The Maximum Agreement of Two Nested Phylogenetic Networks

Jesper Jansson and Wing-Kin Sung

School of Computing, National University of Singapore, 3 Science Drive 2,
Singapore 117543
{jansson, ksung}@comp.nus.edu.sg

Abstract. Given a set $\mathcal{N}$ of phylogenetic networks, the maximum agreement phylogenetic subnetwork problem (MASN) asks for a subnetwork contained in every $N_i \in \mathcal{N}$ with as many leaves as possible. MASN can be used to identify shared branching structure among phylogenetic networks or to measure their similarity. In this paper, we prove that the general case of MASN is NP-hard already for two phylogenetic networks, but that the problem can be solved efficiently if the two given phylogenetic networks exhibit a nested structure. We first show that the total number of nodes $|V(N)|$ in any nested phylogenetic network N with n leaves and nesting depth d is $O(n(d+1))$. We then describe an algorithm for testing if a given phylogenetic network is nested, and if so, determining its nesting depth in $O(|V(N)| \cdot (d+1))$ time. Next, we present a polynomial-time algorithm for MASN for two nested phylogenetic networks N_1, N_2. Its running time is $O(|V(N_1)| \cdot |V(N_2)| \cdot (d_1+1) \cdot (d_2+1))$, where d_1 and d_2 denote the nesting depths of N_1 and N_2, respectively. In contrast, the previously fastest algorithm for this problem runs in $O(|V(N_1)| \cdot |V(N_2)| \cdot 4^f)$ time, where $f \geq \max\{d_1, d_2\}$.

1 Introduction

Phylogenetic trees are commonly used to describe evolutionary relationships among a set of objects (e.g., biological species, proteins, viruses, or languages) produced by an evolutionary process, and can help scientists to understand the mechanisms of evolution as well as to classify the objects being studied and to organize information [15, 19]. However, evolutionary events such as horizontal gene transfer or hybrid speciation (often referred to as *recombination events*) which suggest convergence between objects cannot be adequately represented in a single tree structure [3, 10, 11, 12, 16, 17, 18, 21]. Phylogenetic networks were introduced in order to solve this shortcoming by allowing internal nodes to have more than one parent so that each recombination event may be represented by a node with indegree greater than one. Various methods for constructing and comparing phylogenetic networks have been proposed recently [3, 4, 10, 12, 13, 16, 17, 18, 21].

Phylogenetic network *comparison* has many uses; one application described in [16] is to assess the topological accuracy of different phylogenetic network con-

R. Fleischer and G. Trippen (Eds.): ISAAC 2004, LNCS 3341, pp. 581–593, 2004.
© Springer-Verlag Berlin Heidelberg 2004

struction methods[1]. Another application for network comparison is to identify a subnetwork with as many leaves as possible which is contained in all of the networks in a given set (obtained, for example, by different construction methods or by using the same method on alternative data sets) to determine which ancestral relationships are present in all networks. Moreover, the size of such a subnetwork provides a measure of how similar the networks in a given set are. This problem was formalized as a computational problem called *the maximum agreement phylogenetic subnetwork problem* (MASN) and initially studied in [4].

The general case of MASN is NP-hard for three or more phylogenetic networks [4]. In fact, it is NP-hard even for just *two* networks, as we prove in this paper. Fortunately, recombination events usually do not occur in an unrestricted manner [10, 21]. It is therefore important to know under what structural restrictions on the input networks the problem becomes efficiently solvable. Here, we investigate the computational complexity of MASN for two phylogenetic networks whose merge paths are *nested*, which is a natural generalization of rooted, leaf-labeled trees and so called galled-trees previously studied in [10, 13, 17, 21] (see below for definitions), and prove that this case can be solved by a polynomial-time algorithm. The decomposition technique for nested phylogenetic networks we develop here may also be applicable to other computational and combinatorial problems related to phylogenetic network construction and comparison.

1.1 Problem Definition and Terminology

A *phylogenetic network* is a connected, rooted, simple, directed acyclic graph in which: (1) each node has outdegree at most 2; (2) each node has indegree 1 or 2, except the root node which has indegree 0; (3) no node has both indegree 1 and outdegree 1; and (4) all nodes with outdegree 0 are labeled by elements from a finite set L in such a way that no two nodes are assigned the same label. From here on, nodes of outdegree 0 are referred to as *leaves* and identified with their corresponding elements in L. We denote the set of all nodes and the set of leaves in a phylogenetic network N by $V(N)$ and $\Lambda(N)$, respectively.

