Supplementary material
Maximum Likelihood Inference of Reticulate Evolutionary Histories

Y. Yu[1,*], J. Dong[1], K. Liu[1,2], and L. Nakhleh[1,2,*]

[1]Department of Computer Science, Rice University

[2]Department of Ecology and Evolutionary Biology, Rice University

[*]Contact authors: {yy9,nakhleh}@rice.edu

Contents

1 Phylogenetic networks

In order to account for both hybridization and incomplete lineage sorting, we use the phylogenetic network model given in [17], which is described briefly below.

Definition 1 *A* phylogenetic $\mathcal{X}$-network, *or $\mathcal{X}$-network for short, Ψ is a directed, acyclic graph (DAG) with $V = \{r\} \cup V_L \cup V_T \cup V_N$, where*

- $indeg(r) = 0$ *(r is the* root *of Ψ);*
- $\forall v \in V_L$, $indeg(v) = 1$ *and* $outdeg(v) = 0$ *(V_L are the* external tree nodes*, or* leaves*, of Ψ);*
- $\forall v \in V_T$, $indeg(v) = 1$ *and* $outdeg(v) \geq 2$ *(V_T are the* internal tree nodes *of Ψ); and,*
- $\forall v \in V_N$, $indeg(v) = 2$ *and* $outdeg(v) = 1$ *(V_N are the* reticulation nodes *of Ψ),*

$E \subseteq V \times V$ are the network's edges, including reticulation edges *whose heads are reticulation nodes, and* tree edges *whose heads are tree nodes., and $\ell : V_L \rightarrow \mathcal{X}$ is the* leaf-labeling *function, which is a bijection from V_L to $\mathcal{X}$.*

We use $V(\Psi)$ and $E(\Psi)$ to denote the set of nodes and edges of phylogenetic network Ψ respectively. In addition to the topology of a phylogenetic network Ψ, each edge $b = (u, v)$ in $E(\Psi)$ has a length λ_b measured in coalescent units, which is the number of generations divided by effective population size on that branch. We use Ψ to refer to both the topology and branch lengths of the phylogenetic network.

2 Distribution of gene tree topologies

Given a phylogenetic network Ψ, the gene tree topology is a random variable whose probability mass function (pmf) was given in [3] for the case where the topology of Ψ is a tree, and in [36] for the case where the topology of Ψ is a network. We now briefly review the pmf given in [36].

We denote by Ψ_u the set of nodes that are reachable from the root of Ψ via at least one path that goes through node $u \in V(\Psi)$. Then given a phylogenetic network Ψ and a gene tree G for some locus j, a coalescent history is a function $h : V(G) \to E(\Psi)$ such that the following two conditions hold:

- if v is a leaf in G, then $h(v) = (x, y)$ where y is the leaf in Ψ with the label of the species from which the allele labeling leaf v in G is sampled;
- if v is a node in G_u, and $h(u) = (p, q)$, then $h(v) = (x, y)$ where $y \in \Psi_q$.

In Fig. 1, we show an example of all the possible coalescent histories for a given gene tree and phylogenetic network.

Given a phylogenetic network Ψ and a gene tree G for locus j, we denote by $H_\Psi(G)$ the set of all coalescent histories of G within the branches of Ψ. Then the pmf of the gene tree is given by

$$\mathbf{P}(G|\Psi, \Gamma) = \sum_{h \in H_\Psi(G)} \mathbf{P}(h|\Psi, \Gamma), \tag{1}$$

where Γ is the inheritance probabilities matrix (see the main text) and $\mathbf{P}(h|\Psi, \Gamma)$ gives the pmf of the coalescent history random variable, which can be computed as

$$\mathbf{P}(h|\Psi, \Gamma) = \frac{w(h)}{d(h)} \prod_{b \in E(\Psi)} \frac{w_b(h)}{d_b(h)} \Gamma[b, j]^{u_b(h)} p_{u_b(h)v_b(h)}(\lambda_b). \tag{2}$$

In this equation, $u_b(h)$ and $v_b(h)$ denote the number of lineages enter and exit edge b of Ψ under coalescent history h. The term $p_{u_b(h)v_b(h)}(\lambda_b)$ is the probability of $u_b(h)$ gene lineages coalescing into $v_b(h)$ during time λ_b [28]. And $w_b(h)/d_b(h)$ is the proportion of all coalescent scenarios resulting from $u_b(h) - v_b(h)$ coalescent events that agree with the topology of the gene tree [3]. This quantity without the b subscript corresponds to the root

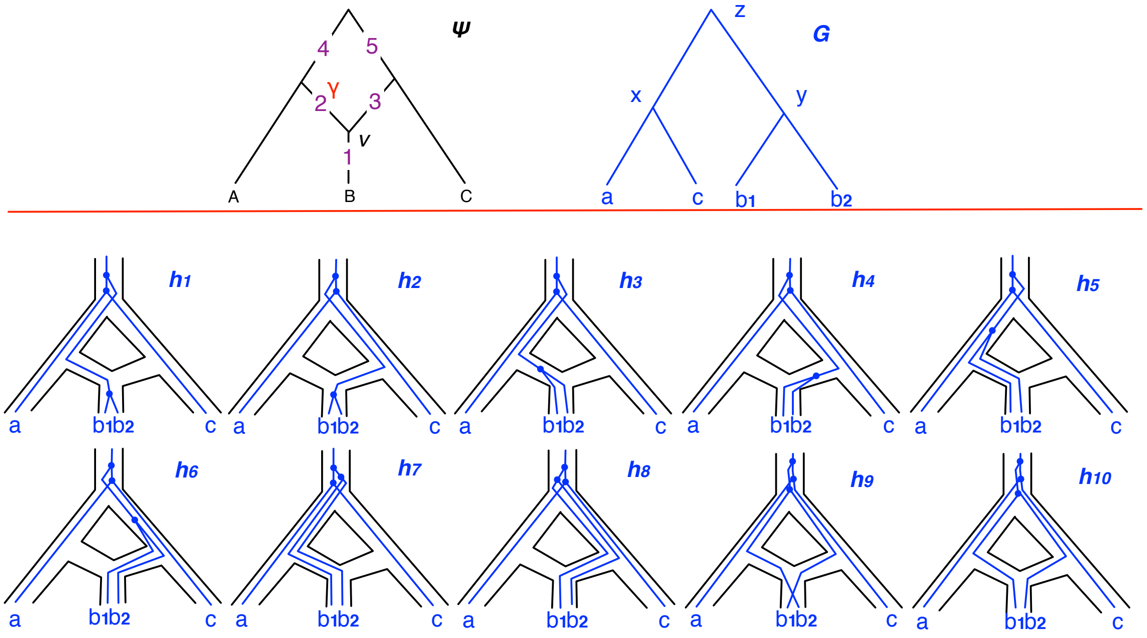


Figure 1: A phylogenetic network Ψ, a gene tree G, and the eight possible coalescent histories of G within the branches of Ψ. Here, one allele is sampled from taxa A and C, and two alleles from taxon B.

of Ψ. In Table 1, we gave an example of how Eq. 2 is computed given the phylogenetic network Ψ and G in Fig. 1.

Recently, we proposed the first method for computing $\mathbf{P}(G|\Psi,\Gamma)$ based on the concept of *MUL-tree* [36]. Basically, the phylogenetic network is first converted to a MUL-tree, and then the probability is calculated as the sum of the probabilities of observing the gene tree within the branches of the MUL-tree under all *allele mappings*. Later, we proposed another more efficient way of computing $\mathbf{P}(G|\Psi,\Gamma)$ based on the concept of *weighted ancestral configuration* [37]. It is an bottom-up algorithm working on the network Ψ directly without explicitly enumerating any coalescent history.

Table 1: The probabilities of all coalescent histories in Fig. 1. For every coalescent history h, columns from 2 to 7 list the probability of having h on every branch of the species network Ψ, where t_i is the branch length of branch i and $g_{uv}(t_i)$ is the probability of u gene lineages coalescing into v within time t_i [3]. Branch 6 corresponds to the branch incident into the root of the species network Ψ. A dash means no gene lineages enter that branch. Therefore, the total probability of a coalescent history is the product taken over all branches of the species network. In Fig. 1, coalescent events y and z can only happen above the root of Ψ. For every coalescent history, the highlighted cell shows where coalescent event x happens.

	Probability of each branch					
	1	2	3	4	5	6
h_1	$g_{21}(t_1)$	γ	$-$	$g_{22}(t_4)$	1	$\frac{1}{3}$
h_2	$g_{21}(t_1)$	$-$	$1-\gamma$	1	$g_{22}(t_5)$	$\frac{1}{3}$
h_3	$g_{22}(t_1)$	$\gamma^2 g_{21}(t_2)$	$-$	$g_{22}(t_4)$	1	$\frac{1}{3}$
h_4	$g_{22}(t_1)$	$-$	$(1-\gamma)^2 g_{21}(t_3)$	1	$g_{22}(t_5)$	$\frac{1}{3}$
h_5	$g_{22}(t_1)$	$\gamma^2 g_{22}(t_2)$	$-$	$\frac{1}{3} g_{32}(t_4)$	1	$\frac{1}{3}$
h_6	$g_{22}(t_1)$	$-$	$(1-\gamma)^2 g_{22}(t_3)$	1	$\frac{1}{3} g_{32}(t_5)$	$\frac{1}{3}$
h_7	$g_{22}(t_1)$	$\gamma^2 g_{22}(t_2)$	$-$	$g_{33}(t_4)$	1	$\frac{1}{9}$
h_8	$g_{22}(t_1)$	$-$	$(1-\gamma)^2 g_{22}(t_3)$	1	$g_{33}(t_5)$	$\frac{1}{9}$
h_9	$g_{22}(t_1)$	γ	$1-\gamma$	$g_{22}(t_4)$	$g_{22}(t_5)$	$\frac{1}{9}$
h_{10}	$g_{22}(t_1)$	γ	$1-\gamma$	$g_{22}(t_4)$	$g_{22}(t_5)$	$\frac{1}{9}$

3 Distribution of gene trees with their branch lengths

Given a species tree, the probability density function (pdf) of a gene tree with branch lengths was given in [20]. Now we propose the first method for computing this pdf when the given species phylogeny is a network. We discuss in the main text the conversion between branch lengths of gene trees in units of expected numbers of mutations and branch lengths of phylogenetic networks in coalescent units. Therefore, neither the phylogenetic network nor the gene trees have to be ultrametric in our model, unless time (in both cases) is measured in standard units (calendar time).

Given a gene tree G and a species tree Ψ (both given by their topologies and branch lengths), there is only one way of reconciling G within the branches of Ψ. However, when the species phylogeny is a network, there might be more than one reconciliation due to different paths that the gene lineages can take at reticulation nodes of Ψ when tracing them backwards in time.

We use $\tau_\Psi(v)$ to denote the height of node v in phylogeny Ψ with branch lengths λ. Given a gene tree G whose branch lengths are given by λ' and a phylogenetic network Ψ whose branch lengths are given by λ, we define a coalescent history with respect to coalescence times to be a function $h : V(G) \to E(\Psi)$, such that the following condition holds:

- for $h \in H_\Psi(G)$, if $h(v) = (x, y)$ and $\tau_\Psi(x) > \tau_G(v) \geq \tau_\Psi(y)$, then $h(v) = (x, y)$.

The quantity $\tau_G(v)$ indicates at which point of branch (x, y) coalescent event v happens. We denote the set of coalescent histories with respect to coalescence times for gene tree G and phylogenetic network Ψ by $H_\Psi(G)$. Clearly, in this case, the set H depends on λ and λ'. To illustrate this, an example is shown in Fig. 2, where the same phylogenetic network and gene tree are used as the ones in Fig. 1, but with branch lengths. We can see that there are only two coalescent histories with respect to coalescence times, h_1 and h_2, resulting from different paths b_1 and b_2 took at the reticulation node. And their corresponding coalescent histories in Fig. 1 are h_5 and h_6, respectively. It is important to note that some λ and λ' may result in $H_\Psi(G) = \emptyset$, which means G cannot be reconciled within the branches of Ψ with respect to their coalescence times.

Given a phylogenetic network Ψ, the pdf of the gene tree (topology and branch lengths) random variable is given by

$$p(G|\Psi, \Gamma) = \sum_{h \in H_\Psi(G)} \mathbf{P}(h|\Psi, \Gamma), \tag{3}$$

where $p(h|\Psi, \Gamma)$ gives the pdf of the coalescent history (with respect to coalescence times) random variable.

