Fast computation of minimum hybridization networks

Benjamin Albrecht, Celine Scornavacca, Alberto Cenci, Daniel Huson

Fast computation of minimum hybridization networks

Benjamin Albrecht[1]*, Celine Scornavacca[1*†], Alberto Cenci[2] and Daniel H. Huson[1†]

[1]Center for Bioinformatics (ZBIT), Tübingen University, Sand 14, 72076 Tübingen, Germany
[2]IRD, Institut de Recherche pour le Développement, BP 64501, 34394 Montpellier Cedex 5, France

ABSTRACT

Motivation: Hybridization events in evolution may lead to incongruent gene trees. One approach to determining possible interspecific hybridization events is to compute a hybridization network that attempts to reconcile incongruent gene trees using a minimum number of hybridization events.

Results: We describe how to compute a representative set of minimum hybridization networks for two given bifurcating input trees, using a parallel algorithm and provide a user-friendly implementation. A simulation study suggests that our program performs significantly better than existing software on biologically relevant data. Finally, we demonstrate the application of such methods in the context of the evolution of the Aegilops/Triticum genera.

Availability and Implementation: The algorithm is implemented in the program Dendroscope 3, which is freely available from `www.dendroscope.org` and runs on all three major operating systems.

Contact: scornava or huson @informatik.uni-tuebingen.de

1 INTRODUCTION

Speciation by hybridization (Gross and Rieseberg, 2005; Mallet, 2007) is a widespread phenomenon in plants (Rieseberg *et al*., 2000; Soltis and Soltis, 2009), but also occurs in some other types of organisms (Schwenk *et al*., 2008; Giraud *et al*., 2008). When two individuals from distinct species hybridize and merge their sets of chromosomes, the hybrid organism is often sterile and does not produce any progeny. Differences among homologous chromosomes prevent correct meiotic pairing and viable gamete production. However, the eventual doubling of the chromosome number could restore a correct pairing (each chromosome pair with its double), and the fertile genotype could give rise to a new allopolyploid species. In this case, homologous chromosomes do not mix and the number of the genes is doubled. In other words, the genes from both parental species coexist in the polyploid genome and evolve independently.

In the case that the parental species are genetically similar enough, the pairing between homologous chromosomes is not completely prevented and balanced meiosis could take place. In this case, the hybrid genotype could produce viable hybrid gametes (containing portions of both parental chromosomes) and progeny. If the progeny remains reproductively isolated from the parental genotypes, it could give rise to a new species (homoploid hybrid speciation).

This implies that, in a hybrid species, different genes may have different evolutionary histories, in which case they will give rise to incongruent gene trees. Thus, using a single rooted phylogenetic tree to represent the evolutionary history of a set of taxa may be inaccurate in the presence of interspecific hybridization. A more precise description may be possible using a rooted phylogenetic network, in which internal *tree* nodes (nodes of indegree 1) represent putative speciation events, whereas *reticulate* nodes (nodes of indegree ≥ 2) represent possible hybridization events.

Suppose we are given a collection of species for which we suspect that hybridization events have played an important role in their evolution. One way to determine a set of possible hybridization events is to compute a *hybridization network* for a given set of gene trees that aims at explaining the incongruences between the different trees using a minimum number of putative hybridization events. A hybridization network for a set of trees $\mathcal{T}$ is simply a rooted phylogenetic network containing the trees in $\mathcal{T}$. In computational terms, the problem can be formulated as follows: Given a set $\mathcal{T}$ of two or more rooted phylogenetic trees, determine all hybridization networks for $\mathcal{T}$ that are minimum in the sense that they have a minimum *reticulate number* (see below). This problem is known to be a computationally hard problem even for the case of determining only one hybridization network for two bifurcating trees on the same set of taxa (Bordewich and Semple, 2007b).

In this paper we present an algorithm that takes as input two bifurcating, rooted phylogenetic trees T_1 and T_2 on the same taxon set $\mathcal{X}$ and produces as output a *representative set* of minimum hybridization networks N on $\mathcal{X}$ that contain both trees. Such a *representative set* is defined to contain exactly one network derived from each of the possible *maximum acyclic agreement forests* (MAAF), as defined below. The number of reticulation nodes in any such minimum network is called the *hybridization number*, denoted by $h(T_1, T_2)$, for T_1 and T_2.

Our algorithm is based on previous work described in (Bordewich and Semple, 2005; Baroni *et al*., 2006; Bordewich and Semple, 2007a), which aims at computing the hybridization number, and on the work reported in (Whidden and Zeh, 2009; Whidden *et al*., 2010) on the *rooted SPR* distance computation. We extend the published approach and provide a parallel implementation to compute a representative set of minimum hybridization networks containing one network per MAAF. In this context, a number of theoretical

*These authors contributed equally to this work.

†To whom correspondence should be addressed.

issues arise and we will show how to address them in a forthcoming paper (Scornavacca *et al*., 2011).

In our experience, the number of resulting networks can be quite large and so we provide methods for showing how the two input trees are embedded in the networks and for determining the number of different networks that contain a specific reticulation. We also provide variants of the algorithm that can be used to compute the *rooted SPR* distance (defined below) or the hybridization number of two bifurcating, rooted phylogenetic trees.

