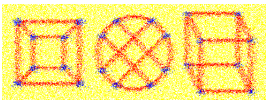
Phylogenetic incongruence through the lens of Monadic Second Order logic

Steven Kelk, Leo van Iersel, Celine Scornavacca, Mathias Weller


Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 20, no. 2, pp. 189–215 (2016)
DOI: 10.7155/jgaa.00390

Phylogenetic incongruence through the lens of Monadic Second Order logic

Steven Kelk[1] *Leo van Iersel*[2]
Celine Scornavacca[3] *Mathias Weller*[4]

[1]Department of Knowledge Engineering (DKE), Maastricht University, P.O. Box 616, 6200 MD Maastricht, The Netherlands
[2]Delft Institute of Applied Mathematics, Delft University of Technology, P.O. Box 5, 2600 AA Delft, The Netherlands
[3]Institut des Sciences de l'Evolution (Université de Montpellier, CNRS, IRD, EPHE), Place E. Bataillon CC 064 - 34095 Montpellier Cedex 5, France
[4]Institut de Biologie Computationnelle (IBC), Laboratory of Informatics, Robotics, and Microelectronics of Montpellier (LIRMM), Université de Montpellier II, 161 rue Ada 34392 Montpellier Cedex 5, France

Submitted:	Reviewed:	Revised:	Accepted:	Final:
August 2015	December 2015	January 2016	February 2016	February 2016

Published:
February 2016

Article type:
Regular paper

Communicated by:
G. Liotta

E-mail addresses: steven.kelk@maastrichtuniversity.nl (Steven Kelk) l.j.j.v.iersel@gmail.com (Leo van Iersel) celine.scornavacca@univ-montp2.fr (Celine Scornavacca) mathias.weller@lirmm.fr (Mathias Weller)

Abstract

Within the field of phylogenetics there is growing interest in measures for summarising the dissimilarity, or *incongruence*, of two or more phylogenetic trees. Many of these measures are NP-hard to compute and this has stimulated a considerable volume of research into fixed parameter tractable algorithms. In this article we use *Monadic Second Order* logic (MSOL) to give alternative, compact proofs of fixed parameter tractability for several well-known incongruence measures. In doing so we wish to demonstrate the considerable potential of MSOL - machinery still largely unknown outside the algorithmic graph theory community - within phylogenetics, introducing a number of "phylogenetics MSOL primitives'' which will hopefully be of use to other researchers. A crucial component of this work is the observation that many incongruence measures, when bounded, imply the existence of an *agreement forest* of bounded size, which in turn implies that an auxiliary graph structure, the *display graph*, has bounded treewidth. It is this bound on treewidth that makes the machinery of MSOL available for proving fixed parameter tractability. Due to the fact that all our formulations are of constant length, and are articulated in the restricted variant of MSOL known as MSO_1, we actually obtain the stronger result that all these incongruence measures are fixed parameter tractable purely in the treewidth (in fact, if an appropriate decomposition is given: the cliquewidth) of the display graph. To highlight the potential importance of this, we re-analyse a well-known dataset and show that the treewidth of the display graph grows more slowly than the main incongruence measures analysed in this article[1].

1 Introduction

The central goal of phylogenetics is to accurately infer the evolutionary history of a set of species (or *taxa*) X from incomplete information. Classically, phylogenetic reconstruction has access to information about each element in X, such as DNA data, and seeks to infer a phylogenetic tree - a tree whose leaves are bijectively labeled by X - that best fits this data. There is a vast literature available on this topic and many different algorithms exist for constructing phylogenetic trees [20, 32]. In practice, it is not uncommon for phylogenetic analysis to generate multiple phylogenetic trees as output. This can occur for various reasons, ranging from software engineering choices (many tree-building packages are designed to generate multiple optimal and near-optimal solutions) to more structural explanations (*reticulate* evolutionary signals that are comprised of multiple distinct tree signals). Given two (or more) distinct phylogenetic trees, it is natural to compare them to determine whether the difference is significant. This explains the interest of the phylogenetics community for measures that can quantify the dissimilarity, or *incongruence*, of phylogenetic trees [25].

[1] A preliminary version of this article can be found at [29]. There all MSOL formulations were in MSO_2 rather than MSO_1 and some formulations did not have constant length, leading to weaker results than presented here. Also, [29] did not contain any experiments.

Some of these measures (such as Tree Bisection and Reconnection distance [1]) are studied to better understand how local-search heuristics, based on rearrangement operations, navigate the space of phylogenetic trees (e.g., [10]). Others, such as Hybridization Number [9], are studied because they assist with the inference of phylogenetic networks, which generalise phylogenetic trees to directed acyclic graphs [25, 26].

Unfortunately, many of these measures are NP-hard and APX-hard to compute. On the positive side, however, the phylogenetics community has been quite successful in proving that these measures are fixed parameter tractable (FPT) when parameterized by the measure itself. Informally, this means that a measure that evaluates to k can be computed in time $f(k) \cdot poly(n)$ where f is some function that only depends on k and n is the size of the instance (often taken to be $|X|$). Such running times have the potential to be much faster than running times of the form $O(n^{f(k)})$ when the measure in question is comparatively small (see e.g. [19] for more background on FPT). A number of state-of-the-art phylogenetics software packages are based on FPT algorithms, such as the software used in [34]. Most FPT results in the phylogenetics literature are based on classical proof techniques such as kernelization and bounded-search.

Parallel to all of this, algorithmic graph theorists have made great steps forward in identifying sufficient, structural conditions under which NP-hard problems on graphs become (fixed parameter) tractable. At the heart of this research lies the width parameter, the most famous example being *treewidth*. Informally, treewidth is a measure that quantifies the dissimilarity of a graph from being a tree. The notion of treewidth, which is most famously associated with the celebrated Graph Minors project of Robertson and Seymour [30], has had a profound impact upon algorithm design. A great many NP-hard problems turn out to become tractable on graphs of bounded treewidth, using broadly similar proof techniques i.e. dynamic programming on tree decompositions [4]. This contributed to the rise of meta-theorems, the archetypal example being *Courcelle's Theorem* [16, 2]. This states, when combined with the result from [5], that any graph property that can be abstractly formulated as a length ℓ sentence of *Monadic Second Order* logic (MSOL), can be tested in time $f(t, \ell) \cdot O(n)$ on graphs of treewidth t, where n is the number of vertices in the graph. When t and ℓ are both bounded by a function of a single parameter p, this yields a running time of the form $f(p) \cdot O(n)$ i.e. linear-time fixed parameter tractability in parameter p. This is an extremely powerful technique in the sense that it completely abstracts away from ad-hoc algorithm design and permits highly compact, "declarative" proofs that a problem is FPT. Courcelle's Theorem (and its variants) are more than two decades old, but their potential is rarely exploited by the phylogenetics community. One exception is the literature on *unrooted compatibility*, which asks whether a set of unrooted phylogenetic trees are compatible, i.e. whether there exists an unrooted tree that contains all input trees as minors. The FPT proof by Bryant and Lagergren [11] proves that the *display graph* (the graph obtained by identifying all taxa with the same label) has bounded treewidth (in the number of input trees), and then gives an MSOL formulation which tests compatibility. A follow-up result by the

present authors applies a similar approach [31].

In this article we show that this technique has much broader potential within phylogenetics. To clarify the exposition we focus on binary trees (both rooted and unrooted) on the same set of taxa X. We begin by proving that if two trees have an *agreement forest* of size k – essentially a partition of the trees into k non-overlapping isomorphic subtrees – the treewidth of the display graph is bounded by a function of k. This simple observation is significant because of the prominent role of agreement forests within the phylogenetics literature. We use this insight to re-analyse three well-known NP-hard phylogenetics problems that were previously shown to be FPT using more conventional analysis. In particular, we give MSOL formulations for (1) Unrooted Maximum Agreement Forest (uMAF), which is equivalent to the problem of computing Tree Bisection and Reconnection distance (TBR) on unrooted trees, (2) rooted Maximum Agreement Forest (rMAF), which is equivalent to the problem of computing Rooted Subtree Prune and Regraft distance (rSPR) on rooted trees, and (3) Hybridization Number (HN) on rooted trees. The formulations for uMAF and rMAF are based on explicitly modelling agreement forests using quartets, triplets and edge cuts. The formulation for HN builds on the rMAF formulation by constraining the agreement forest to be "acyclic", thus leveraging the well-known link between optimal solutions to HN and optimal solutions to the Maximum Acyclic Agreement Forest (MAAF) problem. Finally we consider the (4) Maximum Parsimony Distance on Binary Characters problem. This fairly new distance, d^2_{MP} for short, asks for a binary character f on X that maximizes the absolute difference between the parsimony score (to be defined later) of f on the two trees. It is NP-hard but not known to be FPT (in the distance itself). Here we give an MSOL formulation which shows that the problem is FPT in parameter uMAF. Although this does not settle whether the problem is FPT in the distance itself, it does demonstrate a number of interesting principles. Firstly, it demonstrates the power of "simulating" the execution of polynomial-time algorithms (in this case, Fitch's algorithm [22]) within MSOL. Secondly, any subsequent proof that TBR distance is at most a bounded distance above d^2_{MP} distance and/or that d^2_{MP} distance induces bounded treewidth display graphs, will automatically prove that d^2_{MP} distance is FPT in the distance itself.

