus (i) holds.

For (ii), suppose that there exists a chain common to both $\mathcal{S}$ and $\mathcal{S}'$ that has at least 18 elements. Without loss of generality, we may assume that this chain is maximal. Let A denote the label set of this common chain. Analogous to (i), because of maximality, if A intersects a set in P, then that set is a subset of A. Moreover, if this intersection involves a set that was part of a sequence of reductions to reduce a common chain in $\mathcal{T}_1$ and $\mathcal{T}_1'$, then all of the associated sets in P are subsets of A. Using Lemma 5.18 to get a contradiction, a similar argument used to establish (i) can now be used to establish (ii).

Now consider (iii). Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$ of minimum weight. Let $\mathcal{B}$ and $\mathcal{B}'$ be two binary refinements of $\mathcal{S}$ and $\mathcal{S}'$, respectively, so that $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{B}$ and $\mathcal{B}'$. By Lemma 5.6, such binary refinements exist. If $\mathcal{B}$ and $\mathcal{B}'$ have a common pendant subtree with label set A and $|A| \geq 2$, then this subtree is a common binary refinement of $\mathcal{S}|A$ and $\mathcal{S}'|A$, contradicting (i). Thus $\mathcal{B}$ and $\mathcal{B}'$ have no such pendant subtree. Furthermore, if $\mathcal{B}$ and $\mathcal{B}'$ have a common chain with label set A and $|A| \geq 18$, then this implies that $\mathcal{S}$ and $\mathcal{S}'$ have such a chain, contradicting (ii). Hence, any chain common to both $\mathcal{B}$ and $\mathcal{B}'$ has at most 17 elements. With these restrictions on $\mathcal{B}$ and $\mathcal{B}'$, we can now use the argument for the analogous result for binary trees in Bordewich and Semple (2007b) to complete the proof of (iii). The only modification necessary is to replace chains of size 2 with chains of size at most 17. Making this change and working through the straightforward algebra gives $\sum_i |\mathcal{L}_i| \leq 89k - 51$. By definition of f and Proposition 5.17, $k \leq f(\mathcal{S}, \mathcal{S}') = f(\mathcal{T}, \mathcal{T}')$. Since P is initially empty, $f(\mathcal{T}, \mathcal{T}') = h(\mathcal{T}, \mathcal{T}')$ and the result follows. □

Before proving Theorem 5.2, we need one further lemma which points out an efficient way for checking whether two rooted phylogenetic trees have a common refinement. A collection J of non-empty subsets of a finite set X is a *hierarchy* if, for all $A, B \in J$, the set $A \cap B \in \{\emptyset, A, B\}$. It is well-known that the set of (edge) clusters of a rooted phylogenetic tree is a hierarchy and, moreover, if J is a hierarchy, then there is a rooted phylogenetic tree whose set of non-trivial (edge) clusters is J (Semple and Steel, 2003).

Lemma 5.20. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees. Then $\mathcal{T}$ and $\mathcal{T}'$ have a common refinement if and only if $\mathcal{C}(\mathcal{T}) \cup \mathcal{C}(\mathcal{T}')$ is a hierarchy.*

Proof. Let $\mathcal{S}$ be a common refinement of $\mathcal{T}$ and $\mathcal{T}'$ or, in other words, let $\mathcal{S}$ be a rooted phylogenetic X-tree such that $\mathcal{C}(\mathcal{T}) \subseteq \mathcal{C}(\mathcal{S})$ and $\mathcal{C}(\mathcal{T}') \subseteq \mathcal{C}(\mathcal{S})$. This implies that $\mathcal{C}(\mathcal{T}) \cup \mathcal{C}(\mathcal{T}') \subseteq \mathcal{C}(\mathcal{S})$. Since $\mathcal{C}(\mathcal{S})$ is a hierarchy on X, the same holds for $\mathcal{C}(\mathcal{T}) \cup \mathcal{C}(\mathcal{T}')$. This gives the first direction of the lemma. For the converse, let $\mathcal{C}(\mathcal{T}) \cup \mathcal{C}(\mathcal{T}')$ be a hierarchy on X. Hence, there exists a rooted phylogenetic X-tree $\mathcal{T}''$ such that $\mathcal{C}(\mathcal{T}'') = \mathcal{C}(\mathcal{T}) \cup \mathcal{C}(\mathcal{T}')$. Now it is easily checked that each rooted phylogenetic X-tree $\mathcal{S}$ with $\mathcal{C}(\mathcal{T}'') \subseteq \mathcal{C}(\mathcal{S})$ is a common refinement of $\mathcal{T}$ and $\mathcal{T}'$. □

Proof of Theorem 5.2. Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted phylogenetic X-trees, and let P be an initially empty collection of subsets of X. Let k be an integer. Let $\mathcal{S}$ and $\mathcal{S}'$ be the weighted rooted phylogenetic X'-trees obtained from $\mathcal{T}$ and $\mathcal{T}'$ by repeatedly applying the subtree reduction until no further reduction is possible and, then, for each maximal chain common to both resulting trees, repeatedly applying the long-chain and short-chain reductions. As P is initially empty, $h(\mathcal{T}, \mathcal{T}') = f(\mathcal{T}, \mathcal{T}')$ and so, by Proposition 5.17,

$$h(\mathcal{T}, \mathcal{T}') = f(\mathcal{T}, \mathcal{T}') = f(\mathcal{S}, \mathcal{S}').$$

It is clear that $\mathcal{S}$ and $\mathcal{S}'$ can be found in time polynomial in $|X|$, say $p(|X|)$. By Lemma 5.19 (iii), $|X'| \leq 89h(\mathcal{T}, \mathcal{T}')$ and so, if $|X'| > 89k$, we declare that $h(\mathcal{T}, \mathcal{T}') > k$.

Now suppose that $|X'| \leq 89k$. The time taken to check whether a partition of $X' \cup \{\rho\}$ is a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$ takes time polynomial in k. By Lemma 5.20, note that for deciding if two rooted phylogenetic trees $\mathcal{T}_1$ and $\mathcal{T}_1'$ have a common binary refinement, one simply needs to check whether or not $\mathcal{C}(\mathcal{T}_1) \cup \mathcal{C}(\mathcal{T}_1')$ is a hierarchy. Furthermore, as $|X'| \leq 89k$, the number of forests with at most $k+1$ parts is bounded by a computable function in k, say $f(k)$. If one of these forests is a legitimate-agreement forest for $\mathcal{S}$ and $\mathcal{S}'$ with weight at most k, then we declare $h(\mathcal{T}, \mathcal{T}') \leq k$; otherwise, we declare $h(\mathcal{T}, \mathcal{T}') > k$. Hence, we can answer the Minimum Hybridization decision problem for $\mathcal{T}$ and $\mathcal{T}'$ in time $O(f(k) + p(|X|))$. Thus Minimum Hybridization is fixed-parameter tractable. □

Concluding remarks.

1. While one could explicitly give a function in k that bounds the number of partitions to consider in the proof of Theorem 5.2, it is unlikely to be the best theoretically and we expect in practice much better methods.

2. In this chapter, we reduced a chain using two types of chain reductions. However, we believe that it is possible to do this with a single type of chain reduction. The drawback of such a reduction is that the number of possibilities for a legitimate-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ increases. Since the goal of this chapter is to show that Minimum Hybridization is fixed-parameter tractable, we decided to use the two types of reductions, thereby reducing the complexity and lengths of the proofs.

3. In collaboration with Joshua Collins (University of Canterbury), the implementation of an algorithm for finding the minimum number of hybridization events for two (arbitrary) rooted phylogenetic trees that considers the results of this chapter is subject of ongoing research. We expect to implement this algorithm in a similar way than HybridNumber (see Appendix A.1), but also to include some crucial differences. For example, the order of the reductions which is used in HybridNumber needs to be modified to guarantee that no common short- or long-chain arises once the namesake reductions have been applied. This is of importance since, otherwise, due to consecutive leaves that have previously been involved in a short- or long-chain reduction, the weighting scheme gets more complex. We also intend to use rekernalization, which is often termed *interleaving*, for the exhaustive search part of the algorithm. This technique repeatedly applies the reduction rules as part of the exhaustive search. Detailed information about interleaving are given by Niedermeier and Rossmanith (2000).

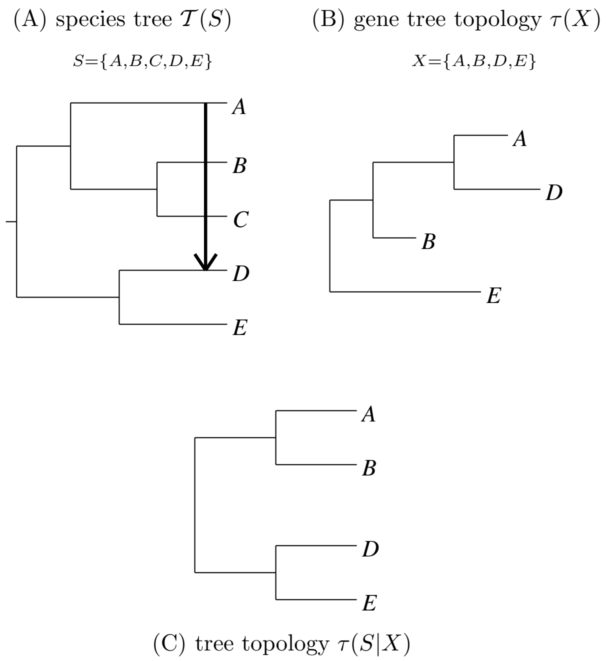
6 A Likelihood Framework to Measure Horizontal Gene Transfer

In this chapter, we focus on the process of horizontal gene transfer (HGT). Like hybridization, HGT can cause gene tree incongruence and, currently, a lot of effort is put into new developments to analyze the extent to which HGT has influenced the evolution of a set of present-day species. Here, we present a likelihood-based approach to estimate a rate of HGT in a simplified setting. To this end, we assume that the number of HGT events within a given time interval follows a Poisson process. To obtain estimates for the rate of HGT, distributions of tree topologies for different numbers of HGT events on a clocklike species tree are simulated. Using these simulated distributions, we estimate an HGT rate for a collection of gene trees representing a set of taxa. As an illustrative example, we apply this approach to the "Clusters of Orthologous Groups of Proteins (COGs)" (Tatusov *et al.*, 2001). Additionally, inaccuracies due to gene tree reconstruction methods are analyzed.

6.1 Introduction

It is well known that gene trees reconstructed for different genetic loci for the same set of taxa do not necessarily have the same branching pattern. In fact, these may be different from each other and different from the species tree (Pamilo and Nei, 1988). Such discrepancies are not always due to problems in the tree reconstruction method, but rather due to biological processes like hybridization, gene duplication and deletion, or HGT (Syvanen, 1994). Here, we will focus on the latter of these processes.

The effect of one HGT event is depicted in Figure 6.1A where a species tree of the five taxa A, B, C, D, and E is shown. This tree indicates a close relation between A and the most recent common ancestor of B and C. In many cases, the species tree also explains the phylogeny of single genes, but sometimes a gene has a different evolutionary history than the species tree (Pamilo and Nei, 1988). For such a gene the gene tree is displayed in Figure 6.1B. One possible explanation for this kind of difference is HGT. In the course of such a process, a piece of DNA (e.g. a gene) is transferred from one organism to another which is not its offspring and often the new genetic material is stably incorporated into the acceptor genome afterwards. In the depicted case, the arrow shows a gene transfer from species A (donor) to species D (acceptor). Consequently, the gene tree for this gene shows a close relationship between A and D.



(A) species tree $\mathcal{T}(S)$

(B) gene tree topology $\tau(X)$

(C) tree topology $\tau(S|X)$

Figure 6.1: Comparison of a species tree (A) and a corresponding gene tree (B) after a single HGT event. The arrow indicates a gene transfer from species A (donor) to species D (acceptor). To check whether the gene tree is a subtree of the species tree, we compute the tree topology $\tau(S|X)$ derived from the species tree (C).

Several approaches have been published which discover single HGT events (Lerat *et al.*, 2003, 2005), whereas another kind of approach estimates the amount of genes that have been acquired through HGT for a given genome. The latter type of analysis is reviewed in Ochman *et al.* (2000) for 19 completely sequenced genomes. For these species, the amount of adopted genes varies between virtually none in organisms with a small genome size, for example *Rickettsia prowazekii*, *Borrelia burgdorferi*, and *Mycoplasma genitalium*, to almost 17 % in *Synechocystis* PCC6803. Another way of detecting horizontally transferred genes uses bacterial genome sequences to examine the nucleotide composition (GC content) and usage of different codons (Lawrence and Ochman, 1997).

In contrast to these approaches, we estimated an overall rate of HGT for a given set of species based on simulating a likelihood curve for the reconstructed species tree. We constructed a clocklike species tree reflecting the actual evolutionary history of the analyzed organisms (Pamilo and Nei, 1988) and simulated different numbers of HGT events that are implemented as sequences of rooted subtree prune and regraft operations on the species tree (see Section 1.4.4). Simulations with different numbers of HGT events lead to a distribution of tree topologies which are comparable with the gene tree distribution to estimate an HGT rate supported by a likelihood framework.

As the number of tree topologies increases exponentially with the number of species, the probability to get a specific topology is very low. In trying to overcome this problem, we worked with quartet subtrees instead of the complete gene tree topologies (see Section 6.2).

6.2 Materials and Methods

6.2.1 Notation

In addition to the usual terminology of phylogenetic trees that is given in Section 1.4, we next introduce some further notion that is needed exclusively for this chapter. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of taxa, and let X be a subset of S. To describe all binary tree topologies with label set X, we use $\mathcal{P}(X)$. Furthermore, let $\tau(X)$ denote an element of $\mathcal{P}(X)$. A *species tree* with taxa set S, denoted $\mathcal{T}(S)$, is an element of $\mathcal{P}(S)$. For a species tree $\mathcal{T}(S)$ with edge set E, let $l : E \rightarrow \mathbb{R}^{\geq 0}$ be a length function on the elements of E such that each element $e \in E$ is assigned a non-negative value, the branch length. The *tree length* $L(\mathcal{T}(S))$ is the sum over all branch lengths of $\mathcal{T}(S)$. We assume a *species tree* $\mathcal{T}(S)$ to be binary, rooted, leaf-labeled, and clocklike (i.e. each leaf has the same distance to the root) and interpret the distance between any vertex and the root as time which has passed since the first speciation event at the root. Moreover, a *gene tree* is a tree topology of a leaf-labeled tree which evolves within a species tree and contains at most all taxa of S. The *restriction of S to X*, denoted by $\tau(S|X)$, is the minimal unrooted tree topology obtained from $\mathcal{T}(S)$ by connecting all leaves labeled with taxa of X and suppressing all vertices of degree two (Figure 6.1C). Since we obtain an unrooted tree $\tau(S|X)$, note that this definition differs from the definition of a restriction which is given in Section 1.4 and used throughout the rest of this thesis.

6.2.2 Modeling Horizontal Gene Transfer

To model the process of HGT, some pivotal assumptions are needed:

(1) A species tree $\mathcal{T}(S)$ is given.
(2) Differences between a gene tree and $\mathcal{T}(S)$ are only caused by HGT events.
(3) The HGT rate λ is homogeneous per gene and unit time.
(4) Genes are transferred independently.
(5) One copy of the transferred gene still remains in the donor genome.

(6) The transferred gene replaces any existing orthologous counterpart in the acceptor genome.

As mentioned above, the effect of an HGT event can result in a branching pattern of a gene tree which differs from a given species tree (Figure 6.1). From a computational point of view, we model each HGT event as a rooted subtree prune and regraft operation (see Section 1.4.4). As we assume a homogeneous HGT rate λ, the transfer events are uniformly distributed along all branches of the tree. Loosely speaking, for each HGT event, we randomly choose a starting point in the clocklike species tree, determine the corresponding time in this tree, search for all branches that coexist at that point of time, and randomly select one as acceptor branch. To prevent gene transfer from species to their own ancestors, note that single transfer events between species are only possible if they coexist in time. This biologically well-motivated restriction is not considered in some current research on HGT models. Furthermore, it is easily seen that not every HGT event changes the branching pattern of the species tree, e.g. if the process takes place between lineages that share the parent vertex.

