Reducing Distortion in Phylogenetic Networks

Daniel H. Huson[1], Mike A. Steel[2], and Jim Whitfield[3]

[1] Center for Bioinformatics (ZBIT), Tübingen University, Germany
huson@informatik.uni-tuebingen.de
[2] Allan Wilson Centre, University of Canterbury, Christchurch, New Zealand
m.steel@math.canterbury.ac.nz
[3] Department of Entomology, University of Illinois at Urbana-Champaign, USA
jwhitfie@life.uiuc.edu

Abstract. When multiple genes are used in a phylogenetic study, the result is often a collection of incompatible trees. Phylogenetic networks and super-networks can be employed to analyze and visualize the incompatible signals in such a data set. In many situations, it is important to have control over the amount of imcompatibility that is represented in a phylogenetic network, for example reducing noise by removing splits that do not recur among the source trees. Current algorithms for computing hybridization networks from trees are based on a combinatorial analysis of the arising set of splits, and are thus sensitive to false positive splits. Here, a filter is desirable that can identify and remove splits that are not compatible with a hybridization scenario. To address these issues, the concept of the distortion of a tree relative to a split is defined as a measure of how much the tree needs to be modified in order to accommodate the split, and some of its properties are investigated. We demonstrate the usefulness of the approach by recovering a plausible hybridization scenario for buttercups from a pair of gene trees that cannot be obtained by existing methods. In a second example, a set of seven gene trees from microgastrine braconid wasps is investigated using filtered networks. A user-friendly implementation of the method is provided as a plug-in for the program SplitsTree4.

1 Introduction

In systematics, the evolution of different species is of interest, however, phylogenetic inference is often based on the DNA or protein sequence of homologous genes and the resulting *gene trees* are usually interpreted as estimations of an underlying *species tree*. A common observation is that different genes give rise to different trees, even in the absence of tree-reconstruction errors, and this fact can usually be explained by mechanisms such as incomplete lineage sorting, duplication-and-loss, horizontal gene transfer (e.g. in bacteria) or hybridization (e.g. in plants).

Although phylogenies based on single gene analysis [32] continue to play a central role in phylogenetics, biologists interested in the evolution of specific groups of taxa often sequence and use more than one gene to infer the phylogeny

P. Bücher and B.M.E. Moret (Eds.): WABI 2006, LNBI 4175, pp. 150–161, 2006.

of the taxa [23], the hope being that as more data is brought into the analysis, a better "species-signal" to "gene-noise" ratio will be obtained and that deviating signals from individual genes can be filtered out.

If the goal is simply to obtain a good estimation of the species tree and if there is evidence that a majority of the genes under study have evolved in a similar way along the same species tree, then one approach is to concatenate the alignments given for each of the genes to produce one large dataset, to which tree-building methods are then applied [23,25]. If each of the genes is long enough to contain strong phylogenetic signals for the group of taxa under investigation, then a second approach is to compute individual gene trees, to summarize them using a (usually somewhat unresolved) consensus tree and then to interpret the consensus as a representation of the well-supported parts of the species tree [30,10,26].

In both cases, the final result suppresses all incompatible signals. However, if the actual incongruencies of the individual gene trees are themselves of interest, then a representation of the data set that maintains (some of) the incompatible signals may be useful. Such a representation is given by the concept of a "split network" [1] and methods for computing such networks are presented in [8] and are implemented in the program SplitsTree4 [15].

To obtain an explicit model of reticulate evolution, reticulate networks are used [15] that explain a given set of trees in terms of hybridization, horizontal gene transfer or recombination events [13,7,19,17,18]. Current methods for determining a hybridization scenario that explains a given set of trees operate by performing a combinatorial analysis of the total set of splits of the trees to identify a hybridization network that generates the trees [22,17]. By definition, combinatorial methods are very sensitive to false positive splits, that is, splits that are incompatible to other splits in the input due to reasons such as homoplasy, tree-estimation error, incomplete lineage sorting etc.

Given a collection (or *profile*) P of k gene trees all inferred on the same set of taxa X, one approach to constructing a set of splits that summarize the set of trees, without eliminating all incompatibilities, is given by the consensus network method [2,14]. This method consists of returning all splits that occur in at least αk of the given input trees, for a given threshold $\alpha \in [0, 1]$.

A main drawback of the consensus network approach is that in practice typical data sets often consist of *partial trees*, that is, gene trees that each only mention some subset X' of the total taxon set X. Partial trees arise because the sequence data for some gene has not yet been sequenced, or because the gene is not present in the genome, for some taxon.