We report on a simulation study that we have undertaken to compare our implementation with other competing methods. This study indicates that our approach is much faster than existing methods. Moreover, to illustrate how one may apply our method to a practical problem, we use it to investigate the evolution of the Aegilops/Triticum genera.

The algorithm presented in this paper is implemented in our program Dendroscope 3 (Huson and Scornavacca, 2011), which is freely available from `www.dendroscope.org` and runs on all three major operation systems.

2 METHODS

Throughout this paper, we follow the terminology and notation defined in (Huson *et al*., 2011) and assume that the reader is familiar with graphs and related terminology. Let T be a phylogenetic tree on $\mathcal{X}$ and let $\mathcal{X}' \subset \mathcal{X}$ be a subset of taxa. We use $T(\mathcal{X}')$ to denote the minimum connected subgraph of T that contains all leaves that are labeled by elements of $\mathcal{X}'$. The *restriction* of T to $\mathcal{X}'$ is defined as the phylogenetic tree $T|_{\mathcal{X}'}$ that is obtained from $T(\mathcal{X}')$ be suppressing all nodes that have both in- and outdegree 1.

We define a *rooted phylogenetic network* on $\mathcal{X}$ as a directed acyclic graph with a single node with indegree zero (the *root*), no nodes with both indegree and outdegree equal to 1, and nodes with outdegree zero (the *leaves*) bijectively labeled by the set $\mathcal{X}$.

Given a rooted bifurcating phylogenetic tree T, a *rooted Subtree Prune and Regraft move* (rSPR-move for short) on T is performed by first detaching a subtree of T rooted at the target of $e_1 = (v_1, w_1)$ by deleting the edge e_1 and re-grafting the subtree on a different branch e_2 of T, by first creating a new node z_2 in e_2 and then a new edge (z_2, w_1). Finally, any node with both in- and outdegree 1 is suppressed. Note that, in the case of regrafting above the root ρ, a new root node ρ' has to be created, as well as two new edges (ρ', ρ) and (ρ', w_1).

Let T_1 and T_2 be two rooted bifurcating phylogenetic trees on a taxon set $\mathcal{X}$. The *rSPR distance* between T_1 and T_2 is defined as the minimum number of rSPR-moves required to transform T_1 into T_2. The problem of computing the rSPR distance between two rooted bifurcating phylogenetic trees on the same taxon set is known to be NP-hard, but fixed-parameter tractable (FPT) (Bordewich and Semple, 2005).

Let T_1 and T_2 be two rooted bifurcating phylogenetic trees on $\mathcal{X}$. For technical purposes, we assume that the root ρ of both trees is a pendant node that has been adjoined to the original root and no re-grafting is permitted above ρ. An *agreement forest* for T_1 and T_2 on $\mathcal{X} \cup \{\rho\}$ is a set of phylogenetic trees, called also a *forest*, $\mathcal{F} = \{F_\rho, F_1, \ldots, F_{h-1}\}$ on $\mathcal{X} \cup \{\rho\}$ that has the following properties:

1. Each tree F_i in $\mathcal{F}$ is the restriction of T_1, and also of T_2, to the set of taxa $\mathcal{X}_i$ that appear in F_i.
2. The root ρ is contained in F_ρ.
3. The trees in $\{T_1(\mathcal{X}_i) \mid i \in \{\rho, 1 \ldots, h-1\}\}$ and $\{T_2(\mathcal{X}_i) \mid i \in \{\rho, 1, \ldots, h-1\}\}$ are node disjoint subtrees of T_1 and T_2, respectively.

An agreement forest with minimum cardinality is called a *maximum agreement forest* (*MAF* for short). The concepts of rSPR distance and MAFs are closely related. Indeed, a pair of rooted bifurcating phylogenetic trees $\{T_1, T_2\}$ has an rSPR distance equals to d if and only if there exists a MAF $\mathcal{F}(T_1, T_2)$ with size $d + 1$ (Bordewich and Semple, 2005; Hein *et al*., 1996). Hence, to determine the rSPR-distance between T_1 and T_2, it suffices to compute a MAF for these two trees.

Recall that a *hybridization network* for two rooted bifurcating phylogenetic trees T_1 and T_2 on $\mathcal{X}$, is a rooted phylogenetic network that contains both trees. Given a rooted phylogenetic network $N = (V, E)$, the *reticulate number* of N is defined as

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

where $\delta^-(v)$ denotes the indegree of v. In the special case that N is *bicombining*, that is, all nodes have indegree at most two, then this is simply the number of reticulation nodes.