For a given species tree with total length $L(\mathcal{T}(S))$ and a fixed λ, the tree topology $\tau(S)$ occurs with a certain probability $P(\tau(S) \mid \mathcal{T}(S), \lambda, L(\mathcal{T}(S)))$ or, in short, $P(\tau(S) \mid \lambda)$, since λ is the parameter of interest. As stated in the introduction, the number of HGT events is Poisson distributed with parameter $\Lambda = \lambda \cdot L(\mathcal{T}(S))$ for a fixed species tree. Thus the probability for $\tau(S)$ given λ is

$$P(\tau(S)|\lambda) = \sum_{h=0}^{\infty} \left(\frac{e^{-\Lambda} \cdot \Lambda^h}{h!} \cdot P(\tau(S) \mid \text{HGT} = h) \right). \tag{6.1}$$

The Poisson distribution describes the probability that h HGT events happen on the species tree $\mathcal{T}(S)$ with $L(\mathcal{T}(S))$ and λ, whereas the second factor is the probability to observe $\tau(S)$ as tree topology after h HGT events. While the Poisson distribution is easy to compute, the probability distribution of the gene trees for a fixed number of HGT events is hard to calculate, except for trivial cases like $h \in \{0, 1\}$.

For a fixed arbitrary subset $X \subseteq S$, we can compute the probability for a subtree $\tau(X)$ as follows

$$P(\tau(X)|\lambda) = \sum_{\tau(S) \in \mathcal{P}(S)} \left(\delta_{(\tau(X), \tau(S|X))} \cdot P(\tau(S)|\lambda) \right). \tag{6.2}$$

The Kronecker delta $\delta_{(\tau(X),\tau(S|X))}$ is one if the topology of the induced subtree $\tau(S|X)$ with respect to $X \subseteq S$ is identical to $\tau(X)$ and, otherwise, it is zero.

Equations 6.1 and 6.2 allow for an estimation of λ in a likelihood framework. Therefore, we assume that λ acts on each gene independently. If m gene trees $\tau_1(S), \ldots, \tau_m(S)$ are given, the likelihood of λ is

$$\text{lik}(\lambda|\tau_1(S), \ldots, \tau_m(S)) = \prod_{i=1}^{m} P(\tau_i(S)|\lambda). \tag{6.3}$$

We maximize Equation 6.3 with respect to λ and interpret the result as the most likely transfer rate.

This approach turns out to be computationally infeasible since a reliable estimation of $P(\tau(S)|\lambda)$ is only possible for a small number of taxa. Hence, we resort to an approximation of the likelihood. We consider a collection $\{X_1, \ldots, X_m\}$ of subsets of S together with the probability distribution induced by Equation 6.2 and assume that the occurrences of the gene trees $\tau(X_1), \ldots, \tau(X_m)$ are mutually independent for different randomly chosen subsets $X_1, \ldots, X_m$. In this case, the joint probability of $\tau(X_1), \ldots, \tau(X_m)$ is

$$P(\tau(X_1), \ldots, \tau(X_m)) \approx \prod_{i=1}^{m} P(\tau(X_i)|\lambda). \tag{6.4}$$

Although Equation 6.4 is an approximation to Equation 6.3, the simulations show that it is good enough for the practical application and that we can also apply the described equations to estimate $\hat{\lambda}$ and $\hat{\Lambda}$, respectively.

6.2.3 Estimating the Probability Distribution of Gene Trees

From the previous paragraph, it is obvious that it is difficult to find an analytical expression for any of the equations. However, Equation 6.1 suggests an efficient simulation. For a fixed number h of HGT events, we can approximate the distribution $P(\tau(S)|\text{ HGT} = h)$ reasonably well. Therefore, we simulate $N = 100,000$ times h HGT events on a species tree with $0 \leq h \leq 60$ and calculate how often each gene tree occurs in the simulated trees. The resulting probability distribution, denoted by $P^*(\cdot)$, represents the results for the i^{th} gene tree in the i^{th} column and the results for a fixed number h of HGT events in the h^{th} row. The final likelihood estimation is then based on $P^*(\cdot)$.

While $P(\tau(S)|\lambda)$ can be estimated for small taxa sets, it gets intractable for biologically interesting numbers because too many tree topologies exist and, hence, it is impossible to simulate enough trees for a reliable estimation within a reasonable time span. In such situations, the probability for different subsets of S proves more successful. Thus we reduce the calculated probability distribution $P^*(\cdot)$ to a subset of randomly chosen quartet topologies of the given set of gene trees.

6.2.4 The COG Data Set

The whole COG data set, which is available via the NCBI website[4], comprises 3,167 protein families of 44 species (2 eukaryotes, 9 archaea, and 33 bacteria) to which we will refer as COG taxa set S_{COG} in the following. As the analysis currently only considers single-copy genes, we extracted those families. To obtain enough phylogenetic information to reconstruct the gene trees, we only used COG families with a minimum alignment length of 100 amino acids (Nei, 1996) and required at least four species per COG family. After applying those three criteria, 780 protein families remained. For each of these families, a gene tree was reconstructed by using TREE-PUZZLE (Schmidt *et al.*, 2002) and the Dayhoff substitution model (Dayhoff *et al.*, 1978).

To construct a species tree that considers the information of all 780 protein families, we built the three unrooted binary trees for all possible quartets (A, B, C, D) and computed the corresponding log-likelihood values ℓ as sum of the log-likelihoods over all COG families (g_i).

$$\begin{aligned} \ell(AB|CD) &= \sum_{i=1}^{780} \ell_{g_i}(AB|CD) \\ \ell(AC|BD) &= \sum_{i=1}^{780} \ell_{g_i}(AC|BD) \\ \ell(AD|BC) &= \sum_{i=1}^{780} \ell_{g_i}(AD|BC). \end{aligned} \tag{6.5}$$

All three log-likelihood values ℓ_{g_i} are set to be 0 if at least one of the species A, B, C, or D has not been sequenced for the corresponding COG family g_i. Afterwards, TREE-PUZZLE was used to construct a species tree $\mathcal{T}(S_{\text{COG}})$ of the so-obtained log-likelihood values.

To assign branch lengths to $\mathcal{T}(S_{\text{COG}})$, we performed a clock test (see Felsenstein, 1988) for all 780 protein families. The results contained 443 clocklike and 337 non-clocklike COG

[4] http://www.ncbi.nlm.nih.gov/

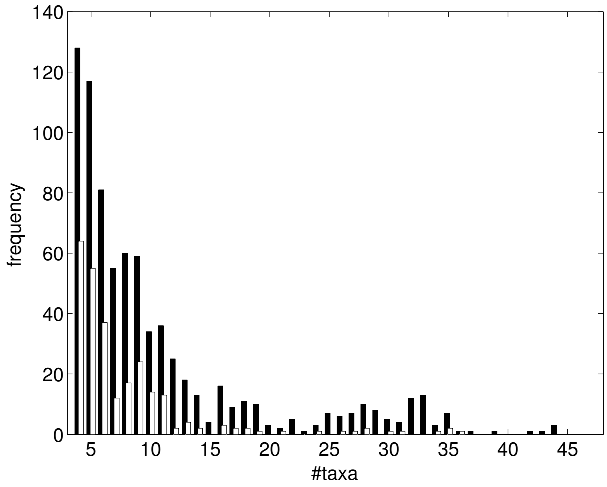


Figure 6.2: Distribution of the COG data set according to the number of sequences in each protein family (black bars: all 780 protein families are considered; white bars: only those families are considered whose corresponding gene tree is a subtree of $\mathcal{T}(S_{\text{COG}})$).

families. Only three of all 780 families consist of 44 sequences (COG0013: Alanyl-tRNA synthetase, COG0092: Ribosomal protein S3, COG0541: Signal recognition particle GTPase Ffh), but none of them evolved clocklike. Therefore, we used an appropriate set of gene trees which covers all 44 species. For each clocklike evolving COG family with taxa set X, we reconstructed the corresponding subtree $\tau(S_{\text{COG}}|X)$ with a total branch length measured in numbers of substitutions per site. Furthermore, we identified a set G of subtrees fulfilling the following conditions: (a) the union of label sets of trees in G contains all taxa of $\mathcal{T}(S_{\text{COG}})$ and (b) each branching point of $\mathcal{T}(S_{\text{COG}})$ is determined by at least one subtree of G. Such a coverage was found for the three clocklike evolving families: COG0419 (ATPase involved in DNA repair), COG0173 (Aspartyl-tRNA synthetase) and COG1242 (uncharacterized FeS oxidoreductase). As some of the splitting times are given by two or three of these families and each of them evolved with a different rate, we computed the ratio of these rates to estimate the splitting times relative to one protein family, in this case COG0419. The obtained species tree with $L(\mathcal{T}(S_{\text{COG}})) = 29.9$ is shown in Appendix A.2 (Figure A.2.1). Finally, the reconstructed species tree $\mathcal{T}(S_{\text{COG}})$ was used to simulate distributions of tree topologies for different numbers of HGT events.

An overview of the COG data set is summarized in Figure 6.2, where the black bars represent the distribution of protein families according to the number of sequences and, hence, to the number of taxa represented by the corresponding gene tree. It is easily seen that the majority of families contains less than 10 sequences. Additionally, the white bars only consider those protein families, whose associated gene trees are subtrees of $\mathcal{T}(S_{\text{COG}})$.

Since the reconstruction of $\mathcal{T}(S_{\text{COG}})$ was quite complex and since it remains unclear if this tree represents the evolutionary history of the COG taxa set correctly, a second clocklike species tree $\mathcal{T}(S_{\text{RNA}})$—based on 16/18 S rDNA sequences—was reconstructed. The 16 and 18 S rDNA genes are homologous to each other (Rubtsov *et al.*, 1980) and encode for the RNA of the small ribosomal subunit. For the tree reconstruction, we downloaded the 16 S rDNA sequence for all 33 bacteria and the 18 S rDNA sequence for all 11 eukaryotes and archaea, respectively, from the RefSeq (NCBI Reference Sequence) data base (Pruitt *et al.*, 2005), calculated a multiple sequence alignment with the Mafft program (Katoh *et al.*, 2005), and reconstructed $\mathcal{T}(S_{\text{RNA}})$ by using TreePuzzle. In Appendix A.2 (Figure A.2.2), the resulting tree with $L(\mathcal{T}(S_{\text{RNA}})) = 4.8$ is shown. A comparison of $L(\mathcal{T}(S_{\text{COG}}))$ and $L(\mathcal{T}(S_{\text{RNA}}))$ indicates that the 16/18 S rDNA sequences were highly conserved during evolution. Note that Figure A.2.2 shows a tree that represents three distinct clades of the kingdoms bacteria, eukaryotes, and archaea as suggested by Woese and Fox (1977).

6.2.5 Comparing Trees

Before detailing a further analysis, we recall that two quartet trees $A_1|B_1$ and $A_2|B_2$ are compatible if at least one of the sets $A_1 \cap A_2$, $A_1 \cap B_2$, $B_1 \cap A_2$, and $B_1 \cap B_2$ is the empty set (Semple and Steel, 2003). Note that in the case of quartet trees each of the sets A_1, B_1, A_2, and B_2 contains two taxa. To compare the most frequent gene tree with $\mathcal{T}(S_{\text{COG}})$, we extracted all quartet topologies from all 780 gene trees and summarized the information in a descending sorted list representing each quartet tree by the number of its occurrences. Afterwards, we built a pairwise compatible quartet set that finally consisted of 35.7 % of the initially extracted quartet topologies. Starting with the most frequent quartet topology, we put each successively following quartet tree in the current set if the resulting set of quartet trees was pairwise compatible. Based on the final quartet set, we reconstructed a tree using Tree-Puzzle. To compare the obtained tree topology with the COG species tree, we built a consensus tree using the program Consense of the Phylip package (Felsenstein, 1989).

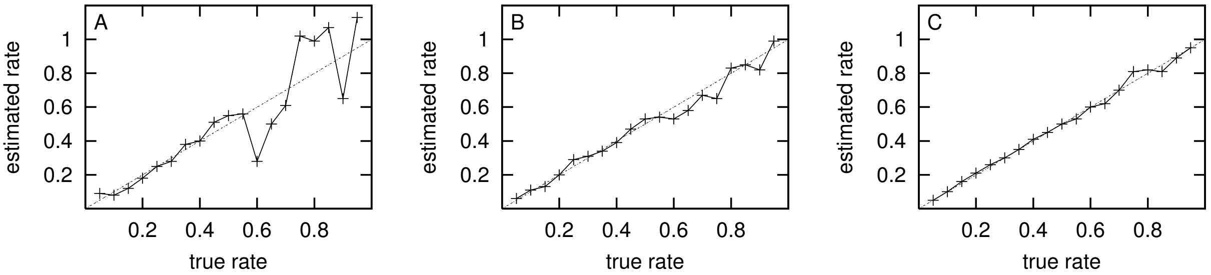


Figure 6.3: Quality of the rate estimation in dependence on the number of quartet topologies: (A) 100, (B) 1,000 and (C) 10,000. $N = 100,000$ and $h_{\max} = 60$ are fixed. Each displayed value is based on one estimation.

6.3 Results

6.3.1 Quality Check

Before estimating an HGT rate for the COG data set, we performed several analyses to check which parameter settings allow reliable results. To this end, we used a program that simulates HGT events with rate λ on $\mathcal{T}(S_{\text{COG}})$. The corresponding number of HGT events was drawn from the Poisson distribution. This kind of simulation generates a new data set which is comparable to the 780 gene trees of the COG data. Since we know the true HGT rate λ, we can check the reliability of the estimation procedure. In the course of such a procedure, we first estimated the probability $P^*(\tau(X) \mid \text{HGT} = h)$ to get the tree topology $\tau(X)$ if exactly h HGT events happened on $\mathcal{T}(S_{\text{COG}})$. Assuming that $P^*(\tau(X) \mid \text{HGT} = h)$ is the relative occurrence of the topology $\tau(X)$, we simulated N times h HGT events on $\mathcal{T}(S_{\text{COG}})$.

To analyze the influence of the size of the quartet set, we generated $1,000$ gene trees for several HGT rates λ, extracted all quartet topologies, and used a randomly chosen subset of these topologies to estimate the HGT rate λ. Repeating this for the quartet set sizes 100, $1,000$, and $10,000$, we got the results visualized in Figure 6.3, where the true HGT rate λ which was used to generate the gene trees and the estimated rate are shown. It turned out that a set of $10,000$ topologies was large enough to get reliable results.

For a second test, we used $10,000$ quartet topologies and varied the value of $h_{\max}$ (the maximal number of simulated transfers on $\mathcal{T}(S_{\text{COG}})$) with $N = 100,000$. Figure 6.4 shows the estimation results based on $h_{\max} \in \{20, 30, 40, 60\}$. For each $h_{\max}$, there exists a maximum rate which can be estimated reliably while rates above get underestimated.

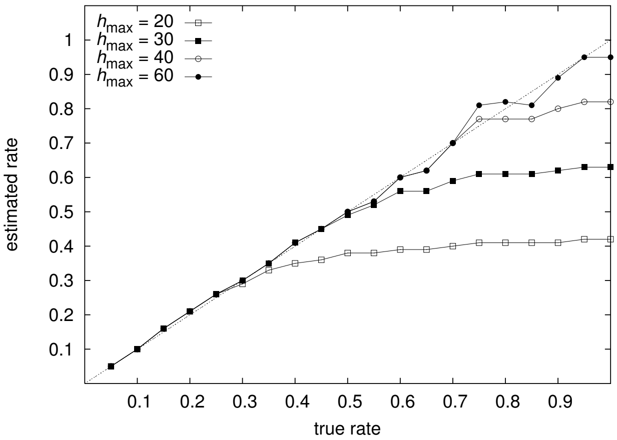


Figure 6.4: Quality of the rate estimation as a function of the maximum number of simulated HGT events with $N = 100,000$ and $10,000$ quartet topologies. Each displayed value is based on one estimation.

6.3.2 The Most Frequent Gene Tree

To determine whether the most frequent gene tree is similar to the reconstructed species tree $\mathcal{T}(S_{\text{COG}})$, we compared both trees. We computed a quartet set of all quartet topologies of the 780 COG gene trees which only consisted of pairwise compatible quartet topologies (see Section 6.2.5). A comparison of the tree reconstructed for this quartet set with the species tree $\mathcal{T}(S_{\text{COG}})$ led to the consensus tree depicted in Figure 6.5. Both trees support all bifurcations except for two vertices indicated by multifurcations in the consensus tree. We can conclude that both trees are almost equal and that the most common gene tree is very similar to $\mathcal{T}(S_{\text{COG}})$.

6.3.3 Estimating the HGT Rate λ for the COG Data Set

The quality tests described in Section 6.3.1 have shown that an HGT rate λ of 0.7 can reliably be estimated if we randomly choose $10,000$ quartet topologies of the 780 COG gene trees and use the parameter setting $N = 100,000$ and $h_{\text{max}} = 60$.