Let us now consider a locus j, whose gene tree is G and an arbitrary $h \in H_\Psi(G)$. For an edge $b = (x, y) \in E(\Psi)$, we define $T_b(h)$ to be a vector of the elements in the

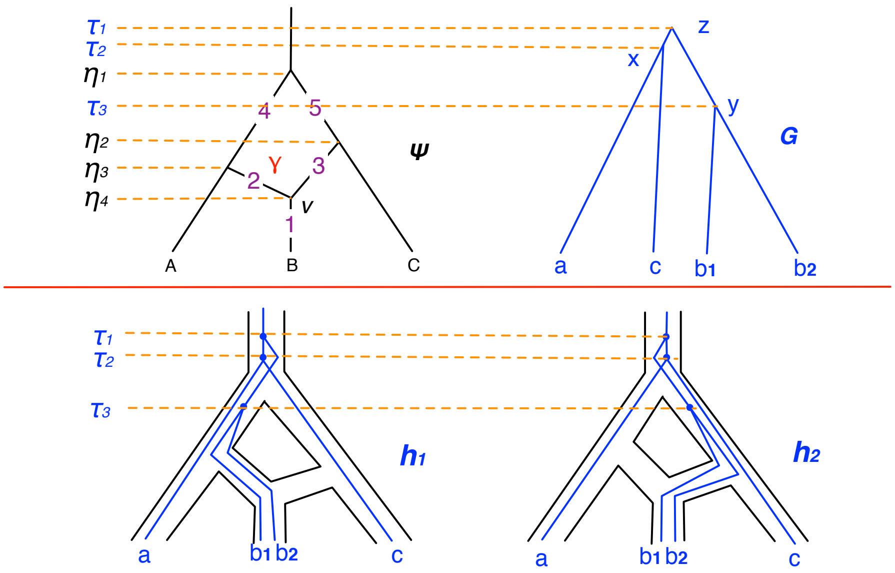


Figure 2: A phylogenetic network Ψ, a gene tree G, and the two possible coalescent histories with respect to coalescence times of G within the branches of Ψ. One allele is sampled from taxa A and C, and two alleles from taxon B. As shown in the figure, τ_1, τ_2 and τ_3 are the heights of the three internal nodes of G, and η_1, η_2, η_3 and η_4 are the heights of four internal nodes of Ψ.

set $\{\tau_G(w) : w \in h^{-1}(b)\} \cup \{\tau_\Psi(y)\}$ in increasing order. We denote by $T_b(h)[i]$ the i-th element of the vector. Furthermore, we denote by $u_b(h)$ the number of gene lineages entering edge b and $v_b(h)$ the number of gene lineages leaving edge b under h. Then we have

$$p(h|\Psi,\Gamma) = \prod_{b=(x,y)\in E(\Psi)} \left[\prod_{i=1}^{|T_b(h)|-1} f_c(u_b(h)-i+1, T_b(h)_{i+1}-T_b(h)_i) \times \frac{1}{\binom{u_b(h)-i+1}{2}} \right] \times f_n(v_b(h), \tau_\Psi(x) - T_b(ht)_{|T_b(h)|}) \times \Gamma[b,j]^{u_b(h)}, \qquad (4)$$

where $f_c(j,t)$ is the pdf of the waiting time t ($t \geq 0$) for j lineages to coalesce into $j-1$ [11, 12]

$$f_c(j,t) = \binom{j}{2} e^{-\binom{j}{2}t}. \qquad (5)$$

Furthermore, $1/\binom{u_b(h)-i+1}{2}$ is the probability of a particular pair of gene lineages among $u_b(h)-i+1$ lineages coalescing in a manner that is consistent with the topology of G.

Table 2: The individual terms of the pdf of all coalescent histories with respect to coalescence times in Fig. 2. For every h, the six columns labeled 1—6 give the term for having the coalescence events given by h on every branch of the species network Ψ. Since there is only one reticulation node and we are illustrating an arbitrary locus, we replace Γ by a single γ value for edge 2 in the network and $1-\gamma$ for edge 3 (the Γ values for every tree edge in the network is 1). Branch 6 is the one incident into the root of the species network. A dash means no gene lineages enter that branch. The relative likelihood for the coalescent history random variable to take on the value of a specific history is the product of all the six terms in the corresponding row of that coalescent history. In Fig. 2, coalescent events y and z can only happen above the root of Ψ. For every h, the highlighted cell shows where coalescent event x happens.

	Phylogenetic network branch-specific individual terms of the coalescent history pdf					
	1	2	3	4	5	6
h_1	$e^{-\eta_4}$	$\gamma^2 e^{-(\eta_3-\eta_4)}$	$-$	$3e^{-(\tau_3-\eta_3)}e^{-(\eta_1-\tau_3)}$	1	$3e^{-(\tau_2-\eta_1)}e^{-(\tau_1-\tau_2)}$
h_2	$e^{-\eta_4}$	$-$	$(1-\gamma)^2 e^{-(\eta_2-\eta_4)}$	1	$3e^{-(\tau_3-\eta_2)}e^{-(\eta_1-\tau_3)}$	$3e^{-(\tau_2-\eta_1)}e^{-(\tau_1-\tau_2)}$

And $f_n(j,t)$ is the probability of no coalescent events happening among j gene lineages for time t which can be computed as [11, 12]

$$f_n(j,t) = e^{-\binom{j}{2}t} \tag{6}$$

After substituting Eq. 5 and Eq. 6 into Eq. 4, we have

$$p(h|\Psi,\Gamma) = \prod_{b\in E(\Psi)} \left[\prod_{i=1}^{|T_b(h)|-1} e^{-\binom{u_b(h)-i+1}{2}(T_b(h)_{i+1}-T_b(h)_i)} \right] \times e^{-\binom{v_b(h)}{2}(\tau_\Psi(b)-T_b(h)_{|T_b(h)|})} \times \Gamma[b,j]^{u_b(h)}. \tag{7}$$

In Table 2, we give an example of how Eq. 7 is computed given the phylogenetic network Ψ and gene tree G in Fig. 2.

We can use the same technique in [36] but with Eq. 3 to calculate $p(G|\Psi,\Gamma)$. Basically, we first convert the phylogenetic network to a MUL-tree. Then under every allele mapping, we compute the set of coalescent histories with respect to coalescence times and use Eq. 7 to compute the probability of every coalescent history. Note that as in [36] special attention needs to be paid to the sets of edges in the MUL-tree that come from the same edge in the original network.

Additionally, we can also compute $p(G|\Psi, \Gamma)$ based on weighted ancestral configurations, which is faster than computing it based on MUL-trees. The main idea is similar to the method we proposed in [37], which was built on the work of [32] for the case of phylogenetic trees. We now describe this method in detail.

We first describe briefly the concept of weighted ancestral configuration (AC, or simply configuration) we introduced in [37]. An ancestral configuration can be associated with a node v of Ψ, denoted by AC_v, or an edge b of Ψ, denoted by AC_b. It is a triplet (B, a, w) interpreted as follows:

- B: a set of lineages that exist at the point with which the AC is associated.
- $a[i]$, for $1 \leq i \leq |V_N|$: an index for the AC split that occurred at reticulation node i and gave rise to B.
- w: weight of the AC which is the probability of observing B at the point with which the AC is associated.

Given two ACs, $AC_1 = (B_1, a_1, w_1)$ and $AC_2 = (B_2, a_2, w_2)$, if $B_1 = B_2$ and $a_1 = a_2$, we say that AC_1 and AC_2 are *identical*. Furthermore, we define two ACs to be *compatible* if for each i, $1 \leq i \leq q$, either $a_1[i] = a_2[i]$ or $a_1[i] \cdot a_2[i] = 0$.

Basically, we traverse the nodes of Ψ in post-order and build a set of ancestral configurations for every node we visit. We denote a set of ACs of node v by $\mathscr{AC}_v$, and a set of ACs of edge b which are about to leave b by $\mathscr{AC}_b$. After $\mathscr{AC}$ is built for the root of the network, we could obtain $p(G|\Psi, \Gamma)$. There are three important operations during this process:

- Merging ACs whenever an internal tree node is encountered. Let u be an internal tree node with two child nodes x_1 and x_2. To construct $\mathscr{AC}_u$, every compatible AC pair (AC_1, AC_2) where $AC_1 = (B_1, a_1, w_1) \in \mathscr{AC}_{(u,x_1)}$ and $AC_2 = (B_2, a_2, w_2) \in \mathscr{AC}_{(u,x_2)}$ are merged into one AC (B, a, w) where $B = B_1 \cup B_2$, $a[i] = \max\{a_1[i], a_2[i]\}$ for all $1 \leq i \leq q$ and $w = w_1 \cdot w_2$.
- Splitting ACs whenever an reticulation node is encountered. Let u be the k_{th} reticulation node whose two parent nodes are y_1 and y_2. For each reticulation node i of

N, we have a counter o_i initialized to 0. Let (B, a, w) be an AC in $\mathscr{AC}_u$. Then we split B it into all possible ordered pairs (B_1, B_2), such that $B_1 \cup B_2 = B$ and $B_1 \cap B_2 = \emptyset$. For each pair, we make $AC_1 = (B_1, a_1, w)$ and $AC_2 = (B_2, a_2, 1)$, where $a_1 = a_2 = a$ except for $a_1[k] = a_2[k] = o_k + 1$ and o_k is incremented by 1. AC_1 then is a configuration about to go along edge (y_1, u), and AC_2 is a configuration about to go along edge (y_2, u).

- Coalescing ACs along an edge. We define a function called **CoalACs** which takes a gene tree G, an edge (x, y) of Ψ and a set of configurations $\mathscr{AC}_y$ that enter edge (x, y), and returns a set of configurations $\mathscr{AC}_{(x,y)}$ that leave edge (x, y). See Alg. 1 for details.

The algorithm for computing $p(G|\Psi, \Gamma)$ is shown in Alg. 2. Further, we can use the same technique we introduced in [37] to reduce the number of configurations at articulation nodes of the network.

Algorithm 1: CoalACs.

```
Input: a gene tree $G$, an edge $(x, y) \in E(\Psi)$, a set of ACs $\mathscr{AC}_y$
Output: a set of ACs $\mathscr{AC}_{(x,y)}$
Let $V(G)$ be the set of internal nodes of $G$ ordered by their heights in increasing order;
$\mathscr{AC}_{(x,y)} \leftarrow \emptyset$;
foreach $(B, a, w) \in AC_y$ do
    $t \leftarrow \tau_\Psi(y)$;
    $B^+ \leftarrow B$;
    $p \leftarrow \lambda_{(x,y)}^{|B|}$;
    foreach $v \in V(G)$ do
        if $\tau_\Psi(y) \leq \tau_G(v) < \tau_\Psi(x)$ then
            Let $L_v$ be the set of taxa under node $v$ in $G$;
            Let $L_B$ be the set of taxa that coalesce into $B$;
            if $L_v \subseteq L_B$ then
                $p \leftarrow p \cdot e^{-\binom{|B^+|}{2}(\tau_G(v)-t)}$;
                $t \leftarrow \tau_G(v)$;
                Apply the coalescent event represented by $v$ to $B^+$ and the resulting $B^+$ contains one less lineages;
            else if $L_v \cap L_B \neq \emptyset$ then
                $p \leftarrow 0$;
                Break;
    if $p \neq 0$ then
        if $|B^+| \neq 1$ then
            $p \leftarrow p \cdot e^{-\binom{|B^+|}{2}(\tau_\Psi(x)-t)}$;
        $\mathscr{AC}_{(x,y)} \leftarrow \mathscr{AC}_{(x,y)} \cup (B^+, a, w \cdot p)$;
return $\mathscr{AC}_{(x,y)}$;
```

Algorithm 2: CalProb.

```
Input: Phylogenetic network $\Psi$, gene tree $G$, and inheritance probabilities matrix $\Gamma$
Output: $p(G|\Psi, \Gamma)$
while traversing the nodes of $\Psi$ in post-order do
    if node $v$ is a leaf, whose parent is $u$ then
        $\mathscr{AC}_v \leftarrow \{(B, a, 1)\}$ where $B$ is the set of leaves in $G$ sampled from the species associated with $v$ and $a$ is a vector of $q$ 0's;
        $\mathscr{AC}_{(u,v)} \leftarrow$ CoalACs($G$, $(u, v)$, $\mathscr{AC}_v$);
    else if node $v$ is a reticulation node, who has child $w$, and two parents $u_1$ and $u_2$ then
        $\mathscr{AC}_v \leftarrow \mathscr{AC}_{(v,w)}$;
        $S_1 \leftarrow \emptyset$;
        $S_2 \leftarrow \emptyset$;
        foreach $AC \in \mathscr{AC}_v$ do
            Split $AC$ in every possible way into pairs of ACs, and for each pair, add one to $S_1$ and the other to $S_2$ ;
        $\mathscr{AC}_{(u_1,v)} \leftarrow$ CoalACs($G$, $(u_1, v)$, $S_1$);
        $\mathscr{AC}_{(u_2,v)} \leftarrow$ CoalACs($G$, $(u_2, v)$, $S_2$);
    else if node $v$ is an internal tree node, who has two children $w_1$ and $w_2$ then
        foreach pair $(AC_1, AC_2)$ of compatible ACs in $\mathscr{AC}_{(v,w_1)} \times \mathscr{AC}_{(v,w_2)}$ do
            Merge $AC_1$ and $AC_2$ and add the resulting AC to $\mathscr{AC}_v$;
        if node $v$ is an internal tree node, and its parent is $u$ then
            $\mathscr{AC}_{(u,v)} \leftarrow$ CoalACs($G$, $(u, v)$, $\mathscr{AC}_v$);
        else
            Create a dummy node $r'$ with height $+\infty$;
            $\mathscr{AC}_{(r',v)} \leftarrow$ CoalACs($G$, $(r', v)$, $\mathscr{AC}_v$);
            return $\sum_{(B,a,w)\in\mathscr{AC}_{(r',r)}} w$;
```

4 Inference of networks and inheritance probabilities

The search consists of (1) optimizing a candidate network's branch lengths and inheritance probabilities, and (2) searching the network topologies space. We assume here that all loci share the same inheritance probability (denoted by $\boldsymbol{\gamma}$) for a given branch in the phylogenetic network. Extending this to allow for varying these probabilities across loci is straightforward (of course, while increasing the running time).

4.1 Optimizing branch lengths and inheritance probabilities of a phylogenetic network

In this section, we describe our approach for optimizing branch lengths $\boldsymbol{\lambda}^*$ and inheritance probabilities $\boldsymbol{\gamma}^*$ for a fixed network topology Ψ ($\boldsymbol{\lambda}^*$ is part of Ψ here), given a set $\mathcal{G}$ of gene trees, in order to maximize $p(G|\Psi, \gamma)$. We discuss separately the cases of using gene tree topologies alone and using gene tree topologies and branch lengths.