The hybridization number for T_1 and T_2 is the minimum reticulation number obtained over all hybridization networks N for T_1 and T_2. The problem of computing the hybridization number for two rooted bifurcating phylogenetic trees on the same taxon set is known to be NP-hard, but FPT (Bordewich and Semple, 2007a; Linz and Semple, 2010). The hybridization number can be calculated by computing a maximum *acyclic* agreement forest. An agreement forest $\mathcal{F}_a(T_1, T_2)$ for T_1 and T_2 is called *acyclic* if its *ancestor-descendant graph* $AG(T_1, T_2, \mathcal{F}_a(T_1, T_2))$ does not contain any directed cycle. This graph is defined as the directed graph whose vertex set is $\mathcal{F}_a(T_1, T_2)$ and for which an edge (F_i, F_j) exists precisely whenever $i \neq j$, and either

1. the root of $T_1(\mathcal{X}_i)$ is an ancestor of the root of $T_1(\mathcal{X}_j)$ in T_1, or
2. the root of $T_2(\mathcal{X}_i)$ is an ancestor of the root of $T_2(\mathcal{X}_j)$ in T_2.

where $\mathcal{X}_i, \mathcal{X}_j \subseteq \mathcal{X}$ are the sets of taxa that appear in F_i and F_j, respectively. An acyclic agreement forest with a minimum number of components is called a *maximum acyclic agreement forest* (*MAAF* for short). If a MAAF with h components exists, then a hybridization network with reticulation number $h - 1$ containing both T_1 and T_2 exists (Baroni *et al*., 2005). For example, if a MAAF with only one component exists, then we have that T_1 and T_2 are congruent and 0 reticulations are needed.

2.1 The algorithm

In this section, we give a high level description of an algorithm that takes as input two rooted bifurcating phylogenetic trees T_1 and T_2 on the same taxon set $\mathcal{X}$, and produces as output a representative set of minimum hybridization networks N on $\mathcal{X}$ that contain both trees, providing exactly one network per MAAF. The problem of computing the rSPR-distance or the hybridization number between two trees is algorithmically a much simpler problem than

computing the hybridization networks and the algorithm that we have implemented deals with these two problems, too.

The algorithm consists of three phases, namely, a *reduction* phase, an *exhaustive* search phase and a *final* phase. In the latter, either the rSPR-distance or hybridization number is reported, or the final hybridization networks are constructed. Whereas the first and the third phases can be executed in polynomial time, the exhaustive search phase is known to be NP-hard (Bordewich and Semple, 2007b). The main aim of the initial reduction phase is to decrease the practical running time by reducing the size of the two trees that are passed to the exhaustive search phase.

2.1.1 Reduction Phase In the first phase of the algorithm, certain patterns present in both T_1 and T_2 are identified and used to reduce the instance of the problem. There are three types of reductions (see Huson *et al*., 2011, for a review). A *subtree reduction* reduces pendant subtrees that are common to both trees (Bordewich and Semple, 2005). This simplification preserves both the rSPR distance and the hybridization number. A *chain reduction* reduces maximal chains of at least three leaves. Chain reductions for the MAF and the MAAF problem are described in (Bordewich and Semple, 2005) and (Bordewich and Semple, 2007a), respectively. Finally, a *cluster reduction* divides the problem into a number of smaller subproblems using the set of minimal clusters common to both trees. A cluster reduction for the computation of MAAFs can be found in (Baroni *et al*., 2006). Recently, a cluster reduction for the computation of MAFs has been proposed (Linz and Semple, 2010).

2.1.2 Exhaustive Search Phase The first phase of the algorithm will usually subdivide the original problem into several smaller subproblems. For each such subproblem (T_1', T_2'), we must compute a MAF, a MAAF or the set of all MAAFs, depending on whether we want to compute the rSPR-distance, the hybridization number or a set of hybridization networks, respectively. To compute a single MAF we use the FPT-algorithm described in (Whidden and Zeh, 2009). A lower bound for the rSPR-distance can be found using the 3-approximation algorithm of (Whidden *et al*., 2010). In (Whidden and Zeh, 2009), the authors also describe an FPT-algorithm for computing acyclic agreement forests. Unfortunately, that work is based on the incorrect assumption that it suffices to avoid all cycles of length two so as to obtain an acyclic agreement forest. In a forthcoming paper (Scornavacca *et al*., 2011, available at `http://arxiv.org/abs/1109.3268`) we will show how to correct and extend their algorithm so as to obtain all MAAFs for a given pair of trees. Broadly speaking, the algorithm works as follows: Suppose that we are interested in computing a MAAF for two trees T_1 and T_2. Our algorithm takes as input a tree R and a forest $\mathcal{F}$ and it proceeds in a bounded-search type fashion by recursively deleting an edge in $\mathcal{F}$ or reducing a common cherry of R and $\mathcal{F}$ until the resulting forest $\mathcal{F}$ is a forest for T_1 and T_2. (The algorithm is called the first time with $R = T_1$ and $\mathcal{F} = T_2$). More precisely, each recursion starts by picking an arbitrary cherry $\{a, c\}$ in R, i.e., a pair of leaves a and c adjacent to a common vertex. Depending on whether $\{a, c\}$ is a common cherry of R and $\mathcal{F}$ or not, and whether a and c are vertices of the same component in $\mathcal{F}$ or not, the algorithm branches into at most three computational paths by recursively calling itself. Regardless of whether $\{a, c\}$ is a common cherry of R and $\mathcal{F}$ or not, the algorithm branches into two new computational paths that correspond to deleting one of the edges entering a and c in $\mathcal{F}$, denoted e_a and e_c, respectively. Additionally, if $\{a, c\}$ is not a cherry in $\mathcal{F}$ and a and c are in the same connected component in $\mathcal{F}$, then the algorithm branches into a third computational path that corresponds to deleting an edge whose starting node lies on the shortest path connecting a and c in $\mathcal{F}$, e_a and e_c excluded. Similarly, if $\{a, c\}$ is a common cherry of R and $\mathcal{F}$, then the algorithm branches into a third path that corresponds to reducing the cherry $\{a, c\}$ to a new leaf labeled $\{a \cup c\}$ both in R and $\mathcal{F}$. If only one MAAF is required, then the algorithm is terminated as soon as the first MAAF is found.