Given a phylogenetic network N and a set L', *the topological restriction* of N to L', denoted by $N \,|\, L'$, is defined as the phylogenetic network obtained by first deleting all nodes which are not on any directed path from the root to a leaf in L' along with their incident edges, and then, for every node with outdegree 1 and indegree less than 2, contracting its outgoing edge (any resulting set of multiple edges between two nodes is replaced by a single edge).

Given a set $\mathcal{N} = \{N_1, \ldots, N_k\}$ of phylogenetic networks, an *agreement subnetwork of* $\mathcal{N}$ is a phylogenetic network A such that $\Lambda(A) \subseteq \bigcap_{N_i \in \mathcal{N}} \Lambda(N_i)$ and for every $N_i \in \mathcal{N}$, A is isomorphic to a subgraph of $N_i \,|\, \Lambda(A)$ in which zero or more of the edges have been deleted and each outgoing edge from a node with

[1] To evaluate a construction method $\mathcal{M}$, repeat the following steps a number of times. First, randomly generate a network N and evolve a sequence down the edges of N according to some chosen model of evolution, then build a network N' for the resulting set of sequences using $\mathcal{M}$, and finally measure the similarity between N' and N.

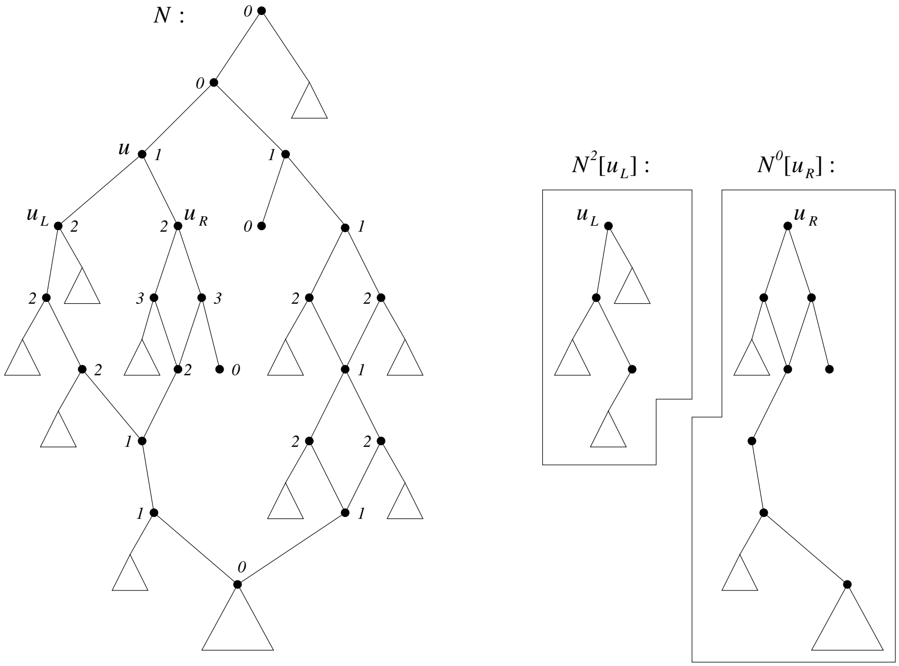

Fig. 1. N is a nested phylogenetic network with nesting depth 3 and u is a split node in N. The numbers shown next to the nodes of N are their respective nesting depths. $N^2[u_L]$ and $N^0[u_R]$ are the subgraphs of N displayed on the right

resulting outdegree 1 has been contracted. A *maximum agreement subnetwork of* $\mathcal{N}$ is an agreement subnetwork of $\mathcal{N}$ with the maximum possible number of leaves. *The maximum agreement phylogenetic subnetwork problem* (MASN) is: Given a set $\mathcal{N} = \{N_1, \ldots, N_k\}$ of phylogenetic networks, find a maximum agreement subnetwork of $\mathcal{N}$. A leaf can appear in a maximum agreement subnetwork of $\mathcal{N}$ only if it is present in every network in $\mathcal{N}$, so we assume without loss of generality that $\Lambda(N_1) = \ldots = \Lambda(N_k)$ and call this leaf set L. Throughout this paper, we let n denote the number of different leaves and k the number of input networks, i.e., $n = |L|$ and $k = |\mathcal{N}|$ in the problem definition above.