All our MSOL formulations are of constant length and are articulated in the more restricted variant of MSOL known as MSO_1, which only allows quantification over (sets of) vertices, contrasting with MSO_2 which also allows quantification over (sets of) edges. The use of constant length formulations (rather than formulations whose length grows as a function of the incongruence measure we are calculating: the HN formulation in [29] grew in length in this way) means that all incongruence measures considered in this article are fixed parameter tractable purely in the treewidth of the display graph (in fact: purely in the cliquewidth of the display graph, assuming an appropriate decomposition is given; this is the significance of MSO_1). This is a stronger result than proving fixed parameter tractability in the incongruence measure itself, because the treewidth of the display graph could potentially be much smaller than the in-

congruence measure. Indeed, in the last section of this article we re-analyse the well known *Poaceae* dataset and show that for these pairs of trees the treewidth of the display graph grows more slowly than the main incongruence measures considered in this article. For a number of incongruence measures not considered in this article, such as Unrooted Subtree Prune and Regraft distance (uSPR) and Nearest Neighbour Interchange (NNI) distance, this effect will also be observed, because TBR is a lower bound on both uSPR and NNI.

Summarizing, our formulations show the potential for MSOL to generate compact, logical FPT proofs for phylogenetics problems. MSOL does not yield practical algorithms but it is an excellent classification tool. Once the existence of FPT algorithms has been confirmed via MSOL one can then switch efforts to finding a *good* FPT algorithm by more direct analysis, possibly (but not exclusively) through direct analysis of tree decompositions. Our experiments on the *Poaceae* dataset suggest that this could be a very fruitful direction for future research.

2 Preliminaries

In this section, we define the main objects that will be manipulated in this paper.

An *unrooted phylogenetic tree* T (unrooted tree for short) is a tree in which no vertex has degree 2 and in which the leaves are bijectively labeled by a label set $\mathcal{L}(T)$. The leaf labels are often called *taxa* and the symbol X is frequently used as shorthand for $\mathcal{L}(T)$. Internal vertices are not labeled. A *rooted* phylogenetic tree (rooted tree for short) is defined similarly, except that it has exactly one vertex, called the *root* of the tree, that is permitted to have degree 2, and edges are directed away from the root. An unrooted tree is *binary* if every internal vertex has degree 3, and a rooted tree is binary if each internal vertex has indegree 1 and outdegree 2, and the root has outdegree 2 and indegree 0.

Given an unrooted tree T and a subset $Y \subseteq \mathcal{L}(T)$, we use $T(Y)$ to denote the minimal subtree of T connecting Y. Moreover, we denote by $T|_Y$ the tree obtained from $T(Y)$ when suppressing vertices of degree 2. We say that $T|_Y$ is the subtree of T *induced* by Y. In graph theory terms, $T|_Y$ is a label-preserving topological minor of T. Induced subtrees are defined in the same way for rooted trees, except that the root of $T|_Y$ becomes the vertex in the minimal connecting subgraph that is closest to the root of T, and we suppress all degree-2 vertices except the new root. We write $T - Y$ to denote $T|_{\mathcal{L}(\mathcal{T})-Y}$.

Given a label set X, a *bipartition* (or *split*) $A|B$ on X is a partition of X into two non-empty sets. Each edge $\{u, v\}$ of a tree T induces a split $\mathcal{L}(T_u)|\mathcal{L}(T_v)$, where T_u and T_v are the two trees obtained from T when $\{u, v\}$ is deleted.

Given an unrooted binary tree T and a set of four distinct labels $\{u, v, w, y\}$ in $\mathcal{L}(T)$, $T|_{\{u,v,w,y\}}$ will be exactly one of the three possible unrooted binary trees on $\{u, v, w, y\}$. These are called *quartets* and are denoted respectively by $uv|wy$, $uw|vy$ and $wv|uy$, depending on the bipartition induced by the central edge. In Figure 1(a) we see $uv|wy$ and $uw|vy$. Given a rooted binary tree T

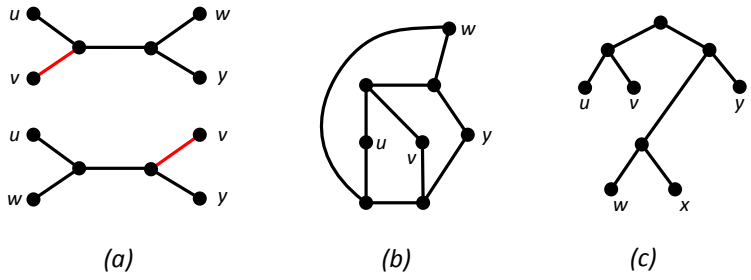


Figure 1: *(a) Two unrooted binary phylogenetic trees on* $\{u, v, w, y\}$*. A maximum agreement forest (uMAF) for these two trees contains 2 components, and can be obtained by cutting the single red edge in both trees and then suppressing the resulting degree-2 vertices. (b) The display graph for the two trees from (a), obtained by identifying leaves with the same label. (c) A rooted binary phylogenetic tree on* $\{u, v, w, x, y\}$*.*

and a set of three labels $\{u, v, w\}$ in $\mathcal{L}(T)$, $T|_{\{u,v,w\}}$ will be exactly one of the three possible rooted binary trees on $\{u, v, w\}$. These are called *triplets* and are denoted respectively by $uv|w$, $uw|v$ and $wv|u$, where $ij|k$ means that the leaf labelled k is incident to the root of the three-leaves tree. For example, if T is the tree shown in Figure 1(c), $T|_{\{x,v,y\}}$ is the triplet $xy|v$.

Let $\mathcal{T} = \{T_1, T_2, \ldots, T_k\}$ be a collection of unrooted trees, not necessarily on the same set of taxa. The *display graph* of $\mathcal{T}$ is obtained from the disjoint graph union of all trees in $\mathcal{T}$ by identifying vertices with the same label; see Figure 1(b) and also Figures 2 and 3.

Given an undirected graph $G = (V, E)$, a *bag* is simply a subset of V. A *tree decomposition* of G consists of a tree $T_G = (V(T_G), E(T_G))$ where $V(T_G)$ is a collection of bags such that the following holds: (1) every vertex of V is in at least one bag, (2) for each edge $\{u, v\} \in E$, there exists some bag that contains both u and v, and (3) for each vertex $u \in V$, the bags that contain u induce a connected subtree of T_G. The *width* of a tree decomposition is equal to the cardinality of its largest bag, minus 1. The *treewidth* of a graph G is equal to the minimum width, ranging over all possible tree decompositions of G. A tree with at least one edge has treewidth 1. For a fixed value of k one can determine in linear time whether a graph has treewidth at most k [5].

Similarly to treewidth measuring the distance of a graph to being a tree, the *cliquewidth* measures the distance of a graph to being a (disjoint union of) clique(s). While the precise definition of cliquewidth is somewhat complex, it is also not relevant to this paper. Suffice it to say that the cliquewidth cw is a stronger parameter than the treewidth tw, that is, there is an (exponential) function f such that $cw(G) \leq f(tw(G))$ for each graph G [18, 15]. However, the converse is false, as cliques have cliquewidth-1 and arbitrary treewidth.

A monadic second order logic formula (MSO_1 formula for short) over a class

of graphs is, loosely speaking, a formula quantifying only over vertices or sets of vertices. If, additionally, quantification over edges and sets of edges is allowed, we say the formula is MSO_2. Courcelle [16] showed that problems expressible in MSO_1 can be solved in linear time on graphs of bounded cliquewidth and problems expressible in MSO_2 can be solved in linear time on graphs of bounded treewidth. However, these results have two shortcomings: first, the implied algorithms are not practical, as the dependence on the treewidth/cliquewidth grows superexponentially and, second, these formulations do not capture problems asking for a solution of certain size. The second issue was resolved for MSO_2 (and treewidth) by Arnborg et al. [2], who showed that the above holds for an extension (called EMS) of MSO_2 that allows, for a given graph G and formula φ, finding a system of sets $(X_1, X_2, \ldots)$ that, among all sets with $G \models \varphi(X_1, X_2, \ldots)$, optimizes a linear function in the sizes $|X_1|, |X_2|, \ldots$, where the binary relation $\models$ is used to denote that the structure on the left of the symbol satisfies the set of sentences on the right of it. Courcelle et al. [17] showed the same for an analogous extension (called LinEMSOL) of MSO_1 (for cliquewidth). More specifically, let $\mathfrak{S} := (V, E, P_1, P_2, \ldots)$ such that V is the vertex set of a graph G, the binary predicate E is true for (u, v) if and only if $\{u, v\}$ is an edge of G, and P_i are *unary* predicates on V (that is, vertex sets), let φ be a formula with free variables $X_1, X_2, \ldots$ using only the predicates $E, P_1, P_2, \ldots$, and let $f_1, f_2, \ldots$ be a function family. The LinEMSOL optimization problem

$$\min\{\sum_i f_i(X_i) \mid \mathfrak{S} \models \varphi(X_1, X_2, \ldots)\}$$

can be solved in linear time on any class of graphs of bounded cliquewidth.

3 Main results

Unless stated otherwise, we assume that $T_1 = (V^1, E^1)$ and $T_2 = (V^2, E^2)$ are both unrooted binary trees on X. Note that the display graph of T_1 and T_2 has $3|X| - 4$ vertices and $4|X| - 6$ edges if $|X| > 2$.

In the subsections that follow we will prove the following results. We begin with a theorem that links treewidth to agreement forests (which will be defined in the next section).

Theorem 1 *Let T_1, T_2 be two unrooted binary trees on the same set of taxa X such that an agreement forest of size k for these two trees exists. Then, the treewidth of their display graph D is at most $k + 1$.*

We then move on to the main FPT results:

Theorem 2 *The incongruence measures uMAF, TBR, rMAF, rSPR, MAAF, HN and d^2_{MP} can all be computed for two binary trees T_1 and T_2 on X in time $O(f(tw) \cdot |X|)$ where tw is the treewidth of the display graph of T_1 and T_2 and f is some computable function that depends only on tw.*

The above theorem will be established by giving, for each measure, a constant-length MSO_1 formulation, and observing that the size of the display graph is linear in $|X|$. Combining this with the result of Bodlaender [5] and Courcelle et al. [17] completes the proof. In fact, by using MSO_1, we obtain the following, more general, theorem. Recall that bounded treewidth implies bounded cliquewidth, but the converse does not hold [15].