2.1.3 Output phase In the case that our aim is only to compute the rSPR-distance or the hybridization number, we simply report the number of components in the MAF or MAAF, respectively, minus one. Otherwise, if our aim is to generate a representative set of hybridization networks, then we construct a phylogenetic network on $\mathcal{X}$ for each different MAAF computed in the exhaustive search phase as described in (Baroni *et al*., 2006). To do this, we first need to undo all the reductions performed in the reduction phase.

2.1.4 Parallelization and additional analysis

For the computation of the hybridization number or networks, each application of a cluster reduction in the reduction phase of the algorithm gives rise to two subproblems. Since the hybrid number for the unreduced problem is equal to the sum of the hybrid number for the two subproblems (Baroni *et al*., 2006), the exhaustive search can be run independently on the two subproblems. In our parallel implementation of the algorithm, each subproblem produced in this way is placed in a queue and the subproblems in the queue are dispatched to individual cores subject to availability.

For the parallelization of the computation of the rSPR distance, it is not that simple and a cluster hierarchy $\mathcal{H}$ has to be computed (Linz and Semple, 2010). A parallel analysis of the subproblems is then possible, but respecting the cluster hierarchy: a subproblem in the hierarchy is analyzed only once all its "descendent subproblems" are. Note that the sum of MAF sizes of each subproblem only provides an upper bound of the rSPR distance. For computing the exact rSPR distance some additional steps are required, see (Linz and Semple, 2010).

When constructing a hybridization network in the output phase of the algorithm we assign the number 1 or 2 to each reticulate edge, depending on whether the edge corresponds to tree T_1 or T_2, respectively. With this information we can highlight the edges of T_1 or T_2 in each of the computed networks.

The exhaustive search phase is also performed in a parallel fashion, launching different threads to search for agreement forests of increasing sizes. When an agreement forest of size k has been found, then all threads searching for an agreement forest of larger size are aborted. A similar strategy is used when searching for a single MAAF or for the set of all MAAFs.) Thus, parallelism is used even when the two input trees do not share any common cluster.

Let $\mathcal{N}$ be the set of output networks. For each network N_i in $\mathcal{N}$ and for each reticulate node r_z in N_i, we compute the set of leaves $\bar{L}(r_z)$ that can be reached by direct paths from r_z without crossing any reticulate node. (Note that this leaf set represents the leaf set of a component of the underlying MAAF). Given two networks $N_i, N_j \in \mathcal{N}$ and a reticulate node r_z in N_i, we say that N_j *contains* r_z if there exists a reticulate node r_l in N_j such that $\bar{L}(r_z) = \bar{L}(r_l)$. Then, for each reticulate node in one of the output

networks $N_1, \ldots, N_t$, we determine how many networks contain that particular node and then label the node by this number divided by the total number of computed networks t. Thus, a reticulate node that occurs in all computed t minimum hybridization networks obtains a support value of 1, whereas a reticulate node that occurs only once has a support value of $\frac{1}{t}$. In addition, the computed networks are listed in descending order with respect to the sum of the support values of their reticulate nodes. When interpreting support values, remember that only one network per MAAF is computed, when there might be many networks per MAAF.

3 SIMULATION STUDY

To study the performance of our approach and implementation, we undertook a simulation study. We generated 2000 synthetic datasets, each consisting of a pair of rooted bifurcating phylogenetic trees. The datasets are based on three parameters, namely the number of taxa n, the number of rSPR-moves k used to obtain the second tree from the first[1], and the *tangling* degree d (as defined below). For each of the following choices of the parameters, n in $\{20, 50, 100, 200\}$, k in $\{5, 10, ..., 50\}$ and d in $\{3, 5, 10, 15, 20\}$, we constructed ten pairs of trees, thus obtaining 2000 different datasets in total.

The *tangling degree* is an ad-hoc concept that we introduce to control how tangled the resulting network will be and it influences the way we obtain the second tree from the first using rSPR-moves. More precisely, let R be an rSPR-move performed on a tree T by choosing two edges e_1 and e_2, pruning the subtree of T rooted at the source of $e_1 = (v_1, w_1)$ and re-grafting it on $e_2 = (v_2, w_2)$. We say that R respects a *tangling degree* of d, if the path from the lowest common ancestor of the nodes v_1 and v_2 to the node v_1 contains at most d edges. The tangling degree is a useful concept because the smaller it is, the more likely it is that one or more cluster reductions can be performed on the resulting dataset. As we will see, the number of common clusters in the two input trees has a major effect on the performance of our algorithm. Indeed, the more cluster reductions we can perform, the smaller the problem instances are and the more effective our parallelization is.

