Running head: MPR PROBLEM UNDER ILS AND HYBRIDIZATION IS NP-HARD

Research Article:

The Most Parsimonious Reconciliation Problem in the Presence of Incomplete Lineage Sorting and Hybridization is NP-Hard

Matthew LeMay¹,

Yi-Chieh Wu²,

Ran Libeskind-Hadas²

1. Department of Mathematics, Harvey Mudd College, Claremont, California 91711, USA

2. Department of Computer Science, Harvey Mudd College, Claremont, California 91711, USA

* Corresponding author:

Yi-Chieh Wu

301 Platt Blvd

Claremont, California 91711

Phone: 909-621-8127

E-mail: yjw@cs.hmc.edu

Abstract: The maximum parsimony phylogenetic reconciliation problem seeks to explain incongruity between a gene phylogeny and a species phylogeny with respect to a set of evolutionary events. While the reconciliation problem is well-studied for species and gene trees subject to events such as duplication, transfer, loss, and deep coalescence, recent work has examined species phylogenies that incorporate hybridization and are thus represented by networks rather than trees. In this paper, we show that the problem of computing a maximum parsimony reconciliation for a gene tree and species network is NP-hard even when only considering deep coalescence. This result suggests that future work on maximum parsimony reconciliation for species networks should explore approximation algorithms and heuristics.

Keywords: phylogenetics, reconciliation, deep coalescence, hybridization, NP-hardness

1 Introduction

Genes evolve via several evolutionary processes operating at various evolutionary timescales. Nucleotides can mutate, and domains can recombine. Genes can be generated, lost, or replaced through gene duplication, gene loss, horizontal gene transfer, and gene conversion. Populations can diverge or combine through speciation and hybridization. In addition to these events, within a population, polymorphisms can persist across speciation events, leading to a phenomenon known as incomplete lineage sorting. Thus, the history of a set of genes may differ from the history of the species in which they evolved.

In phylogenetics, *reconciliations* attempt to explain these differences by mapping gene histories within species histories to infer the evolutionary events that shaped that gene family. The simplest and most common approach seeks a most parsimonious reconciliation (MPR) (Goodman et al. 1979, Page 1994, Maddison 1997), in which each type of event in the model has an associated non-negative cost and the objective is to find a reconciliation of minimum total cost.

The time complexity of the MPR problem depends on the events being modeled and the set of constraints being considered. For example, the lowest common ancestor mapping, which can be computed in polynomial time, solves the MPR problem when considering only duplications (Górecki and Tiuryn 2006), only duplications and losses (Górecki and Tiuryn 2006), and only deep coalescence (Wu and Zhang 2011). When considering duplications, transfers, and losses, the MPR problem can be solved in polynomial time or is NP-hard depending on whether the species tree is undated, partially dated, or fully dated, and on whether the reconciliation is constrained to be time-consistent (Ovadia et al. 2011, Tofigh et al. 2011). Similarly, depending on details of the underlying model, the MPR problem can be solved in polynomial time when considering duplications, transfers, losses, and coalescence (Stolzer et al. 2012, Chan et al. 2017), or is NP-hard when considering duplications, losses, and coalescence (Bork et al. 2017). The MPR problem is also NP-hard when simultaneously modeling the evolution of domains, genes, and species (Li and Bansal 2019).

Though the species history is often represented by a tree, several authors have recently considered reconciliations with species networks which model species hybridization. For example, several methods exist for the related problem of inferring a species network that

minimizes deep coalescence (Yu et al. 2011; 2013a;b). However, the authors did not analyze the time complexity of their algorithms. Furthermore, their approaches require searching over the space of species network topologies, in contrast to the problem considered here, in which the species network is assumed to be known. Other authors have focused on the problem of inferring MPRs between gene trees and species networks, showing that the problem can be solved in polynomial time when minimizing the duplication-transfer-loss cost (Libeskind-Hadas and Charleston 2009) or the duplication-loss cost (To and Scornavacca 2015). There is little previous work on the problem of inferring MPRs between a gene tree and species network in which incongruence is due to deep coalescence, and the question of whether this problem can be solved in polynomial time remained open.

In this paper, we show that the problem of inferring an MPR between a gene tree and species network in the presence of incomplete lineage sorting is NP-hard. Our results suggests that future work on this problem should focus on developing heuristics or approximation algorithms.