Theorem 3 *The incongruence measures uMAF, TBR, rMAF, rSPR, MAAF, HN and d^2_{MP} can all be computed for two binary trees T_1 and T_2 on X in time $O(W + f(cw) \cdot |X|)$ where cw is the cliquewidth of the display graph of T_1 and T_2, W is the time required to compute a clique-width decomposition of value cw of the display graph, and f is some computable function that depends only on cw.*

We have included the W term because, unlike treewidth, it is not known whether computation of cliquewidth itself is FPT. Note that the LinEMSOL machinery is constructive, i.e. it constructs the algorithm that returns an optimal solution [17, Theorem 4].

To establish fixed-parameter tractability in the respective natural parameters, we combine Theorem 2 with a proof that, for each measure k, the treewidth of the display graph is bounded by a function of k. The foundation of all such proofs is Theorem 1.

Theorem 4 *The incongruence measures uMAF, TBR, rMAF, rSPR, MAAF and HN can all be computed for two binary trees T_1 and T_2 on X in time $O(f(k) \cdot |X|)$ where k is the measure itself and f is some computable function that depends only on k.*

Theorem 4 does not hold for d^2_{MP} because at the present time we do not know whether the treewidth of the display graph can be bounded by a function of d^2_{MP}. Instead we obtain the following weaker result.

Theorem 5 *The incongruence measure d^2_{MP} can be computed for two binary trees T_1 and T_2 on X in time $O(f(d_{TBR}) \cdot |X|)$ where d_{TBR} is the TBR distance of T_1 and T_2 and f is some computable function that depends only on d_{TBR}.*

3.1 TBR / MAF on unrooted trees, and a treewidth bound

We will start by giving the definitions of a TBR move and of the TBR distance between two unrooted binary trees.

Definition 1 (TBR move) *Given an unrooted binary tree T, a* tree bisection and reconnection *(TBR) move on T consists of removing an edge of T, say $\{u, v\}$, and then reconnecting the subtrees T_u and T_v obtained from T when $\{u, v\}$ is deleted as follows: subdividing an edge of T_u with a new vertex p; subdividing an edge of T_v with a new vertex q; connecting p to q; and finally suppressing any vertices of degree 2.*

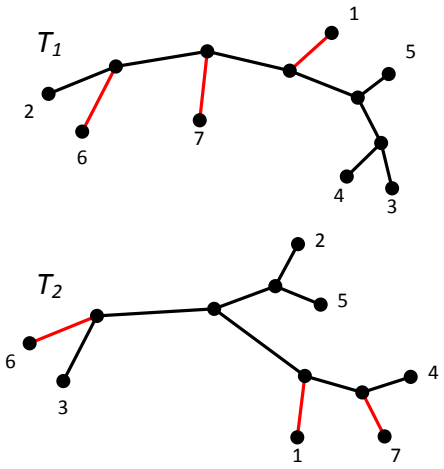


Figure 2: *Two unrooted trees T_1, T_2 on taxa $X = \{1,2,3,4,5,6,7\}$. A uMAF $\{\{2,3,4,5\},\{1\},\{6\},\{7\}\}$ can be obtained by deleting the 3 edges marked red in each tree. Since a uMAF of T_1 and T_2 contains 4 components, the TBR distance between T_1 and T_2 is 3.*

TBR distance is then defined naturally as follows:

Problem: $d_{TBR}(T_1, T_2)$
Input: Two unrooted binary trees T_1, T_2 on the same set of taxa X.
Output: The minimum number of TBR moves required to transform T_1 into T_2.

We will now give the definition of an uMAF for two unrooted binary trees T_1, T_2 on X. Any collection of trees whose label sets partition X is said to be a *forest on* X. Furthermore, we say that a set $\mathcal{F} = \{F_1, \ldots, F_k\}$ of unrooted binary phylogenetic trees – with $|\mathcal{F}|$ referred to as the *size* of $\mathcal{F}$ – is a *forest of* T if $\mathcal{F}$ can be obtained from T by deleting a $(k-1)$-sized subset E of $E(T)$, suppressing any unlabeled leaves, and then finally suppressing any vertices with degree 2. To ease reading, we write $\mathcal{F} = T - E$ if $\mathcal{F}$ can be obtained in this way.

Definition 2 (uMAF) *A set $\mathcal{F}$ of unrooted trees is an agreement forest for T_1 and T_2 (denoted uAF) if $\mathcal{F}$ is a forest of both T_1 and T_2. An unrooted* maximum *agreement forest (uMAF), is an uAF of minimum size.*

So, the uMAF problem is defined as follows:

Problem: $uMAF(T_1, T_2)$
Input: Two unrooted binary trees T_1, T_2 on the same set of taxa X.
Output: An uMAF for T_1 and T_2.

The two problems defined above are closely related, and known to be NP-hard [1].

Theorem 6 ([1]) *Given two unrooted binary trees T_1, T_2 on the same set of taxa X, we have that $d_{TBR}(T_1, T_2) = |uMAF(T_1, T_2)| - 1$.*

Fortunately, they have been proved to be FPT in their natural parameterizations [1], and fast algorithms have been recently proposed [14, 33]. In this section, we will give a more compact proof of their fixed parameter tractability using the trees in Figure 2 as an example; their display graph is shown in Figure 3. We begin with a bound on the treewidth of the display graph.

Theorem 1. *Let T_1, T_2 be two unrooted binary trees on the same set of taxa X such that a uAF of size k for these two trees exists. Then, the treewidth of their display graph D is at most $k + 1$.*

Proof: From [23], we know that the display graph of two identical trees has treewidth 2 (or 1 in the case that both trees consist of a single vertex). Thus, if we have an uAF $\mathcal{F} = \{F_1, \ldots, F_k\}$ of size k, this means that the display graph D_0 of $\mathcal{F}$ (which we define as the display graph constructed from two disjoint copies of $\mathcal{F}$) has k connected components, and treewidth at most 2. This is because the treewidth of a disconnected graph is equal to the largest treewidth ranging over its connected components. Now, we can construct a tree decomposition of D from the tree decomposition of $\mathcal{F}$ as follows: suppose $\mathcal{F}$ can be obtained by removing from T_1, respectively T_2, a subset of edges K^1, respectively K^2, and suppressing vertices with degree 2 and unlabeled leaves. First, note that we can reintroduce the suppressed vertices (and their corresponding edges) in $\mathcal{F}$, obtaining a new forest $\mathcal{F}'$, without changing the treewidth. Indeed, given an edge $\{u, v\}$ in $\mathcal{F}$ that corresponded to a path $(u, x_1, \cdots, x_j, v)$ before the suppression of the vertices with degree 2, we know that there exists a bag B in the tree decomposition of D_0 such that u and v are in B. Then we can add a set of bags $\{B_1, \cdots, B_j\}$ such that $B_1 = \{u, x_1, v\}$, $B_2 = \{x_1, x_2, v\}$, $\cdots$, $B_j = \{x_{j-1}, x_j, v\}$, and add edges $\{B, B_1\}$, $\{B_1, B_2\}$, $\cdots$, $\{B_{j-1}, B_j\}$ to the tree decomposition. For the suppressed unlabeled leaves, say u, this is even easier: we add a bag $\{u, v\}$ as child of any of the bags containing v, where v is the vertex from which the suppressed leaf was hanging. It is easy to see that this is a tree decomposition of the display graph of $\mathcal{F}'$ with treewidth 2. Now, we can easily reintroduce the $k - 1$ edges in K^1 to the display graph, again without changing the treewidth, by, for each edge $\{u, v\}$ in K^1, adding a bag $\{u, v\}$ between two existing bags, one containing u and the other containing v. Note that the obtained decomposition is still a tree, since we are connecting two components of $\mathcal{F}'$. Now, when adding back the edges of K^2, this is not true anymore. In this case, there exists at least a path in the tree decomposition, connecting a bag containing u to a bag containing v. Then, taking the shortest of these paths and adding u to its bags not containing u, we increase the treewidth by at most 1. If we do this for all edges in K^2, we obtain a tree decomposition

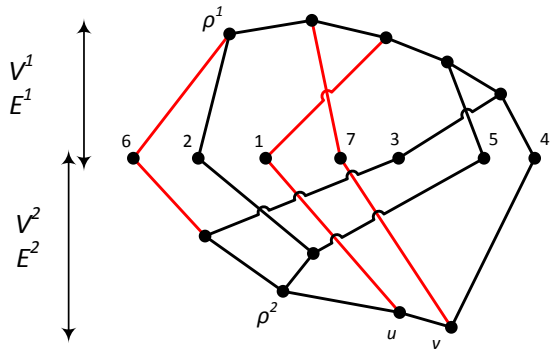


Figure 3: *The display graph $D = (V, E)$ obtained from the trees shown in Figure 2. Note that $V = V^1 \cup V^2$, $E = E^1 \cup E^2$ and $X = V^1 \cap V^2$. In this unrooted context ρ^1 and ρ^2 do not represent roots in the traditional phylogenetic sense: they are arbitrarily selected vertices which simply allow us to use vertices to encode edges within the logical formulation. For example, the edge $\{u, v\}$ can be represented by vertex v because it is the first edge on the unique path (inside T_2) from v to ρ^2. To obtain the uMAF described in Figure 2 the red edges need to be deleted, which (with respect to this choice of ρ^1 and ρ^2) are represented as $K^1 = K^2 = \{1, 6, 7\}$.*

for the display graph of T_1 and T_2 with treewidth at most $2 + (k - 1) = k + 1$. Note that this bound is tight, as the following example shows: an uMAF of two quartets with different topologies, $uv|wx$ and $ux|vw$ say, contains 2 components, and the display graph of these two quartets has treewidth 3 (see also [23]). □