In more detail, each pair (T_1, T_2) of rooted bifurcating phylogenetic trees for a given set of parameters n, k, and d is created as follows: The first tree T_1 on $\mathcal{X} = \{x_1, \ldots, x_n\}$ is generated by first creating a set of n leaf nodes bijectively labeled by the set $\mathcal{X}$. Then, two nodes u and v, both with indegree 0, are randomly picked and a new node w, along with two new edges (w, u) and (w, v), is created. This is done until only one node with no ancestor, the root, is present. The second tree T_2 is obtained from T_1 by applying k rSPR-moves, in each move respecting the given tangling degree d.

We compared our implementation with the best available software for computing the exact hybridization number between two rooted bifurcating trees on the same taxon set, which is HybridNET (Chen and Wang, 2010), available from `http://www.cs.cityu.edu.hk/~lwang/software/Hn/`. (The underlying algorithm is described in (Chen and Wang, 2011)). We do not consider the program HybridInterleave (Collins *et al*., 2011) because it has been shown to be much slower than the HybridNET software (Chen and Wang, 2010). Both programs have been run on a AMD Phenom X4 955 Processor with 4GB RAM.

We present the results of our simulation in Figures 1 and 2. Each program run that took more than 20 minutes was aborted and then counted as if it took 20 minutes. In Figure 1, aborted runs have been included in the averages and, in all plots, the percentages report the proportion of the executions that were completed within 20 minutes. In Figure 2, aborted runs have *not* been included in the averages because their hybridization number is unknown. In the latter figure, we report the number of the executions that were completed within 20 minutes.

Figures 1 and 2 show that the average running time (as function of the number of leaves, of the number of rSPRs or of the hybridization number) of our implementation is always lower than that of HybridNET. For both programs, the running time increases with the number of taxa (see Figure 1a-b), and also with the rSPR distance (approximated by the number of rSPR-moves) (see Figure 1c-d) and the hybridization number (see Figure 2). Figure 2 also shows that, in our simulations, HybridNet was unable to compute the hybridization number for any tree-pair with hybridization number greater than 21, while our program produces results for tree-pairs having a hybridization number up to 40.

Note that the number of common clusters has a significant effect on the performance of our algorithm (see Figure 1e-f), and this explains much of the performance advantage over HybridNet. However, even in the case that no parallelization is performed, namely when the number of common clusters is equal to 1, our implementation is still faster than HybridNet (see Figure 1e-f). Note that the plot in Figure 1f is very erratic when the number of common clusters is greater than 15. This is due to the fact that few datasets share such a high number of common clusters.

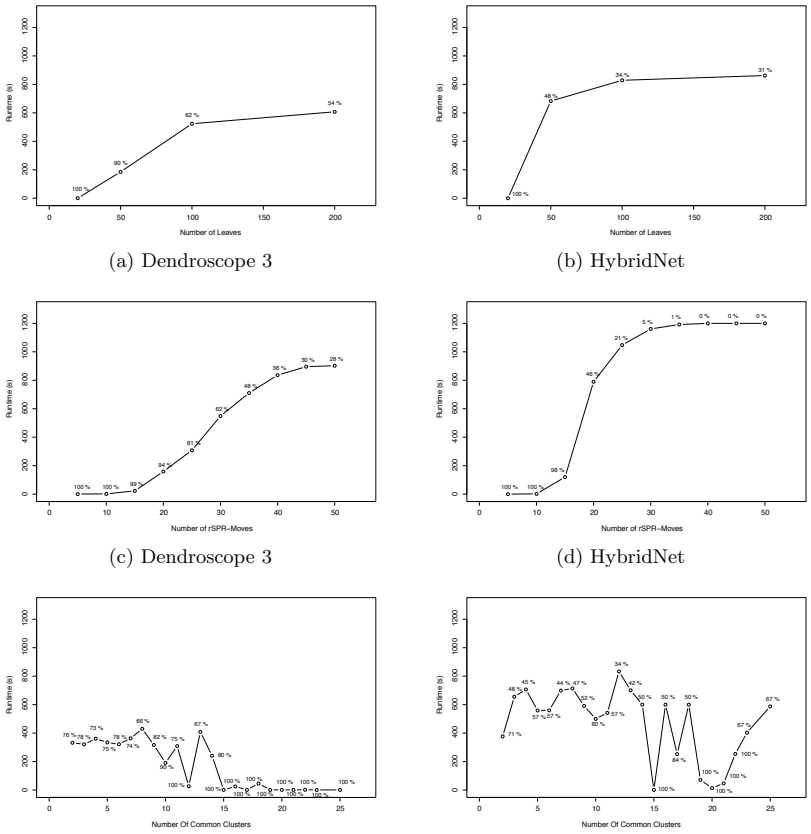
Using the same datasets, we also studied the performance of our implementation for the problems of computing the rSPR-distance for two rooted bifurcating phylogenetic trees. We compared the running time of our implementation with that of the best available software for computing the exact rSPR distance between two rooted bifurcating trees on the same taxon set, i.e., rSPR (Whidden *et al*., 2010), available at `http://kiwi.cs.dal.ca/Software/RSPR`. In all conditions, the program rSPR performs better than our implementation (data not shown). Since the underlying algorithms are the same in this case, we suspect the difference in performance may be due to the fact that our program is implemented in Java, whereas the rSPR program is written in C++.