We applied this procedure to the COG data, repeated the estimation for 50 randomly chosen sets of quartet topologies, and obtained results for $\hat{\lambda}$ between 0.43 and 0.48, and for $\hat{\Lambda}$ between 12.86 and 14.35 presented in Figure 6.6A . Since Λ is the parameter of the Poisson distribution which describes the occurrence of HGT events in time, $\hat{\Lambda}$ is the expected value for the number of HGT events that happened on $\mathcal{T}(S_{\text{COG}})$. The estimated HGT rate $\hat{\lambda}$ is relative to the number of substitutions in the protein family COG0419 (ATPase involved in DNA repair) which was used to assign branch lengths to $\mathcal{T}(S_{\text{COG}})$.

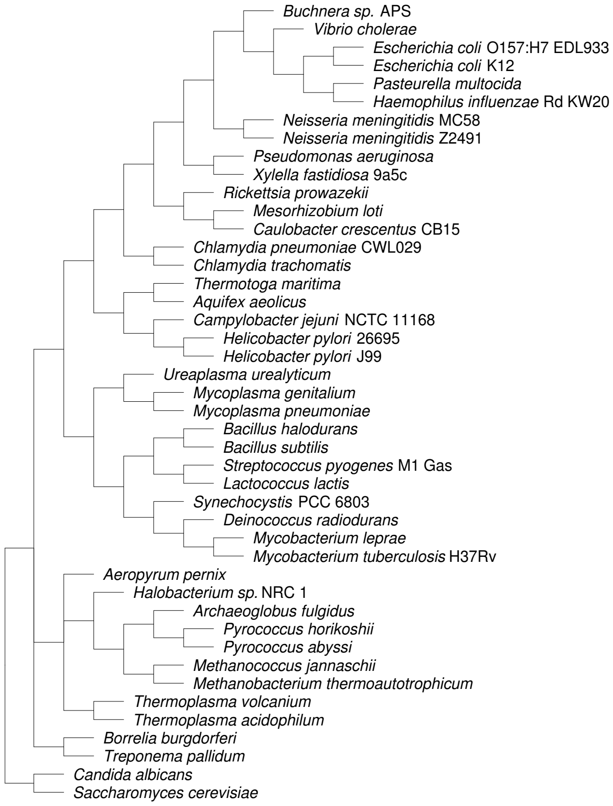


Figure 6.5: Consensus tree of the COG species tree and the tree reconstructed of the quartet set that represents the most frequent quartet subtrees of the 780 gene trees. Only two multifurcations exist which indicate discordance between both trees. This tree was reconstructed with the strict consensus option of the Consense program (Felsenstein, 1989).

To test the reliability of these results, we checked if the estimated HGT rates differ from those estimated for quartet sets randomly chosen from all quartet topologies of the 44 species. Figure 6.6A shows the estimation results of quartet topologies which could be found in the 780 gene trees and Figure 6.6B represents estimations over all $\binom{44}{4} \cdot 3$ quartet topologies. The graph indicates an estimated HGT rate $\hat{\lambda}$, which is about 10 times higher, between 4.66 and 4.7. These rates are higher because the set consists of quartet topologies which are not subtrees of any gene tree, and so more HGT events are necessary to get the distribution.

Furthermore, we simulated HGT events on $\mathcal{T}(S_{\text{RNA}})$ and estimated an HGT rate for 20 quartet sets, where each such set contained $10,000$ randomly chosen quartet topologies of the 780 COG gene trees. The obtained results indicate that $\hat{\Lambda} = 21$ on average.

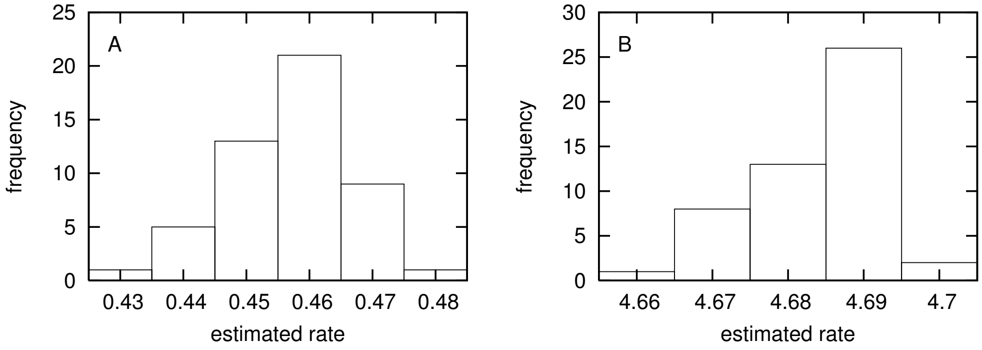


Figure 6.6: Distribution of the estimated HGT rates $\hat{\lambda}$ for the COG data for randomly chosen quartet sets (A) of the 780 gene trees, and (B) over all 44 species tree taxa.

6.3.4 Rate Correction

We performed a further analysis that takes into account the inaccuracies of the gene tree reconstruction method. For each protein family g_i with $i \in \{1, 2, \ldots, 780\}$ representing a taxa set X_{g_i}, we restricted $\mathcal{T}(S_{\text{COG}})$ to X_{g_i}, denoted by $\tau(S_{\text{COG}}|X_{g_i})$, and assigned branch lengths to all of those tree topologies using Tree-Puzzle. Afterwards, we simulated protein sequences of the same size than the corresponding COG sequences along the calculated trees with Seq-Gen (Rambaut and Grassly, 1997) using the Dayhoff substitution model (Dayhoff *et al.*, 1978). We repeated this step five times, calculated gene trees for the simulated sequences, and repeated the estimation procedure. As the newly simulated sequences are based on trees which are subtrees of $\mathcal{T}(S_{\text{COG}})$, we expected to estimate an HGT rate $\hat{\lambda}$ of about zero.

After the estimation of ten randomly chosen quartet sets for each of the five simulated data sets, we got the distribution which is shown in the stacked histogram of Figure 6.7. Each of the five colors represents one data set. The estimation results are nearly constant, at about 0.1 (0.1 ± 0.01). This result could be interpreted as a kind of background noise due to inaccuracies in the applied gene tree reconstruction method (see Section 6.2). Since $\tau(S_{\text{COG}}|X_{g_i})$ is a subtree of $\mathcal{T}(S_{\text{COG}})$, for all $i \in \{1, 2, \ldots, 780\}$, this implies that the estimated average HGT rate $\hat{\lambda}$ of about 0.46 per gene and unit time is about 22 % too high. This would decrease the total amount of HGT events which is necessary to transform $\mathcal{T}(S_{\text{COG}})$ into one gene tree from 14 to 11 events per gene on average.

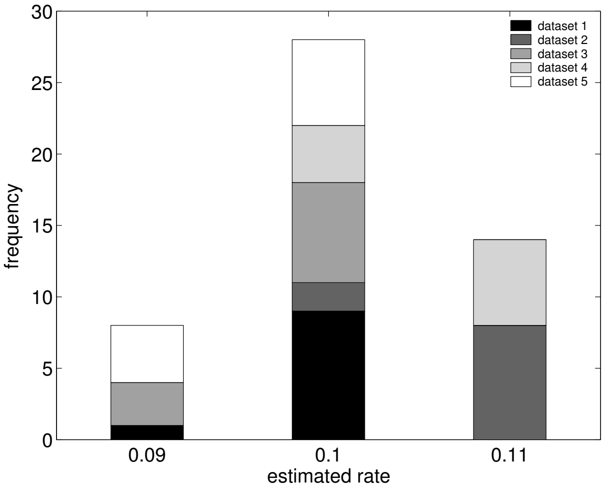


Figure 6.7: Distribution of the estimated HGT rates $\hat{\lambda}$ for five simulated data sets. Each data set is based on the 780 protein families and their corresponding subtrees in $\mathcal{T}(S_{\text{COG}})$. For each data set, ten randomly chosen quartet sets were estimated.

6.4 Discussion

In the previous sections, we have described some results based on a new approach to estimate an overall rate of HGT with the help of a likelihood framework. This procedure allows to estimate a rate of HGT under the assumptions that all differences between a gene tree and an associated species tree have been caused by HGT and that the HGT rate is homogeneous over the whole tree. Note that we did not make any statement about the probability if a gene is transferred at all, but how many events have occurred within one COG family on average. Thus we are assuming that every gene is transferred with the same probability.

A recent publication by Ge *et al.* (2005) also analyzed the COG data set and detected HGT in 33 out of 297 protein families. To do so, they used a novel test statistic based on tree topology comparisons. Unfortunately, they did not say anything about how many HGT events happen in each of those 33 detected COGs, which would be interesting to compare their results with ours.

There are several other approaches trying to estimate an HGT rate. For example, Huelsenbeck *et al.* (2000) developed a Bayesian framework for the analysis of cospeciation which could also be used to estimate rates of genetic transfer. Suchard (2005) published two stochastic models with the same purpose. The first model, developed by Suchard, is based on subtree prune and regraft operations and is applicable if the number of taxa under consideration is small, while the other approach is a random walk over complete graphs and offers a solution for an increasing number of taxa. In both publications, the fact that the corresponding framework can deal with gene and species tree topologies which are not known without error is highlighted. Furthermore, both HGT models require that all gene and species trees are based on the same set of present-day species. In contrast, the new approach, which we have introduced here, can incorporate gene/protein families whose taxa sets are subsets of the species tree taxa set by using quartet subtrees. As an example, the COG data set only comprises three protein families which represent sequences for all 44 taxa (see Figure 6.2). Furthermore, this new framework also takes into account the inaccuracies in the gene tree reconstruction method.

As genomes are not only shaped by HGT, but also by processes like hybridization, gene loss, and duplication (Snel *et al.*, 2002), it becomes clear that the estimated rate of about 11 events per gene and unit time is a kind of upper bound because we assume that all conflicts in the gene tree topologies are caused by HGT. However, it remains as yet unclear, how the rate estimate changes if multi-copy genes were included in the analysis. The high estimates can be explained by the fact that a lot of HGT events will not change the tree topology (events between two vertices that share parents). This is of importance since 71 % of the total branch length of the COG species tree can be involved in HGT events which do not change the branching pattern. As it is most likely that the majority of HGT events in nature takes place between closely related taxa it becomes clear that the number of these events would be underestimated by just counting differences in the branching pattern between two given trees. Moreover, if one gene is transferred back and forth between two lineages, these events will not be detected either. The importance to take unobservable HGT events into account is supported by the fact that the topologies of 264 (34 %) of the 780 COG gene trees are subtrees of $\mathcal{T}(S_{\text{COG}})$ (see white bars in Figure 6.2). This means that the gene tree topology can be explained without any single HGT event. As the number of taxa of those 264 trees differs widely, and even gene trees with up to 36 taxa are equal to the corresponding COG species tree restriction, we can assume that HGT events happened during the evolution of the corresponding genes although we cannot see any of them. This is also supported by the fact that it is still not known if a core of non-transferable genes exists (Nesbø *et al.*, 2001). Summing up, the importance of simulating HGT events on a given species tree, instead of counting visible differences between a species and gene tree, becomes obvious and separates our approach from some previous work on estimating an HGT rate. To get an impression of the probability that an HGT event does not change the tree topology, we counted the simulated trees which are equal to the COG species tree. The result indicates that this probability is 9 % (0.9 %) for the simulated trees after one (two) transfer(s). As our approach includes simulations on a species tree which gave us a distribution of trees after different numbers of HGT events, we automatically include unobservable HGT events and, therefore, the estimated rate is higher than in other approaches. Of course, this high rate also indicates that HGT influences the tree topologies strongly, as described by Doolittle (1999).

Many other approaches (e.g. see Hao and Golding, 2006, Dagan and Martin, 2007) exist which also estimate an HGT rate. All those methods are quite different from one another and it is difficult to compare their results with ours. The two mentioned publications are based on gene present and absent patterns, while the method, which we have introduced here, uses the information of reconstructed gene trees to calculate an HGT rate. Dagan and Martin (2007) have presented a method in which they inferred a conservative lower-bound estimate of about 1.1 HGT events per gene family and gene family lifespan considering the genome size of present day species. As already explained above, the estimates represented here are a kind of upper bound and, therefore, they are higher. Since both methods (Hao and Golding, 2006, Dagan and Martin, 2007) are tested on different data sets, it would be interesting to see how much the results really differ when both are applied to the same data set.

Finally, a comparison of the estimated rates (uncorrected) for the COG species tree with those for the RNA species tree shows that about seven additional HGT events are necessary to explain the incongruence between $\mathcal{T}(S_{\text{RNA}})$ and any of the 780 gene trees than for the same gene tree and $\mathcal{T}(S_{\text{COG}})$. This is in line with our expectations because the species tree $\mathcal{T}(S_{\text{COG}})$ represents the evolutionary history of all 780 analyzed COG families and, hence, it is not guaranteed that this tree depicts the species history of the COG taxa set, whereas $\mathcal{T}(S_{\text{RNA}})$ was obtained from 16/18 S rDNA sequences which are often used to reconstruct universal species trees for a set of given taxa (Woese, 2000, and references therein).

6.5 Outlook

The newly developed likelihood framework to estimate a rate of HGT gives rise to a number of further studies. Since this framework is based on several key assumptions (Section 6.2.2) including some that might be not reasonable from a more biological point of view, it would be interesting to consider more biologically relevant aspects of HGT in the future, e.g., like Suchard (2005), the possibility to include heterogeneous HGT rates in the analysis. Such rates are important to take into account since genes belonging to different functional categories have different transferabilities (Nakamura *et al.*, 2004). Another interesting and important extension for the simulation would be to include uncertainties of the species tree branch lengths. So far, it is assumed that these lengths are exactly known. Currently, the introduced framework exclusively deals with trees that represent a single gene copy per species. Since phylogenies often present several distantly related copies for a given organism, the HGT estimates based on orthologs only could be too low. Hence, another task of future work can be the allowance for multi-copy genes.

7 A New Result for Computing the Rooted Subtree Prune and Regraft Distance

7.1 Introduction

Beside the tree rearrangement operations NNI and TBR, the SPR operation is frequently used in many areas of evolutionary research. Among other applications, (a) the rSPR distance between two rooted binary phylogenetic trees on the same taxa set provides a lower bound on the number of reticulation events (Baroni *et al.*, 2005) and (b) the SPR operation is used to find the best tree in a heuristic search over a tree space (e.g. see Chapter 4 of Felsenstein, 2004).

As shown by Bordewich and Semple (2004), calculating the rSPR distance between two rooted binary phylogenetic trees is an NP-hard problem and exact algorithms which calculate this distance and have a reasonable running time are rare. Hence, one often resorts to approximation algorithms. Two such algorithms that have recently been developed are the 5-approximation algorithm described in Bonet *et al.* (2006) and the 3-approximation algorithm described in Bordewich *et al.* (2008). The latter publication also gives an attractive fixed-parameter tractable algorithm for the problem of computing the rSPR distance between two trees exactly.

We state the optimization problem of computing the rSPR distance between two rooted binary phylogenetic trees as follows:

Minimum rSPR
Instance: Two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$.
Goal: Find a minimum length sequence of single rSPR operations that transform $\mathcal{T}$ into $\mathcal{T}'$.
Measure: The length of this sequence.

Bordewich and Semple (2004) established the following result:

Theorem 7.1. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted binary phylogenetic X-trees. Then*

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') = m(\mathcal{T},\mathcal{T}').$$

Furthermore, they gave a fixed-parameter tractable algorithm to compute $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}')$ by applying slightly modified versions of the subtree and chain reduction to $\mathcal{T}$ and $\mathcal{T}'$.