4.1.1 Using gene tree topologies alone

A heuristic for finding the optimal branch lengths for a fixed species tree topology was introduced in [32]. Here, we are using the same method but in our case of phylogenetic networks we are optimizing not only branch lengths but also inheritance probabilities. In particular, an initial value of likelihood is first calculated with every branch length initialized to be 1.0 and inheritance probability initialized to be 0.5. Then the elements in $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$ are optimized one by one separately using Brent's method [2]. More specifically, while Brent's method is varying the value of one element in $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$ in order to find a local optimum, the values of all other elements are fixed. After the local optimum is found, the element is replaced by this new value and then Brent's method moves to the next element for optimization. Updating all elements in $[\boldsymbol{\lambda}, \boldsymbol{\gamma}]$ once is called a round. After each round of optimization, we compare the likelihood of the network with the newly updates branch lengths and inheritance probabilities to the likelihood from the previous round. If the improvement is smaller than some pre-specified threshold or some pre-specified maximum number of rounds is reached, then the new branch lengths and inheritance probabilities are declared to be optimal and the optimization process terminates. This processes is repeated multiple times to handle the issue of local optima. All parameters used in this optimization process, including those for Brent's method, are listed in Section 8.

4.1.2 Using both topologies and branch lengths of gene trees

As we mentioned, the set of coalescent histories of a gene tree G within the branches of a phylogenetic network Ψ does not change with the branch lengths or inheritance probabilities if only the topology of gene tree G is considered. However, this is not the case if both topologies and branch lengths of the gene trees need to be taken into account. The main reason for this is that the coalescence times in the gene tree provide constraints on where coalescence events could take place within the branches of the phylogenetic network. Further, it is important here to note that the time units in the gene tree and those in the species network must be matched for our method below to work. Consider a branch b with length λ_b (in coalescent units) in a phylogenetic network Ψ. Now, consider a branch d with length τ_d (in units of expected number of mutations) in a gene tree G. The length of branch d in coalescent units is

$$\tau_d \times \frac{2}{\theta}$$

where $\theta = 4N\mu$ is the population mutation rate.

In our discussion below, as well as in our implementation, we assume that θ is the same across all loci and all branches, and that the population size and generation time are the same across all branches. This implies that both the network and gene tree must be ultrametric. However, it is important to notice that removing these assumptions does not affect the model, but rather increase the running time of the method as more parameters require optimization.

In order to guarantee the ultrametricy requirement, instead of optimizing branch lengths of phylogenetic network Ψ and inheritance probabilities, we optimize the height of every internal node of Ψ and inheritance probabilities. Second, in order to ensure that the resulting phylogenetic network allows for embedding the gene trees in the input, we use the coalescence times from the gene trees to compute upper bounds on the heights of nodes in Ψ, and use these upper bounds to constrain the search for height values. Then the iterative process for optimization itself is similar to what we described in Section 4.1.1. The full details of the optimization procedure are available in open source in the software package PhyloNet [30].

4.2 Inferring an ML phylogenetic network

For inferring an ML phylogenetic network, we couple the optimization procedures of the previous section with a procedure for traversing the phylogenetic network space, which we now describe.

4.2.1 Neighborhood of a phylogenetic network

For a fixed number of taxa n, the space of phylogenetic networks, denoted by $\Omega(n)$, consists of an infinite set of non-overlapping subspaces, each of which contains phylogenetic networks that have the same number of reticulation nodes. We denote each subspace of $\Omega(n)$ by $\Omega(n, k)$, where k is the number of reticulation nodes. In particular, $\Omega(n, 0)$ is the tree space.

Given a phylogenetic network $\Psi \in \Omega(n, k)$, we define four types of operations for network rearrangement as follows.

- Adding a reticulation edge (δ_1):

    1. Let (u_1, v_1) and (u_2, v_2) be two distinct edges in Ψ such that v_2 is not a predecessor of u_1.
    2. Delete the two edges (u_1, v_1) and (u_2, v_2).
    3. Add two new nodes x_1 and x_2 and five new edges (u_1, x_1), (x_1, v_1), (u_2, x_2), (x_2, v_2), and (x_1, x_2) to network Ψ.

- Removing a reticulation edge (δ_2):

    1. Let (u, v) be an edge in Ψ such that v is a reticulation node and u is a tree node.

2. Delete the two nodes u and v and the five edges (w, u), (u, z), (u, v), (x, v) and (v, y), where w is the parent node of u, z is the child node of u other than v, x is the parent node of v other than u, and y is the child node of v.
3. Add two new edges (w, z) and (x, y) to network Ψ.

- Relocating the destination of a reticulation edge (δ_3):
  1. Let (u_1, v_1) and (u_2, v_2) be two distinct edges in Ψ such that v_1 is a reticulation node and v_2 is not a predecessor of u_1.
  2. Delete node v_1 and the four edges (u_1, v_1), (u_2, v_2), (w, v_1), and (v_1, z), where w is the parent node of v_1 other than u_1 and z is the child node of v_1.
  3. Add a new nodes x and four new edges (u_2, x), (x, v_2), (u_1, x), and (w, z) to network Ψ.
- Relocating the source of an edge (δ_4):
  1. Let (u_1, v_1) and (u_2, v_2) be two distinct edges in Ψ such that u_1 is neither a reticulation node nor a predecessor of v_2.
  2. Delete node u_1 and the four edges (u_1, v_1), (u_2, v_2), (w, u_1), and (u_1, z), where w is the parent node of u_1 and z is a child node of u_1 other than v_1.
  3. Add a new nodes x and four new edges (u_2, x), (x, v_2), (x, v_1), and (w, z) to network Ψ.

We denote the set of phylogenetic networks that can be obtained by applying operation δ_i to Ψ by $\delta_i(\Psi)$, where $1 \leq i \leq 4$. Clearly, $\Psi' \in \Omega(n, k+1)$ if $\Psi' \in \delta_1(\Psi)$, $\Psi' \in \Omega(n, k)$ if $\Psi' \in \delta_3\Psi$ or $\Psi' \in \delta_4(\Psi)$ and $\Psi' \in \Omega(n, k-1)$ if $\Psi' \in \delta_2(\Psi)$. Finally, we define the neighborhood of a phylogenetic network Ψ, denoted by $\Delta(\Psi)$, to be $\bigcup_{1 \leq i \leq 4} \delta_i(\Psi)$. So a phylogenetic network Ψ' is a neighbor of Ψ if Ψ' can be obtained by applying any operation defined above to Ψ.

4.3 Search heuristic

We employ a hill-climbing heuristic to search the network space in order to find an optimal phylogenetic network Ψ from a set $\mathcal{G}$ of gene trees. Starting from some network, the search proceeds by sampling networks from the neighborhood of the current network, optimizing its branch lengths as well as the inheritance probabilities, and accepting the proposed network if its likelihood improves upon the current one. The process terminates if no neighboring network improves upon the current one (our implementation allows for pre-specifying a number of failed neighbor proposals, since the number of neighbors can be very large for large numbers of leaves).

For the starting network, it is reasonable to start the search from some species trees, e.g., the set of all binary resolutions of majority consensus of the input gene trees, or the optimal species tree under the MDC criterion [13, 29, 38, 39]. For moving from a current network, a neighbor can be generated by applying one of the four types of operations of network rearrangement we defined in the Section 4.2.1. We associate each of these four operations a weight. When we propose a random neighbor of a network, the type of operation to be applied to generate the neighbor is first randomly selected according to their weights and the edges involved in that operation are then randomly chosen. The entire search process is repeated multiple times to handle the problem of local optima.

An illustration of the search is given in Fig. 3.

5 Assessing phylogenetic networks

5.1 Information criteria

The Akaike Information Criterion [1] (AIC) is defined as follows for a phylogenetic network:

$$AIC = 2k - 2\ln L \tag{8}$$

where k is the number of free parameters which includes both branch lengths and inheritance probabilities, and L is the likelihood of the network.

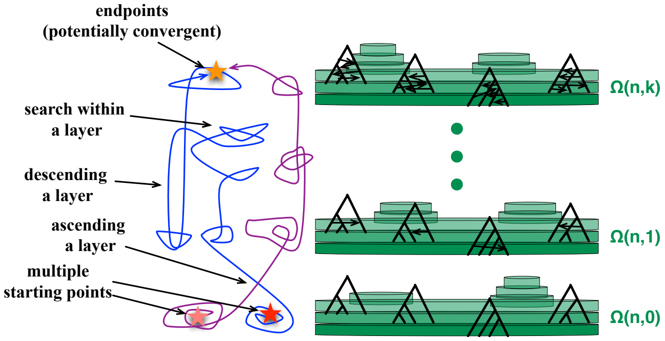


Figure 3: A schematic diagram of searching the n-taxon phylogenetic network space. Layer $\Omega(n, k)$ contains all n-taxon phylogenetic networks with k reticulation nodes. Four simple transformations enable searching this entire space and guarantee reachability of any point from any other point (see main text and SI). These operations allow for the search to proceed within a give layer, ascend a layer, or descend a layer. Weighting, or assigning rates to the the transformations, allows for controlling the behavior of the search (e.g., never descend a layer, or ascend a layer with very low probability). When searching for optimal networks, multiple searches can be initiated from different starting points; these searches may or may not converge onto a single optimal point estimate. Further, the number and features of local optima vary from one layer to another.

The Bayes Information Criterion [24] (BIC) is defined as follows for a phylogenetic network:

$$BIC = -2 \ln L + k \ln n \tag{9}$$

where k and L are defined as in AIC, and n is the number of gene trees in the set.

5.2 Cross-validation

Cross-validation is another model validation technique that assesses how well the model fits a data set. K-fold cross-validation partitions a data set into K equal-size subsets. It uses one set, the training set, which consists of $K-1$ subsets, to infer the model parameters and use the remaining subset, the validation set, to assess prediction. The difference between predictions and the real data in the validation set can be computed. To reduce variability, K

rounds of cross-validations are performed using different partitions, and the differences are averaged over the number of rounds. If there are multiple models, the one with the lowest average difference is the most appropriate model. In our case, for each distinct gene tree topology in the validation subset, we compute its frequency, as well as its probably under the learned network. The difference between these two values is taken to reflect the quality of the model.

5.3 Parametric bootstrap

With the increasing interest in reconstruction of phylogenetic trees, in order to evaluate how confident one should be in a reconstructed phylogeny, bootstrapping has been widely used for decades since it was first proposed as a method for obtaining confidence limits on phylogenies [5]. Here, we employ parametric bootstrap assess support for the branches in an inferred phylogenetic network (illustrated in Fig. 4).

The idea is as follows. An inferred phylogenetic network Ψ is used to generate k sets of gene trees independently, each of which has the same size as the number of loci in the input (in PhyloNet, the default value of k is 100). Then from each simulated set of gene trees, a phylogenetic network is inferred using the same method and settings as the one used to obtain the original phylogenetic network Ψ from gene trees $\mathcal{G}$. Finally, by comparing the k inferred phylogenetic networks with Ψ, we obtain the support of every edge in Ψ.

This parametric bootstrap works for both inference from gene trees with and without branch lengths. In PhyloNet, for ML inference using only the topologies of gene trees, we implemented our own simulator to generate topologies of gene trees from a given phylogenetic network. And for ML inference using both the topologies and branch lengths of gene trees, we called an external software `ms` [6] to simulate gene trees with branch lengths.

The bootstrap value of a branch in the inferred phylogenetic network Ψ is calculated as the proportion of networks in $\Psi_1, \ldots, \Psi_k$ that contain the same branch. We say that branch b_1 in network Ψ_1 and branch b_2 in network Ψ_2 are the same if they satisfy the following two conditions:

- b_1 and b_2 induce the same set of softwired clusters [9],

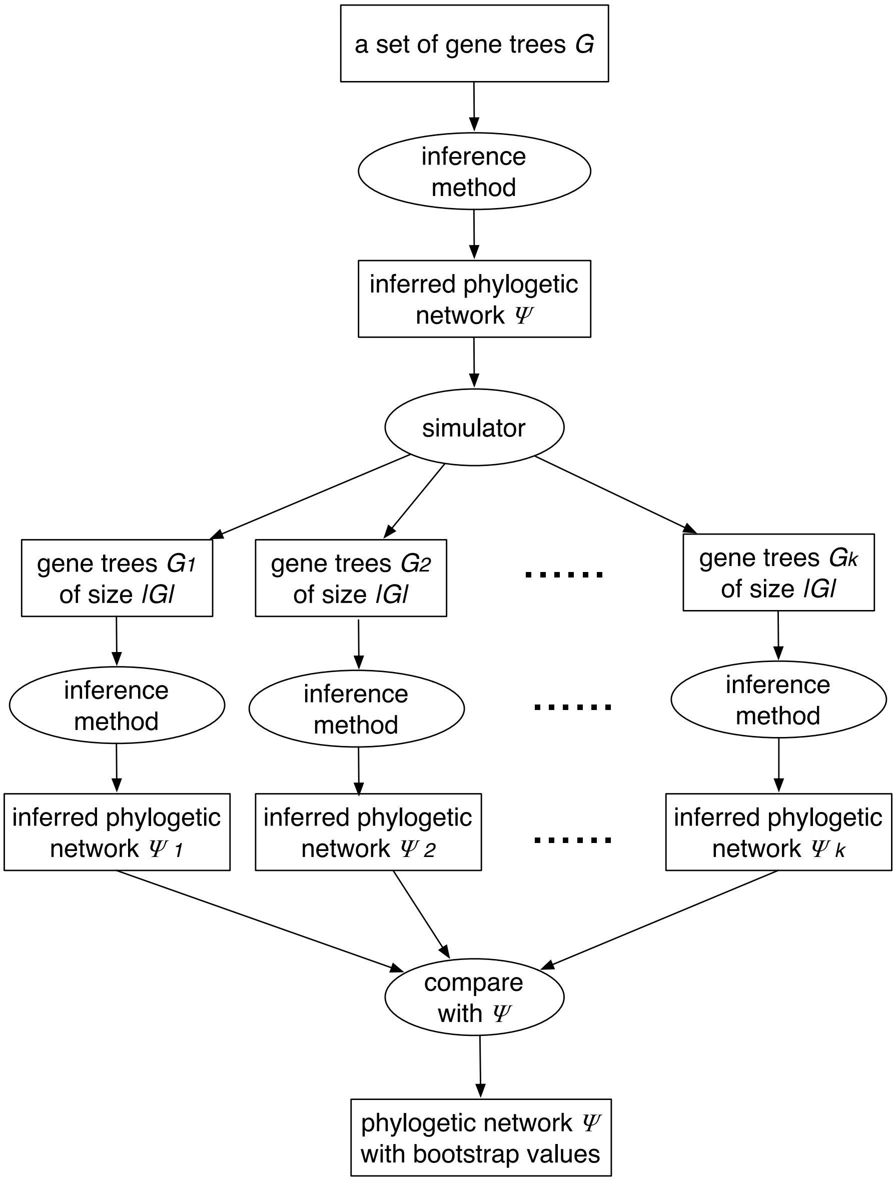

a set of gene trees G
inference method
inferred phylogetic network Ψ
simulator
gene trees G_1 of size $|G|$
gene trees G_2 of size $|G|$
......
gene trees G_k of size $|G|$
inference method
inference method
......
inference method
inferred phylogetic network Ψ_1
inferred phylogetic network Ψ_2
......
inferred phylogetic network Ψ_k
compare with Ψ
phylogetic network Ψ with bootstrap values

Figure 4: Illustration of parametric bootstrap to assess support for an inferred phylogenetic network's branches.