Given a profile of partial gene trees, the Z-closure method [16] computes a *super network* on the full taxon X that summarizes all the input trees. This approach first uses an inference rule to construct a set of splits on the full taxon set and then, as above, a network construction algorithm [8] is employed to obtain a split network. A practical weakness of this method is that it does not provide a natural parameter (such as α above) with which one can control the amount of incompatibility that is represented in the resulting network.

The goal of this paper is to develop an adjustable parameter than can be used with any super network method or consensus method to generate split networks that represent a controlled amount of incompatible signals. The approach that we take is to filter splits by the amount of "distortion" that they generate. We have implemented this approach as a plug-in `FilteredSuperNetwork` for the SplitsTree4 program [15].

This concept is particularly useful in the context of computing hybridization networks from gene trees, because it can be used to remove splits from a data set that are not compatible with a simple hybridization scenario. This is due to the fact that the distortion of a split equals the number of SPR or TBR operations required to modify a tree to accommodate the split, which will be small for incompatibilities caused by hybridization.

We illustrate this use of a distortion filter for a set of 46 *Ranunculus* (buttercup) species, represented by two gene trees, one based on a chloroplast *JSA* region, and the other based on a nuclear *ITS* region [20]. Although this dataset is known to contain examples of both allopolyploid and diploid hybridization events (Pete Lockhart, personal communication), past attempts to compute a corresponding hybridization network from the two trees have failed [17]. Here we demonstrate that a plausible hybridization network can be computed when employing an appropriate distortion filter.

A second example is given by a set of seven gene trees for 45 species of wasps [3]. Mixed-model Baysian analysis [24] of the combined data set indicates that there is little support for internal edges of the phylogeny and here we show how filtered network methods can be used to investigate whether this lack of support is due to conflict between the different gene trees, or whether it represents a lack of real coherent signal in the data.

In the following Section 2 we provide the necessary formal definitions, and then introduce the concept of distortion and explore some of its properties. Then, in Section 3, we present an algorithm for efficiently computing the distortion of a tree relative to a split. Finally, in Section 4, we illustrate the application of the algorithm to two different biological data sets.

We are grateful to the Cass Field Station of the University of Canterbury, where we developed the main ideas of this paper. D.H.H. would like to thank the DFG and the Erskine Programme for funding. J.W. would like to thank the Allan Wilson Centre for sponsoring his trip to NZ, and National Science Foundation Grant DEB 0316566 for funding the generation of the wasp data. Thanks to Pete Lockhart for providing the buttercup trees and for many useful discussions.

2 The Distortion of a Tree Relative to a Split

We mostly follow the notation of [29]. By a *partial X–tree* we mean a tree $\mathcal{T}$ together with a labeling map ϕ from some subset X' of X into the vertices of $\mathcal{T}$ so that each vertex of degree at most 2 receives at least one label. Given an X–split $\sigma = A|B$ we may regard σ as a map from X into $\{0, 1\}$ (where elements of A are sent to 0 and elements of B are sent to 1) and so, by restricting σ to X', we may view σ as a binary character for $\mathcal{T}$.

If $\mathcal{T}$ is a *phylogenetic tree* (that is, the only vertices of $\mathcal{T}$ labeled by X' are the leaves and these each receive exactly one label), then let $h(\mathcal{T}, \sigma)$ denote the *homoplasy score* of the binary character σ on $\mathcal{T}$, that is, the parsimony score of σ, minus 1.

For any X-split σ and partial X–tree $\mathcal{T}$, we define the *distortion of* $\mathcal{T}$ *relative to* σ as

$$\partial(\mathcal{T}, \sigma) := \min_{\mathcal{T}' \in Phy(\mathcal{T})} h(\mathcal{T}', \sigma),$$

where $Phy(\mathcal{T})$ denotes the set of *phylogenetic refinements of* $\mathcal{T}$, that is, the phylogenetic trees with the same label set as $\mathcal{T}$ and that contain all the splits of $\mathcal{T}$.