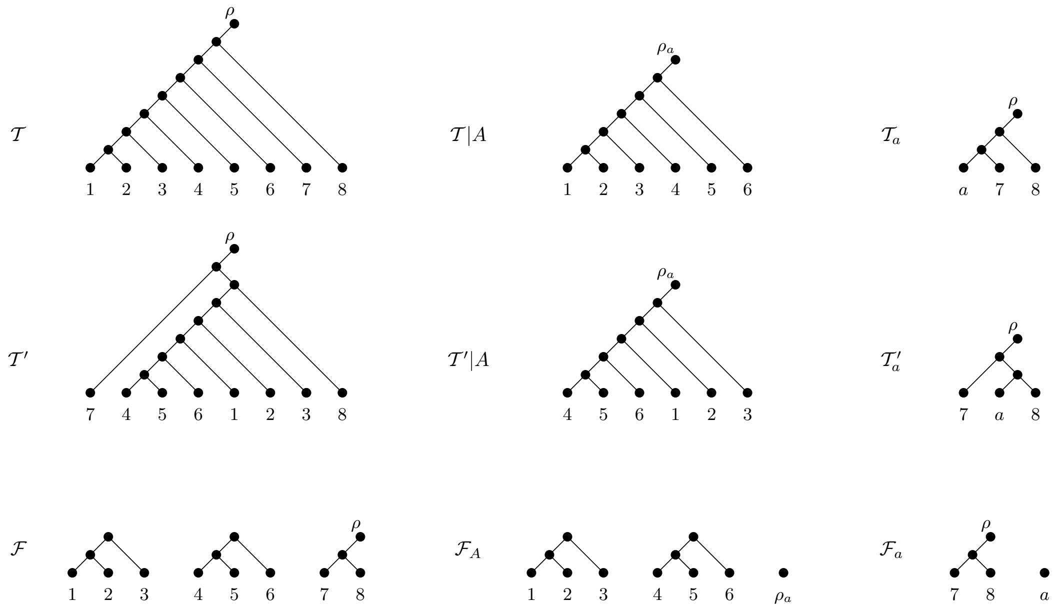


Figure 7.1: A maximum-agreement forest $\mathcal{F}$ for the two rooted binary trees $\mathcal{T}$ and $\mathcal{T}'$ on the same set of taxa (left). A maximum-agreement forest $\mathcal{F}_A$ for the cluster-tree pair $\mathcal{T}|A$ and $\mathcal{T}'|A$ (middle) and a maximum-agreement forest $\mathcal{F}_a$ for the cluster-reduced tree pair $\mathcal{T}_a$ and $\mathcal{T}'_a$ (right) resulting from applying the cluster reduction to the cluster $A = \{1, 2, \ldots, 6\}$ of $\mathcal{T}$ and $\mathcal{T}'$.

However, a similar result for the cluster reduction in the context of calculating the rSPR distance is not yet established. In the remainder of this chapter, we show how the problem of calculating exactly the rSPR distance between two rooted binary phylogenetic X-trees can be broken into two smaller subproblems by a single application of the cluster reduction (Section 7.2) before showing two examples that point out some difficulties in trying to apply this reduction more than once (Section 7.3).

Let $\mathcal{F}$ be a maximum-agreement forest for two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ that have a cluster A with $|A| \geq 2$ in common. Suppose that the cluster reduction (see Section 2.4) is applied to $\mathcal{T}$ and $\mathcal{T}'$ such that the resulting two tree pairs are the cluster-reduced tree pair $\mathcal{T}_a$ and $\mathcal{T}'_a$ obtained from $\mathcal{T}$ and $\mathcal{T}'$, respectively, by replacing the subtree having leaf set A with a single vertex labeled a and the corresponding cluster-tree pair $\mathcal{T}|A$ and $\mathcal{T}'|A$ whose label set is A. Furthermore, let $\mathcal{F}_A$ be a maximum-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$, and let $\mathcal{F}_a$ be such a forest for $\mathcal{T}_a$ and $\mathcal{T}'_a$.

Theorem 2.5 shows that the hybridization number of $\mathcal{T}$ and $\mathcal{T}'$ is equal to the sum of the hybridization numbers of the cluster-tree pair and the cluster-reduced tree pair that result from applying the cluster reduction once. By considering that the rSPR distance between $\mathcal{T}$ and $\mathcal{T}'$ is equal to the size of a maximum-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ minus one (see Theorem 2.1 of Bordewich and Semple, 2004), it is suggesting to conjecture that the cluster reduction can also be used to calculate the rSPR distance between $\mathcal{T}$ and $\mathcal{T}'$, denoted by $d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}')$. However, Figure 7.1 shows that the result does not hold in this context since

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') = |\mathcal{F}| - 1 = 2$$

and

$$d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) + d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}_a') = |\mathcal{F}_A| - 1 + |\mathcal{F}_a| - 1 = 2 + 1 = 3.$$

As stated in Proposition 11.7 of Semple (2007), the rSPR distance only satisfies the following weaker result

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') \leq d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) + d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}_a') \leq d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') + 1.$$

Recalling Lemma 2.2, no label set of a maximum-acyclic-agreement forest exclusively contains the root label, whereas Figure 7.1 indicates that a maximum-agreement forest for two rooted binary phylogenetic X-trees can contain an isolated vertex that represents the root. Hence, there can exist two maximum-agreement forests $\mathcal{F}_A$ and $\mathcal{F}_a$ such that $\mathcal{F}_A$ contains an isolated vertex representing the root, labeled ρ_a say, and $\mathcal{F}_a$ contains an isolated vertex labeled a. As we will shortly see, this situation is crucial in establishing a cluster result for the calculation of $d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}')$. Furthermore, it is important to note now that we do not impose a minimality criteria on the size of the cluster that is reduced in the course of a cluster reduction to break up the problem of computing $d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}')$.

7.2 A Single Application of the Cluster Reduction

In this section, we first establish two lemmas that are needed to prove the main theorem of this chapter. This theorem shows how one can calculate the rSPR distance between two rooted binary phylogenetic trees on the same label set by calculating this distance for the two tree pairs resulting from applying the cluster reduction.

Suppose that $\mathcal{F} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$ is a maximum-agreement forest for two rooted binary phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$ that have a cluster A with $|A| \geq 2$ in common. We say that $\mathcal{F}$ contains a *mixed tree* $\mathcal{T}_m$ if there exists an element $m \in \{\rho, 1, 2, \ldots, k\}$ such that $\mathcal{L}(\mathcal{T}_m) \cap A \neq \emptyset$ and $\mathcal{L}(\mathcal{T}_m) \cap ((X - A) \cup \{\rho\}) \neq \emptyset$. Note that there can exist at most one such tree in $\mathcal{F}$ since, otherwise, the trees $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}$ are not vertex-disjoint in $\mathcal{T}$ and the trees $\{\mathcal{T}'(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}$ are not vertex-disjoint in $\mathcal{T}'$, respectively.

Depending on whether there exists a maximum-agreement forest $\mathcal{F}$ containing a mixed tree, we next prove two lemmas which provide lower bounds on calculating $d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}')$.

Lemma 7.2. *Let $\mathcal{F}$ be a maximum-agreement forest for two rooted binary phylogenetic X-tree $\mathcal{T}$ and $\mathcal{T}'$ that have a cluster A in common with $|A| \geq 2$. Furthermore, let $\mathcal{T}_a$ and $\mathcal{T}_a'$ be the cluster-reduced tree pair obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the cluster reduction to A, and let $\mathcal{T}|A$ and $\mathcal{T}'|A$ be the corresponding cluster-tree pair. If $\mathcal{F}$ contains a mixed tree, then*

$$d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') \geq d_{\mathrm{rSPR}}(\mathcal{T}_a,\mathcal{T}_a') + d_{\mathrm{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A). \qquad (7.1)$$

Proof. Let $\mathcal{T}_m$ be the mixed tree of $\mathcal{F}$. The minimal rooted subtree of $\mathcal{T}$ which contains the label set of the mixed tree includes the root of $\mathcal{T}|A$ and, similarly, this also holds for $\mathcal{T}'$. Since $\mathcal{F}$ is an agreement forest, this implies that $\mathcal{T}_m$ is the unique tree in $\mathcal{F}$ with the described properties. Let $\mathcal{T}_a$ be the tree obtained from $\mathcal{T}_m$ by replacing the pendant subtree $\mathcal{T}_m|(A \cap \mathcal{L}(\mathcal{T}_m))$ with a single leaf labeled a. Furthermore, let the tree $\mathcal{T}_{\rho_a}$ be obtained from $\mathcal{T}_m$ by adding a vertex labeled ρ_a at the end of a pendant edge adjoined to the root of $\mathcal{T}_m|(\mathcal{L}(\mathcal{T}_m) \cap A)$. Then, as $\mathcal{F}$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$,

$$\mathcal{F}_a = \{\mathcal{T}_j \in \mathcal{F} : \mathcal{L}(\mathcal{T}_j) \subseteq ((X - A) \cup \{\rho\})\} \cup \{\mathcal{T}_a\}$$

is an agreement forest for $\mathcal{T}_a$ and $\mathcal{T}_a'$ and

$$\mathcal{F}_A = \{\mathcal{T}_j \in \mathcal{F} : \mathcal{L}(\mathcal{T}_j) \subseteq A\} \cup \{\mathcal{T}_{\rho_a}\}$$

is such a forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$. It is easily checked that $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 1$. Then it follows from Theorem 2.1 of Bordewich and Semple (2004) that

$$\begin{aligned} d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') &= |\mathcal{F}| - 1 \\ &= |\mathcal{F}_a| + |\mathcal{F}_A| - 1 - 1 \\ &\geq d_{\mathrm{rSPR}}(\mathcal{T}_a,\mathcal{T}_a') + d_{\mathrm{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A). \end{aligned}$$

This inequality gives the desired result and completes the proof of the lemma. □

Lemma 7.3. *Let $\mathcal{F}$ be a maximum-agreement forest for two rooted binary phylogenetic X-tree $\mathcal{T}$ and $\mathcal{T}'$ that have a cluster A in common with $|A| \geq 2$. Furthermore, let $\mathcal{T}_a$ and $\mathcal{T}_a'$ be the cluster-reduced tree pair obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the cluster reduction to A, and let $\mathcal{T}|A$ and $\mathcal{T}'|A$ be the corresponding cluster-tree pair. If $\mathcal{F}$ contains* *no mixed tree, then*

$$d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') \geq d_{\mathrm{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1. \tag{7.2}$$

Proof. Let $\{a\}$ and $\{\rho_a\}$ be used to denote the trees consisting of an isolated vertex labeled a and ρ_a, respectively. All trees $\mathcal{T}_j \in \mathcal{F}$ have the property that either $\mathcal{L}(\mathcal{T}_j) \subseteq A$ or $\mathcal{L}(\mathcal{T}_j) \subseteq ((X-A) \cup \{\rho\})$. Since $\mathcal{F}$ is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$, it is easily seen that

$$\mathcal{F}_a = \{\mathcal{T}_j \in \mathcal{F} : \mathcal{L}(\mathcal{T}_j) \subseteq ((X-A) \cup \{\rho\})\} \cup \{a\}$$

is an agreement forest for $\mathcal{T}_a$ and $\mathcal{T}'_a$ and

$$\mathcal{F}_A = \{\mathcal{T}_j \in \mathcal{F} : \mathcal{L}(\mathcal{T}_j) \subseteq A\} \cup \{\rho_a\}$$

is such a forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$. Thus $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 2$. Again, by Theorem 2.1 of Bordewich and Semple (2004), we can deduce that

$$\begin{aligned} d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') &= |\mathcal{F}| - 1 \\ &= |\mathcal{F}_a| + |\mathcal{F}_A| - 2 - 1 \\ &\geq d_{\mathrm{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1 \end{aligned}$$

and the result follows immediately. □

Having the results of the above two lemmas, we are now in a position to state the central theorem of this chapter. Given all maximum-agreement forests for a cluster-tree pair and all such forests for the corresponding cluster-reduced tree pair, it shows how one can calculate the rSPR distance between the two initial unreduced trees.

Theorem 7.4. *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rooted binary phylogenetic X-trees and suppose that A is a common cluster of $\mathcal{T}$ and $\mathcal{T}'$ with $|A| \geq 2$. Let $\mathcal{T}_a$ and $\mathcal{T}'_a$ be the cluster-reduced tree pair obtained from $\mathcal{T}$ and $\mathcal{T}'$ by applying the cluster reduction to A, and suppose that the label set of both trees is $(X-A) \cup \{a\}$, where $a \notin X$. Furthermore, let $\mathcal{T}|A$ and $\mathcal{T}'|A$ be the corresponding cluster-tree pair. If there exists a maximum-agreement forest $\mathcal{F}_a$ for $\mathcal{T}_a$ and $\mathcal{T}'_a$ in which a labels an isolated vertex and if there exists a maximum-agreement forest $\mathcal{F}_A$ for $\mathcal{T}|A$ and $\mathcal{T}'|A$ in which the root labeled ρ_a is an isolated vertex, then*

$$d_{\mathrm{rSPR}}(\mathcal{T},\mathcal{T}') = d_{\mathrm{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1$$

and, otherwise,

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') = d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A).$$

Proof. Let $\{a\}$ and $\{\rho_a\}$ be used to denote the trees consisting of an isolated vertex labeled a and ρ_a, respectively. First, we show that the inequality

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') \leq d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) - 1 \tag{7.3}$$

holds. Let $\mathcal{F}_a$ be a maximum-agreement forest for $\mathcal{T}_a$ and $\mathcal{T}'_a$ that contains an isolated vertex labeled a, and let $\mathcal{F}_A$ be a maximum-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$ that contains an isolated vertex labeled ρ_a. Then

$$\mathcal{F} = (\mathcal{F}_a - \{a\}) \cup (\mathcal{F}_A - \{\rho_a\})$$

is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ with $|\mathcal{F}| = |\mathcal{F}_a| - 1 + |\mathcal{F}_A| - 1$ and as a result, we have

$$\begin{aligned} d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) - 1 &= |\mathcal{F}_a| - 1 + |\mathcal{F}_A| - 1 - 1 \\ &= |\mathcal{F}| - 1 \\ &\geq d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}'). \end{aligned}$$

This establishes Equation 7.3.

We next show that

$$d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') \leq d_{\mathrm{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A). \tag{7.4}$$

Let $\mathcal{F}_a$ be a maximum-agreement forest for $\mathcal{T}_a$ and $\mathcal{T}'_a$, and let $\mathcal{F}_A$ be a maximum-agreement forest for $\mathcal{T}|A$ and $\mathcal{T}'|A$. Suppose that a and ρ_a do not both label isolated vertices in $\mathcal{F}_a$ and $\mathcal{F}_A$, respectively. Let $\mathcal{T}_a$ be the unique tree in $\mathcal{F}_a$ with a vertex labeled a, and let $\mathcal{T}_{\rho_a}$ be the unique tree in $\mathcal{F}_A$ with a vertex labeled ρ_a. Then the tree $\mathcal{T}_{a,\rho_a}$ can be obtained in one of the following three ways.

(i) If neither a nor ρ_a labels an isolated vertex in $\mathcal{F}_a$ and $\mathcal{F}_A$, respectively, then $\mathcal{T}_{a,\rho_a}$ is obtained from $\mathcal{T}_{\rho_a}$ and $\mathcal{T}_a$ by adjoining $\mathcal{T}_{\rho_a}$ to $\mathcal{T}_a$ via a new edge joining the vertex labeled a with the root labeled ρ_a, removing the labels a and ρ_a, and suppressing any degree two vertices apart from the root.

(ii) If a labels an isolated vertex in $\mathcal{F}_a$ and ρ_a does not label an isolated vertex in $\mathcal{F}_A$, then $\mathcal{T}_{a,\rho_a}$ is obtained from $\mathcal{T}_{\rho_a}$ by deleting the edge which is incident with the vertex labeled ρ_a and the vertex labeled ρ_a itself.

(iii) If a does not label an isolated vertex in $\mathcal{F}_a$ and ρ_a labels an isolated vertex in $\mathcal{F}_A$, then $\mathcal{T}_{a,\rho_a}$ is obtained from $\mathcal{T}_a$ by deleting the pendant edge ending in the vertex labeled a and the vertex labeled a itself.

Since $\mathcal{F}_a$ and $\mathcal{F}_A$ are agreement forests, we have

$$\mathcal{F} = (\mathcal{F}_a \cup \mathcal{F}_A - \{\mathcal{T}_a, \mathcal{T}_{\rho_a}\}) \cup \{\mathcal{T}_{a,\rho_a}\}$$

is an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. With $|\mathcal{F}| = |\mathcal{F}_a| + |\mathcal{F}_A| - 1$ it now follows that,

$$\begin{aligned} d_{\text{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) &= |\mathcal{F}_a| - 1 + |\mathcal{F}_A| - 1 \\ &= |\mathcal{F}| - 1 \\ &\geq d_{\text{rSPR}}(\mathcal{T}, \mathcal{T}'). \end{aligned}$$

This establishes Equation 7.4.

For the second part of this proof, let $\mathcal{F}$ be a maximum-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. There are two cases to consider:

(i) $\mathcal{F}$ contains no mixed tree and

(ii) $\mathcal{F}$ contains a mixed tree.