- b_1 and b_2 are either both tree edges or both reticulation edges.

6 Simulations

6.1 Settings for the simulations in the main text (Figure 2)

Phylogenetic network. We used the phylogenetic network shown in Fig. 5. It captures a scenario where there is divergence followed by a hybridization with inheritance probability being 0.1 as shown in the figure.

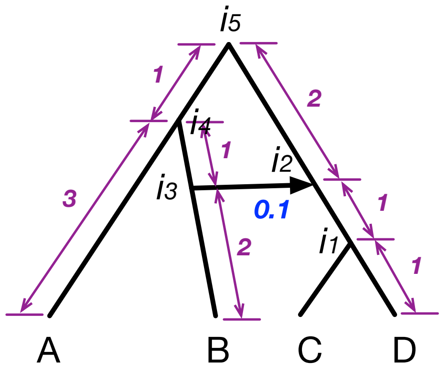


Figure 5: A phylogenetic network used in the simulations reported in the main text. Branch lengths are in coalescent units and the inheritance probability is 0.1 for the reticulation edge (i_3, i_2).

.

True gene trees. Gene trees were simulated using software ms [6]. See command below. We varied the number of loci by $loci = \{10, 20, 40, 80, 160\}$, for each of which we generate 30 sets of gene trees.

ms 4 $loci$ -T -I 4 1 1 1 1 -ej 0.5 4 3 -es 1.0 3 0.5 -ej 1.0 2 5 -ej 1.5 5 1 -ej 2.0 3 1

Sequences. Sequences were generated using Seq-gen [19] under the GTR model. We used a population mutation rate of $\theta = 0.036$. More specifically, for gene trees contained in file $gtFile$, the following command was used:

seq-gen -mGTR -s0.018 -f$baseFreq$ -r$rates$ -l$seqLen < gtFile$

where $baseFreq = 0.300414, 0.191363, 0.196748, 0.311475$ are the base frequencies of the nucleotides A, C, G and T, and $rates = 1.24284, 3.47484, 0.48667, 1.07118, 4.38510, 1.0$ are the relative rates of substitutions. We also varied the length of the sequences through $seqLen = \{250, 500, 1000\}$.

Estimated gene trees. Gene trees were estimated using PAUP* [27] under maximum likelihood. For each sequence alignment, we randomly generated 100 bootstrap replicates. And for each of them, we used the following commands in PAUP* to reconstruct an ultrametric gene tree:

```
execute seqFile;
nj;
lscore 1/tratio=estimate nst=6 rmatrix=estimate;
set criterion=likelihood;
lset tratio=estimate nst=6 rmatrix=estimate clock=yes;
hsearch addseq=asis;
```

where $seqFile$ is a NEXUS file that contains the sequence alignment. All branch lengths of the reconstructed gene tree were then multiplied by $2/\theta$ to convert them into coalescent units.

Experiments. We inferred the optimal networks from (i) true gene tree topologies, (ii) estimated gene tree topologies, (iii) true gene tree topologies and branch lengths, and (iv) estimated gene tree topologies and branch lengths. Default settings were used for inference (See Table. 4). See main text for results.

Furthermore, to study the effect of branch lengths of the phylogenetic network and the inheritance probability on the performance of the method, we investigated two cases. More specifically, assuming the network in Fig. 5 is Ψ_1, we considered two networks Ψ_2 and Ψ_3, where Ψ_2 is obtained by doubling the lengths of the internal branches (i_5, i_4), (i_5, i_2), (i_4, i_3) and (i_2, i_1) of Ψ_1 and not changing the inheritance probability, and Ψ_3 is obtained by changing the inheritance probability of Ψ_1 from 0.1 to 0.5 and keeping the branch lengths unmodified. Then from Ψ_2 and Ψ_3 we generated gene trees using the same settings as above. Finally, we used the topologies of those gene trees to infer networks. The results are shown in Fig. 6. We can see that, as expected, increasing branch lengths and increasing inheritance probability result in improved accuracy of the method.

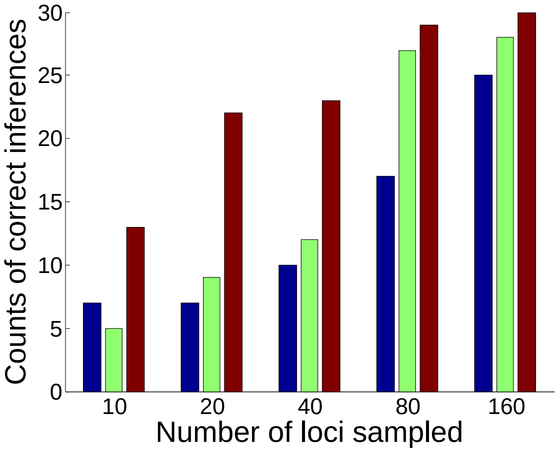


Figure 6: The number of correct inferences using true gene tree topologies simulated from species network Ψ_1, Ψ_2 and Ψ_3. The blue, green and red bars are the results for Ψ_1, Ψ_2 and Ψ_3, respectively.

Varying the number of individuals. We have also varied the number of individuals (lineages) sampled from each of taxa C and D and considered 1, 2, and 4 lineages (for each of the two taxa). For each case, we used the true gene tree topologies and true gene tree topologies and branch lengths to infer the networks. The accuracy results are shown in Fig. 7. The results show that increasing the number of individuals sampled from each of C and D improves the accuracy of the method, as expected (except for the case when using the gene tree topologies alone on 10 loci). Doubling the number of loci used in the inference results in a bigger improvement in accuracy in general than doubling the number of alleles (there are a few exceptions to this that can be seen in Fig 7). However, it is important to caution here that these results are obtained from relatively small networks (4 taxa) and using true gene trees. As the number of taxa increases, the number of reticulations increases, and the branch lengths get shorter, it would be expected that sampling more individuals would show more significant gains in accuracy.

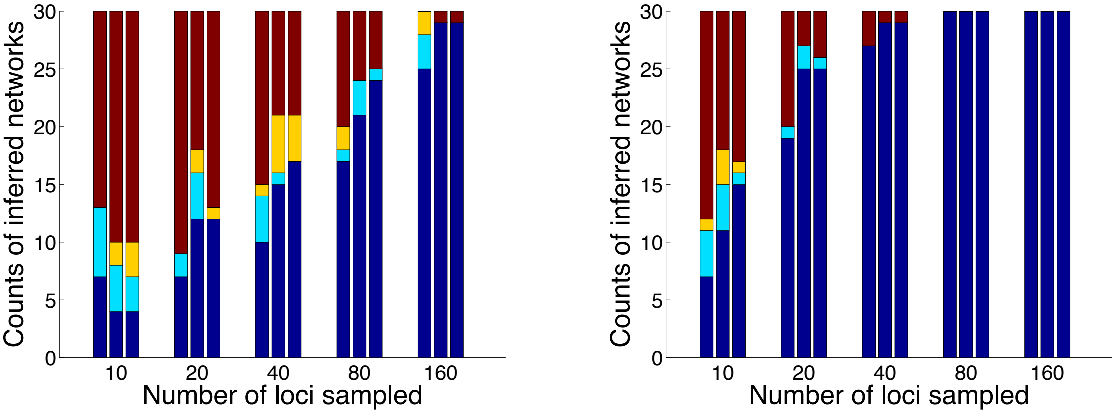


Figure 7: Accuracy of the method on simulated data with varying alleles. The data were generated down the phylogenetic network Ψ_1 (Fig. 5 and also shown in Fig. 2A in the main text). Results based on true gene tree topologies and true gene tree topologies and branch lengths are shown in the left and right panels, respectively. For every number of loci, the bars from left to right correspond to cases of 1, 2, and 4 individuals sampled from each of the two taxa C and D, respectively. The dark blue, cyan, and yellow regions correspond to the number of times each of the networks Ψ_1, Ψ_2, and Ψ_3, respectively (Fig.2A in the main text), were inferred. The maroon region corresponds to the number of times any other network with a single reticulation was inferred.

6.2 Other simulations

We also conducted other simulations under scenarios that are “easier” for the inference method (longer sequences, longer branch lengths, higher inheritance probabilities, and no speciation following hybridization).

The simulations make use of several tools and programs:

- PhyloNet [31], which has implementation of all our methods.
- Hybrid-Lambda [40] which simulates the evolution of gene trees within the branches of a phylogenetic network under the coalescent model.

- Seq-gen [19] which simulates the evolution of DNA sequences down a given (gene) tree.

- Fasttree [15, 16], which infers a maximum likelihood phylogenetic tree from a sequence alignment.

Using these tools, we conducted three types of simulations (Fig. 8).

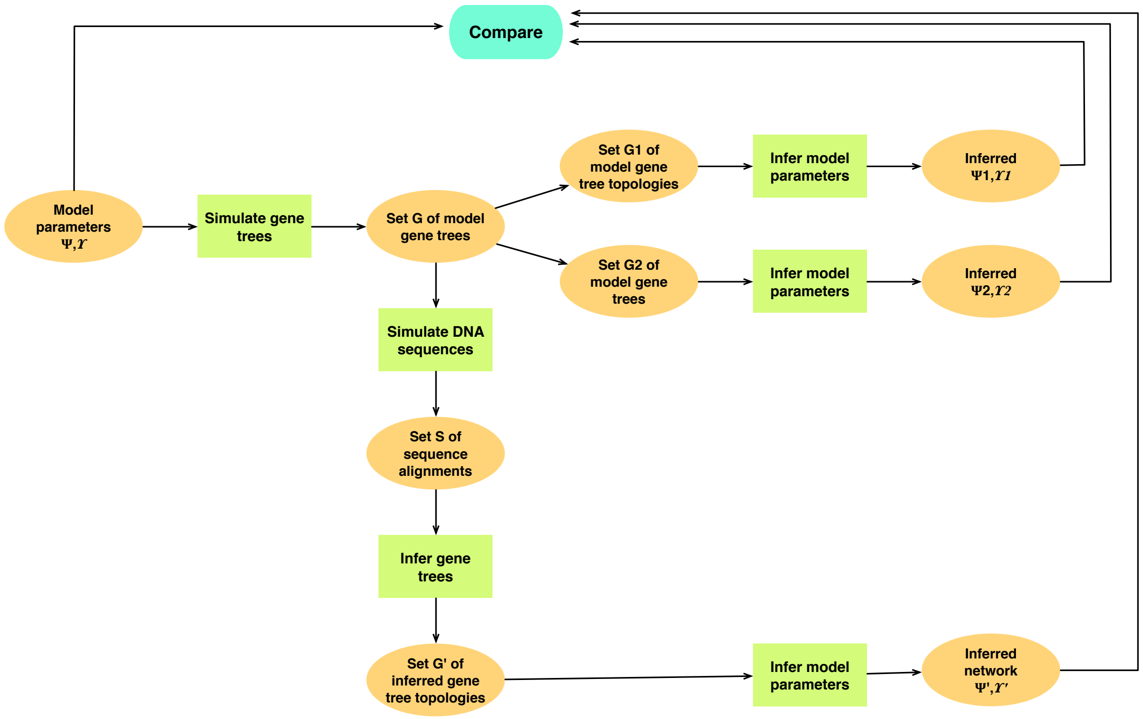
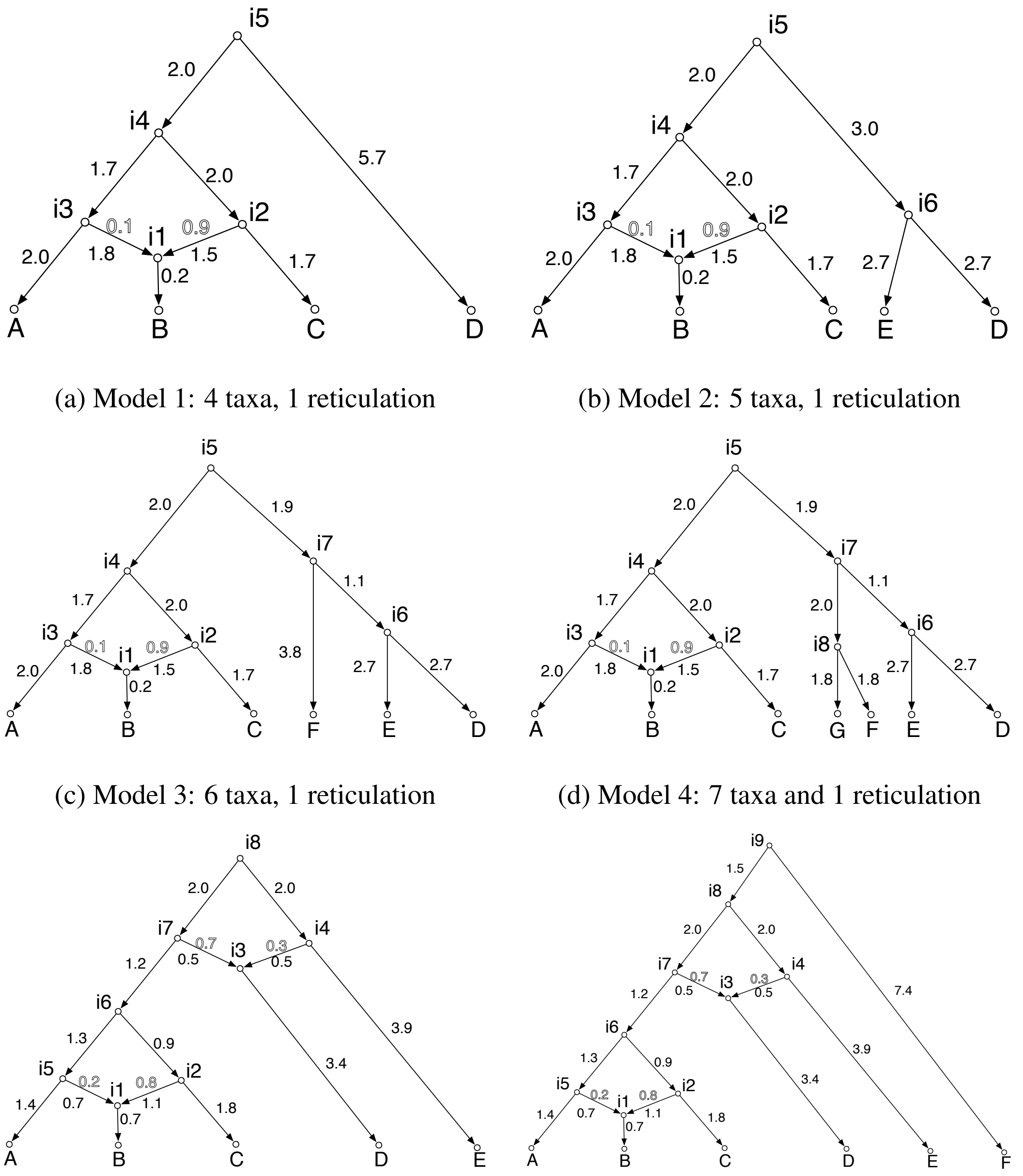