The high-level idea of the MSOL formulation for uMAF is that an uAF $\mathcal{F}$ with k components can be represented by edge sets $K^1 \subseteq E^1$ and $K^2 \subseteq E^2$ such that $|K^1| = |K^2| = k - 1$ and $\mathcal{F}_1 = T_1 - K^1 = \mathcal{F}_2 = T_2 - K^2$. To ensure that edges we delete correctly model an agreement forest we require that (1) the two forests $\mathcal{F}_1$ and $\mathcal{F}_2$ induce an identical partition of X and (2) the components of the two induced forests must have the same topology. To enforce (1) we observe that (in, say, T_1) two taxa x_1 and x_2 are in the same component of the forest resulting from deletion of K^1 if and only if they can still reach each other inside T_1 after deletion of those edges. In turn, this occurs if and only if there is a path from x_1 to x_2 entirely contained inside T_1 which avoids all the edges in K^1. To enforce (2) we demand that a quartet is in the first forest (i.e. the quartet is contained inside one of the trees in the forest) if and only the quartet is in the second forest. This uses the fact that two unrooted binary trees on the same set of taxa are topologically identical if and only if they induce identical sets of quartets [13]. The main technicality stems from the fact in MSO_1 we do not have the vertex-edge incidence relation available and we cannot quantify over subsets of edges. However, as we shall see it is not too difficult to overcome this problem.

In the following, $\forall_{x\in Y}\phi(x)$ is short for $\forall_x(x \in Y) \rightarrow \phi(x)$ and $\forall_{Y\subseteq Z}\phi(Y)$ for $\forall_Y(\forall_{x\in Y} x \in Z) \rightarrow \phi(Y)$. Also, $x \in Y \cup Z$ is short for $x \in Y \vee x \in Z$.

Let $T_1 = (V^1, E^1)$ and $T_2 = (V^2, E^2)$ be the two input trees and $D = (V, E)$ their display graph. We fix two arbitrary vertices $\rho^1 \in V^1$ and $\rho^2 \in V^2$. By arbitrarily "rooting" the trees in this way we can unambiguously use vertices to represent edges. In particular, we will model K^1 not as a set of edges but as a set of vertices v representing the deletion of the first edge on the unique v-ρ^1-path in T_1 (we call these edges "edges of K^1 with respect to ρ^1"). We define K^2 equally for T_2. The structure over which we optimize is $\mathfrak{S}_{\text{uAF}} := (V, E, \rho^1, \rho^2, V^1, V^2, X)$ where $V = V^1 \cup V^2$, $E = E^1 \cup E^2$ and $X = V^1 \cap V^2$ is used to distinguish the taxa from the inner vertices of the display graph. (See Figure 3 for clarification of the terms used in the structure.) Note that we cannot distinguish E^1 from E^2 directly (since only *vertex*-subsets are allowed), but we use $\mathrm{E}^i(x, y) \equiv E(x, y) \wedge x \in V^i \wedge y \in V^i$ with $i \in \{1, 2\}$ instead. Then, we use the following primitives (only stated for T_1 here, but completely analogous for T_2):

Z is connected in T_1:

$$\text{connected}^1(Z) \equiv \forall_{Y\subseteq Z}\left(\forall_{y\in Y}\forall_{z\in Z}\,\mathrm{E}^1(y, z) \rightarrow z \in Y\right) \rightarrow (Y = Z)$$

the unique x-y-path in T_1 contains z:

$$\text{in\_path}^1(x, y, z) \equiv \forall_{Y\subseteq V^1}\left(x \in Y \wedge y \in Y \wedge \text{connected}^1(Y)\right) \rightarrow (z \in Y)$$

the quartet in T_1 corresponding to the four distinct taxa $x_1, \ldots, x_4$ has the topology "x_1 is closer to x_2 than to x_3 and x_4" $\left(\begin{smallmatrix} x_1 & & x_3 \\ & \diagdown\,\diagup & \\ x_2 & & x_4 \end{smallmatrix}\right)$:

$$\text{Quartet}^1(x_1, \ldots, x_4) \equiv \exists_{v\in V^1}(\neg\,\text{in\_path}^1(x_1, x_2, v) \wedge \bigwedge_{i=3,4} \text{in\_path}^1(x_1, x_i, v)) \wedge \bigwedge_{1\leq i<j\leq 4} x_i \neq x_j$$

z is the "LCA" of x and y wrt. ρ^1 in T_1:

$$\text{LCA}^1(x, y, z) \equiv \text{in\_path}^1(\rho^1, x, z) \wedge \text{in\_path}^1(\rho^1, y, z) \wedge \text{in\_path}^1(x, y, z)$$

x and y are connected by a path in T_1 that avoids deleted edges of K^1 wrt. ρ^1:

$$\text{PAC}^1(x, y, K^1) \equiv \forall_{z\in K^1}\left(\text{in\_path}^1(x, y, z) \rightarrow \text{LCA}^1(x, y, z)\right)$$

Deleting the edges of K^1 wrt. ρ^1 in T_1 and the edges of K^2 wrt. ρ^2 in T_2 yields an agreement forest:

$$\begin{aligned}\text{uAF}(K^1, K^2) \equiv\ & \left(\forall_{x,y\in X}\,\text{PAC}^1(x, y, K^1) \leftrightarrow \text{PAC}^2(x, y, K^2)\right) \wedge \\ & \forall_{x_1,\ldots,x_4\in X}\Big(\bigwedge_{i,j\leq 4} \text{PAC}^1(x_i, x_j, K^1)\Big) \rightarrow \\ & (\text{Quartet}^1(x_1, \ldots, x_4) \leftrightarrow \text{Quartet}^2(x_1, \ldots, x_4))\end{aligned}$$

Before completing the formulation, we need to show that the result does not depend on our choice of ρ^1 and ρ^2:

Lemma 1 *Let ρ^1, ρ^2, K^1, and K^2 be such that $\mathfrak{S}_{uAF} \models \mathrm{uAF}(K^1, K^2)$. Then, for each $r^1 \in V^1$, there is a set K' with $|K'| = |K^1|$ and $\mathfrak{S}'_{uAF} := (V, E, r^1, \rho^2, V^1, V^2, X)$ is a model for $\mathrm{uAF}(K', K^2)$.*

Proof: Let K^1_E denote the set of edges of K^1 wrt. ρ^1. Then, we define

$$K' := \{u \mid uv \in K^1_E \wedge \text{the unique } u\text{-}r^1\text{-path in } T_1 \text{ contains } v\}.$$

and note that K^1_E is also the set of edges of K' wrt. r^1. We show for all x and y that $\mathfrak{S}'_{\mathrm{uAF}}$ is a model for $\mathrm{PAC}^1(x, y, K')$ if and only if $\mathfrak{S}_{\mathrm{uAF}}$ is a model for $\mathrm{PAC}^1(x, y, K^1)$. Let p denote the unique x-y-path in T_1.

"$\Rightarrow$": Assume that $\mathfrak{S}_{\mathrm{uAF}}$ is not a model for $\mathrm{PAC}^1(x, y, K^1)$ and let z denote a vertex of K^1 on p that is not the LCA of x and y with respect to ρ^1. Then, p contains both z and its parent z' with respect to ρ^1. If z' is also the parent of z with respect to r^1, then $z \in K'$ and, since p contains both z and z', we know that z is also not the LCA of x and y with respect to r^1. This contradicts $\mathfrak{S}'_{\mathrm{uAF}}$ being a model for $\mathrm{PAC}^1(x, y, K')$. Otherwise, z is the parent of z' with respect to r^1. Then, $z' \in K'$ and, since its parent is in p, we know that z' is not the LCA of x and y with respect to r^1, leading to the same contradiction.

"$\Leftarrow$": This direction of the proof is completely analogous to the other direction.

□

Finally, the LinEMSOL-formulation for uMAF is "minimize $|K^1|+|K^2|$ such that $\mathfrak{S}_{\mathrm{uAF}} \models \mathrm{uAF}(K^1, K^2)$". To see that this objective function is correct, we sketch that, for each $k \in \mathbb{N}$, there are K^1, K^2 with $\mathfrak{S}_{\mathrm{uAF}} \models \mathrm{uAF}(K^1, K^2)$ and $|K^1| + |K^2| \leq k$ if and only if there is an agreement forest with at most $k/2 + 1$ components. For the "$\Rightarrow$"-direction, let K^1, K^2 be such that $\mathfrak{S}_{\mathrm{uAF}} \models \mathrm{uAF}(K^1, K^2)$ and $|K^1|+|K^2|$ is minimum among all such pairs. By minimality and the formulation of primitive uAF, the forest $T_i - K^i$ has *exactly* $|K^i| + 1$ connected components, and each such component contains *at least* one taxon ($i \in \{1, 2\}$). It follows that $|K^1| = |K^2|$ (because the two forests induce the same reachability relation between taxa) and the connected components in the forests induce identical sets of quartet topologies. In other words: K^1 and K^2 together induce a valid agreement forest with $\frac{k}{2} + 1$ components. For the other direction, suppose an agreement forest $\mathcal{F}$ with $k/2+1$ components exists. Then, let K^1 be the set of roots in T_1 wrt. ρ^1 of the components of $\mathcal{F}$ that do not contain ρ^1 (and K^2 analogously). Then $|K^1| + |K^2| = k$ and it can be verified that $\mathrm{uAF}(K^1, K^2)$ evaluates to true.