Suppose that there exists a maximum-agreement forest $\mathcal{F}_a$ for $\mathcal{T}_a$ and $\mathcal{T}'_a$ that contains an isolated vertex labeled a and that there exists a maximum-agreement forest $\mathcal{F}_A$ for $\mathcal{T}|A$ and $\mathcal{T}'|A$ that contains an isolated vertex labeled ρ_a. We next show that the inequality

$$d_{\text{rSPR}}(\mathcal{T}, \mathcal{T}') \geq d_{\text{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) - 1 \qquad (7.5)$$

holds. If no mixed tree exists, then Equation 7.5 follows directly from Lemma 7.3. On the other hand, if a mixed tree $\mathcal{T}_m$ exists, assume that the Inequality 7.5 does not hold; that is

$$d_{\text{rSPR}}(\mathcal{T}, \mathcal{T}') \leq d_{\text{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) - 2.$$

Linking this inequality with the result of Lemma 7.2, we have

$$d_{\text{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) \leq d_{\text{rSPR}}(\mathcal{T}, \mathcal{T}') \leq d_{\text{rSPR}}(\mathcal{T}_a, \mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) - 2.$$

This gives a contradiction. By combining both cases with Equation 7.3, we can deduce that

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') = d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1. \tag{7.6}$$

This establishes the first part of the theorem.

Now suppose that there exists no combination of two maximum-agreement forests $\mathcal{F}_a$ and $\mathcal{F}_A$ such that a labels an isolated vertex in $\mathcal{F}_a$ and ρ_a labels an isolated vertex in $\mathcal{F}_A$. We next show that the inequality

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') \geq d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) \tag{7.7}$$

holds. If a mixed tree exists in $\mathcal{F}$, Equation 7.7 follows directly from Lemma 7.2. Otherwise, by Lemma 7.3, a maximum-agreement forest $\mathcal{F}$ which does not contain a mixed tree has the property that

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') \geq d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1.$$

Recalling our assumption that no combination of two maximum-agreement forests $\mathcal{F}_a$ and $\mathcal{F}_A$ exists such that a labels an isolated vertex in $\mathcal{F}_a$ and ρ_a labels an isolated vertex in $\mathcal{F}_A$, we can follow that $\mathcal{F}_a$ or $\mathcal{F}_A$ is not of smallest size. This can be easily checked by considering the construction of $\mathcal{F}_a$ or $\mathcal{F}_A$ in the proof of Lemma 7.3, where a and ρ_a both label isolated vertices. Hence, $\mathcal{F}_a$ and $\mathcal{F}_A$ can be obtained from $\mathcal{F}$ in a way such that $|\mathcal{F}| > |\mathcal{F}_a| + |\mathcal{F}_A| - 2$. Thus

$$\begin{aligned} d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') &= |\mathcal{F}| - 1 \\ &> |\mathcal{F}_a| + |\mathcal{F}_A| - 2 - 1 \\ &\geq d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1 \end{aligned} \tag{7.8}$$

or, in short, $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') > d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1$. For a contradiction, now assume that Inequality 7.7 does not hold; that is

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') \leq \ d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1.$$

Combining this assumption with Inequality 7.8, we have

$$d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1 < d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') \leq d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A) - 1.$$

This inequality gives a contradiction. By considering both cases and Equation 7.4, we can follow that

$$d_{\text{rSPR}}(\mathcal{T},\mathcal{T}') = d_{\text{rSPR}}(\mathcal{T}_a,\mathcal{T}'_a) + d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A).$$

This completes the proof of the theorem. □

7.3 Some First Insight into Repeated Applications of the Cluster Reduction

Since a running time analysis of the HybridNumber algorithm (see Section 3) has shown that the cluster reduction has a considerable positive impact on its speed, it is likely that such a framework can make a contribution to the development of exact algorithms to calculate the rSPR distance between two rooted binary phylogenetic trees. Consequently, it is of interest to apply the cluster reduction more than once. In the following, we provide the reader with some first insight into such a framework by trying to extend the result of the previous section. We will shortly see some examples pointing out that this is not easily achievable. For example, it is not sufficient to calculate maximum-agreement forests only, but agreement forests up to a certain size need to be considered.

We first introduce some new definitions. Let $\mathcal{T}_0$ and $\mathcal{T}'_0$ be two rooted binary phylogenetic X-trees. We say that the cluster reduction has been applied t *times* if, for all $j \in \{1,2,\dots,t\}$, the j^{th} cluster reduction replaces $\mathcal{T}_{j-1}$ and $\mathcal{T}'_{j-1}$, with two new tree pairs:

(i) the *cluster-reduced tree pair* $\mathcal{T}_j$ and $\mathcal{T}'_j$ which has been obtained from $\mathcal{T}_{j-1}$ and $\mathcal{T}'_{j-1}$ by replacing a subtree whose label set A_j is a common cluster (with $|A_j| \geq 2$) of $\mathcal{T}_{j-1}$ and $\mathcal{T}'_{j-1}$ with a single vertex labeled $l_j \notin (X \cup \{\rho, l_1, l_2, \dots, l_{j-1}\})$ and

(ii) the *cluster-tree pair* $\mathcal{T}_{j-1}|A_j$ and $\mathcal{T}'_{j-1}|A_j$.

In the following, we refer to the tuple

$$\mathcal{R} = (\{\mathcal{T}_0|A_1,\mathcal{T}'_0|A_1\},\{\mathcal{T}_1|A_2,\mathcal{T}'_1|A_2\},\dots,\{\mathcal{T}_{t-1}|A_t,\mathcal{T}'_{t-1}|A_t\},\{\mathcal{T}_t,\mathcal{T}'_t\})$$

as a *cluster-tree collection* for $\mathcal{T}_0$ and $\mathcal{T}'_0$. To calculate an agreement forest for two rooted binary phylogenetic trees on the same label set, a root vertex is added at the end of a pendant edge adjoined to the original roots (see Figure 2.2). We label this vertex with ρ_j for all cluster-tree pairs $\mathcal{T}_{j-1}|A_j$ and $\mathcal{T}'_{j-1}|A_j$ with $j \in \{1,2,\dots,t\}$ and with ρ for $\mathcal{T}_t$ and $\mathcal{T}'_t$.

Let $\mathcal{F}^*$ be a collection of forests such that each member of $\mathcal{F}^*$ is a maximum-agreement

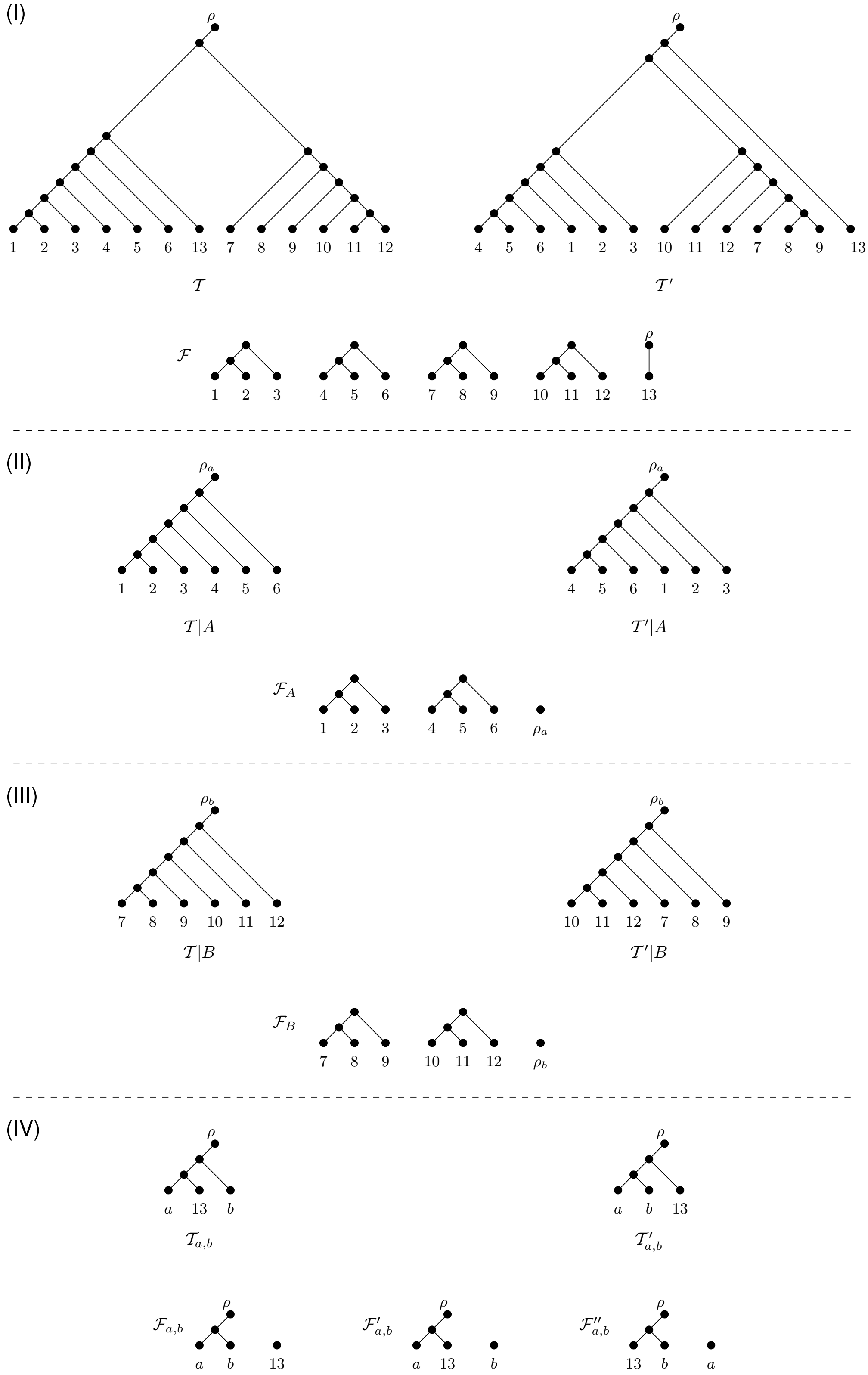


Figure 7.2: Counterexample to Conjecture 7.5 with $A = \{1, 2, \ldots, 6\}$ and $B = \{7, 8, \ldots, 12\}$. (For details, see text.)

forest for a tree pair in $\mathcal{R}$. Then, for all $j \in \{1, 2, \ldots, t\}$, set s_j to be 1 if there exists a maximum-agreement forest in $\mathcal{F}^*$ with an isolated vertex labeled l_j and if there exists a maximum-agreement forest in $\mathcal{F}^*$ with an isolated vertex labeled ρ_j and, otherwise, set s_j to be 0. In the following, we refer to

$$S = \sum_{j=1}^{t} s_j$$

as *isolation score for* $\mathcal{F}^*$.

Conjecture 7.5. *Let $\mathcal{T}_0$ and $\mathcal{T}_0'$ be two rooted binary phylogenetic X-trees, and let*

$$\mathcal{R} = (\{\mathcal{T}_0|A_1, \mathcal{T}_0'|A_1\}, \{\mathcal{T}_1|A_2, \mathcal{T}_1'|A_2\}, \ldots, \{\mathcal{T}_{t-1}|A_t, \mathcal{T}_{t-1}'|A_t\}, \{\mathcal{T}_t, \mathcal{T}_t'\})$$

be a cluster-tree collection for $\mathcal{T}_0$ and $\mathcal{T}_0'$. Furthermore, let $\mathcal{F}^*$ be precisely the collection of maximum-agreement forests that contains all such forests for each tree pair in $\mathcal{R}$, and let S be the isolation score associated with $\mathcal{F}^*$. Then

$$d_{\mathrm{rSPR}}(\mathcal{T}_0, \mathcal{T}_0') = d_{\mathrm{rSPR}}(\mathcal{T}_t, \mathcal{T}_t') + \sum_{j=1}^{t} d_{\mathrm{rSPR}}(\mathcal{T}_{j-1}|A_j, \mathcal{T}_{j-1}'|A_j) - S. \qquad (7.9)$$

If $t = 1$, then, by recalling Theorem 7.4 and the definition of the isolation score S, it is easily checked that the conjecture holds. However, this is not necessarily the case if $t > 1$. To see this, consider Figure 7.2 which shows two rooted binary trees $\mathcal{T}$ and $\mathcal{T}'$ on the same label set whose rSPR distance is $|\mathcal{F}| - 1 = 4$ (see (I)). The three tree pairs of the cluster-tree collection

$$\mathcal{R} = (\{\mathcal{T}|A, \mathcal{T}'|A\}, \{\mathcal{T}|B, \mathcal{T}'|B\}, \{\mathcal{T}_{a,b}, \mathcal{T}_{a,b}'\})$$

for $\mathcal{T}$ and $\mathcal{T}'$ are shown in (II), (III), and (IV). Note that there only exists one maximum-agreement forest for each of the tree pairs visualized in (I), (II), and (III), whereas there are three such forests for $\mathcal{T}_{a,b}$ and $\mathcal{T}_{a,b}'$. Since a, b, ρ_a, and ρ_b label an isolated vertex in some maximum-agreement forest, we calculate the following rSPR distance by applying Equation 7.9:

$$\begin{aligned} d_{\mathrm{rSPR}}(\mathcal{T}_{a,b}, \mathcal{T}_{a,b}') + d_{\mathrm{rSPR}}(\mathcal{T}|A, \mathcal{T}'|A) + d_{\mathrm{rSPR}}(\mathcal{T}|B, \mathcal{T}'|B) - S &= 1 + 2 + 2 - 2 \\ &< d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}') \\ &= 4. \end{aligned}$$

Having this result, it is easily seen that one needs to choose precisely one agreement forest for each tree pair contained in $\mathcal{R}$. Hence, the crucial question is which agreement forests one has to choose to calculate the rSPR distance between the two initial two trees.

To see why it is not sufficient to consider maximum-agreement forests only, a second example is presented in Figure 7.3. Again, two rooted binary trees $\mathcal{T}$ and $\mathcal{T}'$ on the same label set are shown whose rSPR distance is $|\mathcal{F}|-1=5$ (see (I)). Further below, three tree pairs (see (II), (III), and (IV)) are depicted that are contained in the associated 3-tuple

$$\mathcal{R}=(\{\mathcal{T}|A,\mathcal{T}'|A\},\{\mathcal{T}|B,\mathcal{T}'|B\},\{\mathcal{T}_{a,b},\mathcal{T}'_{a,b}\}).$$

Since each of these tree pairs has exactly one maximum-agreement forest and neither a and ρ_a nor b and ρ_b both label isolated vertices, it is straightforward to calculate the following rSPR distance:

$$\begin{aligned} d_{\text{rSPR}}(\mathcal{T}_{a,b},\mathcal{T}'_{a,b})+d_{\text{rSPR}}(\mathcal{T}|A,\mathcal{T}'|A)+d_{\text{rSPR}}(\mathcal{T}|B,\mathcal{T}'|B) &= 2+2+2\\ &> d_{\text{rSPR}}(\mathcal{T},\mathcal{T}')\\ &= 5. \end{aligned}$$

The result shows that $\mathcal{F}_{a,b}$, $\mathcal{F}_B$, and $\mathcal{F}_A$ do not lead to the correct rSPR distance between $\mathcal{T}$ and $\mathcal{T}'$ and that one needs to consider agreement forests up to a certain size. For example, if all agreement forests for $\mathcal{T}_{a,b}$ and $\mathcal{T}'_{a,b}$ with at most four trees are calculated, then one of these forests is $\mathcal{F}^*_{a,b}$ (see (IV)). By considering that a, b, ρ_a, and ρ_b label isolated vertices, it is then easily checked that a maximum-agreement forest $\mathcal{F}^*$ for $\mathcal{T}$ and $\mathcal{T}'$ can be obtained from $\mathcal{F}^*_{a,b}$, $\mathcal{F}_B$, and $\mathcal{F}_A$ with $\mathcal{F}=\mathcal{F}^*$. Consequently, we have $d_{\text{rSPR}}(\mathcal{T},\mathcal{T}')=|\mathcal{F}^*|-1$.

It is subject of ongoing research to consider these results and to develop an appropriate framework that will finally allow for repeated applications of the cluster reduction. Intuitively, one wants to find exactly one agreement forest for each tree pair in

$$\mathcal{R}=(\{\mathcal{T}_0|A_1,\mathcal{T}'_0|A_1\},\{\mathcal{T}_1|A_2,\mathcal{T}'_1|A_2\},\ldots,\{\mathcal{T}_{t-1}|A_t,\mathcal{T}'_{t-1}|A_t\},\{\mathcal{T}_t,\mathcal{T}'_t\})$$

such that the number of trees over all $t+1$ forests minus the number of trees over all $t+1$ forests whose label sets are subsets of $\{l_1,l_2,\ldots,l_t,\rho_1,\rho_2,\ldots,\rho_t\}$ is minimized over all such collections of $t+1$ agreement forests.