Figure 8: Simulation Flow Chart. Hybrid-Lambda [40] is used to simulate gene trees within a network. Seq-gen [19] is used to simulate the evolution of DNA sequences down gene trees. Fasttree [15, 16] is used to estimated gene trees from sequence alignments. PhyloNet [31] is used to infer phylogenetic networks.

.

In each simulation run, we varied the number of gene trees: 10, 50, 100 and 500. We also consider multiple scenarios of phylogenetic network topologies, branch lengths, and inheritance probabilities, as shown in Fig. 9. For each setting, we conducted 30 runs and averaged the results.



(a) Model 1: 4 taxa, 1 reticulation

(b) Model 2: 5 taxa, 1 reticulation

(c) Model 3: 6 taxa, 1 reticulation

(d) Model 4: 7 taxa and 1 reticulation

(e) Model 5: 5 taxa and 2 reticulations

(f) Model 6: 6 taxa and 2 reticulations

Figure 9: Models used in the simulation. These models are ultrametric networks, required by Hybrid-Lambda.

6.2.1 Results

The plots are arranged as follows. First we show the six models we used for the short flow simulation. Figures 9a, 9b, 9c and 9d have one reticulation node with four, five, six, and seven taxa respectively. Figures 9e and 9f has two reticulation nodes with five and six taxa respectively.

Although limited and empirical in nature, some observations are made from the plots.

- In the topology only case, cross-validation captures much better the correct reticulation nodes than both BIC and AIC. More trees help. But even with very few trees, cross validation is still superior to the other two. It shows that cross-validation is an effective way to determine the proper number of reticulation events.

- In the case of gene trees with branch lengths, AIC and BIC perform similarly when there is a single reticulation node. BIC does a better job than AIC when the number of reticulations is two.

- The cluster distance between the original network and the inferred network with correct network nodes is always smaller than that between the original network and the inferred network with incorrect reticulations when the number of gene trees grows larger.

- Using gene tree branch lengths results in better estimates of the inheritance probabilities. When the number of trees increases, the estimates improve. When using gene tree topologies alone, the inheritance probability estimates are slightly less accurate.

6.2.2 A complete simulation involving network, gene trees and sequences

We use a network as shown in Figure 16 to generate gene trees and nucleotide sequences. Then the process is inverted to use these sequences to estimate gene trees and network. Several public domain software are used as well as PhyloNet. Hybrid-Lambda [40] uses the network to generate gene trees which form the input of Seq-gen [19] to produce sequences. For each gene tree, only a sequence is generated. These sequences are fed into Fasttree [15, 16] to generate unrooted rooted gene trees. PhyloNet reroots these gene trees via the outgroup and later infers the network.

The gene trees are organized the same way as in the simulation section. The Hybrid-Lambda parameter settings used to generate gene trees from the network are the same as well. The Seq-gen settings used to generate sequences from gene trees is

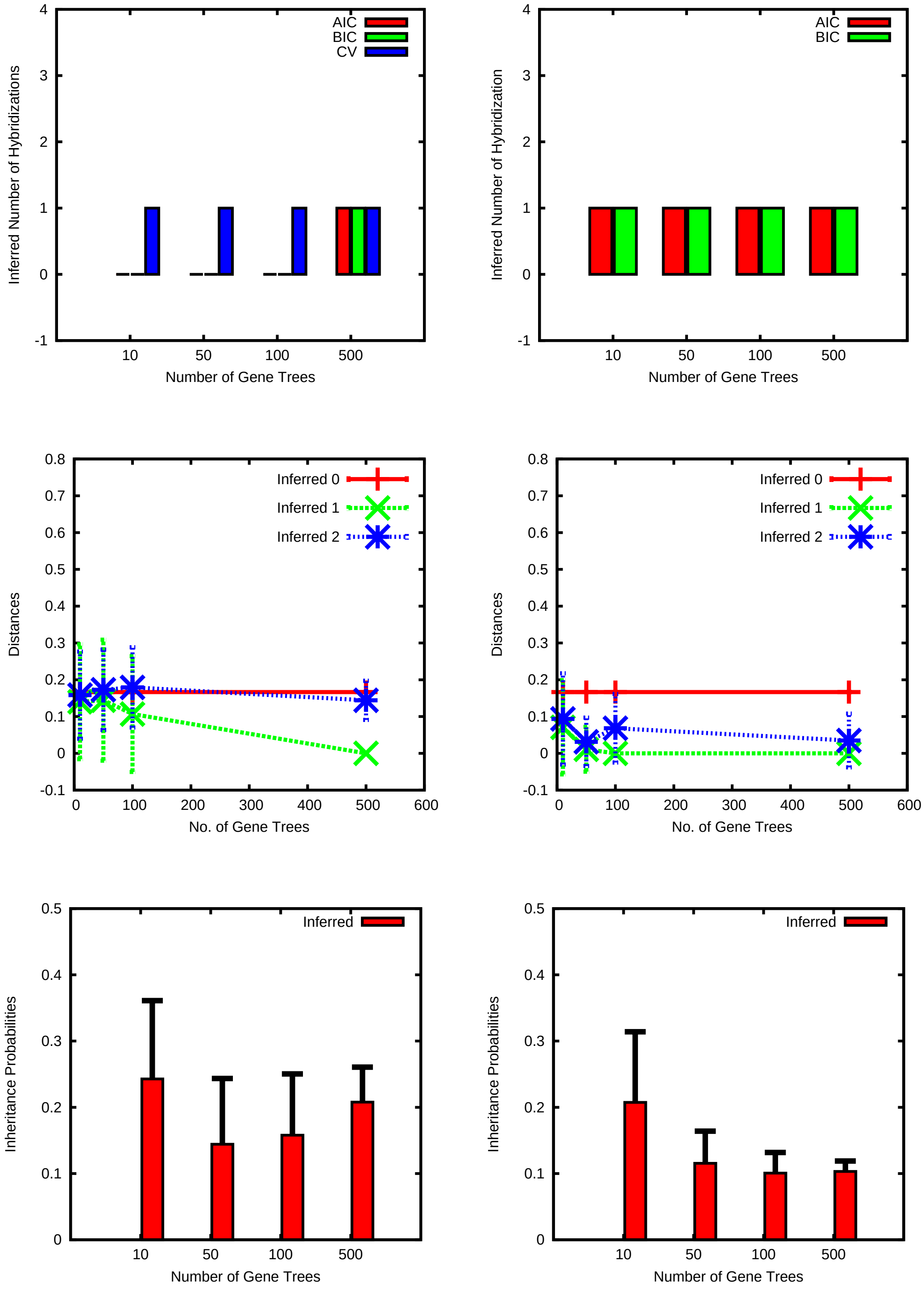


Figure 10: Results for Model 1. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

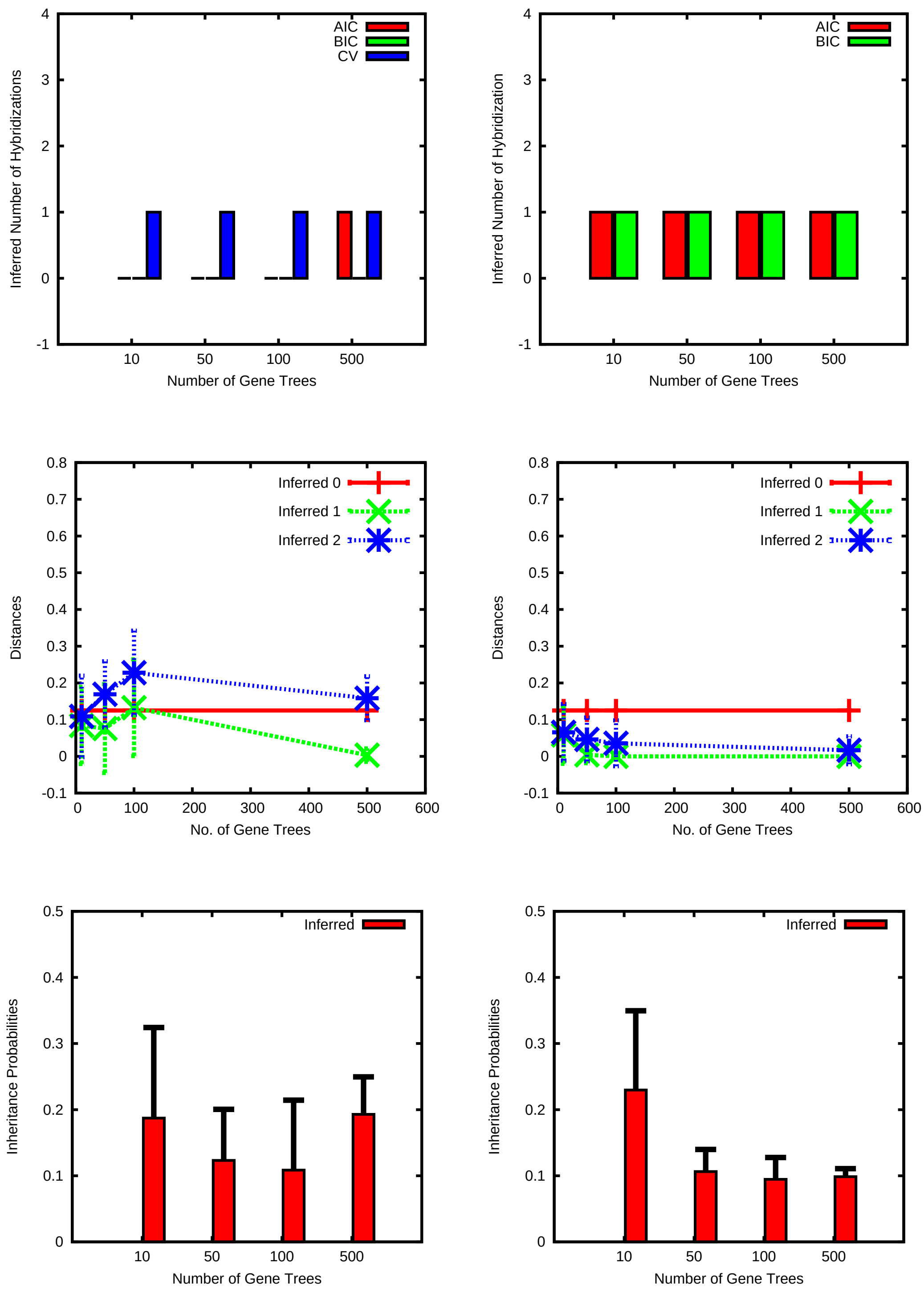


Figure 11: Results for Model 2. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