The following result provides an interpretation of the distortion as a measure of how much a tree needs to be modified in order to accommodate the split σ, see Figure 1. Recall that two commonly-used ways to transform trees are by

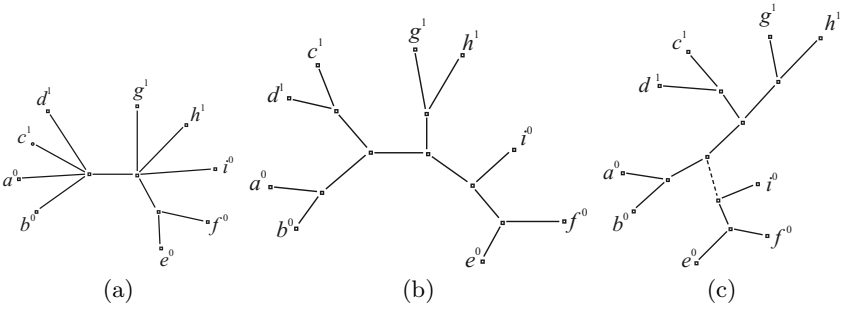

Fig. 1. (a) A tree $\mathcal{T}$ labeled by taxa $X = \{a, \ldots, i\}$, with superscript 0 or 1 indicating that the taxon lies in part A or B, of the split $\sigma = A \mid B = \{a, b, e, f, i\} \mid \{c, d, g, h\}$; we have $h(\mathcal{T}, \sigma) = 3$. (b) A refinement $\mathcal{T}'$ of $\mathcal{T}$, with $h(\mathcal{T}', \sigma) = 1$, leading to $\partial(\mathcal{T}, \sigma) = 1$. (c) $\partial(\mathcal{T}, \sigma) = h(\mathcal{T}', \sigma) = 1$ matches the transformation of $\mathcal{T}'$ into $\mathcal{T}''$ on which σ is compatible, using one SPR move.

SPR ('subtree prune and regraft') and TBR ('tree bisection and reconnection') operations, which are explained further in [29,9]. In particular, the result explains why a filter based on distortion will be a useful tool for removing false positive splits when computing a hybridization network.

Proposition 1. *For any partial X–tree $\mathcal{T}$ and X–split σ, the value $\partial(\mathcal{T}, \sigma)$ equals the smallest number of (SPR or TBR) tree rearrangement operations required to transform at least one phylogenetic refinement of $\mathcal{T}$ into a tree that has the split σ.*

Proof. The result follows from Theorem 5.2 of Bryant [6]. □

A tree $\mathcal{T}' \in Phy(\mathcal{T})$ that minimizes $h(\mathcal{T}', \sigma)$ is the optimal refinement of $\mathcal{T}$, with respect to maximum parsimony, for the binary character that corresponds to σ, in the sense of [4]. Moreover, the value of $\partial(\mathcal{T}, \sigma)$ is unaltered if one replaces in the definition the set $Phy(\mathcal{T})$ by the set of binary phylogenetic refinements of $\mathcal{T}$. Notice also that if we replace the partial X–tree $\mathcal{T}$ by its minimal phylogenetic

refinement $\mathcal{T}_p$ (i.e. the partial phylogenetic X–tree whose splits consist of the splits of $\mathcal{T}$ together with the trivial splits on the label set of $\mathcal{T}$) then we have

$$\partial(\mathcal{T}, \sigma) = \partial(\mathcal{T}_p, \sigma),$$

so it suffices to describe an algorithm for computing ∂ for partial phylogenetic trees.

The score ∂ has a dual 'max-flow' description. Let $p(\mathcal{T}, \sigma)$ denote the maximum number of vertex-disjoint paths that each connect an A–type leaf to a B–type leaf. By Menger's Theorem (see [12]) this is equal to the minimum number of vertices of $\mathcal{T}$ that need to be deleted from $\mathcal{T}$ in order to separate each A–type leaf from each B–type leaf.

Theorem 1. *For any phylogenetic X–tree and X–split, σ,*

$$\partial(\mathcal{T}, \sigma) = p(\mathcal{T}, \sigma) - 1.$$

Proof. Omitted due to space restrictions. □

Given a collection ('profile') of partial X–trees $P = \{\mathcal{T}_1, \ldots, T_k\}$ define the *distortion of P relative to σ* as follows:

$$\partial(P, \sigma) := \sum_{i=1}^{k} \partial(\mathcal{T}_i, \sigma).$$

Proposition 1 implies that $\partial(P, \sigma)$ is the minimum total number of transformations required on refinements of trees in P so that σ is a split of each resulting tree. In Section 3 we present an algorithm that efficiently computes $\partial(\mathcal{T}, \sigma)$ directly from σ and $\mathcal{T}$.