To establish Theorem 4 for uMAF/TBR it remains to prove that the display graph has treewidth bounded by a function of uMAF/TBR. This is immediate from Theorem 1.

3.2 rSPR / MAF on rooted trees

In this section, we will give analogous results for the computation of rSPR distance. Before that, we need to introduce some definitions.

Definition 3 (rSPR move) *Given a rooted binary tree T, a* subtree prune and regraft *(rSPR) move on T consists of removing an edge of T, say (u, v), yielding two trees T_u and T_v, and then reconnecting them as follows: subdividing some edge of T_u with a new vertex p; adding an edge directed from p to v, and then suppressing any vertices with indegree and outdegree both equal to 1.*

rSPR distance is defined analogously to TBR distance, and a rMAF for two rooted binary trees T_1, T_2 is defined similarly to an uMAF, but in a rooted framework. We refer to [7] for precise definitions. The main difference is that a forest consists of *rooted* binary trees and this has to be taken into account when comparing the topology of the components. In the rooted context, agreement forests are mainly studied because of their close relationship to rSPR distance. To accurately model rSPR distance it is necessary to slightly modify each input tree T_i as follows: we add a vertex with special label ρ at the end of a pendant edge adjoined to the original root of T_i, see Figure 4. We then consider ρ to be part of the label set of the tree. In other words, we explicitly assume $\rho \in X$. Note that the addition of ρ means that we can equivalently view each T_i as an unrooted binary tree, with ρ acting as a placeholder for the root location, and this is how the trees will be modelled in the display graph (see Figure 5).

The close relationship between rMAF and rSPR distance is summarized by the following well-known result.

Theorem 7 ([7]) *Given two rooted binary trees T_1, T_2 on the same set of taxa X, and assuming that an extra taxon ρ has been appended to the roots of the two trees, we have that $d_{rSPR}(T_1, T_2) = |rMAF(T_1, T_2)| - 1$.*

Note that these problems have been proved NP-hard and FPT in their natural parameterizations [7].

The MSO_1 formulation for rMAF is very similar to the uMAF formulation.The structure over which we optimize is $\mathfrak{S}_{\mathrm{AF}} := (V, E, \rho^1, \rho^2, V^1, V^2, X)$. In the uMAF formulation, ρ^1 and ρ^2 were selected arbitrarily to induce an orientation on the edges, and thus to allow us to unambiguously represent edges with vertices. They also have this technical function here (because we remain within MSO_1) but now we additionally want them to indicate the true location of the tree roots. For this reason we assign $\rho := \rho^1 = \rho^2$. For simplicity we abbreviate $\forall_{x \in X}(x \neq \rho \rightarrow \phi(x))$ to $\forall_{x \in X - \rho}\phi(x)$.

Indeed, the main difference with the TBR formulation is that we need primitives for triplets instead of quartets, because we are working in the rooted environment and two rooted binary trees are topologically equivalent if and only if they contain the same set of triplets [12]. Fortunately we can use the fact that triplet $xy|z$ is in T_i $(x, y, z \in X)$ if and only if quartet $xy|\rho z$ is in the unrooted interpretation of T_i. This yields the primitive (which as usual can be analogously defined for T_2):

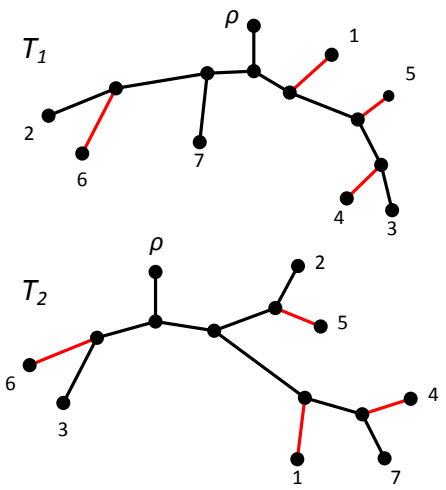


Figure 4: *Two rooted trees T_1, T_2 obtained by rooting the trees shown in Figure 2. The rooting is achieved by adding an explicit taxon ρ, so $X = \{\rho, 1, 2, 3, 4, 5, 6, 7\}$. A rMAF $\{\{2, 3, 7\}, \{1\}, \{4\}, \{5\}, \{6\}\}$ can be obtained by deleting the 4 edges marked red in each tree. Since a rMAF of T_1 and T_2 contains 5 components, the rSPR distance between T_1 and T_2 is 4. Moreover, the above-given forest is acyclic, so it is also a MAAF, and hence the hybridization number of T_1 and T_2 is also 4.*

the triplet in T_1 corresponding to the three distinct taxa x_1, x_2, x_3 has topology "x_1 is closer to x_2 than to x_3" $\left(\begin{smallmatrix} & \wedge & \\ x_1 & x_2 & x_3 \end{smallmatrix} \right)$:

$$\text{Triplet}^1(x_1, x_2, x_3) \equiv \text{Quartet}^1(x_1, x_2, x_3, \rho^1)$$

Subsequently, only the final part of the TBR formulation changes:

Deleting the edges of K^1 wrt. ρ^1 in T_1 and the edges of K^2 wrt. ρ^2 in T_2 yields a rooted agreement forest:

$$\begin{aligned}\text{rAF}(K^1, K^2) \equiv \big(\forall_{x,y \in X}\, \text{PAC}^1(x, y, K^1) \leftrightarrow \text{PAC}^2(x, y, K^2)\big) \wedge \\ \forall_{x_1,x_2,x_3 \in X-\rho}\Big(\bigwedge_{i,j \leq 3} \text{PAC}^1(x_i, x_j, K^1)\Big) \rightarrow \big(\text{Triplet}^1(x_1, x_2, x_3) \leftrightarrow \\ \text{Triplet}^2(x_1, x_2, x_3)\big)\end{aligned}$$

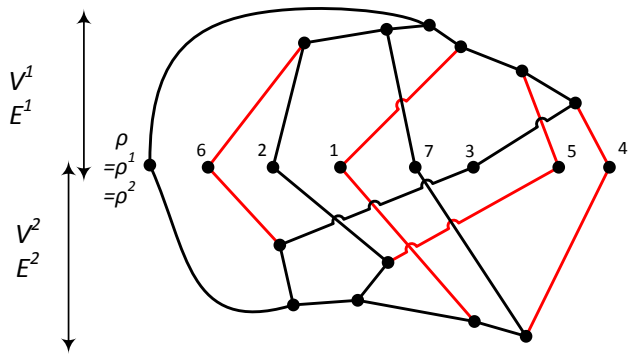


Figure 5: *The display graph obtained from the rooted trees shown in Figure 4. This is used for the rMAF/rSPR and MAAF/HN formulations. To obtain the forest described in Figure 4 it is necessary to delete the red edges, so here* $K^1 = K^2 = \{6, 1, 5, 4\}$.

The LinEMSOL-formulation for rSPR is "minimize $|K^1| + |K^2|$ such that $\mathfrak{S}_{\text{AF}} \models \text{rAF}(K^1, K^2)$". To establish Theorem 4 for rMAF/rSPR, observe that an agreement forest of two rooted trees T_1 and T_2 induces an agreement forest of the same size of the *unrooted* interpretations of these trees, simply by ignoring the orientation of edges. Hence the treewidth bound described in Theorem 1 is still applicable, and Theorem 4 follows.

3.3 Hybridization Number

In this section, we deal again with rooted trees A *rooted phylogenetic network* (rooted network for short) $N = (V(N), E(N))$ on a set of taxa X is any rooted acyclic digraph in which no vertex has degree 2 (except possibly the root) and whose leaves are bijectively labeled by elements of X. The *hybridization number* of N, denoted by $h(N)$, is defined as

$$h(N) = \sum_{v \in V(N): \delta^-(v) > 0} (\delta^-(v) - 1) = |E(N)| - |V(N)| + 1$$

where $\delta^-(v)$ denotes the indegree of v.

Given a rooted network N on X and a rooted binary tree T on X', with $X' \subseteq X$, we say that T is *displayed* by N if T can be obtained from N by deleting a subset of its edges and any resulting degree-0 vertices, and then suppressing vertices with $\delta^-(v) = \delta^+(v) = 1$.

We are now ready to define the hybridization number problem:

Problem: $HN(T_1, T_2)$
Input: Two rooted binary trees T_1, T_2 on the same set of taxa X.

Output: A rooted network N displaying T_1 and T_2 such that $h(N)$ is minimum over all rooted networks with this property.

The hybridization number for T_1 and T_2, denoted by $h(T_1, T_2)$, is defined as the hybridization number of this minimum network. As done for TBR and rSPR, we can give a characterization of the hybridization number in terms of agreement forests. To do so, we need to define *acyclic* agreement forests.

Let $\mathcal{F} = \{F_1, F_2, \ldots, F_k\}$ be an agreement forest for two rooted binary trees T_1 and T_2 on the same set of taxa X, and let $AG(T_1, T_2, \mathcal{F})$ be the directed graph whose vertex set is $\mathcal{F}$ and for which (F_i, F_j) is an arc iff $i \neq j$, and either

(1) the root of $T_1(\mathcal{L}(F_i))$ is an ancestor of the root of $T_1(\mathcal{L}(F_j))$ in T_1, or

(2) the root of $T_2(\mathcal{L}(F_i))$ is an ancestor of the root of $T_2(\mathcal{L}(F_j))$ in T_2.

We call $\mathcal{F}$ an *acyclic agreement forest* (AAF) for T_1 and T_2 if $AG(T_1, T_2, \mathcal{F})$ does not contain any directed cycle. A *maximum* acyclic agreement forest (MAAF), is an AAF of minimum size.