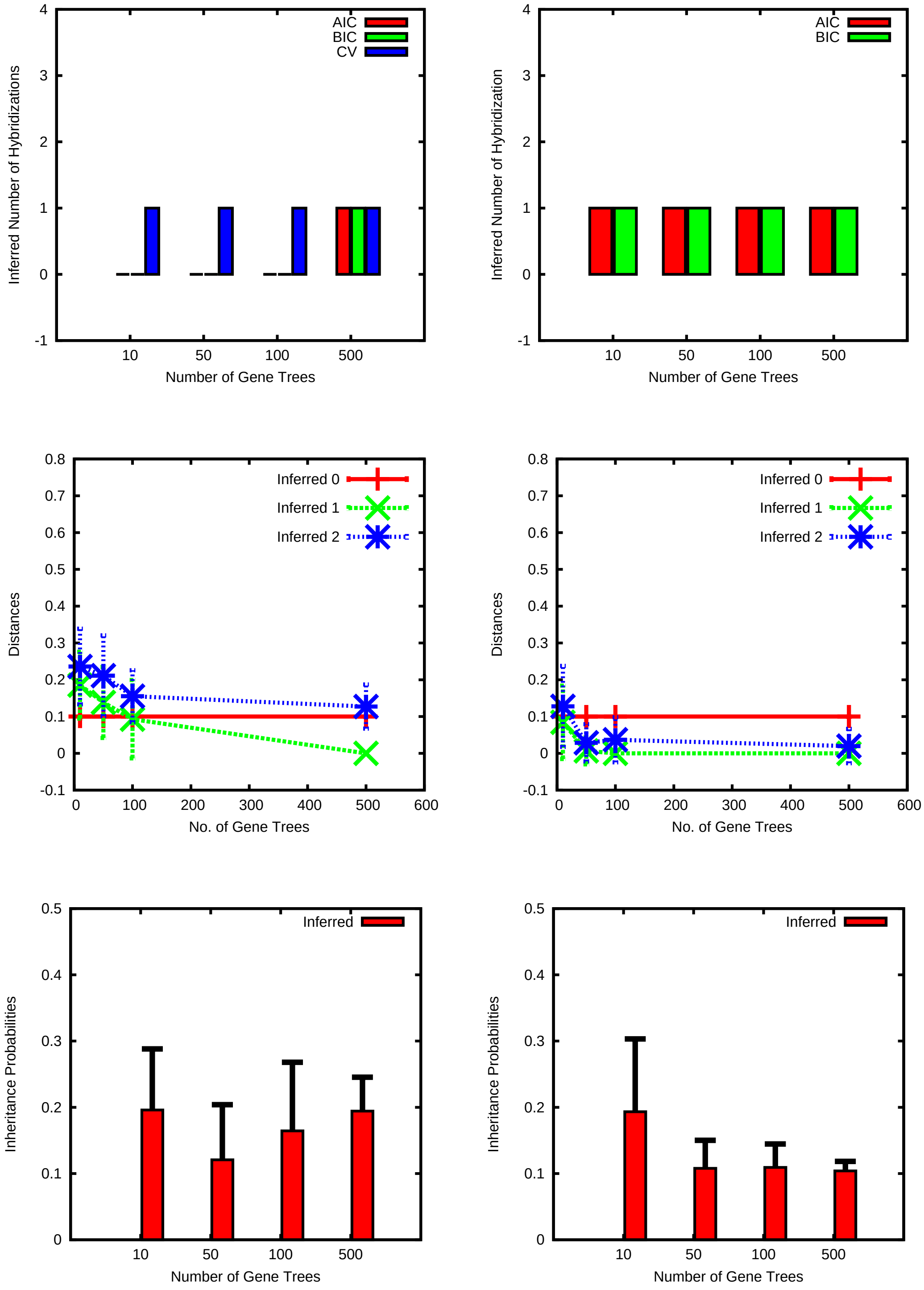


Figure 12: Results for Model 3. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

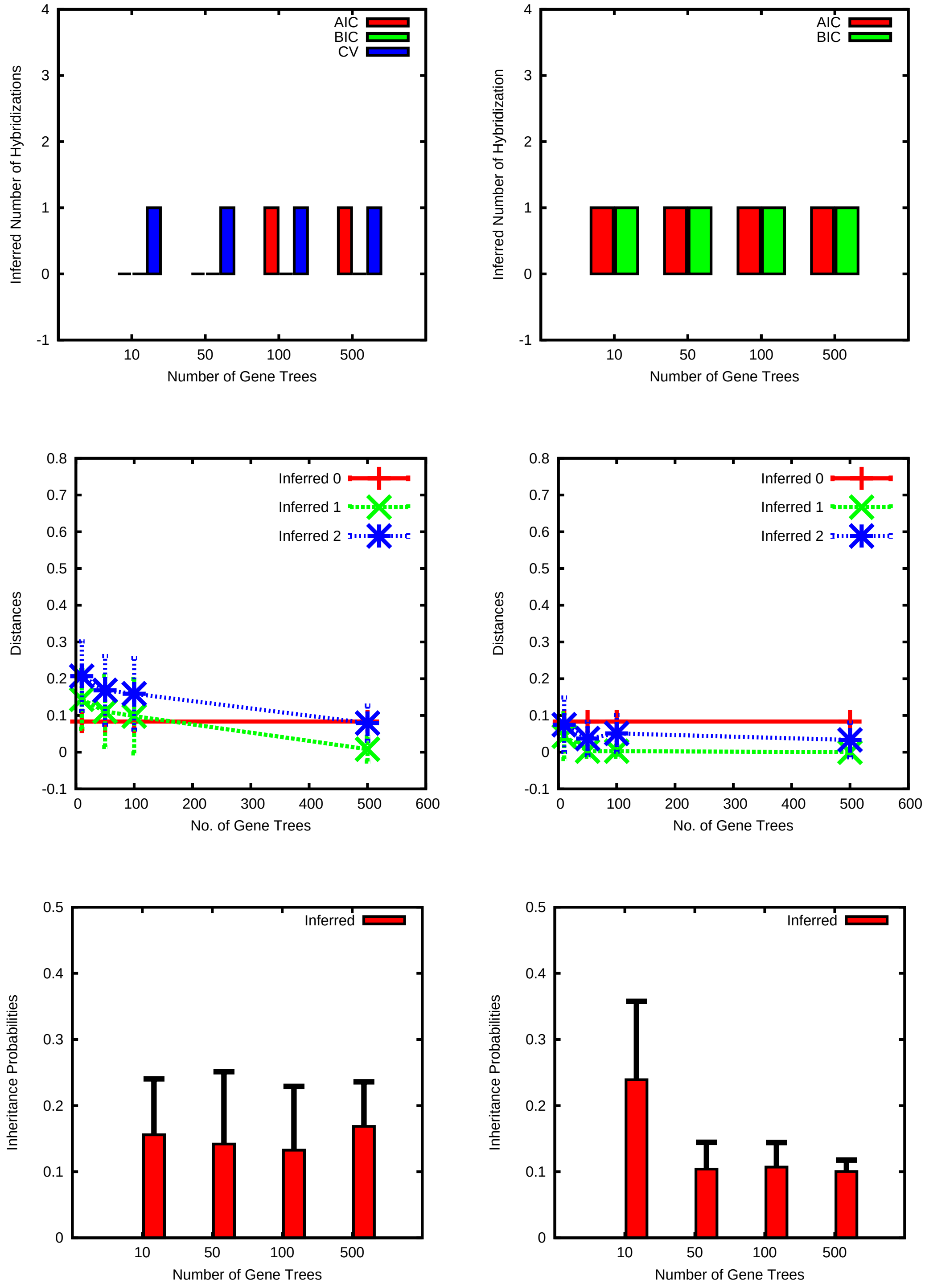


Figure 13: Results for Model 4. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

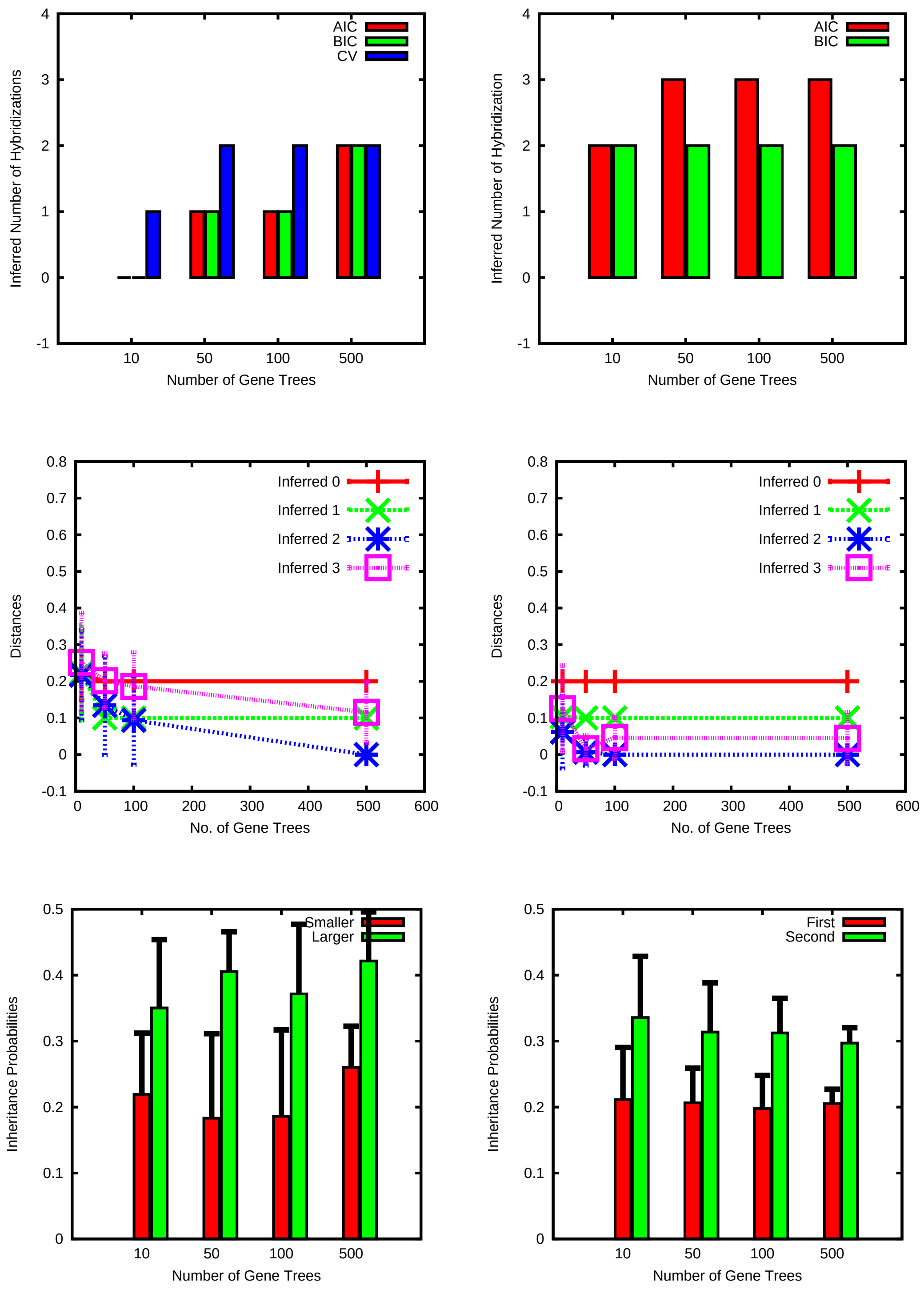


Figure 14: Results for Model 5. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

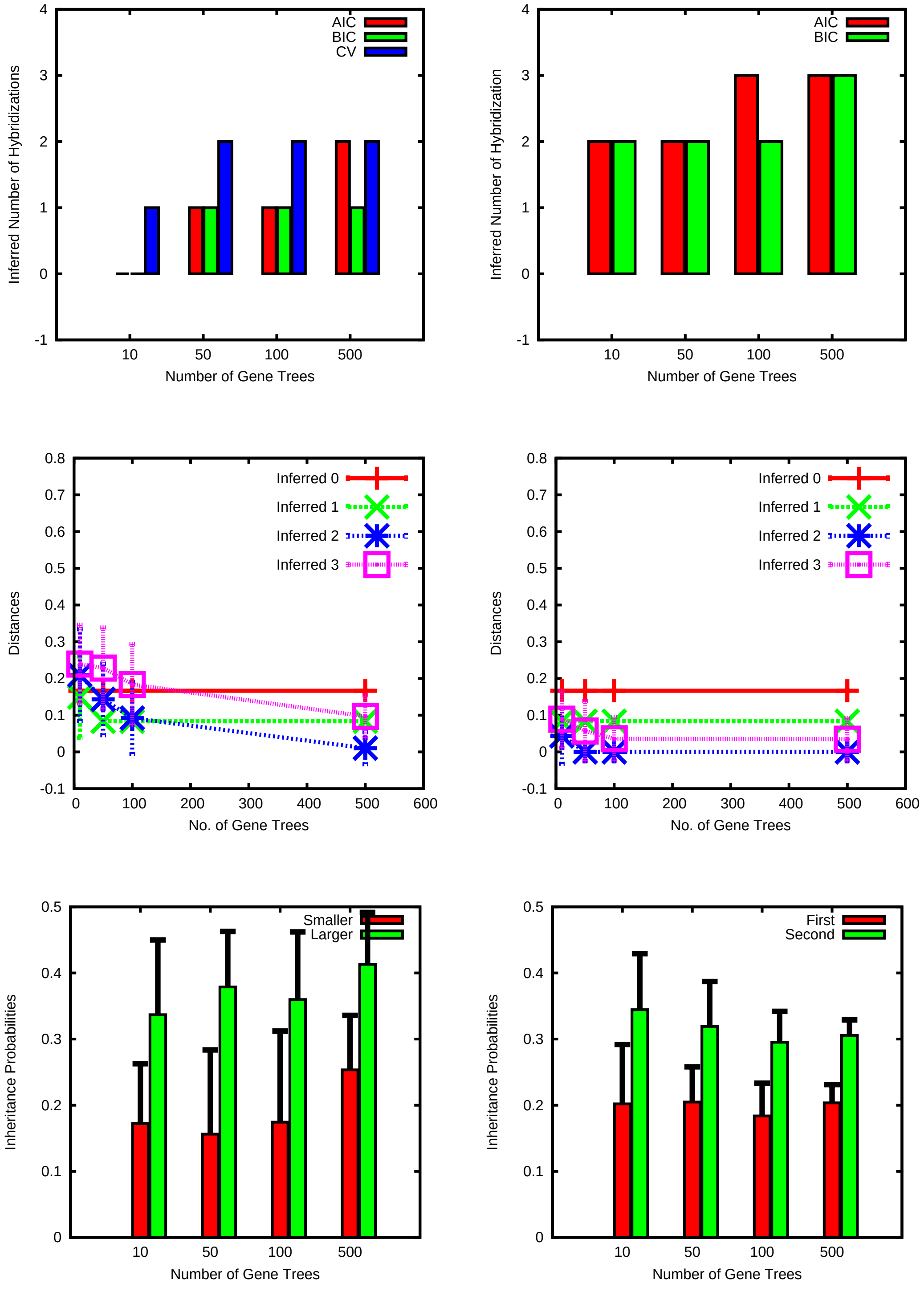


Figure 15: Results for Model 6. Left column: Input consists of true gene tree topologies alone. Right column: Input consists of true gene tree topologies and branch lengths. Top row: inferred number of hybridizations. Middle row: Distance between the true network and inferred network. Bottom row: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

```
seq-gen -mHKY -l5000 -t0.5 -s0.0005 -fe -op < treeFileName > seqFileName
```

For nucleotide substitution model, we used HKY. By setting the nucleotide frequencies equal (with the -fe option) and the transition transversion ratio to 0.5 (with the -t0.5 option), it becomes JC69 as a special case of HKY. The length of character sequence is 5000. The scale factor of 0.0005 equals the expected number of substitutions per site per coalescent unit for each branch is 0.0005. By multiplying with the branch length, we find the expected number of substitutions per site for each branch. The output uses the PHYLIP format.