One approach to super-network construction from a profile P of partial trees would be to identify those X–splits σ for which $\partial(P, \sigma)$ is less than some (adjustable) threshold $k \geq 0$. However this problem seems in general to be intractable due to the following result.

Proposition 2. *The following problem is NP–hard. Given a profile P of partial X–trees, determine whether there exists a non-trivial X–split σ with $\partial(P, \sigma) = 0$.*

Proof. The result follows from the NP–hardness of 'Split-quartet compatibility' by [5]. □

In view of Proposition 2 an alternative approach is to use P to first construct a large set of 'feasible' X–splits, and then to use ∂ to prune this set to a more conservative subset. More concretely, we propose to first use the Z-closure algorithm to compute a set of X–splits for P and then to return all splits σ with $\partial(P, \sigma) \leq k$, for a given integer threshold $k \geq 0$.

Another option for a profile P of partial X–trees – which generalizes the consensus network approach– is, for a non-negative integer r, and real number $\alpha \in [0, 1]$ to consider the set of X–splits defined by:

$$\{\sigma : |\{\mathcal{T} \in P : \partial(\mathcal{T}, \sigma) \leq r\}| \geq \alpha |P|\}.$$

For $r = 0$ and a profile P consisting of binary phylogenetic X–trees, then using the set of all splits contained in P, this corresponds to the consensus network (with threshold α).

Proposition 2 indicates that this is a hard problem, if we do not restrict the set of splits under consideration. For partial trees, one can use the Z-closure to compute a set of candidate splits. We have implemented this approach as a plug-in for SplitsTree [15] and discuss this in detail below.

Finally, assume we are given a profile P of (non-partial) X-trees. For small values of r we can compute all possible X–splits σ with $\partial(P, \sigma) \leq r$ as follows: For each tree $\mathcal{T} \in P$, consider all $O(\binom{n-3}{r})$ possible ways of selecting up to r vertex-disjoint edges in the tree, where $n = |X|$. By placing a change on each selected edge, each such choice of edges defines a binary character σ with distortion $\partial(\mathcal{T}, \sigma) \leq r$. Return all splits whose total score over all trees does not exceed r.

3 Computation of the Distortion

Given a partial X–tree $\mathcal{T}$ and an X–split σ, the definition of $\partial(\mathcal{T}, \sigma)$ in Section 2 does not immediately lead to an algorithm. To compute this value, we describe a modification of Sankoff's algorithm [27,28] for computing the parsimony score of a character on a tree.

In the following, we will assume that $\mathcal{T}$ is a phylogenetic X'–tree, with $X' \subseteq X$. However, our algorithm is easily extended to the case that $\mathcal{T}$ is multi-labeled (i.e., has nodes labeled by more than one taxon), and has labels on (some or all) internal vertices.

Algorithm 2 (Distortion)
Input: *A phylogenetic partial X–tree $\mathcal{T}$ and an X–split σ.*
Output: *The distortion $\partial(\mathcal{T}, \sigma)$.*
Root $\mathcal{T}$ at the midpoint of an edge and let ρ denote the root vertex.
Initialization: For all vertices v and all $a \in \{0, 1\}$ set:

$$S_v(a) = \begin{cases} 0, & \text{if } a = 0 \text{ and } \phi^{-1}(v) \subseteq A, \text{ or } a = 1 \text{ and } \phi^{-1}(v) \subseteq B \\ \infty, & \text{if } a = 0 \text{ and } \emptyset \neq \phi^{-1}(v) \subseteq B, \text{ or } a = 1 \text{ and } \emptyset \neq \phi^{-1}(v) \subseteq A. \end{cases}$$

Compute S_ρ using the following recursion:
For $a, b \in \{0, 1\}$, and a vertex v with children $w_1, \ldots, w_k$, set

$$S_v(a) = \sum_{w_i : S_{w_i}(a) < S_{w_i}(b)+1} S_{w_i}(a) + \sum_{w_i : S_{w_i}(a) \geq S_{w_i}(b)+1} S_{w_i}(b) + \Delta,$$

where

$$\Delta = \begin{cases} 1, & \text{if there exists } w_i : S_{w_i}(a) \geq S_{w_i}(b) + 1; \\ 0, & \text{otherwise.} \end{cases}$$

The result is given by $\partial(\mathcal{T}, \sigma) = \min\{S_\rho(0), S_\rho(1)\} - 1$.