Remark. It is straightforward to check that the result of Theorem 7.4 can be upgraded to two arbitrary rooted phylogenetic trees $\mathcal{T}$ and $\mathcal{T}'$ by taking into account that more than

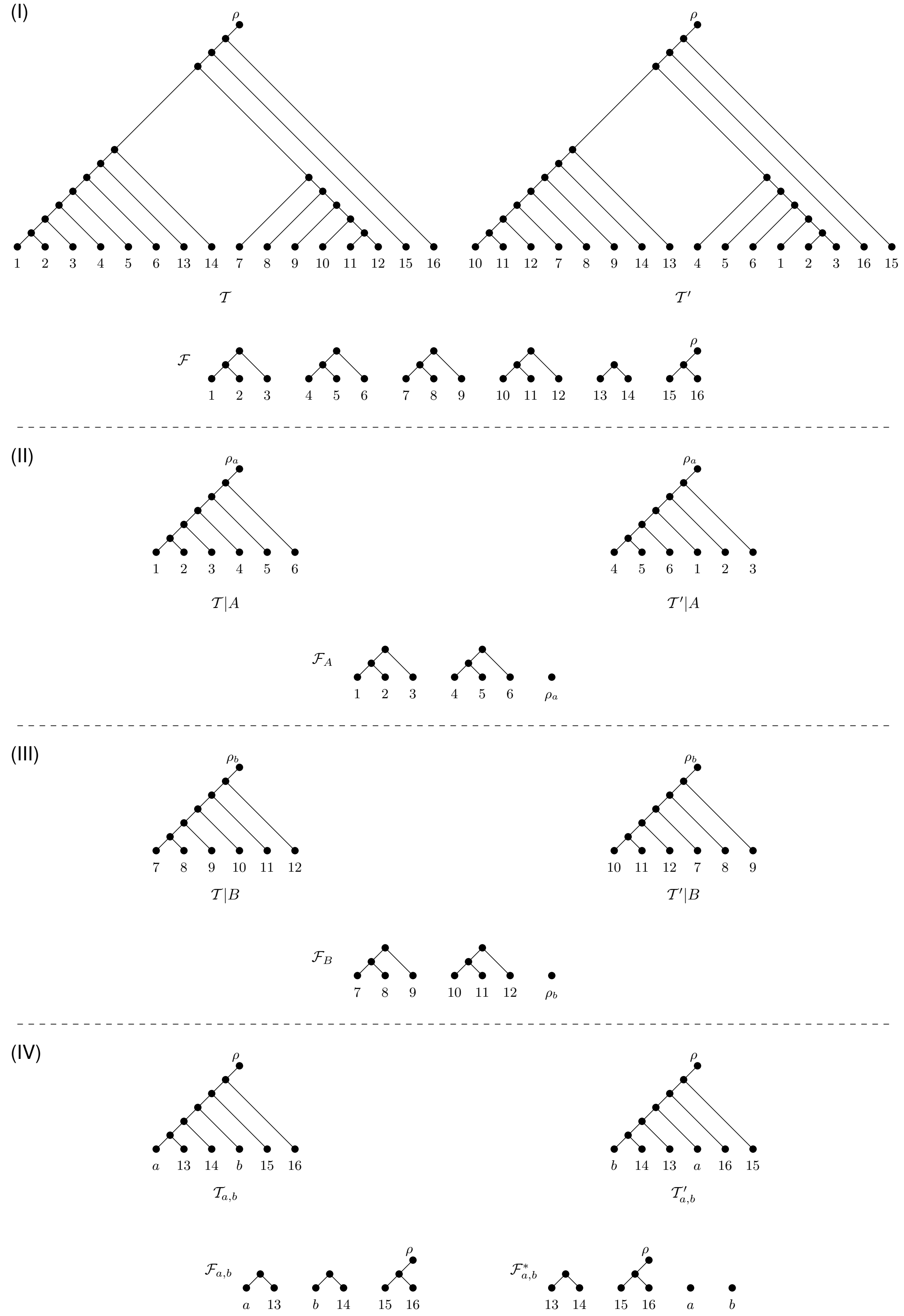


Figure 7.3: Example showing that it is not sufficient to consider maximum-agreement forests only. Here, the cluster reduction is applied to the following two clusters of $\mathcal{T}$ and $\mathcal{T}'$: $A = \{1, 2, \dots, 6\}$ and $B = \{7, 8, \dots, 12\}$. (For details, see text.)

one mixed tree in a maximum-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ can exist. For a definition of $d_{\mathrm{rSPR}}(\mathcal{T}, \mathcal{T}')$ if $\mathcal{T}$ and $\mathcal{T}'$ are rooted phylogenetic X-trees, see page 74. Furthermore, the subtree and chain reduction can be applied in the same way as stated in Section 5.5, but without considering the set P with an associated weight function. Than and Nakhleh (2008) have recently approached the problem of calculating the number of reticulation events for two rooted phylogenetic trees in terms of rSPR operations, thus providing a lower bound on this number. They have suggested a quadratic-time algorithm that refines two trees as much as possible and preserves the number of rSPR operations before applying the subtree reduction and a restricted version of the chain reduction.

Summary

Reticulate evolution—the umbrella term for processes like hybridization, horizontal gene transfer (HGT), and recombination—results in species whose genomes are mosaics of DNA segments derived from distinct ancestral species. Consequently, the analysis of different genetic loci often reveals incompatibilities between gene trees due to different branching patterns. Inferring phylogenies in the presence of reticulation events has turned out to be more complicated because the history of life can probably not be best represented by using evolutionary trees and phylogenetic networks seem to be more appropriate in these cases.

This thesis describes two new mathematical models that aim at calculating the extent to which hybridization and HGT, respectively, have influenced the development of the current diversity of species. More precisely, the following results have been established:

Measuring Hybridization for a Set of Phylogenetic Trees (Chapter 2):
Recently, Bordewich and Semple (2007a,b), and Baroni *et al.* (2005, 2006) have developed a combinatorial framework to calculate the minimum number of hybridization events for two rooted binary phylogenetic X-trees. This approach provides the background for Chapters 3, 4, and 5 of this thesis. Rather than repeating this concept, we have presented a generalization for when the set of trees is arbitrarily large.

HybridNumber**: A Reduction Algorithm for Hybridization (Chapter 3):**
In the context of this chapter, we have shown how the combinatorial approach which is described in Chapter 2 can be used to implement the new and exact algorithm HybridNumber that computes the minimum number of hybridization events for two rooted binary phylogenetic trees on the same taxa set. HybridNumber is based on a characterization of the hybridization number in terms of agreement forests and repeated applications of three reduction rules that reduce the size of the problem instance before calculating the hybridization number exactly. Given that the underlying problem is NP-hard, we have shown that HybridNumber runs efficiently on many instances of a grass data set (Grass Phylogeny Working Group, 2001) and returns the exact solution within a reasonable time.

How Deep is a Hybridization Event? (Chapter 4):
To approach the question when hybridization events have occurred during the evolutionary history of a set of present-day species, we have established some theoretical results which show that a combination of a modified version of HybridNumber with a new algorithm BuildForest is suitable to calculate all combinations of hybridization events

of minimum size for a given tree pair. We applied the algorithms to the previously mentioned grass data set and compared the number of hybridization events at the leaves of an associated network with those at interior vertices of this network. The results indicate that—in line with our expectations—the majority of hybridization events have occurred relatively recently.

Hybridization in Non-Binary Trees (Chapter 5):
Based on the combinatorial framework to calculate the minimum number of hybridization events for rooted binary phylogenetic trees that has been described in Chapters 2 and 3, we have shown that calculating this number for two (arbitrary) rooted phylogenetic X-trees is fixed-parameter tractable. Motivated by the fact that many biological data sets contain trees with polytomies, we have established theoretical results that finally allow for a reduction of the problem instance such that the size of the label set of the resulting two trees is linear in the actual number of hybridization events for the original two trees. This has been achieved by a careful upgrade of the notion of agreement forests and the statement of several reduction rules which are similar to the subtree, chain, and cluster reduction for binary trees but also contain crucial extensions.

A Likelihood Framework to Measure Horizontal Gene Transfer (Chapter 6):
Beside hybridization, HGT is another important process of reticulate evolution that is common among bacteria. Assuming that the number of HGT events within a given time interval follows a Poisson process, we have introduced a new method to simulate different numbers of HGT events on a given species tree. The resulting tree distribution can afterwards be used to estimate a rate of HGT for a set of gene trees in a likelihood-based framework. We applied this newly developed method to the COG data set (Tatusov *et al.*, 2001). Additionally considering inaccuracies due to the gene tree reconstruction method, the results suggest an HGT rate of about 0.36 per gene and unit time or, in other words, 11 HGT events per gene occurred on average among the 44 taxa of the COG species tree.

A New Result for Computing the Rooted Subtree Prune and Regraft Distance (Chapter 7):
The rooted subtree prune and regraft distance has often been used to provide a lower bound on the number of reticulation events. Calculating this distance between two trees is an NP-hard problem and exact algorithms are rare. In this chapter, we have introduced a new reduction that can be used to decompose the problem instance into two smaller and, hence, more tractable subproblems. Although this reduction is similar to the cluster reduction that can be used to compute the number of hybridization events for two trees, there are some crucial differences. Since it is desirable but challenging to apply this

reduction more than once, we give two examples pointing out some difficulties in trying to achieve this.

Zusammenfassung

Hybridisierung, horizontaler Gentransfer (HGT) und Rekombination sind wichtige Prozesse der Evolution, die in Spezies resultieren, deren Genome aus DNA Segmenten verschiedener Arten zusammengesetzt sind. Infolgedessen deckt ein Vergleich mehrere Genbäume, die für verschiedene Gene einer Gruppe von Spezies konstruiert wurden, oft Widersprüche in den Baumtopologien auf. Die Rekonstruktion phylogenetischer Bäume in Gegenwart von Hybridisierung, HGT oder Rekombination ist daher eine komplexe Fragestellung von aktuellem Forschungsinteresse, und es wird diskutiert, ob die Evolution der Arten ausreichend genau mit Hilfe eines phylogenetischen Baumes dargestellt werden kann. Phylogenetische Netzwerke scheinen in diesem Zusammenhang eine geeignetere Wahl zu sein.

Diese Arbeit beschreibt zwei neu entwickelte, mathematische Modelle, mit denen man analysieren kann, wie stark Hybridisierung und HGT die Entwicklung der Arten beeinflusst hat. Im Einzelnen wurden die folgenden Resultate erzielt:

Ein kombinatorischer Ansatz zur Berechnung von Hybridisierung (Kapitel 2):
Bordewich und Semple (2007a,b) sowie Baroni *et al.* (2005, 2006) haben vor kurzem eine Methode vorgestellt, mit der man die minimale Anzahl von Hybridisierungsereignissen (im Folgenden als *Hybridisierungszahl* bezeichnet) für zwei gewurzelte, binäre Bäume berechnen kann. Dieser kombinatorisch ausgerichtete Ansatz ist Grundlage für Kapitel 3, 4 und 5 der vorliegenden Arbeit. Statt die Theorie der genannten Veröffentlichungen zu wiederholen, haben wir ein verallgemeinertes Konzept für eine beliebig große Anzahl von Bäumen vorgestellt.

Der Algorithmus HybridNumber (Kapitel 3):
In diesem Kapitel haben wir gezeigt, wie die in Kapitel 2 vorgestellte Methode genutzt werden kann, um den Algorithmus HybridNumber zu implementieren, der die Hybridisierungszahl für zwei gewurzelte, binäre Genbäume exakt berechnen kann. Grundlage von HybridNumber sind (1) die Charakterisierung der Hybridisierungszahl mit Hilfe identischer Teilbäume (sogenannte *agreement forests*), in welche die beiden Genbäume zerlegt werden können, sowie (2) die wiederholte Anwendung dreier Reduktionsregeln, welche die Taxaanzahl der Genbäume reduzieren, bevor die Hybridisierungszahl berechnet wird. Trotz der Tatsache, dass das zugrunde liegende Problem NP-schwer ist, konnten wir zeigen, dass HybridNumber für einen Gräserdatensatz (Grass Phylogeny Working Group, 2001) in den meisten Fällen effizient arbeitet und das exakte Ergebnis in angemessener Zeit ausgibt.

Verteilung von Hybridisierungsereignissen im Netzwerk (Kapitel 4):
Schwerpunkt dieses Kapitels ist die Fragestellung, wann die einzelnen Hybridisierungsereignisse während der Evolution stattgefunden haben. Wir haben dazu einige theoretische Resultate präsentiert, die zeigen, dass eine Kombination einer erweiterten Version von HybridNumber mit dem neuen Algorithmus BuildForest dazu benutzt werden kann, alle Kombinationen von Hybridisierungsereignissen für zwei gewurzelte, binäre Bäume zu berechnen. Auf Grundlage dieses theoretischen Ergebnisses haben wir die Algorithmen auf den zuvor erwähnten Gräserdatensatz angewandt und die Anzahl der Hybridisierungsereignisse an den Blättern eines Netzwerks mit der entsprechenden Anzahl von Ereignissen verglichen, die an den internen Knoten stattgefunden hat. Unseren Erwartungen entsprechend konnten wir zeigen, dass die Mehrzahl der Hybridisierungsereignisse an den Blättern und somit in jüngster Zeit stattgefunden hat.

Hybridisierung in phylogenetischen Bäumen mit Polytomien (Kapitel 5):
Basierend auf der in den Kapiteln 2 und 3 vorgestellten kombinatorischen Methode zur Berechnung der Hybridisierungszahl für zwei gewurzelte, binäre Bäume, werden in diesem Kapitel Ergebnisse vorgestellt, die zusammengefasst zeigen, dass die Berechnung der Hybridisierungszahl für beliebige, gewurzelte Bäume *fixed-parameter tractable* ist. Durch die Tatsache motiviert, dass für viele biologische Daten nur Bäume mit Polytomien konstruiert werden können, haben wir die agreement forest Notation erweitert und mehrere Reduktionsregeln formuliert, die eine Verkleinerung der Probleminstanz ermöglichen, bis diese linear zur Hybridisierungszahl ist.

Eine Likelihood-Methode zum Simulieren und Schätzen von HGT (Kapitel 6):
HGT ist neben Hybridisierung ein weiterer wichtiger Vorgang, der zu den nicht baumhaft verlaufenden Evolutionsprozessen gehört. Unter der Annahme, dass der Verteilung von HGT Ereignissen in einem gegebenen Zeitintervall ein Poisson Prozess zugrunde liegt, haben wir eine neue Methode vorgestellt, um HGT Ereignisse auf einem Speziesbaum zu simulieren. Die resultierende Baumverteilung kann dazu benutzt werden, um eine HGT Rate für eine Menge von Genbäumen zu schätzen. Das entwickelte Schätzverfahren ist dabei likelihood-basiert. Diese Methode wurde auf den COG Datensatz (Tatusov *et al.*, 2001) angewandt, und zusätzlich wurden Ungenauigkeiten in der Rekonstruktion der Genbäume berücksichtigt. Die erhaltenen Resultate schlagen eine HGT Rate von 0.36 pro Gen und Zeiteinheit vor. Dies bedeutet, dass im Durchschnitt 11 HGT Ereignisse pro Gen zwischen den 44 Taxa des COG Speziesbaumes stattgefunden haben.

Ein neues Ergebnis zur Berechnung der rSPR Distanz (Kapitel 7):
Die *rooted subtree prune and regraft (rSPR)* Distanz zwischen zwei gewurzelten, binären

Bäumen wird häufig dazu benutzt, um eine untere Schranke für die Anzahl von HGT oder Hybridisierungsereignissen zu bestimmen. Auch dieses Problem ist NP-schwer und nur wenige exakte Algorithmen sind bislang entwickelt worden. In diesem Kapitel haben wir eine Reduktion vorgestellt, die zur Zerlegung des Problems in zwei kleinere Teilprobleme genutzt werden kann. Dabei besteht jedes der beiden resultierenden Teilprobleme aus einem neuen Baumpaar mit einer kleineren Taxamenge. Es ist wünschenswert diese Reduktion mehrfach anwenden zu können, jedoch hat sich herausgestellt, dass dies eine komplexere Aufgabe ist als erwartet. Zur Verdeutlichung haben wir zwei Beispiele gezeigt, welche einige Schwierigkeiten aufdecken, die zur Lösung des Problems umgangen werden müssen.

Abbreviations

BLAST	Basic Local Alignment Search Tool
ChR	chain reduction
ClR	cluster reduction
COG	Cluster of Orthologous Groups of Proteins
DNA	desoxyribo nucleic acid
EMBL	European Molecular Biology Laboratory
GC content	guanine-cytosine content
HGT	horizontal gene transfer
mrca	most recent common ancestor
nc	nucleotide
NCBI	National Center for Biotechnology Information
NNI	nearest neighbor interchange
rDNA	ribosomal DNA
RefSeq	NCBI Reference Sequence
rSPR	rooted subtree prune and regraft
SR	subtree reduction
TBR	tree bisection and reconnection
UPGMA	unweighted pair group method with arithmetic mean

Bibliography

Allan, H. H. (1961). *Flora of New Zealand, Volume I, Indigenous tracheophyta: Psilopsida, Lycopsida, Filicopsida, Gymnospermae, Dicotyledones.* Government Printer, Wellington, New Zealand.