2 Definitions

We use the terms *node* and *vertex* interchangeably. A *rooted phylogenetic network* refers to a rooted directed acyclic graph with a single root with in-degree 0 and out-degree 2; additional internal nodes with either in-degree 1 and out-degree 2, called *branch nodes*, or in-degree 2 and out-degree 1, called *hybridization nodes*; and one or more leaves with in-degree 1 and out-degree 0. Edges leading to hybridization nodes are called *hybridization edges*. Given a network N , let $V(N)$ denote its node set and $E(N)$ denote its edge set. Let $L(N) \subset V(N)$ denote its leaf set, $I(N) = V(N) \setminus L(N)$ denote its set of internal nodes, and $r(N) \in I(N)$ denote its root node. For a node $v \in V(N)$, let $c(v)$ denote its set of children (the empty set if v is a leaf), let $p(v)$ denote its set of parents (the empty set if v is the root node), and, if v has a single parent, $e(v)$ denotes the edge $(p(v), v)$. The size of N , denoted by $|N|$, is equal to $|V(N)| + |E(N)|$.

Let $\leq_N$ ($<_N$) be the partial order on $V(N)$ such that $v \leq_N u$ ($v <_N u$) if and only if there exists a path in N from u to v ($v \neq u$); v is said to be *lower or equal to* (lower than) u , and v a (strict) *descendant* of u , and u a (strict) *ancestor* of v . The partial order $\geq_N$ ($>_N$) is defined analogously.

Given two nodes u and v of N such that $v \leq_N u$, a path from u to v in N is a sequence of contiguous edges from u to v in N . Note that if $u = v$, the path from u to v is empty. As there can be multiple paths between pairs of vertices in a network, let $paths_N(u, v)$ denote the set of all paths from u to v . Let $paths(N)$ denote the set of all paths in network N .

A phylogenetic tree is a phylogenetic network with no hybridization nodes; that is, a directed binary tree. In the remainder of this paper, we refer to rooted phylogenetic networks and rooted phylogenetic trees simply as *networks* and *trees*, respectively.

A *species network* S represents the evolutionary history of a set of species and a *gene tree* G represents the evolutionary history of a set of genes sampled from these species. A *leaf map* $Le: L(G) \rightarrow L(S)$ associates each leaf in the gene tree with a corresponding species from which the gene was sampled. The map need not be one-to-one nor onto. Note that gene phylogenies are assumed to be trees whereas species phylogenies may, in general, be

networks.

2.1 Reconciliations

A *reconciliation* for a given gene tree, species network, and leaf map comprises a pair of maps: The *vertex map* associates each node of G with a node of S . The *path map* associates a path in S from the node of S where $p(v)$ is mapped by the vertex map to the node of S where v is mapped by the vertex map. The vertex map must be consistent with the given leaf map and must satisfy the temporal constraints of the species map; namely if a gene node g is mapped to species node s and a child g' of g is mapped to species node s' , then s must be an ancestor of s' . The path map is required because S is a network and thus there may be multiple paths between ancestors and descendants in the network. The formal definition of reconciliations is given in Definition 2.1.

Definition 2.1 (Reconciliation). Given a gene tree G , a species network S , and a leaf map Le , a *reconciliation*¹ R for (G, S, Le) is a pair of maps (R_v, R_p) where $R_v : V(G) \rightarrow V(S)$ is a *vertex map* and $R_p : V(G) \rightarrow paths(S)$ is a *path map* subject to the following constraints:

1. If $g \in L(G)$, then $R_v(g) = Le(g)$.
2. If $g \in I(G)$, then for each $g' \in c(g)$, $R_v(g') \leq_S R_v(g)$.
3. If $g \neq r(G)$, then $R_p(g) \in paths_S(R_v(p(g)), R_v(g))$. Otherwise, $R_p(g) = \emptyset$.

Constraint 1 asserts that R_v extends the leaf map Le . Constraint 2 asserts that R_v satisfies the temporal constraints implied by S . Constraint 3 asserts that the vertex map and path map are consistent.