Proposition 3. *Let $\mathcal{T}$ be a partial phylogenetic X–tree and $\sigma = A \mid B$ be an X-split. Algorithm 2 computes the distortion $\partial(\mathcal{T}, \sigma)$ in linear time.*

Proof. The algorithm considers each parent-child pair of vertices exactly once, and hence the time requirement is linear.

We will prove the result by induction. First, consider the initialization step. The map $S_v(0)$ is set to 0 for every internal vertex v, and otherwise to 0 or ∞, depending on whether the label of the leaf v lies in A or B, respectively. Vice-versa for $S_v(1)$.

Now, consider a vertex v and assume by induction that we have correctly computed $S_{w_i}(a)$ for all children $W = \{w_1, w_2, \ldots, w_k\}$ of v and all $a \in \{0, 1\}$.

Define $W_0 := \{w_i \in W \mid S_{w_i}(0) < S_{w_i}(1)+1\}$ and $W_1 := \{w_i \in W \mid S_{w_i}(0) \geq S_{w_i}(1) + 1\}$.

To compute $S_v(0)$, consider a refinement $\mathcal{T}'$ of $\mathcal{T}$ such that v has one or two out-edges (depending on whether one or both of the sets W_A and W_B are non-empty), $e_0 = (v, u_0)$ and $e_1 = (v, u_1)$, leading to one or two subtrees containing the sets W_0 and W_1, respectively. We choose state 0 and state 1 on the nodes $W_0 \cup \{u_0\}$ and $W_1 \cup \{u_1\}$, respectively, and pay a penalty of 1 for a change along edge e_1, if $W_1 \neq \emptyset$. Note that the degree of u_0 or u_1 may be 2, which we allow for purposes of the proof, as this does not alter the achievable score. We compute $S_v(1)$ in a similar manner. □

4 Implementation and Applications

We have implemented the above ideas as a new plug-in `FilteredSuperNetwork` for the program SplitsTree4 [15]. This method takes as input a profile P of (partial) X-trees and produces as output a filtered set of X–splits Σ. These splits can then visualized as a split network using the algorithm described in [8], or used to compute a hybridization network, using the algorithm described in [17].

The method proceeds by first computing the Z-closure Σ' of all partial X–splits in P and then computing the profile score of every split $\sigma \in \Sigma'$. The user must provide two parameters. The first parameter, `maxDistortion`, determines the maximal distortion $\partial(\mathcal{T}, \sigma)$ acceptable to consider $\sigma \in \Sigma'$ as being *supported* by the tree $\mathcal{T} \in P$. The second parameter, `minSupportingTrees`, determines the minimum number of trees $\mathcal{T} \in P$ that are required to support σ so that σ is present in the set of output splits Σ. Either parameter can be set by a slider that is coupled to a histogram that shows how many splits will be present in the output for any given choice of the parameter, given the current value of the other parameter.

As mentioned above, an important application of the distortion filter is as a preprocessing step in the computation of hybridization networks [17]. Given a set of gene trees that show significant incongruencies due to hybridization events, the goal here is to compute a hybridization network that "explains" the gene trees. Existing approaches perform a combinatorial analysis of the set of trees or splits to derive a network, and thus are very sensitive to false positive splits in the data set. If the underlying hybridization scenario is relatively simple, e.g.

involving only isolated events, then the distortion filter can be used to remove interfering splits.

For example, consider the set $P = \{\mathcal{T}_1, \mathcal{T}_2\}$ of two gene trees on 46 *Ranunculus* (buttercup) species depicted in Figure 2, based on (a) a chloroplast *JSA* region, and (b) a nuclear *ITS* region [20]. The split network representing the set Σ of all splits from either tree is shown in Figure 2(c). Although this dataset