For the sake of completeness, in Figure 1 of the supplementary material we report the running time of Dendroscope 3 and HybridNET when computing the exact hybridization number and a set of hybridization networks, containing a network per MAAF, on a grass (*Poaceae*) dataset provided by the Grass Phylogeny Working Group (Grass Phylogeny Working Group, 2001). This dataset has been often used to evaluate programs computing the hybridization number or hybridization networks for two rooted bifurcating phylogenetic trees on the same taxon set.

4 APPLICATION TO PHYLOGENY OF AEGILOP/TRITICUM GENERA

The Triticeae tribe (Fam. Poaceae) consists of diploid and polyploid grasses with the same basic haploid chromosome number (x = 7),

[1] Note that this number is an upper bound on the true rSPR distance.

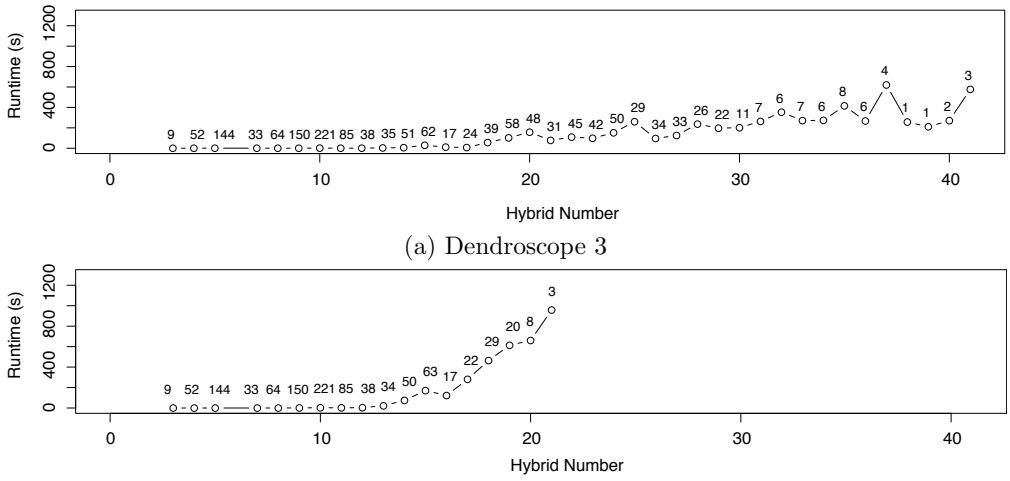


(a) Dendroscope 3 (b) HybridNet

(c) Dendroscope 3 (d) HybridNet

(e) Dendroscope 3 (f) HybridNet

Fig. 1. Comparison of the running time of Dendroscope 3 (on the left) and HybridNet (on the right). (a)-(b) Average running time as function of the number of leaves. (c)-(d) Average running time as function of the rSPR-moves. (e)-(f) Average running time as function of the number of common clusters. In all plots, the percentages report the proportion of the executions that were completed within 20 minutes.



(a) Dendroscope 3

(b) HybridNet

Fig. 2. Average running time as function of the hybridization number for Dendroscope 3 (a) and HybridNet (b). In both plots, we report the number of the executions that were completed within 20 minutes.

	PinA	matK
T. urartu TU55	EU307589	FJ897889
T. monococcum DP57	EU307591	FJ897868
Ae. tauschii DP16	FJ898213	FJ897861
Ae. comosa DP13	FJ898210	FJ897858
Ae. uniaristata DP56	FJ898218	FJ897867
Ae. bicornis DP18	FJ898215	FJ897863
Ae. longissima DP17	FJ898214	FJ897862
Ae. sharonensis DP53	FJ898216	FJ897864
Ae. speltoides SP6	FJ898222	FJ897884
Hordeum vulgare (Morex)	AY643843	EF115541

Table 1. Accession numbers in GenBank of the sequences used for obtaining the trees in Figure 3.

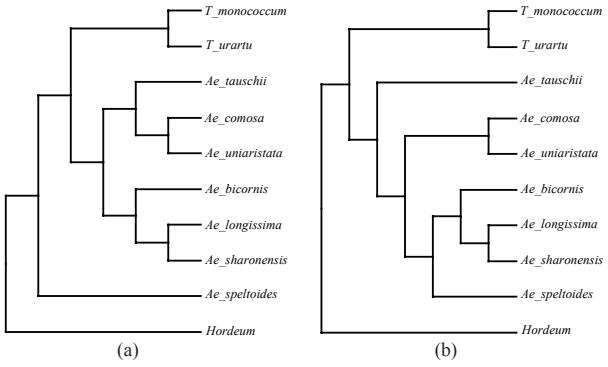


Fig. 3. The two consensus trees computed from 100 bootstrap replicates for the matK (a) and PinA (b) datasets.

and they are distributed worldwide. Phylogenic analysis performed with different sequence datasets show significant inconsistencies (Kellogg *et al*., 1996; Mason-Gamer and Kellogg, 1996; Sasanuma *et al*., 2004). In particular, inconsistencies between chloroplast and genomic data are often detected (Sasanuma *et al*., 2004). We used two datasets of sequences from number of diploid species belonging to the close genera Triticum and Aegilops: matK and PinA, located on the chloroplast and the Triticeae chromosome 5, respectively. Sequence data were obtained from GenBank. The accession numbers can be found in Table 1.