The Fasttree settings used to generate unrooted gene trees from sequences.

```
FastTree -nt < fasttreeInputFileName > fasttreeOutputFileName
```

Fasttree use the Jukes-Cantor + CAT model. The -nt option shows that it works with nucleotide sequences.

We compared the gene trees that are used to generate the sequences and those estimated from sequences. In this model, all the gene trees have the outgroup directly connected to the root. The average Robinson-Foulds distance is 0.32 while its weighted version is 0.081. The standard deviation for RF distance is 0.53 and its weighted version is 0.13. There are totally 19800 gene trees on each side. Among them there are 14060 pairs are exactly the same (71%). If we increase the scale factor or increase the sequence length, the same pair rate will be higher.

When PhyloNet infers the network, the default maximum length of a network branches is 6. Since there are some large branch length values in the models, we assigned the maximum branch length to be 20 for all the models tested in this paper.

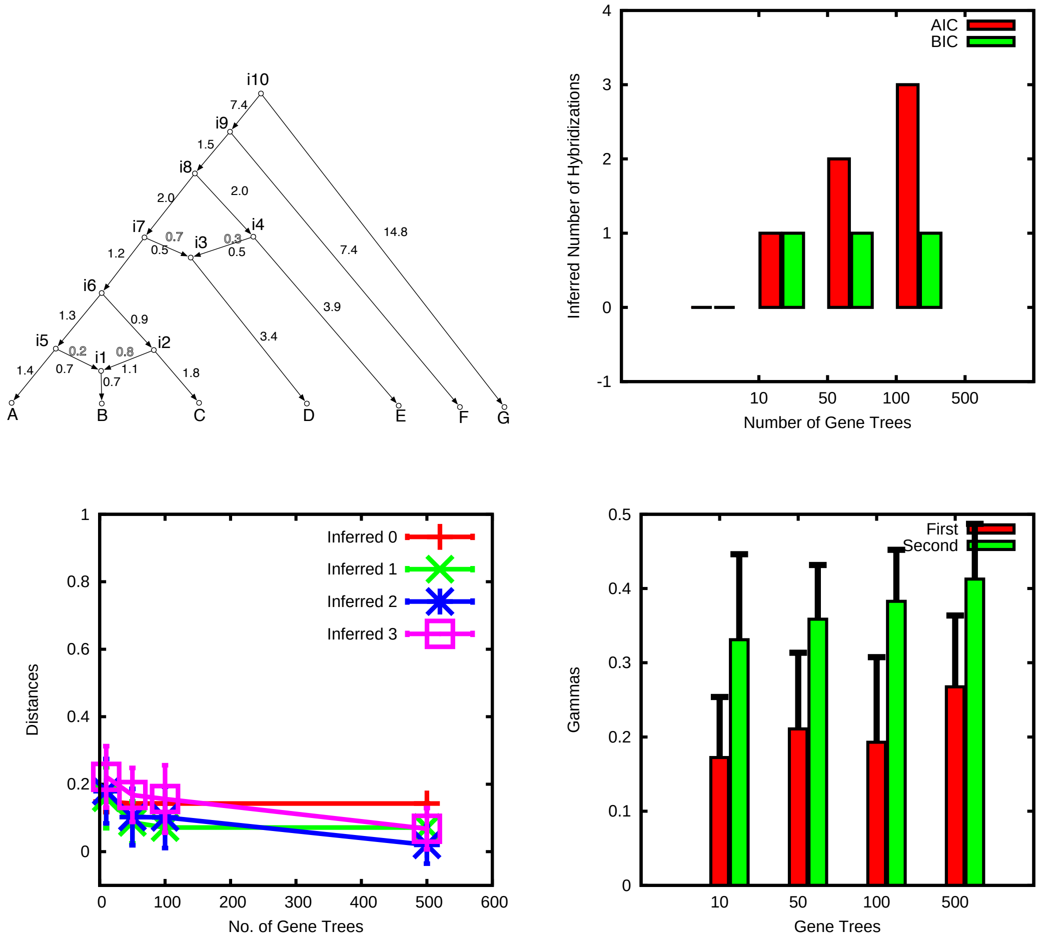


Figure 16: Results for Model 7. Upper left: The model. Upper right: Inferred number of hybridizations with inferred gene tree topologies from sequences. Lower left: Distance between the true network and inferred network. Lower right: Inferred inheritance probabilities (averages over 30 runs with standard deviation bars shown).

7 Analysis of a house mouse (*Mus musculus*) data set

Using our method of inferring a phylogenetic network, we analyzed a house mouse (*Mus musculus*) data set.

M. musculus **samples.** Our *Mus musculus domesticus* samples were provided by the previous study of [26] and represent one population from France (in the Massif Central) and another population from Germany (in the vicinity surrounding Cologne and Bonn). The

Mus musculus musculus samples in our study also came from previous studies [26, 4, 35] and represent a population in Czechoslovakia (Studenec) [26], another population in Kazakhstan (Almaty) [26], and a third population from China (Urumqi in Xinjiang Province) [4, 35]. For simplicity, these five populations will be referred to as *DF*, *DG*, *MZ*, *MK* and *MC*, respectively.

Sequence data. Genome-wide sequence data for our samples was produced using the Mouse Diversity Array [34]. We called genotypes from raw intensity values for the Chinese *M. m. musculus* samples using the procedure described in [4]; genotypes for all other *M. musculus* samples were provided by [26]. Since our computational pipeline was constructed to analyze substitution-based variation, we filtered loci exhibiting short indel or structural variation found in previously reported whole-genome sequencing of *M. musculus* strains (including wild-derived *M. m. domesticus* and *M. m. musculus* strains) [10, 33]. We used the most recent *M. musculus* reference genome coordinates as of this writing (version GRCm38.p2) throughout our study.

The reference *Rattus norvegicus* genome (version RGSC Rnor_5.0) was used as an outgroup in our analyses. Orthology between the *R. norvegicus* genome and *M. musculus* was determined using the BLASTZ-produced [23] pairwise genome alignment provided by the UCSC Genome Browser [14].

In total, 387,923 loci from the *M. musculus* genome were sampled in our data sets.

Local phylogeny estimation. Genotypes were phased into haplotypes and missing bases were imputed using fastPHASE [22]. A larger superset of 416 *M. musculus* samples from the studies of [35, 4, 26] were used for this purpose.

We then estimated local phylogenies along haplotype sequences using a custom analytical pipeline. To satisfy the assumption of no intralocus recombination required by our new method, breakpoints inducing recombination-free intervals were inferred on haplotypes using the Four-Gamete Test [7]. We inferred phylogenies between breakpoints using the maximum likelihood method implemented in [18]. The maximum likelihood phylogenetic analysis used the General Time Reversible substitution model [21] with the CAT model of rate variation across sites [25]. Local phylogenies were rooted using *R. norvegicus* as an outgroup. To satisfy the assumption of free recombination between loci, as required by our new method, local phylogenies were sampled at 100 kb intervals so that linkage disequilibrium was negligible [26]. In total, 20639 local phylogenies were reconstructed.

Species phylogeny inference. From the reconstructed gene trees, we inferred the optimal phylogenetic networks with 0, 1, 2 and 3 reticulation nodes, respectively, using our method described in Section 2, 4.1 and 4.2 (only topologies of gene trees were used). For each of them, the search was run 50 times and top 5 networks were saved. All other parameters were set to their default values as listed in Section 8.

Since all five populations under analysis are closely related, most of the reconstructed gene trees were not binary due to identical sequences of multiple alleles. As bootstrap is not doable in this case, given the very short sequences for each locus and the low signal, we treated uncertainty differently. Consider a non-binary gene tree G that is inferred for some locus. Then, we use the following equation for the probability of G:

$$P(G|\Psi,\Gamma) = \sum_{g' \in b(G)} P(g'|\Psi,\Gamma), \tag{10}$$

where $b(G)$ is the set of all binary resolutions of G. We then used this term in the likelihood formulation based on gene tree topologies.

The results are shown in Fig. 17. Furthermore, to account for model complexity, we calculated the values of three information criteria, AIC, AICc and BIC, as well as the error of cross-validation, for the optimal inferred networks with the number of reticulation nodes from 0 to 3 respectively, as shown in Table 3. More specifically, we did 10-fold cross-validation and only binary gene trees in the validation sets were used to calculate the error. We can see in the table that the error keeps decreasing from optimal network with 0 reticulation node to the one with 2 reticulation nodes, and there is no improvement from optimal network with 2 reticulation nodes to the one with 3 reticulation nodes. Similar trend holds for all three information criteria, where the improvement from the optimal network with 2 reticulation nodes to the one with 3 is relatively small compared to that from the optimal network with 0 reticulation node to the one with 1, as well as the optimal network with 1 reticulation node to the one with 2.

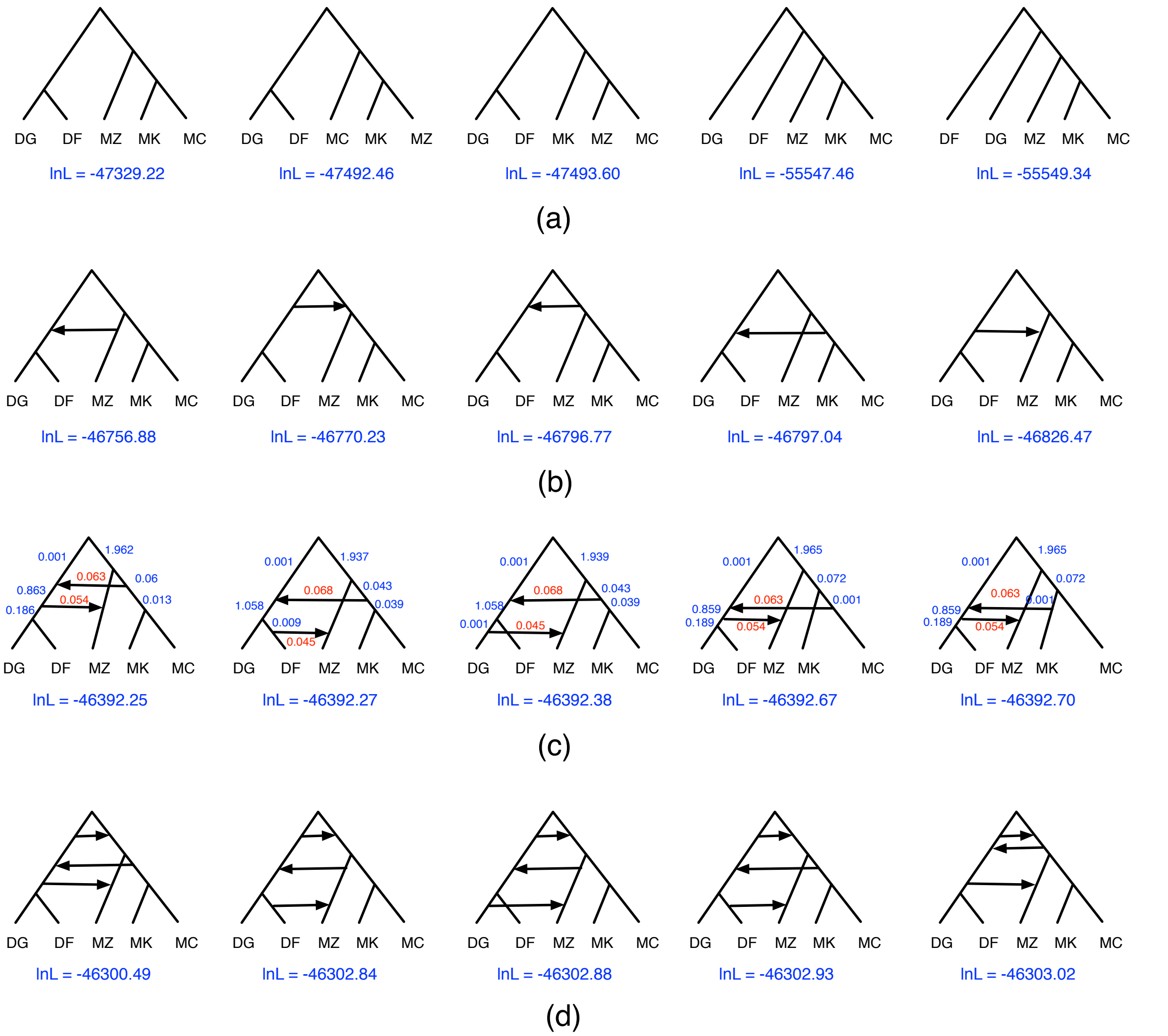


Figure 17: The inferred phylogenetic networks of the *M. musculus* dataset. The rows from top to bottom contain top 5 phylogenetic networks with 0, 1, 2 and 3 reticulation nodes, respectively. In each row, networks are listed from left to right with an decreasing value of log likelihood shown under each of them. Branch lengths and inheritance probabilities are shown for the networks with two reticulations.