In a multispecies coalescent process, evolution in the species network is viewed backward in time, from the leaves toward the root. Then, given a reconciliation R , we can count the number of gene lineages “passing through” each edge e of the species network. Specifically, given edge $e \in E(S)$,

$$\mathbf{L}_R(e) = |\{g \in V(G) : e \in R_p(g)\}|$$

The number of “extra lineages” is defined to be:

$$\mathbf{XL}_R(e) = \max(0, \mathbf{L}_R(e) - 1).$$

Note, for example, that if two gene paths pass through a species edge, there is one extra lineage on that edge.

Finally, the *deep coalescence cost* of a reconciliation is the sum of extra lineages across all edges of the species network:

$$\mathbf{DC}_R = \sum_{e \in E(S)} \mathbf{XL}_R(e).$$

This value is the *reconciliation cost* in this model. Finally, we formalize these optimization and decision problems:

---

¹When explaining topological incongruence through only deep coalescence, a reconciliation is sometimes called a *coalescent history* (Elworth et al. 2019).

Figure 1. The gene tree in the NP-hardness construction. The shaded subtree G^x on the left contains a variable gadget G_i^x for each variable x_i in the 3SAT instance, and the shaded subtree G^c on the right contains a single leaf ℓ_j for each clause j in the 3SAT instance.

Problem 2.1 (Most Parsimonious Reconciliation (MPR)). Given a gene tree G , a species network S , and a leaf map Le , find a reconciliation R for (G, S, Le) such that the deep coalescence cost $\mathbf{DC}_R$ is minimized.

Problem 2.2 (Most Parsimonious Reconciliation Decision Problem (MPRD)). Given a gene tree G , a species network S , a leaf map Le , and an integer k , is there a reconciliation R for (G, S, Le) such that $\mathbf{DC}_R \leq k$?

3 NP-hardness

Theorem 3.1. *MPRD is NP-hard.*

In the proof that follows, it will be convenient to consider the gene tree as the collection of all paths P from its root to its leaves. For a given leaf map $Le : L(G) \rightarrow L(S)$, a *lineage map* with respect to Le is a mapping M from each path $p_\ell \in P$ whose endpoint is leaf ℓ to a path in S from $r(S)$ to $Le(\ell)$. Each reconciliation has a corresponding lineage map, M . Specifically, let $R = (R_v, R_p)$ be a reconciliation and let $r(G), g_1, \dots, g_k, g_k = \ell$, denote the nodes on the unique path from $r(G)$ to leaf ℓ . Then, M associates this path in G with path $R_p(g_1)R_p(g_2) \dots R_p(g_k)$ in S . Note that multiple different reconciliations may induce the same lineage map since there are, in general, different mappings of nodes of G to nodes of S than induce the same set of paths. For simplicity, when referring to lineage maps we use the notation $M(\ell)$ in lieu of $M(p_\ell)$ where p is the unique path from $r(G)$ to leaf ℓ .

Finally, we use the notation (A, B) to represent a binary tree with a root and children A and B , either of which may be leaves or trees themselves.

Proof. Our proof is by a reduction from 3SAT. Consider an instance of 3SAT with n variables and m clauses and, without loss of generality, assume that n and m are both powers of 2.

Figure 2. The species network in the NP-hardness reduction. (A) The shaded subtree S^x on the left contains a variable gadget S_i^x for each variable x_i in the 3SAT instance, and the subtree S^c on the left contains a clause gadget S_j^c for each clause j in the 3SAT instance. (B) The variable gadget on the left is shown in detail. Vertices T_i and F_i are the first vertices on the variable setting paths (indicated in bold) for variable x_i . The clause gadget on the right is shown in detail for clause j . The bold edges are from variable setting paths for the three variables in clause j . Note that the edge $U_{k,j}, V_{k,j}$ indicated on the right is the k^{th} bold edge in the clause gadget for clause j iff variable x_i is the k^{th} variable in clause j . In this example, $U_{k,j}, V_{k,j}$ appears on a true variable setting path, indicating that variable x_i appears unnegated in clause j .

(If not, the 3SAT instance can be trivially padded with dummy variables and clauses to construct an equivalent instance in polynomial time.)