The best-fit model of nucleotide substitution for each dataset was chosen using JModeltest (Posada, 2008) and PhyML 3.0 (Guindon *et al*., 2010) was used to obtain maximum likelihood phylogenetic trees (heuristic search with BIONJ starting tree, SPR and NNI swapping and 100 bootstrap replicates). Majority-rule extended consensus trees were computed from 100 bootstrap replicates by the CONSENSE program of the PHYLIP package (Felsenstein, 2005), using the default parameters in the rooting setting. The two consensus trees are shown in Figure 3.

Two kinds of inconsistencies were found. One involves *Ae. speltoides* and the ancestor of *A. bicornis*, *Ae. longissima* and *Ae. sharonensis* (hereafter referred to as the BLS species). *Ae. speltoides* has a basal position in both trees, but in the matK tree it is isolated, whereas in the PinA gene it is very close to the BLS. In the matK tree, the BLS species are close to species that radiated more recently. This deep phylogenetic inconsistency could be a consequence of a hybridization between *A. speltoides* and an ancestor of BLS species that introgressed into portions of *Ae. speltoides* genome. This hypothesis is supported by the observation of similar inconsistencies involving *Ae. speltoides* and *Ae. longissima* in a larger sample of nuclear genes (Escobar *et al*., 2011). Note that this hypothesis is present in all three hybridization networks obtained by Dendroscope 3 from the two consensus trees (see Figure 4). The other inconsistency involves the relative position *Ae. tauschii* and the ancestors of two groups of species, i.e., *Ae. uniaristata* + *Ae. comosa* and the BLS species. Three alternative hybridizations could be inferred (see Figure 4). However, this inconsistency is more recent and could also be explained by an incomplete allele sorting (that is, allelic variants coexisting in the common ancestor of all these species were randomly fixed in the derived species).

5 DISCUSSION AND ACKNOWLEDGEMENTS

It has become standard practice to base evolutionary studies on multiple genes. When incongruences between different gene trees are small, then they are usually deemed insignificant and are dealt with by performing a consensus analysis. However, when the differences between the gene trees are more significant, and when mechanisms such as hybridization may have played an important role in the evolutionary history of a set of species, then an alternative approach may be to try to reconcile the different gene trees by combining them into a rooted phylogenetic network in which reticulation nodes represent possible hybridization events.

While some papers in the literature have focused on reconciling incongruent gene phylogenies in terms of a network (Koblmuller *et al*., 2007), a major problem has been the lack of software implementing an algorithm for computing and investigating such networks. In this paper, we address the need for such software by providing an algorithm that runs fast on most practical problems and produces a representative set of minimum hybridization networks, containing one network for each MAAF. Our algorithm is implemented in the program Dendroscope 3, which allows the user to visualize and compare the resulting networks, both among each other, and also with the original input trees.

Our simulation study shows that our implementation is faster than existing implementations and the study of grasses, reported in Section 4, shows how one may use the software to obtain different possible hybridization scenarios.

The authors would like to thank Simone Linz for helpful discussions.

REFERENCES

Baroni, M., Grüenewald, S., Moulton, V., and Semple, C. (2005). Bounding the number of hybridization events for a consistent evolutionary history. *Mathematical Biology*, **51**, 171–182.

Baroni, M., Semple, C., and Steel, M. (2006). Hybrids in real time. *Systematic Biology*, **55**(1), 46–56.

Bordewich, M. and Semple, C. (2005). On the computational complexity of the rooted subtree prune and regraft distance. *Annals of Combinatorics*, **8**(4), 409–423.

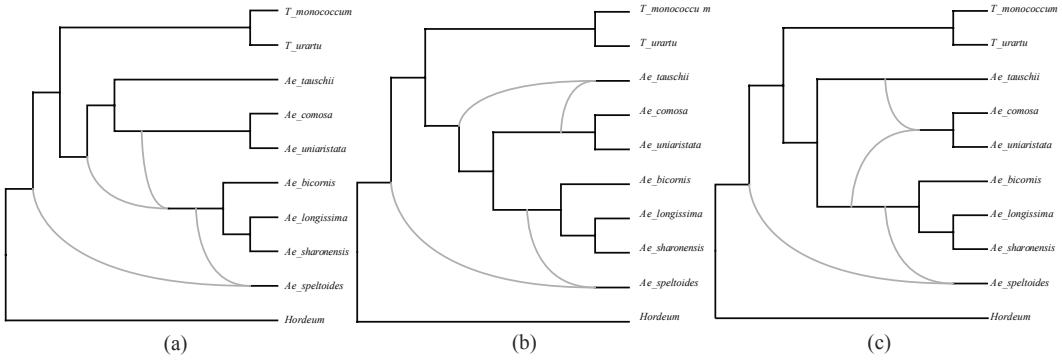


Fig. 4. The three hybridization networks obtained by the described algorithm for the matK and PinA consensus trees of Figure 3.