Allen, B. L. and Steel, M. (2001). Subtree transfer operations and their induced metrics on evolutionary trees. *Annals of Combinatorics*, **5**:1-13.

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, **25**:3389-3402.

Andersson, J. O. (2005). Lateral gene transfer in eukaryotes. *Cellular and Molecular Life Sciences*, **62**:1182-1197.

Arnold, M. L. (1997). *Natural Hybridization and Evolution.* Oxford University Press.

Arnold, M. L. and Meyer, A. (1997). Natural hybridization in primates: One evolutionary mechanism. *Zoology*, **109**:261-276.

Bandelt, H.-J., Forster, P., Sykes, B. C., Richards, M. B. (1995). Mitochondrial portraits of human population using median networks. *Genetics*, **141**:743-753.

Baroni, M., Semple, C., and Steel, M. (2004). A framework for representing reticulate evolution. *Annals of Combinatorics*, **8**:391-408.

Baroni, M., Grünewald, S., Moulton, V., Semple, C. (2005). Bounding the number of hybridization events for a consistent evolutionary history. *Mathematical Biology*, **51**:171-182.

Baroni, M., Semple, C., Steel, M. (2006). Hybrids in real time. *Systematic Biology*, **55**:46-56.

Beiko, R. and Hamilton, N. (2006). Phylogenetic identification of lateral genetic transfer events. *BMC Evolutionary Biology*, **6**:15.

Bergthorsson, U., Adams, K. L., Thomason, B., Palmer, J. D. (2003). Widespread horizontal transfer of mitochondrial genes in flowering plants. *Nature*, **424**:197-201.

Bonet, M. K., St. John, K., Mahindru, R., Amenta, N. (2006). Approximating subtree distances between phylogenies. *Journal of Computational Biology*, **13**:1419-1434.

Bordewich, M. and Semple, C. (2004). On the computational complexity of the rooted subtree prune and regraft distance. *Annals of Combinatorics*, **8**:409-423.

Bordewich, M. and Semple, C. (2007a). Computing the minimum number of hybridization events for a consistent evolutionary history. *Discrete Applied Mathematics*, **155**:914-928.

Bordewich, M. and Semple, C. (2007b). Computing the hybridization number of two phylogenetic trees is fixed-parameter tractable. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **4**:458-466.

Bordewich, M., Linz, S., St. John, K., Semple, C. (2007c) A reduction algorithm for computing the hybridization number of two trees. *Evolutionary Bioinformatics*, **3**:86-98.

Bordewich, M., McCartin, C., Semple, C. (2008). A 3-approximation algorithm for the subtree distance between phylogenies. *Journal of Discrete Algorithms*, in press.

Boucher, Y., Douady, C. J., Papke, R. T., Walsh, D. A., Boudreau, M. E., Nesbø, C. L., Case, R. J., Doolittle, W. F. (2003). Lateral gene transfer and the origins of prokaryotic groups. *Annual Review of Genetics*, **37**:283-328.

Bryant, D. and Moulton, V. (2002). An agglomerative method for the construction of planar phylogenetic networks. In *Proceedings of the Workshop on Algorithms in Bioinformatics (WABI 2002)*, Lecture Notes in Computer Science, Vol. 2454, pp. 375-391.

Bushman, F. (2002). *Lateral DNA Transfer: Mechanisms and Consequences.* Cold Spring Harbor Laboratory Press.

Choi, I.-G. and Kim, S.-H. (2007). Global extent of horizontal gene transfer. *Proceedings of the National Academy of Science of the United States of America*, **104**:4489-4494.

Cortes-Ortiz, L., Duda, T. F. Jr, Canales-Espinosa, D., Garcia-Orduna, F., Rodriguez-Luna, E., Bermingham, E. (2007). Hybridization in large-bodied New World primates. *Genetics*, **176**:2421-2425.

Dagan, T. and Martin, W. (2007). Ancestral genome sizes specify the minimum rate of lateral gene tranfer during prokaryote evolution. *Proceedings of the National Academy of Science of the United States of America*, **104**:870-875.

Dayhoff, M. O., Schwartz, R. M., Orcutt, B. C. (1978). In *Atlas of Protein Sequence and Structure.* Washington DC: National Biomedical Research Foundation. Vol. 5, pp. 345-352.

de la Cruz, F. and Davies, J. (2000). Horizontal gene transfer and the origin of species: Lessons from bacteria. *Trends in Microbiology*, **8**:128-133.

Diruggiero, J., Dunn, D., Maeder, D. L., Holley-Shanks, R., Chatard, J., Horlacher, R., Robb, F. T., Boos, W., Weiss, R. B. (2000). Evidence of recent lateral gene transfer among hyperthermophilic archaea. *Molecular Microbiology*, **38**:684-693.

Doolittle, W. F. (1999). Phylogenetic classification and the universal tree. *Science*, **284**:2124-2129.

Downey, R. and Fellows, M. (1998). *Parameterized Complexity (Monographs in Computer Science).* Springer Verlag.

Dress, A. W. M. and Huson, D. H. (2004). Constructing splits graphs. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, **1**:109-115.

Eisen J. A. (2000). Horizontal gene transfer among microbial genomes: New insights from complete genome analysis. *Current Opinion in Genetics and Development*, **10**:606-611.

Ellstrand, N. C., Whitkus, R., and Rieseberg, L. H. (1996). Distribution of spontaneous plant hybrids. *Proceedings of the National Academy of Science of the United States of America*, **93**:5090-5093.

Fehrer, J., Gemeinholzer, B.,Chrtek Jr, J., Bräutigam, S. (2007). Incongruent plastid and nuclear DNA phylogenies reveal ancient intergeneric hybridization in *Pilosella* hawkweeds (*Hieracium*, Cichorieae, Asteraceae). *Molecular Phylogenetics and Evolution*, **42**:347-361.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, **17**:368-376.

Felsenstein, J. (1988). Phylogenies from molecular sequences: Inference and reliability. *Annual Review of Genetics*, **22**:521-565.

Felsenstein, J. (1989). PHYLIP - Phylogeny Inference Package (Version 3.2). *Cladistics*, **5**:164-166.

Felsenstein, J. (2004). *Inferring Phylogenies.* Sinauer Associates.

Fitch, W. M. (1971). Toward defining the course of evolution: Minimum change for a specific tree topology. *Systematic Zoology*, **20**:406-416.

Garcia-Vallve, S., Romeu, A., Palau, J. (2000). Horizontal gene transfer in bacterial and archaeal complete genomes. *Genome Research*, **10**:1719-1725.

Gascuel, O. (ed.) (2005). *Mathematics of Evolution and Phylogeny.* Oxford University Press.

Ge, F., Wang, L. S., Kim, J. (2005). The cobweb of life revealed by genome-scale estimates of horizontal gene transfer. *PLoS Biology*, **3**:e316.

Gogarten, J. P., Doolittle, W. F., Lawrence, J. G. (2002). Prokaryotic evolution in light of gene transfer. *Molecular Biology and Evolution*, **19**:2226-2238.

Gogarten, J. P. (2003). Gene transfer: Gene swapping craze reaches eukaryotes. *Current Biology*, **13**:R53-R54.

Grant, P. R., Grant, B. R. (1992). Hybridization of bird species. *Science*, **256**:193-197.

Grass Phylogeny Working Group (2001). Phylogeny and subfamilial classification of the grasses (*Poaceae*). *Annals of the Missouri Botanical Garden* , **88**:373-457.

Gusfield, D. and Bansal, V. (2005). A fundamental decomposition theory for phylogenetic networks and incompatible characters. In *Proceedings of the Ninth Annual International Conference on Research in Computational Molecular Biology (RECOMB 2005)*, Lecture Notes in Bioinformatics, Vol. 3500, pp. 217-232.

Hallett, M. and Lagergren, J. (2001). Efficient algorithms for lateral gene transfer problems. In *Proceedings of the Fifth Annual International Conference on Research in Computational Molecular Biology (RECOMB 2001)*, ACM Press, pp. 149-156.

Hao, W. and Golding, G. B. (2006). The fate of laterally tranferred genes: Life in the fast lane to adaptiation or death. *Genome Research*, **16**:636-643.

Harrison, R. G. (1993). Hybrids and hybrid zones: Historical perspectives. In *Hybrid zones and the evolutionary process*. Oxford University Press.

Hein, J., Jing, T., Wang, L., Zhang, K. (1996). On the complexity of comparing evolutionary trees. *Discrete Applied Mathematics*, **71**:153-169.

Hubbs, C. L. (1955), Hybridization between fish species in nature. *Systematic Zoology*, **4**:1-20.

Huelsenbeck, J. P., Rannala, B., Larget, B. (2000). A Bayesian framework for the analysis of cospeciation. *Evolution*, **54**:352-364.

Huson, D. H., Klöpper, T., Lockhart, P. J., Steel, M. A., (2005). Reconstruction of reticulate networks from gene trees. In *Proceedings of the Ninth Annual International Conference on Research in Computational Molecular Biology (RECOMB 2005)*, Lecture Notes in Bioinformatics, Vol. 3500, pp. 233-249.

Huson, D. H. (2007). Split networks and reticulate networks. In *New Mathematical Models for Evolution*. Oxford University Press, Chapter 9, pp. 247-276.

Katoh, K., Kuma, K., Toh, H., Miyata, T. (2005). MAFFT version 5: Improvement in accuracy of multiple sequence alignment. *Nucleic Acids Research*, **33**:511-518.

Kurland, C. G., Canback, B., Berg, O. G. (2003). Horizontal gene transfer: A critical view. *Proceedings of the National Academy of Science of the United States of America*, **100**:9658-9662.

Lawrence, J. G. and Ochman, H. (1997). Amelioration of bacterial genomes: Rates of change and exchange. *Journal of Molecular Evolution*, **44**:383-397.

Lerat, E., Daubin, V., Moran, N. A. (2003). From gene trees to organismal phylogeny in prokaryotes: The case of the γ-Proteobacteria. *PLoS Biology*, **1**:e19.

Lerat, E., Daubin, V., Ochman, H., Moran, N. A. (2005). Evolutionary origins of genomic repertoires in bacteria. *PLoS Biology*, **3**:e130.

Linz, S., Radtke, A., von Haeseler, A. (2007). A likelihood framework to measure horizontal gene transfer. *Molecular Biology and Evolution*, **24**:1312-1319.

Lockhart, P. J. (2007). Private communication.

MacLeod, D., Charlebois, R. L., Doolittle, W. F., Bapteste, E. (2005). Deduction of probable events of lateral gene transfer through comparison of phylogenetic trees by recursive consolidation and rearrangement. *BMC Evolutionary Biology*, **5**:27.

Maddison, W. P. (1989). Reconstructing character evolution on polytomous cladograms. *Cladistics*, **5**:365-377.

Maddison, W. P. (1997). Gene trees in species trees. *Systematic Biology*, **46**:523-536.

Madigan, M. T., Martinko, J. M., Brock, T. D. (2005). *Brock Biology of Microorganisms and Student Companion.* Prentice Hall International.

Maiden, M. C. J. (1998). Horizontal genetic exchange, evolution, and spread of antibiotic resistance in bacteria. *Clinical Infectious Diseases*, **27**:S12-20.

Mallet, J. (2005). Hybridization as an invasion of the genome. *Trends in Ecology and Evolution*, **20**:229-237.

Mallet, J. (2007). Hybrid speciation. *Nature*, **446**:279-283.

Mayr, E. (1992). A local flora and the biological species concept. *American Journal of Botany*, **79**:222-238.

McBreen, K. and Lockhart, P. J. (2006). Reconstructing reticulate evolutionary histories of plants. *Trends in Plant Science*, **11**:398-404.

Nakamura, Y., Itoh, T., Matsuda, H., Gojobori, T. (2004). Biased biological functions of horizontally transferred genes in prokaryotic genomes. *Nature Genetics*, **36**:760-766.

Nakhleh, L., Ruths, D., and Wang, L. S. (2005a). RIATA-HGT: A fast and accurate heuristic for reconstructing horizontal gene transfer. In *Proceedings of the Eleventh International Computing and Combinatorics Conference (COCOON 2005)*, Lecture Notes in Computer Science, Vol. 3595, pp. 84-93.

Nakhleh, L., Warnow, T., Linder, C. R., St. John, K. (2005b). Reconstructing reticulate evolution in species—theory and practice. *Journal of Computational Biology*, **12**:796-811.

Nei, M. (1996). Phylogenetic analysis in molecular evolutionary genetics. *Annual Review of Genetics*, **30**:371-403.

Nelson, K. E., Clayton, R. A., Gill, S. R., et al. (29 co-authors). (1999). Evidence for lateral gene transfer between archaea and bacteria from genome sequence of *Thermotoga maritima*. *Nature*, **399**:323-329.

Nesbø, C. L., Boucher, Y., Doolittle, W. F. (2001). Defining the core of nontransferable prokaryotic genes: The euryarchaeal core. *Journal of Molecular Evolution*, **53**:340-350.

Niedermeier, R. and Rossmanith, P. (2000). A general method to speed up fixed-parameter-tractable algorithms. *Information Processing Letters*, **73**:125-129.

Ochman, H., Lawrence, J. G., Groisman, E. A. (2000). Lateral gene transfer and the nature of bacterial innovation. *Nature*, **405**:299-304.

Olsen, G. J., Matsuda, H., Hagstrom, R., Overbeek, R. (1994). fastDNAmL: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Computer Applications in the Biosciences* , **10**:41-48.

Pamilo, P. and Nei, M. (1988). Relationships between gene trees and species trees. *Molecular Biology and Evolution*, **5**:568-583.

Paun, O., Lehnebach, C., Johansson, J. T., Lockhart, P., Hörandl, E. (2005). Phylogenetic relationships and biogeography of *Ranunculus* and allied genera (Ranunculaceae) in the Mediterranean region and in the European Alpine System. *Taxon*, **54**:911-930.

Pruitt, K. D., Tatusova, T., Maglott, D. R. (2005). NCBI Reference Sequence (RefSeq): A curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Research*, **33**:D501-504.

Rambaut, A. and Grassly, N. C. (1997). Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Computer Applications in the Biosciences*, **13**:235-238.

Rannala, B. and Yang, Z. (1996). Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference. *Journal of Molecular Evolution*, **43**:304-311.

Rieseberg, L. H., Raymond, O., Rosenthal, D. M., Lai, Z., Livingstone, K., Nakazato, T., Durphy, J. L., Schwarzbach, A. E., Donovan, L. A., Lexer, C. (2003). Major ecological transitions in wild sunflowers facilitated by hybridization. *Science*, **301**:1211-1216.

Rubtsov, P. M., Musakhanov, M. M., Zakhaiyev, V. M., Krayev, A. S., Skryabin, K. G., Bayev, A. A. (1980). The structure of the yeast ribosomal RNA genes (part I). The complete nudeotide sequence of the 18 S ribosomal RNA gene from *Saccharomyces cerevisiae*. *Nucleic Acids Research*, **8**:5779-5794.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, **4**:406-425.

Salzberg, S. L., White, O., Peterson, J., Eisen, J. A. (2001) Microbial genes in the human genome: Lateral transfer or gene loss? *Science*, **292**:1903-1906.

Schmidt, H. A., Strimmer, K., Vingron, M., von Haeseler, A. (2002). TREE-PUZZLE: Maximum likelihood phylogenetic analysis using quartets and parallel computing. *Bioinformatics*, **18**:502-504.

Schmidt, H. A. (2003). Phylogenetic trees from large datasets. PhD thesis, Heinrich-Heine-Universität, Düsseldorf.

Seehausen, O. (2004). Hybridization and adaptive radiation. *Trends in Ecology and Evolution*, **19**:198-207.

Semple, C. and Steel, M. (2003). *Phylogenetics*. Oxford University Press.

Semple, C. (2007). Hybridization networks. In *New Mathematical Models for Evolution*. Oxford University Press, Chapter 10, pp. 277-314.

Sneath, P. H. A. and Sokal, R. R. (1973). *Numerical Taxonomy*. Freeman, San Francisco, CA.

Snel, B., Bork, P., Huynen, M. A. (2002). Genomes in flux: The evolution of archaeal and proteobacterial gene content. *Genome Research*, **12**:17-25.