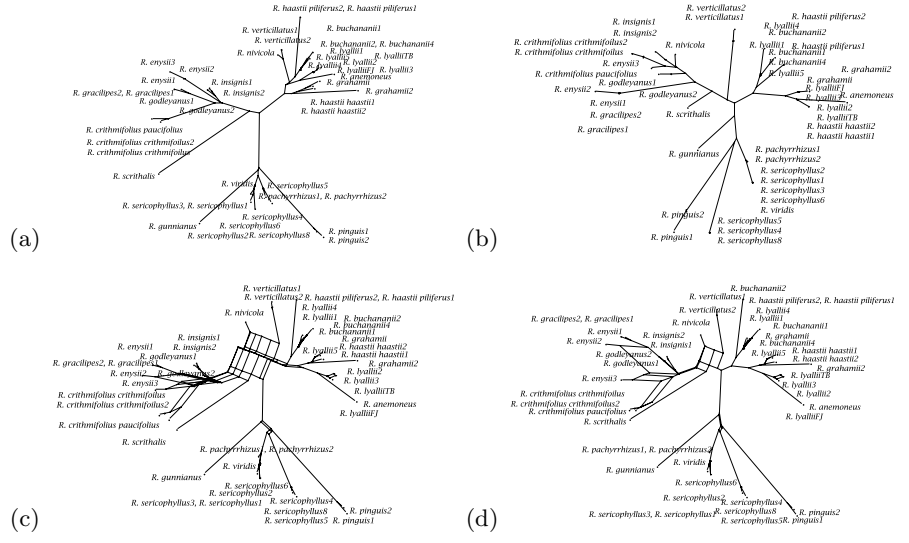

Fig. 2. Two phylogenetic trees for 46 buttercup species, obtained (a) using a nuclear ITS gene and (b) using a chloroplast JSA region [20]. (c) A split network displaying all splits contained in the two trees: (d) The split network for those splits whose distortion is at most 1 on each of the two trees.

is known to contain examples of both allopolyploid and diploid hybridization events (Pete Lockhart, personal communication), previous attempts to compute a corresponding hybridization network from the two trees have failed [17], due to interfering splits.

For this dataset, it makes sense to apply the distortion filter to obtain the set

$$\Sigma' = \{\sigma \in \Sigma \mid \partial(\mathcal{T}, \sigma) \leq 1, \forall \mathcal{T} \in P\},$$

as this consists of every split that is contained in one of the trees, and is also contained in the other, or in a tree that differs by one tree rearrangement from the other. Figure 2(d) shows the split network for the reduced data set Σ'.

Application of a hybridization network algorithm [17] produces the network depicted in Figure 3. The network clearly indicates that *R. nivicola* arises as a (allopolyploid) hybrid between *R. insignis*, and *R. verticallatus*. Moreover, the network indicates two further possible hybridization events, one leading to *R. enysii3* (as this involves a single lineage, probably diploid hybridization), and the other leading to *R. pinguis*.

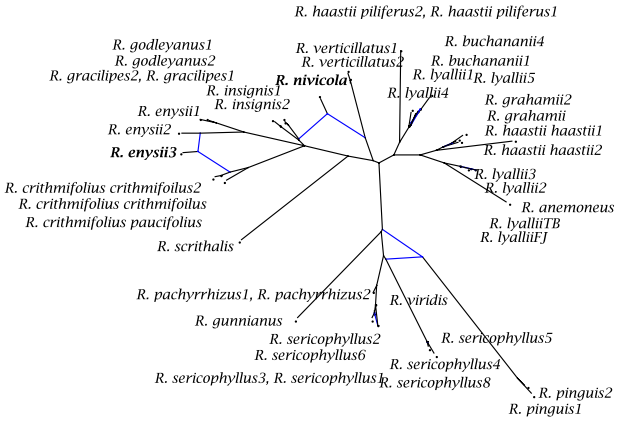

Fig. 3. The hybridization network computed from the filtered set of splits

We now discuss a second example that derives from a study of the phylogeny of microgastrine braconid wasps, a diverse and terrestrially ubiquitous group of small insects that live parasitically as immatures within the bodies of caterpillars. This insect group has been proposed to have diversified rapidly about 50 million years ago into what are now recognized as the modern genera [21,31,3]. At about this time their host insects, and the plants they live upon, were also strongly diversifying [11].

Recent work [3] presents DNA sequence data from seven genes for a set of 45 species of wasps representing a number of microgastrine genera and related subfamilies of wasps. In most cases not all species were successfully sequenced; as many as six (and as few as zero) of the species were missing from a gene tree. Mixed-model Bayesian analysis [24] of the combined seven-gene data set resolved most phylogenetic relationships at the species level (external edges) and among wasp subfamilies (deeply internal edges connecting the ingroup to outgroups), but showed short and relatively poorly-supported internal edges subtending many of the combinations of wasp genera. The internal relationships among wasp genera approximate a "star phylogeny".

It was thus of interest to investigate via filtered network methods whether this star phylogeny pattern is due to conflict between splits supported by different sets of data, or whether it represents a real lack of a coherent signal in data patterns (splits).