The acyclicity condition is used to model the fact that species cannot inherit genetic material from their own offspring. The two problems defined above are closely related, as the following well-known result shows.

Theorem 8 ([3]) *Given two rooted binary trees T_1, T_2 on the same set of taxa X, we have that $h(T_1, T_2) = |MAAF(T_1, T_2)| - 1$.*

The above equivalence formed the basis for results proving that both problems are NP-hard [9] and fixed parameter tractable [8].

For the MSOL formulation, we extend the formulation for rMAF and use the same structure $\mathfrak{S}_{\mathrm{AF}}$. In particular, we introduce[2] ρ and assume again that $\rho \in X$, $\rho^1 = \rho^2 = \rho$. (Hence, Figures 4 and 5 also apply to HN.)

We base the formulation on the one for rMAF. As such, we model the deletion of an arc uv of T_1 by including the vertex v in K^1 and, thus, K^1 can be thought of as the roots of the trees of the target agreement forest in T_1 (likewise for K^2 and T_2). The main difficulty is then to impose acyclicity on $AG(T_1, T_2, \mathcal{F})$, which is only implicitly defined via a directed reachability relation that can "switch" repeatedly between the input trees. To model this, we use an MSOL primitive "corr" that allows us to identify the roots in T_1 and T_2 that correspond to a same tree in $\mathcal{F}$, and combine it with the strict ancestor relations for T_1 and T_2. We can then enforce the existence of a DAG ordering of $K^1 \cup K^2$ with respect to this combined relation.

Unfortunately, modelling the edge-cuts leading to an agreement forest as the vertices to which they are incoming does not give us a handle on the "highest" component of the agreement forest (which contains the root). To this end, we artificially force $\rho^1 \in K^1$ and $\rho^2 \in K^2$, mimicking an edge-deletion incoming to

[2]Strictly speaking, ρ is not necessary for MAAF/HN but it helps with some technicalities concerning the LCA of the "highest" agreement forest component in each tree, so we keep it to clarify the exposition.

the root of the respective tree — this is related to the fact that an agreement forest with k components can be obtained from only $k-1$ edge cuts. Hence in this formulation ρ^1 and ρ^2 do not only function as indicators of root location, they also function as the heads of dummy edge cuts.

x is a strict ancestor of y in T_1 or T_2:

$$x < y \equiv \bigvee_{i=1,2} \text{in\_path}^i(\rho^i, y, x) \wedge (x \neq y)$$

the vertices in $K^1 \cup K^2$ do not induce any "empty" forest components i.e. components without taxa:

$$\text{no\_empty}(K^1, K^2) \equiv \bigwedge_{i=1,2} \forall_{x \in K^i} \exists_{z \in X} \text{PAC}^i(x, z, K^i)$$

vertices $x, y \in K^1 \cup K^2$ are corresponding (i.e. "define" the same component of the forest):

$$\begin{aligned}\text{corr}(x, y, K^1, K^2) \equiv &(\forall_{z \in X} \text{PAC}^i(x, z, K^i) \leftrightarrow \text{PAC}^{3-i}(y, z, K^{3-i})) \wedge \\ &(\bigvee_{i=1,2} (x \in K^i \wedge y \in K^{3-i}))\end{aligned}$$

vertex $x \in K^1 \cup K^2$ or its corresponding vertex (z) is a strict ancestor of y:

$$x <_{K^1, K^2} y \equiv (x < y) \vee \forall_{z \in K^1 \cup K^2} (\text{corr}(x, z, K^1, K^2) \rightarrow z < y)$$

Z contains all vertices in $K^1 \cup K^2$ that are arranged after x in all DAG orderings of $K^1 \cup K^2$:

$$\text{after}(x, Z, K^1, K^2) \equiv \forall_{y, z \in K^1 \cup K^2} ((x = z \vee z \in Z) \wedge z <_{K^1, K^2} y) \rightarrow y \in Z$$

Deleting the edges of K^1 in T_1 and the edges of K^2 in T_2 yields an *acyclic* agreement forest:

$$\begin{aligned}\text{AAF}(K^1, K^2) \equiv &\rho^1 \in K^1 \wedge \rho^2 \in K^2 \wedge \text{rAF}(K^1, K^2) \wedge \text{no\_empty}(K^1, K^2) \wedge \\ &\forall_{x \in K^1 \cup K^2} \exists_{Z \subseteq K^1 \cup K^2} (\text{after}(x, Z, K^1, K^2) \wedge x \notin Z)\end{aligned}$$

The LinEMSOL-formulation is then "minimize $|K^1| + |K^2|$ such that $\mathfrak{S}_{\text{AF}} \models \text{AAF}(K^1, K^2)$". Note that the no_empty primitive is not needed because any set of edge cuts that creates components without taxa cannot be optimal but we left it in the formulation to help verify more easily that "corr" unambiguously encodes a bijection.

In the following, we use $x \leq^1 y$ ($x <^1 y$) to denote "x is a (strict) ancestor of y in T_1".

Lemma 2 *For each k, there are size-k sets K^1 and K^2 such that $\mathfrak{S}_{AF}$ is a model for $\mathrm{AAF}(K^1, K^2)$ if and only if there is an acyclic agreement forest of size k.*

Proof: "⇐": Let $\mathcal{F}$ be a size-k agreement forest and let $(F_1, \ldots, F_k)$ be a DAG ordering of $AG(T_1, T_2, \mathcal{F})$. For each $j \in \{1, \ldots, k\}$ let z_j^1 and z_j^2 denote the LCA of $X \cap V(F_j)$ in T_1 and T_2, respectively. Finally, for $i \in \{1, 2\}$ let $K^i := \{z_j^i \mid 1 \leq j \leq k\}$. Then, for all $j \in \{1, \ldots, k\}$, we have $\mathrm{corr}(z_j^1, z_j^2, K^1, K^2)$ and, therefor, z_j^1 and z_j^2 are incomparable with respect to $<_{K^1,K^2}$. Further, since $\rho \in X$, the LCA in T_1 of $X \cap V(F)$ for the tree F of $\mathcal{F}$ containing ρ is ρ^1. Thus, $\rho^1 \in K^1$ and, likewise, $\rho^2 \in K^2$. Since one can verify that $\mathfrak{S}_{\mathrm{AF}}$ is a model for $\mathrm{rAF}(K^1, K^2)$, it remains to show that $(z_1^1, z_1^2, \ldots, z_k^1, z_k^2)$ is a DAG ordering of $K^1 \cup K^2$ with respect to $<_{K^1,K^2}$. But as z_j^1 and z_j^2 are incomparable for each j and $(F_1, \ldots, F_k)$ is a DAG ordering of $AG(T_1, T_2, \mathcal{F})$, the claim follows. Thus, $\mathfrak{S}_{\mathrm{AF}}$ is also a model for $\mathrm{AAF}(K^1, K^2)$.

"⇒": Let $\mathfrak{S}_{\mathrm{AF}}$ be a model for $\mathrm{AAF}(K^1, K^2)$. Then, by definition of AAF, there is a DAG ordering $(z_1, \ldots, z_k)$ of K^1 (with respect to $<_{K^1,K^2}$). To construct an agreement forest $\mathcal{F} = (F_1, \ldots, F_k)$, let F_i be the subtree of $T_1 - \{z_1, \ldots, z_{i-1}\}$ that is rooted at z_i. Since $\mathfrak{S}_{\mathrm{AF}}$ is a model for $\mathrm{rAF}(K^1, K^2)$, we know that $\mathcal{F}$ is indeed an agreement forest as proved in Section 3.2. In the following, we show that $\mathcal{F}$ is acyclic. Towards a contradiction, assume that there is some $\ell \in \{1, 2\}$ and some i, j such that $i < j$ and the root x_j of $T_\ell(\mathcal{L}(F_j))$ is an ancestor of the root x_i of $T_\ell(\mathcal{L}(F_i))$ in T_ℓ (that is $x_j \leq^\ell x_i$). Without loss of generality, let $\ell = 1$. Note that $z_j \leq^1 x_j$ and, since F_i and F_j are distinct, there is a vertex $z \in K^1$ with $z >^1 x_j$ on the unique x_j-x_i-path in T_1 (possibly $z = z_i$). Then, since there are no vertices of K^1 between x_i and z_i, we have $z_i \geq^1 z >^1 x_j \geq^1 z_j$, contradicting that $(z_1, \ldots, z_k)$ is a DAG ordering of K^1 with $i < j$. □

To establish Theorem 4 it remains to prove that the display graph has treewidth bounded by a function of HN/MAAF. An acyclic agreement forest is trivially an agreement forest, so this is immediate from Theorem 1.

3.4 Maximum parsimony distance on binary characters

Let T be an unrooted binary tree on a set of taxa X. A *binary character* f is simply a function $f : X \to \{red, blue\}$. An *extension* of f to T is a mapping $g : V(T) \to \{red, blue\}$ such that, for all $x \in X$, $g(x) = f(x)$. For a given character f, an *optimal extension* is any extension g of f such that the number of bichromatic edges is minimized. The number of bichromatic edges in an optimal extension is called the *parsimony score* of f with respect to T, and denoted $l_f(T)$. The well-known algorithm by Fitch can be used to compute $l_f(T)$ (and an optimal extension) in polynomial time [22]. We shall describe Fitch's algorithm in due course. The *maximum parsimony distance problem on binary characters*, denoted d_{MP}^2, is defined as follows [21].

Problem: $d^2_{MP}(T_1, T_2)$
Input: Two unrooted binary trees T_1, T_2 on the same set of taxa X
Output: Construct a binary character f on X such that the value $|l_f(T_1) - l_f(T_2)|$ is maximized.