Construction: The gene tree G is constructed as follows: A *gene variable gadget* for a variable x_i comprises a tree G_i^x with three leaves, labelled a_i, b_i, y_i , with the topology $((a_i, b_i), y_i)$. The root of this gadget is labelled x_i . These n variable gadgets are connected via a perfect binary tree G^x . A *gene clause gadget* for a clause c_j consists simply of a single leaf ℓ_j . These m leaves are connected via a perfect binary tree G^c . A k -comb of length k is a binary tree constructed from a path of length k ($k+1$ vertices and k edges) where each of the first k vertices on that path has two children; one is the next vertex on the path and another is a leaf. In total a k -comb gadget then has $k+1$ leaves. The root of the gene tree G has two children, each of which is the root of a $2n$ -comb, one called comb d and the other comb e . One of the two deepest leaves of comb d is the root of tree G^x and the remaining leaves are labeled $d_1, \dots, d_{2n}$ in order of depth from the root. Similarly, one of the two deepest leaves of comb e is the root of G^c and the remaining leaves are labeled $e_1, \dots, e_{2n}$ in order of depth from the root. The structure of the gene tree is depicted in Figure 1.

The species network S is constructed as follows: A *species variable gadget* for variable x_i consists of a subtree S_i^x with four children labeled T_i, A_i, B_i , and F_i , with topology $((T_i, A_i), (F_i, B_i))$. A_i and B_i are leaves. The root of this gadget is labelled X_i . T_i and F_i are the first vertices of paths which correspond to setting x_i to true or false, respectively. We henceforth refer to these paths as the *variable setting paths* for T_i and F_i , respectively. The remaining vertices on these variable settings paths are described in the next paragraph; ultimately these paths join at a hybridization node which has a single leaf child Y_i . The roots of these n variable gadgets are joined via a perfect binary tree S^x .

A *species clause gadget* S_j^c for clause C_j is constructed as follows. Let z_1, z_2 , and z_3 denote the three literals in that clause. If literal z_1 is the unnegated variable x_i then a vertex $U_{1,j}$ and its child $V_{1,j}$ are introduced on the variable setting path for F_i in the species variable gadget for x_i . Conversely, if literal z_1 is the negation of x_i , then vertex $U_{1,j}$ and its child $V_{1,j}$ are introduced on the variable setting path for T_i . The analogous process is used to introduce a pair of vertices $U_{2,j}$ and $V_{2,j}$ for the variable setting path for literal z_2 and a pair of vertices $U_{3,j}$ and $V_{3,j}$ for the variable setting path for literal z_3 . The root of the species clause gadget for clause C_j is a vertex U_j with $U_{1,j}$ as one child and U'_j as the second child whose children are $U_{2,j}$ and $U_{3,j}$. Note that $U_{1,j}, U_{2,j}$, and $U_{3,j}$ are hybridization nodes since they each have two parents. Node $V_{1,j}$ has a second child V_j , $V_{2,j}$ and $V_{3,j}$ have a child V'_j (a hybridization node), V'_j is another parent of V_j (a hybridization node) which, in turn, has a single child ℓ_j , a leaf of the species tree. The roots of the m species clause gadgets are connected with a perfect binary tree S^c .

Finally, the species variable gadget tree S^x and the clause gadget tree S^c are connected using two $2n$ -combs called D and E analogous to the d and e $2n$ -combs in the gene tree. Specifically, the root of the species tree has two children, one of which is the root of comb D and one is the root of comb E . One of the two deepest leaves of comb D is the root of the variable gadget tree S^x and the remaining leaves are labeled $D_1, \dots, D_{2n}$ in increasing depth from the root. Similarly, one of the two deepest leaves of comb E is the root of the variable gadget tree S^c and the remaining leaves are labeled $E_1, \dots, E_{2n}$ in increasing depth

from the root. A representation of the species network is shown in Figure 2.

The leaf map Le is as follows: For the leaves in the variable gadgets, $Le(a_i) = A_i$, $Le(b_i) = B_i$, and $Le(y_i) = Y_i$, for $1 \leq i \leq n$. For the leaves of the $2n$ -comb gadgets, $Le(d_i) = D_i$ and $Le(e_i) = E_i$, for $1 \leq i \leq 2n$. For the leaves in the clause gadgets, $Le(\ell_i) = L_i$, for $1 \leq i \leq m$.

Finally, the value of k in the decision problem is set to be n , the number of variables in the 3SAT instance. It is easily seen that this construction can be performed in time polynomial in the size of the 3SAT instance.