We consider seven unrooted, multifurcating gene trees as independently analyzed using Bayesian analysis (GTR + I + Γ substitution model for the two mtDNA genes 16S and COI, HKY + I + Γ for the nuclear genes EF1α, LW rhodopsin, *wingless*, 28S and argK). The five nuclear genes are widely believed to provide stronger phylogenetic signal for deeper relationships than the two mtDNA genes, which are more widely employed for inference of close species relationships.

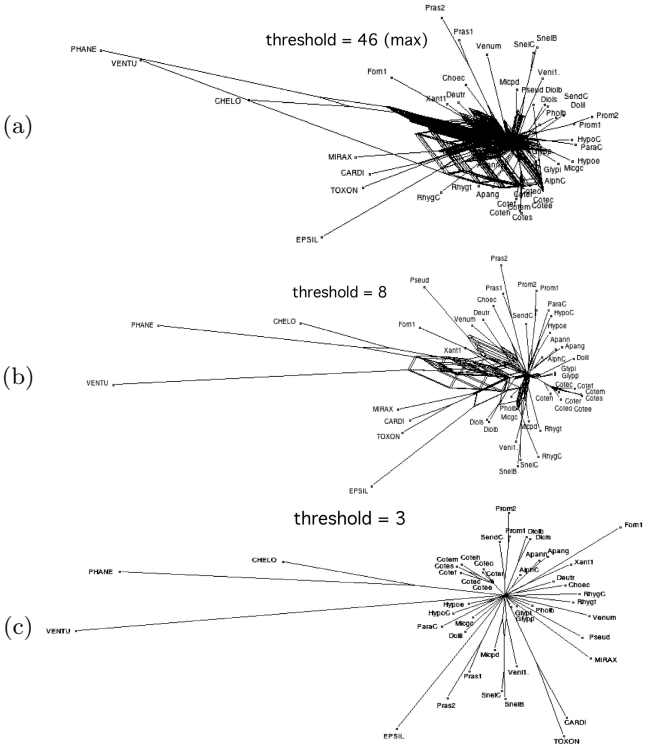

Fig. 4. Effect on Z-closure network of reducing the distortion threshold. (a) With all splits included (distortion threshold 46), (b) with distortion threshold 8, and (c) with distortion threshold 3.

This prediction seems to be borne out by the more distantly related outgroups (shown as all capitals in the taxon labels) more often being erroneously connected closely to ingroups in the mtDNA trees, along with stronger diversification among species within genera (longer edges near the periphery of the tree) than is shown with the more slowly-evolving nuclear genes.

Figure 4 shows the effect of first applying the Z-closure method for combining partial trees into a super-network (a), and then using the distortion filter with different thresholds. When all splits (threshold = 46, encompassing 213 splits) are allowed to contribute to the super-network, the result is a tangled mess, Figure 4(a). Reducing the threshold to 8, Figure 4(b), results in a clear simplification of the network (124 splits), with most of the remaining multidimensionality deriving from uncertainty in relationships between outgroups and the near star-phylogeny of ingroup generic relationships. Reducing the threshold to 3, Figure 4(c), results in only a single remaining reticulation (from 66 splits); at lower values even this uncertainty disappears.