We overload d^2_{MP} to also denote the optimum value of $|l_f(T_1) - l_f(T_2)|$. The problem was recently shown to be NP-hard and APX-hard [28]. It is not known whether the problem is FPT in d^2_{MP}. The following result, however, is already known.

Lemma 3 ([21]) *Let T_1, T_2 be two unrooted binary trees on the same set of taxa X. Then $d^2_{MP}(T_1, T_2) \leq d_{TBR}(T_1, T_2)$.*

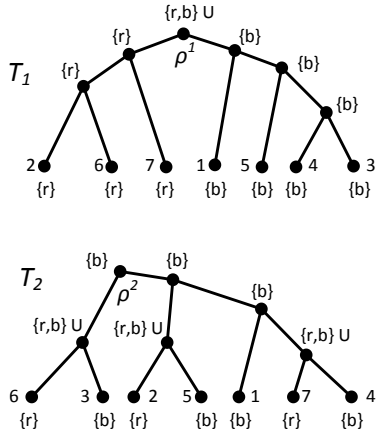


Figure 6: *The two rooted trees from Figure 4, but this time drawn differently and without an explicit extra taxon ρ, which in this context is not necessary. The figure shows the execution of the bottom-up phase of Fitch's algorithm on the two trees when applied to the character f, which maps taxa 2,6,7 to* red *and taxa 1,3,4,5 to* blue. *The label $\{r, b\}\cup$ denotes a union event, so the parsimony score of T_1 with respect to f is 1, and of T_2 with respect to f is 3. Hence, $d^2_{MP}(T_1, T_2) \geq |1 - 3| = 2$. In fact, no other character can improve upon this, so $d^2_{MP}(T_1, T_2) = 2$. Once the two trees are combined into the display graph, R^1 will be the 5 vertices of T_1 labelled $\{r\}$, B^1 will be the 7 vertices of T_1 labelled $\{b\}$, $RB^1_I = \emptyset$ and RB^1_U will be the single vertex of T_1 labelled $\{r, b\}\cup$. R^2 will be the 3 vertices of T_2 labelled $\{r\}$, B^2 will be the 7 vertices of T_2 labelled $\{b\}$, $RB^2_I = \emptyset$ and RB^2_U will be the 3 vertices of T_2 labelled $\{r, b\}\cup$.*

Given two trees T_1, T_2 as input to d^2_{MP}, it is not known whether the display graph D of T_1 and T_2 has treewidth bounded by a function of d^2_{MP}. However, from Lemma 3 and earlier results in this article (Theorems 6 and 1) it is clear that D has treewidth bounded by a function of $d_{TBR}(T_1, T_2)$.

The MSOL formulation we give here, which is based on an ILP formulation from [28], maximizes $l_f(T_1) - l_f(T_2)$. (To compute d^2_{MP} we need to use the MSOL machinery twice, once for $l_f(T_1) - l_f(T_2)$ and once for $l_f(T_2) - l_f(T_1)$, taking the maximum of the two results. The second call only differs in its objective function so we omit details).

The basic idea is to range over all possible binary characters, simultaneously embedding two static formulations[3] of Fitch's algorithm to "compute" $l_f(T_1) - l_f(T_2)$.

Fitch's algorithm proceeds as follows. If T is not rooted, we root it arbitrarily (by subdividing an arbitrary edge). The algorithm then works in two phases, a bottom-up phase which computes $l_f(T)$, and then a top-down phase which actually computes a corresponding extension. In the bottom-up phase, we start by assigning each taxon x the singleton set of colours $S(x) := \{f(x)\}$. For an internal vertex u with children v_1, v_2 we set $S(u) := S(v_1) \cap S(v_2)$ (if $S(v_1) \cap S(v_2) \neq \emptyset$, in which case we say u is an *intersection vertex*) and $S(u) := S(v_1) \cup S(v_2)$ (if $S(v_1) \cap S(v_2) = \emptyset$, in which case we say that u is a *union vertex*). The value $l_f(T)$ is equal to the number of internal vertices that are union vertices. See Figure 6. (We omit a description of the constructive top-down phase as it is not relevant for this article).

To translate this into an MSOL formulation, we begin by arbitrarily rooting T_1 and T_2. (In this case we can avoid introducing an explicit taxon ρ: for $i \in \{1, 2\}$ we simply subdivide an arbitrary edge in T_i and let ρ^i be the subdivision vertex.) The central idea is to partition the vertices of each tree T_i into four possible subsets R^i, B^i, RB^i_I and RB^i_U corresponding to the subset of colours that Fitch allocates to each vertex, and distinguishing union events from intersection events: *red*, *blue*, {*red, blue*} (intersection vertex) and {*red, blue*} (union vertex). See again Figure 6. We subsequently ask the MSOL formulation to instantiate the free set variables R^i, B^i, RB^i_I and RB^i_U ($i \in \{1, 2\}$) such that the expression $|RB^1_U| - |RB^2_U|$ is maximized.
The following primitives will be useful:

v is a child of u in T_i:

$$\text{child}^i(u, v) \equiv \text{E}^i(u, v) \wedge \text{in\_path}^i(\rho^i, v, u)$$

c_1 and c_2 are distinct children of u in T_i:

$$\text{children}^i(c_1, c_2, u) \equiv (c_1 \neq c_2) \wedge \text{child}^i(u, c_1) \wedge \text{child}^i(u, c_2)$$

For each tree T_i we add the following constraints:

The four subsets R, B, RB_I and RB_U partition the vertices of the tree; we omit the formulation as it is trivial:

$$\text{partition}(V^i, R^i, B^i, RB^i_I, RB^i_U)$$

[3] Interestingly, the earlier phylogenetics MSOL articles [11, 31] also used static formulations: in that case the classical polynomial-time algorithm of Aho.

A vertex in X can only be in R or B:

$\forall_{x \in X}(x \in R^i \vee x \in B^i)$

An internal vertex is in R if and only if (one child is in R and the other child is not in B):

$\forall_{u \in V^i \setminus X}(u \in R^i \Leftrightarrow \exists_{c_1,c_2 \in V_i}(\text{children}^i(c_1, c_2, u) \wedge c_1 \in R^i \wedge c_2 \notin B^i))$

An internal vertex is in B if and only if (one child is in B and the other child is not in R):

$\forall_{u \in V^i \setminus X}(u \in B^i \Leftrightarrow \exists_{c_1,c_2 \in V_i}(\text{children}^i(c_1, c_2, u) \wedge c_1 \in B^i \wedge c_2 \notin R^i))$

An internal vertex is in RB_I if and only if (neither child is in R or B):

$\forall_{u \in V^i \setminus X}(u \in RB_I^i \Leftrightarrow \exists_{c_1,c_2 \in V_i}(\text{children}^i(c_1, c_2, u) \wedge c_1 \notin R^i \cup B^i \wedge c_2 \notin R^i \cup B^i))$

An internal vertex is in RB_U if and only if (one child is in R and one child is in B):

$\forall_{u \in V^i \setminus X}(u \in RB_U^i \Leftrightarrow \exists_{c_1,c_2 \in V_i}(\text{children}^i(c_1, c_2, u) \wedge c_1 \in R^i \wedge c_2 \in B^i))$

Finally, we ensure that both trees select exactly the same character:

$$\forall_{x \in X}((x \in R^1 \Leftrightarrow x \in R^2) \wedge (x \in B^1 \Leftrightarrow x \in B^2))$$

We then naturally define $d^2_{MP}(R^1, B^1, RB^1_I, RB^1_U, R^2, B^2, RB^2_I, RB^2_U)$ as the conjunction of all the above constraints. The LinEMSOL-formulation for d^2_{MP} is then "maximize $|RB^1_U| - |RB^2_U|$ such that $(V, E, \rho^1, \rho^2, V^1, V^2, X) \models d^2_{MP}(R^1, B^1, RB^1_I, RB^1_U, R^2, B^2, RB^2_I, RB^2_U)$".

4 Experiments

In this section we re-analyse the well-known *Poaceae* grass dataset [24][4]. The dataset comprises 6 rooted, binary trees which were combined into 15 pairs. Each pair is on the same set of taxa, but the number of taxa varies between pairs due to restriction to common taxa. For each pair we computed hybridization number and rSPR distance exactly using Dendroscope 3 [27]. We also computed TBR distance exactly using an ad-hoc ILP formulation. Computation of TBR distance disregards the root location and treats the trees as being unrooted. The parameter uMAF is then obtained simply by adding one to the TBR distance (recall Theorem 6). For each pair, we used the "Greedy Fill-In" heuristic [6] to compute an *upper bound* on the treewidth of the display graph; exact computation of the treewidth was computationally infeasible. (Here the display graph does not include an extra taxon ρ to encode the root location since ρ has a minimal impact on the treewidth.) See Table 1 for the results. For completeness we also computed d^2_{MP} (using the ILP software from [28])

tree pair		*taxa*	*HN*	*rSPR*	*TBR*	*uMAF*	**TW** $\leq$	*display graph size*	d^2_{MP}
rpoC2	waxy	10	1	1	1	2	**3**	$\|V\|$=28,$\|E\|$=36	1
phyB	waxy	14	3	3	2	3	**3**	$\|V\|$=40,$\|E\|$=52	2
phyB	rbcL	21	4	4	4	5	**3**	$\|V\|$=61,$\|E\|$=80	3
rbcL	waxy	12	7	6	3	4	**3**	$\|V\|$=34,$\|E\|$=44	3
phyB	rpoC2	21	7	6	4	5	**3**	$\|V\|$=61,$\|E\|$=80	3
waxy	ITS	15	8	7	5	6	**4**	$\|V\|$=43,$\|E\|$=56	3
phyB	ITS	30	8	8	7	8	**4**	$\|V\|$=88,$\|E\|$=116	5
ndhF	waxy	19	9	7	4	5	**4**	$\|V\|$=55,$\|E\|$=72	3
ndhF	rpoC2	34	12	11	8	9	**5**	$\|V\|$=100,$\|E\|$=132	6
rbcL	rpoC2	26	13	11	6	7	**5**	$\|V\|$=76,$\|E\|$=100	4
ndhF	rbcL	36	13	10	6	7	**3**	$\|V\|$=106,$\|E\|$=140	4
rbcL	ITS	29	14	13	10	11	**5**	$\|V\|$=85,$\|E\|$=112	6
ndhF	phyB	40	14	12	6	7	**3**	$\|V\|$=118,$\|E\|$=156	6
rpoC2	ITS	31	15	14	10	11	**6**	$\|V\|$=91,$\|E\|$=120	7
ndhF	ITS	46	19	19	15	16	**6**	$\|V\|$=136,$\|E\|$=180	10

Table 1: The results of our experiments with the *Poaceae* grass dataset.

but, because it is not known whether the treewidth of the display graph can be bounded by a function of d^2_{MP}, we have placed it at the periphery of the table.