Furthermore, in order to check how the search covered the space of phylogenetic networks, we exhaustively enumerated all networks with 1 reticulation node and calculated their likelihood scores. More specifically, we first listed all possible 105 binary species trees over 5 taxa (DF, DG, MZ, MK and MC). Then from each of them, say st, we cal-

	lnL	AIC	AICc	BIC	Error of cross-validation
$N(0)$	-47329	94664	94664	94688	7.69×10^{-5}
$N(1)$	-46756	93527	93527	93583	5.36×10^{-5}
$N(2)$	-46392	92806	92806	92893	4.03×10^{-5}
$N(3)$	-46300	92635	92635	92754	4.13×10^{-5}

Table 3: The results of information criteria and cross validation of the optimal inferred species networks of the *M. musculus* dataset. $N(k)$ refers to the optimal inferred species network with k reticulation nodes.

culated the likelihood score of every network in $\delta_1(st)$. We ordered all of them by their likelihood scores, and found the top 5 were exactly what we obtained by our heuristic search in Fig. 17(b).

8 PhyloNet implementation and use

PhyloNet [30] is an open-source software our group developed for phylogenetic analysis. All methods we discussed in this paper are implemented in it. We illustrate the usage of the command *inferNetwork_ML* which infers a phylogenetic network from a set of gene trees. It takes a set of gene trees and the maximum number of reticulations and returns optimal inferred phylogenetic networks along with branch lengths and inheritance probabilities. There are many parameters for the users to specify; See Table 4 for details.

```
InferNetwork_ml (gt1 [, gt2...]) numReticulations [-a taxaMap] [-bl] [-b threshold] [-s startingNetwork] [-n numNetReturned] [-h {s1 [, s2...]}] [-w (w1,w2,w3,w4)] [-f maxFailure] [-x numRuns] [-m maxNetExamined] [-d maxDiameter] [-p (rel,abs)] [-r maxRounds] [-t maxTryPerBr] [-i improveThreshold] [-l maxBL] [-pl numProcessors] [-di]
```

Parameter	Illustration	Default
$(gt_1\ [,\ gt_2\ldots])$	Comma delimited list of gene tree identifiers.	-
numReticulations	Maximum number of reticulations to add to the species network.	-
-a *taxaMap*	Gene tree / species network taxa association.	-
-bl	Use the branch lengths of the gene trees for the inference.	No
-b *threshold*	Gene trees bootstrap threshold. Edges of gene trees whose bootstrap values are under it will be contracted.	100
-s *startingNetwork*	The network to start search from.	MDC tree
-n *numNetReturned*	Number of top optimal networks to return.	1
-h $\{s_1\ [,\ s_2\ldots]\}$	A set of specified hybrid species. The size of this set equals the number of reticulation nodes in the inferred network.	-
-w (w_1, w_2, w_3, w_4)	The weights of operations (δ_1, δ_2, δ_3, δ_4) for network arrangement during the network search.	$(0.15, 0.15, 0.2, 0.5)$
-f *maxFailure*	The maximum number of consecutive failures before the search terminates.	100
-x *numRuns*	The number of runs of the search.	10
-m *maxNetExamined*	Maximum number of network topologies to examine during the search in each run.	$+\infty$
-d *maxDiameter*	Maximum diameter to make an rearrangement during network search.	$+\infty$
-p (*rel, abs*)	The original stopping criterion of Brents algorithm for optimizing branch lengths and inheritance probabilities of a network.	$(0.01, 0.001)$
-r *maxRound*	Maximum number of rounds to optimize branch lengths and inheritance probabilities for a network topology.	100
-t *maxTryPerBr*	Maximum number of trial per branch in one round to optimize branch lengths and inheritance probabilities for a network topology.	100
-i *improveThreshold*	Minimum threshold of improvement to continue the next round of optimization of branch lengths and inheritance probabilities.	0.001
-l *maxBL*	Maximum branch lengths considered during optimization.	6
-pl *numProcessors*	Number of processors if you want the computation to be done in parallel.	1
-di	Output the Rich Newick string of the inferred network that can be read by Dendroscope [8].	No

Table 4: The usage of command *inferNetwork_ML* in PhyloNet. The first two parameters are mandatory and all others are optional.

References

[1] H. Akaike. A new look at the statistical model identification. *IEEE Trans Automat Contr*, 19:716–723, 1974.

[2] R.P. Brent. *Algorithms for Minimization without Derivatives*. Prentice- Hall, Englewood Clifts, New Jersey, 1973.

[3] J.H. Degnan and L.A. Salter. Gene tree distributions under the coalescent process. *Evolution*, 59:24–37, 2005.

[4] John Didion, Hyuna Yang, Keith Sheppard, Chen-Ping Fu, Leonard McMillan, Fernando de Villena, and Gary Churchill. Discovery of novel variants in genotyping arrays improves genotype retention and reduces ascertainment bias. *BMC Genomics*, 13(1):34, 2012.

[5] J. Felsenstein. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39:783–791, 1985.

[6] R. R. Hudson. Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18:337–338, 2002.

[7] Richard R. Hudson and Norman L. Kaplan. Statistical properties of the number of recombination events in the history of a sample of DNA sequences. *Genetics*, 111(1):147–164, 1985.

[8] D.H. Huson, D.C. Richter, C. Rausch, T. Dezulian, M. Franz, and R. Rupp. Dendroscope: An interactive viewer for large phylogenetic trees. *BMC Bioinformatics*, 8(1):460, 2007.

[9] D.H. Huson, R. Rupp, and C. Scornavacca. *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press, New York, 2010.

[10] Thomas M. Keane, Leo Goodstadt, Petr Danecek, Michael A. White, Kim Wong, Binnaz Yalcin, Andreas Heger, Avigail Agam, Guy Slater, Martin Goodson, Nicholas A. Furlotte, Eleazar Eskin, Christoffer Nellaker, Helen Whitley, James Cleak, Deborah Janowitz, Polinka Hernandez-Pliego, Andrew Edwards, T. Grant Belgard, Peter L. Oliver, Rebecca E. McIntyre, Amarjit Bhomra, Jerome Nicod, Xiangchao Gan, Wei Yuan, Louise van der Weyden, Charles A. Steward, Sendu Bala, Jim Stalker, Richard Mott, Richard Durbin, Ian J. Jackson, Anne Czechanski, Jose Afonso Guerra-Assuncao, Leah Rae Donahue, Laura G. Reinholdt, Bret A. Payseur, Chris P. Ponting, Ewan Birney, Jonathan Flint, and David J. Adams. Mouse genomic variation and its effect on phenotypes and gene regulation. *Nature*, 477(7364):289–294, Sep 2011.

[11] J. F. C. Kingman. The coalescent. *Stochast. Proc. Appl.*, 13:235–248, 1982.

[12] J. F. C. Kingman. On the genealogy of large populations. *J. Appl. Prob.*, 19A:27–43, 1982.

[13] W. P. Maddison. Gene trees in species trees. *Syst. Biol.*, 46:523–536, 1997.

[14] Laurence R. Meyer, Ann S. Zweig, Angie S. Hinrichs, Donna Karolchik, Robert M. Kuhn, Matthew Wong, Cricket A. Sloan, Kate R. Rosenbloom, Greg Roe, Brooke Rhead, Brian J. Raney, Andy Pohl, Venkat S. Malladi, Chin H. Li, Brian T. Lee, Katrina Learned, Vanessa Kirkup, Fan Hsu, Steve Heitner, Rachel A. Harte, Maximilian Haeussler, Luvina Guruvadoo, Mary Goldman, Belinda M. Giardine, Pauline A. Fujita, Timothy R. Dreszer, Mark Diekhans, Melissa S. Cline, Hiram Clawson, Galt P. Barber, David Haussler, and W. James Kent. The UCSC Genome Browser database: extensions and updates 2013. *Nucleic Acids Research*, 41(D1):D64–D69, 2013.

[15] Price M.N., P.S. Dehal, and A.P. Arkin. Fasttree: Computing large minimum-evolution trees with profiles instead of a distance matrix. *Molecular Biology and Evolution*, 26:1641–1650, 2009.

[16] Price M.N., P.S. Dehal, and A.P. Arkin. Fasttree 2 – approximately maximum-likelihood trees for large alignments. *PLoS ONE*, 5(3):e9490, 2010.

[17] L. Nakhleh. Evolutionary phylogenetic networks: models and issues. In L. Heath and N. Ramakrishnan, editors, *The Problem Solving Handbook for Computational Biology and Bioinformatics*, pages 125–158. Springer, New York, 2010.

[18] M. Price, P. Dehal, and A. Arkin. FastTree 2 - approximately maximum-likelihood trees for large alignments. *PLoS ONE*, 5(3):e9490, March 2010.

[19] A. Rambaut and N. C. Grassly. Seq-gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Comp. Appl. Biosci.*, 13:235–238, 1997.

[20] B. Rannala and Z. Yang. Bayes estimation of species divergence times and ancestral population size using DNA sequences from multiple loci. *Genetics*, 164:1645–1656, 2003.

[21] F. Rodriguez, J.L. Oliver, A. Marin, and J.R. Medina. The general stochastic model of nucleotide substitution. *Journal of Theoretical Biology*, 142:485–501, 1990.

[22] Paul Scheet and Matthew Stephens. A fast and flexible statistical model for large-scale population genotype data: Applications to inferring missing genotypes and haplotypic phase. *The American Journal of Human Genetics*, 78(4):629 – 644, 2006.

[23] Scott Schwartz, W. James Kent, Arian Smit, Zheng Zhang, Robert Baertsch, Ross C. Hardison, David Haussler, and Webb Miller. Human-mouse alignments with BLASTZ. *Genome Research*, 13(1):103–107, 2003.

[24] G.E. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

[25] A. Stamatakis. Phylogenetic models of rate heterogeneity: A high performance computing perspective. In *Proceedings of IPDPS2006*, HICOMB Workshop, Rhodos, Greece, April 2006.

[26] Fabian Staubach, Anna Lorenc, Philipp W. Messer, Kun Tang, Dmitri A. Petrov, and Diethard Tautz. Genome patterns of selection and introgression of haplotypes in natural populations of the house mouse (*Mus musculus*). *PLoS Genet*, 8(8):e1002891, 08 2012.

[27] D. L. Swofford. PAUP*: Phylogenetic analysis using parsimony (and other methods), 1996. Sinauer Associates, Underland, Massachusetts, Version 4.0.

[28] S. Tavaré. Line-of-descent and genealogical processes, and their applications in population genetics models. *Theor. Pop. Biol.*, 26:119–164, 1984.

[29] C. Than and L. Nakhleh. Species tree inference by minimizing deep coalescences. *PLoS Computational Biology*, 5(9):e1000501, 2009.

[30] C. Than, D. Ruths, and L. Nakhleh. PhyloNet: a software package for analyzing and reconstructing reticulate evolutionary relationships. *BMC Bioinformatics*, 9(1):322, 2008.

[31] C. Than, D. Ruths, and L. Nakhleh. PhyloNet: a software package for analyzing and reconstructing reticulate evolutionary relationships. *BMC Bioinformatics*, 9:322, 2008.

[32] Y. Wu. Coalescent-based species tree inference from gene tree topologies under incomplete lineage sorting by maximum likelihood. *Evolution*, 66:763–775, 2012.

[33] Binnaz Yalcin, Kim Wong, Avigail Agam, Martin Goodson, Thomas M. Keane, Xiangchao Gan, Christoffer Nellaker, Leo Goodstadt, Jerome Nicod, Amarjit Bhomra, Polinka Hernandez-Pliego, Helen Whitley, James Cleak, Rebekah Dutton, Deborah Janowitz, Richard Mott, David J. Adams, and Jonathan Flint. Sequence-based characterization of structural variation in the mouse genome. *Nature*, 477(7364):326–329, Sep 2011.

[34] Hyuna Yang, Yueming Ding, Lucie N. Hutchins, Jin Szatkiewicz, Timothy A. Bell, Beverly J. Paigen, Joel H. Graber, Fernando Pardo-Manuel de Villena, and Gary A. Churchill. A customized and versatile high-density genotyping array for the mouse. *Nat Meth*, 6(9):663–666, Sep 2009.

[35] Hyuna Yang, Jeremy R. Wang, John P. Didion, Ryan J. Buus, Timothy A. Bell, Catherine E. Welsh, Francois Bonhomme, Alex Hon-Tsen Yu, Michael W. Nachman, Jaroslav Pialek, Priscilla Tucker, Pierre Boursot, Leonard McMillan, Gary A. Churchill, and Fernando Pardo-Manuel de Villena. Subspecific origin and haplotype diversity in the laboratory mouse. *Nat Genet*, 43(7):648–655, Jul 2011.

[36] Y. Yu, J.H. Degnan, and L. Nakhleh. The probability of a gene tree topology within a phylogenetic network with applications to hybridization detection. *PLoS Genetics*, 8:e1002660, 2012.

[37] Y. Yu, N. Ristic, and L. Nakhleh. Fast algorithms and heuristics for phylogenomics under ILS and hybridization. *BMC Bioinformatics*, 14:S6, 2013.

[38] Y. Yu, T. Warnow, and L. Nakhleh. Algorithms for MDC-based multi-locus phylogeny inference. *The 15th Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, LNBI 6577:531–545, 2011.

[39] Y. Yu, T. Warnow, and L. Nakhleh. Algorithms for MDC-based multi-locus phylogeny inference: Beyond rooted binary gene trees on single alleles. *Journal of Computational Biology*, 18:1543–1559, 2011.

[40] S. Zhu, J.H. Degnan, and B. Eldon. Hybrid-Lambda: simulation of multiple merger and Kingman gene genealogies in species networks and species trees. *pre-print*, 2013. arXiv:1303.0673 [q-bio.PE].