References

1. H.-J. Bandelt and A.W.M. Dress. A canonical decomposition theory for metrics on a finite set. *Advances in Mathematics*, 92:47–105, 1992.
2. H.-J. Bandelt, P. Forster, B.C. Sykes, and M.B. Richards. Mitochondrial portraits of human population using median networks. *Genetics*, 141:743–753, 1995.
3. J.C. Banks and J.B. Whitfield. Dissecting the ancient rapid radiation of microgastrine wasp genera using additional nuclear genes. *Molecular Phylogenetics and Evolution*, in press, 2006.
4. M. Bonet, M.A. Steel, T. Warnow, and S. Yooseph. Better methods for solving parsimony and compatibility, 1998. Proc. RECOMB'98.
5. D. Bryant. Hunting for trees, building trees and comparing trees: theory and method in phylogenetic analysis. Ph.D. thesis, Dept. Mathematics, University of Canterbury, 1997.
6. D. Bryant. The splits in the neighbourhood of a tree. *Annals of Combinatorics*, 8(1):1–11, 1997.
7. S. Eddhu D. Gusfield and C. Langley. The fine structure of galls in phylogenetic networks. to appear in: INFORMS J. of Computing Special Issue on Computational Biology, 2004.
8. A.W.M. Dress and D.H. Huson. Constructing splits graphs. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, 1(3):109–115, 2004.
9. J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Inc., 2004.
10. S. Gadagkar, M.S. Rosenberg, and S. Kumar. Inferring species phylogenies from multiple genes: concatenated sequence tree versus consensus gene tree. *J. of Experimental Zoology (Mol. Dev. Evol.)*, 304B:64–74, 2005.
11. D. Grimaldi. The co-radiations of pollinating insects and angiosperms in the Cretaceous. *Ann. Missouri Bot. Garden*, 86:373–406, 1999.
12. F. Harary. *Graph Theory*. Series in Mathematics. Addison-Wesley, Reading MA, 1969.
13. J. Hein. Reconstructing evolution of sequences subject to recombination using parsimony. *Math. Biosci.*, pages 185–200, 1990.
14. B. Holland and V. Moulton. Consensus networks: A method for visualizing incompatibilities in collections of trees. In G. Benson and R. Page, editors, *Proc. 3rd Workshop on Algorithms in Bioinformatics WABI'03*, volume 2812 of *LNBI*, pages 165–176. Springer, 2003.
15. D.H. Huson and D. Bryant. Application of phylogenetic networks in evolutionary studies. *Molecular Biology and Evolution*, 23:254–267, 2006. Software available from `www.splitstree.org`.
16. D.H. Huson, T. Dezulian, T. Kloepper, and M.A. Steel. Phylogenetic supernetworks from partial trees. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, 1(4):151–158, 2004.
17. D.H. Huson, T. Kloepper, P.J. Lockhart, and M.A. Steel. Reconstruction of reticulate networks from gene trees. In *Proc. 9th Int'l Conf. on Research in Computational Molecular Biology RECOMB'05*, 2005.
18. D.H. Huson and T.H. Kloepper. Computing recombination networks from binary sequences. *Bioinformatics*, 21(suppl. 2):ii159–ii165, 2005.
19. C.R. Linder and L.H. Rieseberg. Reconstructing patterns of reticulate evolution in plants. *Am. J. Bot.*, 91(10):1700–1708, 2004.
20. P.J. Lockhart, P.A. McLenachan, D. Havell, D. Glenny, D.H. Huson, and U. Jensen. Phylogeny, dispersal and radiation of New Zealand alpine buttercups: molecular evidence under split decomposition. *Ann. Missouri Bot. Garden*, 88:458–477, 2001.

21. P. Mardulyn and J.B. Whitfield. Phylogenetic signal in the COI, 16S and 28S genes for inferring relationships among genera of Microgastrinae (Hymenoptera: Braconidae); evidence of a high diversification rate in this group of parasitoids. *Molecular Phylogenetics and Evolution*, 12:282–294, 1999.
22. L. Nakhleh, T. Warnow, and C.R. Linder. Reconstructing reticulate evolution in species - theory and practice. In *Proc. 8th Int'l Conf. on Research in Computational Molecular Biology RECOMB'04*, pages 337–346, 2004.
23. A. Rokas, B.L. Williams, N. King, and S.B. Carroll. Genome-scale approaches to resolving incongruence in molecular phylogenies. *Nature*, 425:798–804, 2003.
24. F. Ronquist and J.P. Huelsenbeck. Mrbayes3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19:1572–1574, 2003.
25. N.A. Rosenberg. The probability of topological concordance of gene trees and species trees. *Theor. Pop. Biol.*, 61:225–247, 2002.
26. M.J. Sanderson. and A.C. Driskell. The challenge of constructing large phylogenetic trees. *Trends in Plant Sciences*, 8:374–379, 2003.
27. D. Sankoff. Minimal mutation trees of sequences. *SIAM J. of Applied Mathematics*, pages 35–42, 1975.
28. D. Sankoff and P. Rousseau. Locating the vertices of a Steiner tree in an arbitrary metric space. *Mathematical Programming*, 9:240–246, 1975.
29. C. Semple and M.A. Steel. *Phylogenetics*. Oxford University Press, 2003.
30. D.L. Swofford. When are phylogeny estimates from molecular and morphological data incongruent? In M.M. Miyamoto and J. Cracraft, editors, *Phylogenetic Analysis of DNA Sequences*, pages 295–333. Oxford University Press, Oxford UK, 1991.
31. J.B. Whitfield. Estimating the age of the polydnavirus/braconid wasp symbiosis. *Proc. of the National Academy of Sciences USA*, 99:7508–7513, 2002.
32. C.R. Woese. Bacterial evolution. *Microbiol. Rev.*, 51:221–272, 1987.