Correctness: We prove that the constructed MPRD instance has a reconciliation with deep coalescence cost no more than n if and only if there is a satisfying assignment of the variables in the given 3SAT instance.

We begin with several observations about any lineage map M corresponding to a reconciliation for the constructed instance of MPRD. The species network S has unique paths from its root to the leaves $A_1, \dots, A_n$, $B_1, \dots, B_n$, $D_1, \dots, D_{2n}$, and $E_1, \dots, E_{2n}$. Therefore, therefore, all lineage maps share the same unique paths $M(\ell)$ for all leaves ℓ among $a_1, \dots, a_n$, $b_1, \dots, b_n$, $d_1, \dots, d_{2n}$, and $e_1, \dots, e_{2n}$. The only leaves $\ell \in L(G)$ for which $M(\ell)$ has more than one possible path are $y_1, \dots, y_n$ and $\ell_1, \dots, \ell_m$.

Note that in order for a reconciliation to have cost no more than n , the induced lineage map M must satisfy the properties that $M(y_i)$ contains the node X_i , the root of the gadget S_i^x . Similarly, $M(\ell_j)$ contains the node U_j , the root of the gadget S_j^c . To see this, suppose by way of contradiction that there is a variable leaf y_i such that the path $M(y_i)$ (a path from $r(S)$ to Y_i) does not contain the vertex X_i . The only paths from $r(S)$ to Y_i that do not contain X_i are through clause gadgets, and therefore $M(y_i)$ must pass through the E comb. Since there is a unique path from $r(S)$ to A_i in S and that path does not pass through the E comb, $M(a_i)$ cannot contain nodes from the E comb. Therefore, $M(y_i)$ must diverge from $M(a_i)$ at $r(S)$, and therefore also diverges from $M(e_{2n})$ at $r(S)$ since e_{2n} is more distantly related to y_i than a_i is to y_i . But then each of the $2n$ internal nodes of the comb E has at least two lineages, one from $M(y_i)$ and one from $M(e_{2n})$, contributing a cost of at least $2n > n$, contradicting the assumed cost bound. By similar reasoning, $M(\ell_j)$ must contain U_j .

There are, therefore, only two possibilities for $M(y_i)$ – it either includes the variable setting path for T_i or the variable setting path for F_i in the variable gadget for x_i . Both of these options contribute at least one to the total cost since $M(a_i)$ and $M(b_i)$ must diverge at (or above) X_i and, since y_i is more distantly related to a_i and b_i than a_i and b_i are to one another, $M(y_i)$ must diverge from $M(a_i)$ and $M(b_i)$ at (or above) X_i . Thus, $M(y_i)$ must contribute an extra lineage on an edge shared with $M(a_i)$ or $M(b_i)$. Since there are n variables, this contributes a cost of n , so these are necessarily the only extra lineages. There are only three possible lineage maps for each $M(\ell_j)$. Each one passes through the corresponding clause gadget S_j^c , and therefore uses an edge on one of three species variable setting paths that pass through that gadget.

Now suppose there is a satisfying assignment of the variables in the 3SAT instance. Then construct a lineage map M with respect to Le as follows: $M(y_i)$ contains the variable setting path T_i if the variable x_i is set to true, and the variable setting path F_i if the variable x_i is

set to false. For a clause j , let z_k , $k \in \{1, 2, 3\}$, denote one of three literals in that clause that evaluates to true with respect to the given satisfying assignment. If z_k is an unnegated variable x_i , then, by construction, the F_i variable setting path contains the edge $(U_{k,j}, V_{k,j})$ in the clause gadget S_j^c . We then construct $M(\ell_j)$ so that it passes through the clause gadget via that edge. If z_k is a negated variable $\neg x_i$, then, by construction, the T_i variable setting path contains the edge $(U_{k,j}, V_{k,j})$ in the clause gadget S_j^c and $M(\ell_j)$ is chosen to pass through the clause gadget via that edge. A reconciliation inducing this lineage map is trivial since each vertex in the gene tree has a corresponding vertex in the species network. The only cost incurred by this reconciliation is one for each variable gadget as noted above. The total cost is therefore n and thus this is a “yes” instance of MPRD.