Sokal, R. R. and Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, **38**:1409-1438.

Song, Y. and Hein, J. (2003). Parsimonious reconstruction of sequence evolution and haplotype blocks: finding the minimum number of recombination events. In *Proceedings of the Workshop on Algorithms in Bioinformatics (WABI 2003)* Lecture Notes in Bioinformatics, Vol. 2812, pp. 287-302.

Song, Y. and Hein, J. (2005). Constructing minimal ancestral recombination graphs. *Journal of Computational Biology*, **12**:147-169.

Suchard, M. A. (2005). Stochastic models for horizontal gene transfer: Taking a random walk through tree space. *Genetics*, **170**:419-431.

Syvanen, M. (1994). Horizontal gene transfer: Evidence and possible consequences. *Annual Review of Genetics*, **28**:237-261.

Tatusov, R. L., Natale, D. A., Garkavtsev, I. V., Tatusova, T. A., Shankavaram, U. T., Rao, B. S., Kiryutin, B., Galperin, M. Y., Fedorova, N. D., Koonin, E. V. (2001). The COG database: New developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Research*, **29**:22-28.

Than, C. and Nakhleh, L. (2008). SPR-based tree reconciliation: Non-binary trees and multiple solutions. In *Proceedings of the Sixth Asia Pacific Bioinformatics Conference (APBC2008)*, in press.

Venter, J. C., Adams, M. D., Sutton, G. G., Kerlavage, A. R., Smith, H. O., Hunkapiller, M. (1998). Shotgun sequencing of the human genome. *Science*, **280**:1540-1542.

Woese, C. R. and Fox, G. E. (1977). Phylogenetic structure of the prokaryotic domain: The primary kingdoms. *Proceedings of the National Academy of Science of the United States of America*, **74**:5088-5090.

Woese, C. R. (2000). Interpreting the universal phylogenetic tree. *Proceedings of the National Academy of Science of the United States of America*, **97**:8392-8396.

Zwickl, D. J. and Hillis, D. M. (2002). Increased taxon sampling greatly reduces phylogenetic error. *Systematic Biology*, **51**:588-598.

Appendix

A.1 Pseudocode of HybridNumber

Here, we present the pseudocode of HybridNumber. For a rooted binary phylogenetic X-tree $\mathcal{T}$ and a subset A of X, we denote the minimal subtree of $\mathcal{T}$ connecting the elements in A by $\mathcal{T}(A)$. Furthermore, we denote the tree formed by replacing a cluster A with the new leaf c by $\mathcal{T}[A \rightarrow c]$. If B is a subset of X, we use $\mathcal{T}[-B]$ to denote the phylogenetic tree obtained from $\mathcal{T}$ by deleting each of the elements in B and suppressing any resulting vertex of degree two apart from the root. Finally, $\mathcal{F}(\mathcal{T}, E)$ denotes the forest obtained from the tree $\mathcal{T}$ by deleting the edges in the set E. Due to the chain reduction, the input to HybridNumber includes a weight function w on pairs of taxa; this can be taken to be zero for all pairs in the initial input.

```
Algorithm A.1.1: HybridNumber($\mathcal{S}, \mathcal{T}, w$)

($\mathcal{S}, \mathcal{T}, w$) ← SubtreeReduction($\mathcal{S}, \mathcal{T}, w$)
($\mathcal{S}, \mathcal{T}, w$) ← ChainReduction($\mathcal{S}, \mathcal{T}, w$)
if ∃ a minimal common cluster $C$ of $\mathcal{S}$ and $\mathcal{T}$ and
$1 < |C| <$ number of taxa of $\mathcal{S}$
  do { ($\mathcal{S}_1, \mathcal{T}_1, w_1, \mathcal{S}_2, \mathcal{T}_2, w_2$) ← ClusterReduction($\mathcal{S}, \mathcal{T}, w$)
       $h_1$ ← ExhaustiveSearch($\mathcal{S}_1, \mathcal{T}_1, w_1$)
       $h_2$ ← HybridNumber($\mathcal{S}_2, \mathcal{T}_2, w_2$)
       $h \leftarrow h_1 + h_2$
  else
  do $h$ ← ExhaustiveSearch($\mathcal{S}, \mathcal{T}, w$)
return ($h$)
```

```
Algorithm A.1.2: SubtreeReduction($\mathcal{S}, \mathcal{T}, w$)

$A \leftarrow$ maximal common subtree of $\mathcal{S}$ and $\mathcal{T}$
if $|A| > 1$
  do { $\mathcal{S}' \leftarrow \mathcal{S}[A \to a]$
       $\mathcal{T}' \leftarrow \mathcal{T}[A \to a]$
       $w' \leftarrow w$ restricted to pairs of taxa not in $A$
       $(\mathcal{S}, \mathcal{T}, w) \leftarrow$ SubtreeReduction($\mathcal{S}', \mathcal{T}', w'$)
return $(\mathcal{S}, \mathcal{T}, w)$
```

```
Algorithm A.1.3: ChainReduction($\mathcal{S}, \mathcal{T}, w$)

$(a_1, \ldots, a_m) \leftarrow$ maximal common chain of $\mathcal{S}$ and $\mathcal{T}$
if $m \geq 3$
  do { $weight \leftarrow \sum_{i=1}^{m-1} w(a_i, a_{i+1})$
       $w(a, b) \leftarrow weight + (m - 2)$
       $\mathcal{S}' \leftarrow \mathcal{S}[\{a_1\} \to a, \{a_2\} \to b, -\{a_3, \ldots, a_m\}]$
       $\mathcal{T}' \leftarrow \mathcal{T}[\{a_1\} \to a, \{a_2\} \to b, -\{a_3, \ldots, a_m\}]$
       $w' \leftarrow \{w(a, b)\} \cup w$ restricted to pairs not in $\{a_1, \ldots, a_m\}$
       $(\mathcal{S}, \mathcal{T}, w) \leftarrow$ ChainReduction($\mathcal{S}', \mathcal{T}', w'$)
return $(\mathcal{S}, \mathcal{T}, w)$
```

```
Algorithm A.1.4: ClusterReduction($\mathcal{S}, \mathcal{T}, w$)

$C \leftarrow$ minimal common cluster of $\mathcal{S}$ and $\mathcal{T}$
$\mathcal{S}_1 \leftarrow \mathcal{S}(C)$
$\mathcal{S}_2 \leftarrow \mathcal{S}[C \to c]$
$\mathcal{T}_1 \leftarrow \mathcal{T}(C)$
$\mathcal{T}_2 \leftarrow \mathcal{T}[C \to c]$
$w_1 \leftarrow w$ restricted to pairs of taxa in $C$
$w_2 \leftarrow w$ restricted to pairs of taxa not in $C$
return $(\mathcal{S}_1, \mathcal{T}_1, w_1, \mathcal{S}_2, \mathcal{T}_2, w_2)$
```

Algorithm A.1.5: ExhaustiveSearch($\mathcal{S}, \mathcal{T}, w$)

```
if $\mathcal{S} \cong \mathcal{T}$   return $(0)$
$h \leftarrow$ number of leaves of $\mathcal{S}$
$i \leftarrow 0$
repeat
  for each $E$ a subset of the edges of $\mathcal{S}$ such that $|E| = i$
    do
      $\mathcal{F} \leftarrow \mathcal{F}(\mathcal{S}, E)$
      if $\mathcal{F}$ is a legitimate-agreement forest of $\mathcal{S}$ and $\mathcal{T}$
        do
          $P \leftarrow \{(a,b) : a, b$ are isolated taxa in $\mathcal{F}\}$
          $h' \leftarrow i + \sum_{(a,b) \in P} w(a,b)$
          if $h' < h$
            do $h \leftarrow h'$
  $i \leftarrow i + 1$
until $i \geq h$
return $(h)$
```

Remarks

(ii) The actual implemented algorithms contain various small improvements compared to the pseudocode in order to improve running time. While these changes (described in Section 3.2.2) do not affect the theoretical 'worst case' running time, in practice they are beneficial.

(ii) In HybridNumber, following a call to the cluster reduction, the cluster-tree pair $\mathcal{S}_1$ and $\mathcal{T}_1$ cannot be reduced any further using the reductions, in which case we immediately call ExhaustiveSearch. However, it may now be possible to further reduce the cluster-reduced tree pair $\mathcal{S}_2$ and $\mathcal{T}_2$ and so we call HybridNumber.

(iii) In ExhaustiveSearch, if we have found a forest of weight h formed by deleting fewer than h edges, we must run until we have checked all possible forests resulting from the deletion of up to h edges in case there exists one of lower weight. This check is a consequence of the way in which the chain reduction works.

A.2 Species Trees

The following two figures represent the species trees that were used to estimate an HGT rate for the COG data set containing protein sequences for 44 species. The first such tree (Figure A.2.1) was reconstructed from the 780 protein families of the COG data set, whereas the second species tree (Figure A.2.2) was calculated by using the 16/18 S rDNA sequences of the same 44 species.

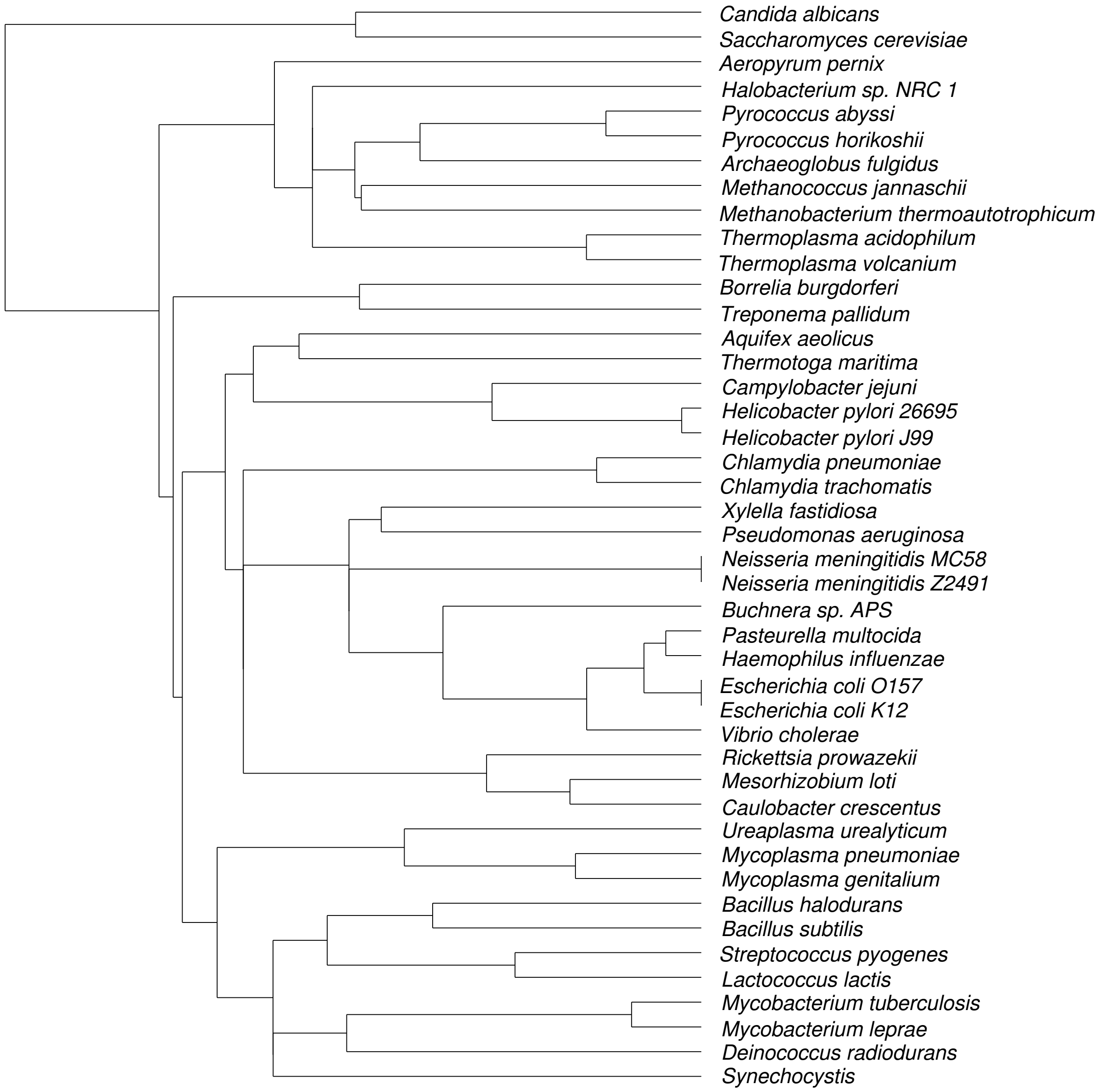


Figure A.2.1: Species tree reconstructed for the COG data set.

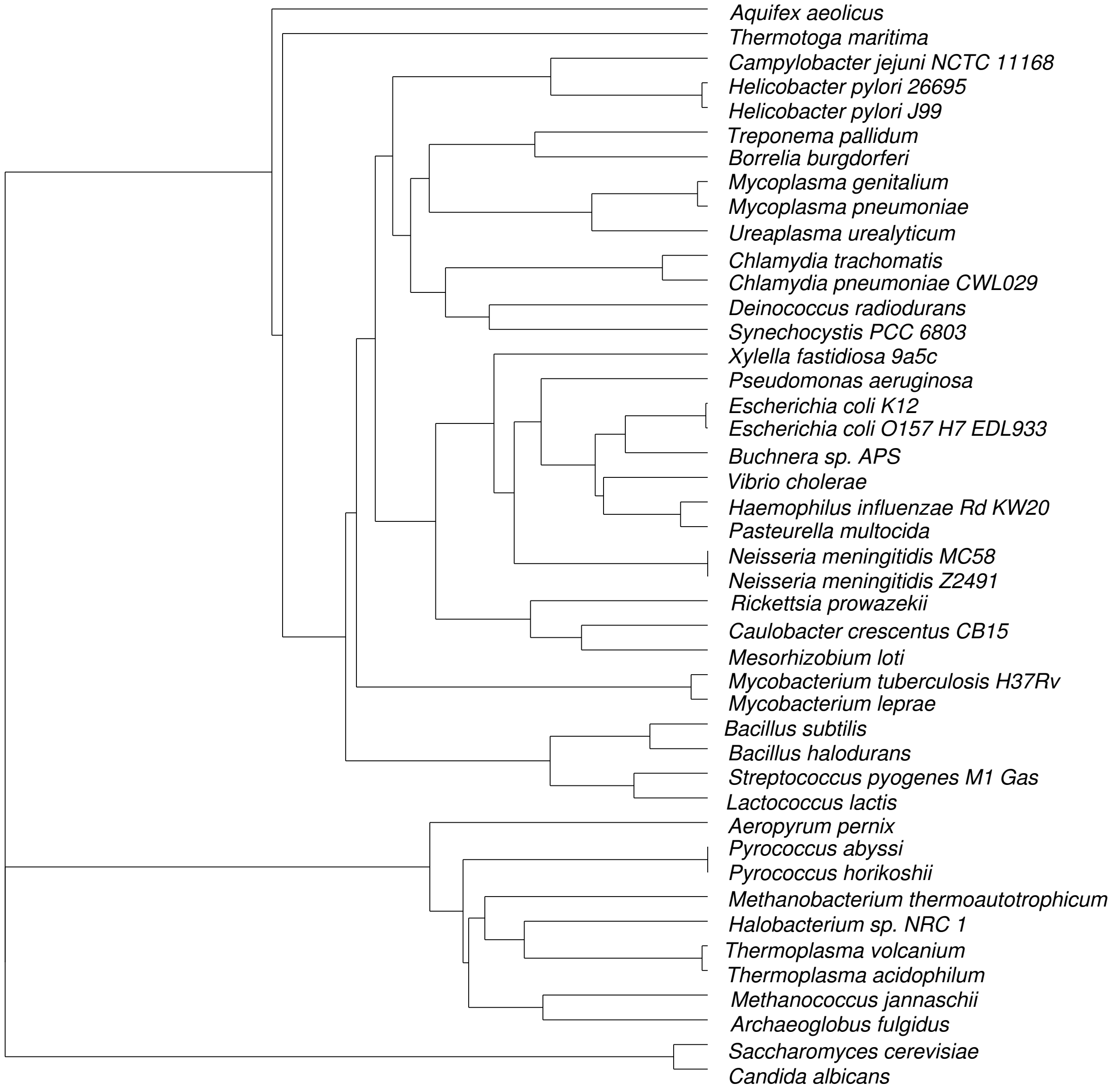


Figure A.2.2: Species tree reconstructed for the 16/18 S rDNA sequences.