Bordewich, M. and Semple, C. (2007a). Computing the hybridization number of two phylogenetic trees is fixed-parameter tractable. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **4**(3), 458–466.

Bordewich, M. and Semple, C. (2007b). Computing the minimum number of hybridisation events for a consistent evolutionary history. *Discrete Applied Mathematics*, **155**(8), 914–928.

Chen, Z.-Z. and Wang, L. (2010). HybridNET: a tool for constructing hybridization networks. *Bioinformatics*, **26**(22), 2912–2913.

Chen, Z.-Z. and Wang, L. (2011). Algorithms for reticulate networks of multiple phylogenetic trees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **99**(PrePrints).

Collins, J., Linz, S., and Semple, C. (2011). Quantifying hybridization in realistic time. *Journal of Computational Biology*. Ahead of print. doi:10.1089/cmb.2009.0166.

Escobar, J., Scornavacca, C., Cenci, A., Guilhaumon, C., Santoni, S., Douzery, E., Ranwez, V., Glémin, S., and David, J. (2011). Multigenic phylogeny and analysis of tree incongruences in Triticeae (Poaceae). *BMC Evolutionary Biology*, **11**, 181.

Felsenstein, J. (2005). Phylip (phylogeny inference package) version 3.6. Distributed by the author. Department of Genome Sciences, University of Washington, Seattle.

Giraud, T., Refregier, G., Le Gac, M., de Vienne, D., and Hood, M. (2008). Speciation in fungi. *Fungal Genet. Biol.*, **45**, 791–802.

Gross, B. and Rieseberg, L. (2005). The ecological genetics of homoploid hybrid speciation. *J. Hered*, **96**, 241–252.

Guindon, S., Dufayard, J., Lefort, V., Anisimova, M., Hordijk, W., and Gascuel, O. (2010). New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of PhyML 3.0. *Systematic Biology*, **59**(3), 307–321.

Hein, J., Jiang, T., Wang, L., and Zhang, K. (1996). On the complexity of comparing evolutionary trees. *Discrete Applied Mathematics*, **71**(1-3), 153–169.

Huson, D. and Scornavacca, C. (2011). Dendroscope 3 - a program for computing and drawing rooted phylogenetic trees and networks. In preparation, software available from: `www.dendroscope.org`.

Huson, D. H., Rupp, R., and Scornavacca, C. (2011). *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press.

Kellogg, E., Appels, R., and Mason-Gamer, R. (1996). When genes tell different stories: the diploid genera of Triticeae (Gramineae). *Syst Bot*, **23**, 321347.

Koblmuller, S., Duftner, N., Sefc, K., Aibara, M., Stipacek, M., Blanc, M., Egger, B., and Sturmbauer, C. (2007). Reticulate phylogeny of gastropod-shell-breeding cichlids from Lake Tanganyika - the result of repeated introgressive hybridization. *BMC Evolutionary Biology*, **7**(1), 7.

Linz, S. and Semple, C. (2010). A cluster reduction for computing the subtree distance between phylogenies. In press, Annals of Combinatorics.

Mallet, J. (2007). Hybridization, introgression, and linkage evolution. *Nature*, **446**, 279–283.

Mason-Gamer, R. and Kellogg, E. (1996). Testing for phylogenetic conflict among molecular data sets in the tribe Triticeae (Gramineae). *Syst Biol*, **45**, 2538.

Posada, D. (2008). jModelTest: Phylogenetic model averaging. *Molecular Biology and Evolution*, **25**, 12531256.

Rieseberg, L. H., Baird, S. J. E., and Gardner, K. A. (2000). Hybridization, introgression, and linkage evolution. *Plant Molecular Biology*, **42**, 205–224.

Sasanuma, T., Chabane, K., Endo, T., and Valkoun, J. (2004). Characterization of genetic variation in and phylogenetic relationships among diploid aegilops species by AFLP: incongruity of chloroplast and nuclear data. *Theor Appl Genet*, **108**, 612–618.

Schwenk, K., Brede, N., and Streit, B. (2008). Introduction. extent, processes and evolutionary impact of interspecific hybridization in animals. *Philos. Trans. R. Soc. Lond., Ser. B: Biol. Sci.*, **363**, 2805–2811.

Scornavacca, C., Linz, S., and Albrecht, B. (2011). A first step toward computing all hybridization networks for two rooted binary phylogenetic trees. In preparation.

Soltis, P. and Soltis, D. (2009). The role of hybridization in plant speciation,. *Annu. Rev. Plant Biol.*, **60**, 561–588.

Whidden, C. and Zeh, N. (2009). A unifying view on approximation and FPT of agreement forests. In *Proceedings of WABI'09*, pages 390–402, Berlin, Heidelberg. Springer-Verlag.

Whidden, C., Beiko, R. G., and Zeh, N. (2010). Fast FPT algorithms for computing rooted agreement forests: Theory and experiments. In P. Festa, editor, *SEA*, volume 6049 of *Lecture Notes in Computer Science*, pages 141–153. Springer.