Conversely, suppose there is some reconciliation with cost at most n and let M be the corresponding lineage map. Then, we induce a setting of each variable x_i based on whether $M(y_i)$ contains the T_i or F_i variable setting path. As noted previously, this induces a cost of n and thus the remaining paths cannot contribute any additional cost. Therefore, for each clause C_j , $M(\ell_j)$ must pass through an otherwise unused edge $(U_{k,j}, V_{k,j})$, $k \in \{1, 2, 3\}$, implying that, by construction, the k^{th} literal in clause C_j has a setting that satisfies that clause. Therefore, the 3SAT instance is satisfied. □

4 Discussion

In this work, we have shown that the problem of inferring an MPR between a gene tree and species network in the presence of incomplete lineage sorting is NP-hard. These results suggest several important directions for future research. First, approximation algorithms and exact fixed-parameter tractable algorithms should be explored for the MPR problem. Second, the problem may be solved effectively in many instances using satisfiability solvers or integer linear programming. Third, heuristics can be explored and tested experimentally.

Acknowledgements

This work was supported by the Department of Computer Science and the Dean of Faculty of Harvey Mudd College. This material is based upon work supported by the National Science Foundation under Grant No. IIS-1751399 to YW and Grant No. IIS-1905885 to RLH. The authors thank Adam Walker for valuable comments that helped improve the paper.

References

- Bork D, Cheng R, Wang J, Sung J and Libeskind-Hadas R. 2017. On the computational complexity of the maximum parsimony reconciliation problem in the duplication-loss-coalescence model. *Algorithm Mol Biol* **12**.
- Chan Y.-b, Ranwez V and Scornavacca C. 2017. Inferring incomplete lineage sorting, duplications, transfers and losses with reconciliations. *J Theor Biol* **432**:1–13.

- Elworth R. A. L, Ogilvie H. A, Zhu J and Nakhleh L. 2019. Advances in computational methods for phylogenetic networks in the presence of hybridization. In T. Warnow, ed., *Bioinformatics and Phylogenetics: Seminal Contributions of Bernard Moret*, 317–360. Springer International Publishing, Cham.
- Goodman M, Czelusniak J, Moore G. W, Romero-Herrera A and Matsuda G. 1979. Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. *Syst Zool* **28**:132–163.
- Górecki P and Tiuryn J. 2006. Dls-trees: A model of evolutionary scenarios. *Theoret Comput Sci* **359**:378–399.
- Li L and Bansal M. S. 2019. An integrated reconciliation framework for domain, gene, and species level evolution. *IEEE/ACM Trans Comput Biol Bioinform* **16**:63–76.
- Libeskind-Hadas R and Charleston M. 2009. On the computational complexity of the reticulate cophylogeny reconstruction problem. *J Comput Biol* **16**:105–117.
- Maddison W. P. 1997. Gene trees in species trees. *Syst Biol* **46**:523–536.
- Ovadia Y, Fielder D, Conow C and Libeskind-Hadas R. 2011. The cophylogeny reconstruction problem is NP-complete. *J Comput Biol* **18**:59–65.
- Page R. D. 1994. Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas. *Syst Biol* **43**:58–77.
- Stolzer M, Lai H, Xu M, Sathaye D, Vernot B and Durand D. 2012. Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees. *Bioinformatics* **28**:409–415.
- To T.-H and Scornavacca C. 2015. Efficient algorithms for reconciling gene trees and species networks via duplication and loss events. *BMC Genomics* **16**:S6.
- Tofigh A, Hallett M and Lagergren J. 2011. Simultaneous identification of duplications and lateral gene transfers. *IEEE/ACM Trans Comput Biol Bioinform* **8**:517–535.
- Wu T and Zhang L. 2011. Structural properties of the reconciliation space and their applications in enumerating nearly-optimal reconciliations between a gene tree and a species tree. *BMC Bioinf* **12**:S7–.
- Yu Y, Barnett R. M and Nakhleh L. 2013a. Parsimonious inference of hybridization in the presence of incomplete lineage sorting. *Syst Biol* **62**:738–751.
- Yu Y, Ristic N and Nakhleh L. 2013b. Fast algorithms and heuristics for phylogenomics under ils and hybridization. *BMC Bioinformatics* **14**:S6.
- Yu Y, Than C, Degnan J. H and Nakhleh L. 2011. Coalescent histories on phylogenetic networks and detection of hybridization despite incomplete lineage sorting. *Syst Biol* **60**:138–149.