The main observation is that, in this data set, the treewidth of the display graph appears to grow much more slowly than TBR distance / uMAF, and thus automatically much more slowly than hybridization number and rSPR distance. For a number of incongruence measures not considered in this article, such as Unrooted Subtree Prune and Regraft distance (uSPR) and Nearest Neighbour Interchange (NNI) distance, this effect will also be observed, because TBR is a lower bound on both uSPR and NNI [1]. Interestingly, despite the fact that we only use an upper bound on the treewidth of the display graph, the bound given in Theorem 1 is still satisfied in all cases, and for larger TBR values is rather pessimistic.

In order to understand whether the low-treewidth phenomenon observed in the Poaceae dataset is the rule rather than the exception, we need to deepen our understanding of how phenomena such as horizontal gene transfer and hybridization impact upon phylogenetic tree topology in practice. However, mathematical models for incongruence are still very much in a developmental phase and it is difficult to construct meaningful experiments based on simulations. For this reason the natural way forward is is to undertake a comprehensive empirical analysis of multiple exisiting biological datasets. This is beyond the scope of this article, and is therefore deferred to future research. Nevertheless, it is already useful to note that, in practice, the phylogenetic trees that are compared to each other are often assumed to have a significant amount of shared history,

[4]Note the trees used here are the ones obtained in [27] by re-analysing the sequence data in [24] with a more powerful software than the one used in the original publication.

and thus topological structure. This creates some hope that, for many credible biological datasets, the treewidth of thedisplay graph will indeed be low.

5 Conclusion

We have demonstrated how agreement forests, which are intensively studied objects in the phylogenetics literature, naturally lead to bounded treewidth in an auxiliary graph structure known as the *display graph*. This opens the door to compact, "declarative" proofs of fixed parameter tractability for a range of phylogenetics problems by formulating them in Monadic Second Order Logic (MSOL). Our formulations have introduced a number of logical primitives and design principles that will hopefully be of use to other phylogenetics researchers seeking to utilize this powerful machinery elsewhere in phylogenetics. Indeed, it is natural to ask: what are the essential characteristics of phylogenetics problems that are amenable to this technique? Can the low treewidth, that was observed in our re-analysis of a well-known dataset, be leveraged to obtain competitive algorithms that operate directly on tree decompositions of the display graph?

References

[1] B. Allen and M. Steel. Subtree transfer operations and their induced metrics on evolutionary trees. *Annals of Combinatorics*, 5:1–15, 2001. doi:10.1007/s00026-001-8006-8.

[2] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12:308 – 340, 1991. doi:10.1016/0196-6774(91)90006-K.

[3] M. Baroni, S. Grünewald, V. Moulton, and C. Semple. Bounding the number of hybridisation events for a consistent evolutionary history. *Mathematical Biology*, 51:171–182, 2005. doi:10.1007/s00285-005-0315-9.

[4] H. L. Bodlaender. A tourist guide through treewidth. *Acta cybernetica*, 11(1-2):1, 1994.

[5] H. L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM Journal of Computing*, 25:1305–1317, 1996. doi:10.1137/S0097539793251219.

[6] H. L. Bodlaender and A. M. C. A. Koster. Treewidth computations I. upper bounds. *Inf. Comput.*, 208(3):259–275, 2010. doi:10.1016/j.ic.2009.03.008.

[7] M. Bordewich and C. Semple. On the computational complexity of the rooted subtree prune and regraft distance. *Annals of Combinatorics*, 8:409–423, 2004. doi:10.1007/s00026-004-0229-z.

[8] M. Bordewich and C. Semple. Computing the hybridization number of two phylogenetic trees is fixed-parameter tractable. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4:458–466, 2007. doi:10.1109/tcbb.2007.1019.

[9] M. Bordewich and C. Semple. Computing the minimum number of hybridization events for a consistent evolutionary history. *Discrete Applied Mathematics*, 155:914–928, Apr. 2007. doi:10.1016/j.dam.2006.08.008.

[10] D. Bryant. The splits in the neighborhood of a tree. *Annals of Combinatorics*, 8(1):1–11, 2004. doi:10.1007/s00026-004-0200-z.

[11] D. Bryant and J. Lagergren. Compatibility of unrooted phylogenetic trees is FPT. *Theoretical Computer Science*, 351:296 – 302, 2006. doi:10.1016/j.tcs.2005.10.033.

[12] D. Bryant and M. Steel. Extension operations on sets of leaf-labeled trees. *Advances in Applied Mathematics*, 16(4):425–453, 1995. doi:10.1006/aama.1995.1020.

[13] O. P. Buneman. The recovery of trees from measures of dissimilarity. *Mathematics in the archaeological and historical sciences*, 1971.

[14] J. Chen, J.-H. Fan, and S.-H. Sze. Parameterized and approximation algorithms for maximum agreement forest in multifurcating trees. *Theoretical Computer Science*, 562:496–512, 2015. doi:10.1016/j.tcs.2014.10.031.

[15] D. G. Corneil and U. Rotics. On the relationship between cliquewidth and treewidth. *SIAM Journal on Computing*, 34(4):825–427, 2005. doi:10.1137/S0097539701385351.

[16] B. Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and Computation*, 85:12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

[17] B. Courcelle, J. A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000. doi:10.1007/s002249910009.

[18] B. Courcelle and S. Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101(1-3):77–114, 2000. doi:10.1016/S0166-218X(99)00184-5.

[19] R. G. Downey and M. R. Fellows. *Fundamentals of parameterized complexity*, volume 4. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

[20] J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Incorporated, 2004.

[21] M. Fischer and S. Kelk. On the maximum parsimony distance between phylogenetic trees. *Annals of Combinatorics*, 2014. preliminary version arXiv preprint arXiv:1402.1553. doi:10.1007/s00026-015-0298-1.

[22] W. Fitch. Toward defining the course of evolution: minimum change for a specific tree topology. *Systematic Biology*, 20(4):406–416, 1971. doi:10.2307/2412116.

[23] A. Grigoriev, S. Kelk, and N. Lekić. On low treewidth graphs and supertrees. *Journal of Graph Algorithms and Applications*, 19(1):325–243, 2015. doi:10.1007/978-3-319-07953-0_6.

[24] G. P. W. Group, N. P. Barker, L. G. Clark, J. I. Davis, M. R. Duvall, G. F. Guala, C. Hsiao, E. A. Kellogg, H. P. Linder, R. J. Mason-Gamer, et al. Phylogeny and subfamilial classification of the grasses (poaceae). *Annals of the Missouri Botanical Garden*, pages 373–457, 2001. doi:10.2307/3298585.

[25] D. H. Huson, R. Rupp, and C. Scornavacca. *Phylogenetic networks: Concepts, Algorithms and Applications*. Cambridge University Press, 2010. doi:10.1017/CBO9780511974076.

[26] D. H. Huson and C. Scornavacca. A survey of combinatorial methods for phylogenetic networks. *Genome biology and evolution*, 3(1):23–35, Jan. 2011. doi:10.1093/gbe/evq077.

[27] D. H. Huson and C. Scornavacca. Dendroscope 3: an interactive tool for rooted phylogenetic trees and networks. *Systematic biology*, page sys062, 2012. doi:10.1093/sysbio/sys062.

[28] S. Kelk and M. Fischer. On the complexity of computing mp distance between binary phylogenetic trees. *arXiv preprint arXiv:1412.4076*, 2014.

[29] S. Kelk., L. van Iersel, and C. Scornavacca. Phylogenetic incongruence through the lens of monadic second order logic. *arXiv preprint arXiv:1503.00368*, February 2015.

[30] N. Robertson and P. D. Seymour. Graph minors. II. Algorithmic aspects of tree-width. *Journal of algorithms*, 7(3):309–322, 1986. doi:10.1016/0196-6774(86)90023-4.

[31] C. Scornavacca, L. van Iersel, S. Kelk, and D. Bryant. The agreement problem for unrooted phylogenetic trees is FPT. *Journal of Graph Algorithms and Applications*, 18:385–392, 2014. doi:10.7155/jgaa.00327.

[32] C. Semple and M. Steel. *Phylogenetics*. Oxford University Press, 2003.

[33] C. Whidden, R. G. Beiko, and N. Zeh. Fixed-parameter algorithms for maximum agreement forests. *SIAM Journal on Computing*, 42(4):1431–1466, 2013. doi:10.1137/110845045.

[34] C. Whidden, N. Zeh, and R. G. Beiko. Supertrees based on the subtree prune-and-regraft distance. *Systematic Biology*, 63(4):566–581, 2014. doi:10.1093/sysbio/syu023.