To describe our results, we need the following terminology. Let N be a phylogenetic network. Recall that nodes with outdegree 0 are called *leaves*. We refer to nodes with indegree 2 as *hybrid nodes*. For any hybrid node h, every ancestor s of h such that h can be reached using two disjoint directed paths starting at the children of s is called a *split node of* h. If s is a split node of h then any path starting at s and ending at h is a *merge path of* h, and any path starting at a child of s and ending at a parent of h is a *clipped merge path of* h.

We say that N is a *nested phylogenetic network* if for every two merge paths P_1, P_2 of two different hybrid nodes h_1, h_2, either P_1 and P_2 are disjoint, one is a subpath of the other, or their intersection equals either h_1 or h_2. For each node u in a nested phylogenetic network N, define the *nesting depth of* u, $d(u)$, as the number of hybrid nodes in N that have a clipped merge path passing through u. See Fig. 1 for an example. The *nesting depth of* N, denoted by $d(N)$, is the maximum value of $d(u)$ over all $u \in V(N)$. Note that if $d(N) = 0$ then N is a tree. Gusfield *et al.* [10] defined a *galled-tree* (also referred to in the literature as a *gt-network* [17] or a *topology with independent recombination events* [21]) as a phylogenetic network in which all clipped merge paths are disjoint. For a discussion on the biological significance of galled-trees, see [10]. Clearly, $d(N) \leq 1$ if and only if N is a galled-tree. Thus, nested phylogenetic networks naturally extend the notion of rooted, leaf-labeled trees and galled-trees.

Finally, given any phylogenetic network N, let $\mathcal{U}(N)$ be the undirected graph obtained from N by replacing each directed edge by an undirected edge. N is said to be a *level-f* phylogenetic network if, for every biconnected component B in $\mathcal{U}(N)$, the subgraph of N induced by the set of nodes in B contains at most f nodes with indegree 2. If $f = 0$ then N is a tree, and we have $f = 1$ if and only if N is a nested phylogenetic network with nesting depth 1. If N is a nested phylogenetic network with nesting depth d then $f \geq d$.

1.2 Previous Results

Median-joining, split decomposition (SplitsTree), PYRAMIDS, statistical parsimony (TCS), molecular-variance parsimony (Arlequin), reticulogram (T-REX), and netting are some general methods for *constructing* phylogenetic networks (see [18] for a survey). More recently presented methods include NeighborNet [3] and Z-closure [12]. Algorithms for some reconstruction problems with additional constraints on the networks were given in [10, 13, 17, 21]; in particular, these papers considered problems involving constructing a network with nesting depth 1.

As for *comparing* two given networks, one method based on the Robinson-Foulds (RF) measure for phylogenetic trees was proposed in [16]. MASN was introduced in [4], where it was shown to be NP-hard if restricted to $k = 3$ and an $O(n^2)$-time algorithm for the special case of two level-1 phylogenetic networks (i.e., having nesting depth 1) was presented. [4] also showed that MASN for two level-f networks N_1 and N_2 can be solved in $O(|V(N_1)| \cdot |V(N_2)| \cdot 4^f)$ time.

MASN generalizes a well-studied problem known as *the maximum agreement subtree problem* (MAST)[2] (see, e.g., [1, 2, 5, 7, 9, 14, 20]) in which the input is a set of distinctly leaf-labeled trees and the goal is to compute a tree embedded in all of the input trees with the maximum possible number of labeled leaves. The fastest known algorithm for MAST for two trees runs in $O(\sqrt{D}\, n \log(2n/D))$ time, where n is the number of leaves and D is the maximum degree of the two input trees [14]. Note that this is $O(n \log n)$ for two trees with D bounded by a constant and $O(n^{1.5})$ for two trees with unbounded D. MAST is NP-hard for three trees with unbounded degrees [1], and solvable in $O(kn^3 + n^\delta)$ time for $k \geq 3$ trees, where δ is an upper bound on at least one of the input trees' degrees [2, 7].

[2] MAST is also known as *the maximum homeomorphic subtree problem* (MHT).

1.3 Our Results and Organization of Paper

In this paper, we focus on MASN for two nested phylogenetic networks. In Section 2, we derive some useful combinatorial properties of nested networks. We first prove that $|V(N)| = O(n(d+1))$ for any nested network N with n leaves and nesting depth d and then show how to test whether a given phylogenetic network is nested, and if so, determine its nesting depth in $O(|V(N)| \cdot (d+1))$ time. In Section 3, we present a simple and fast algorithm for solving MASN for two nested networks N_1 and N_2 running in $O(|V(N_1)| \cdot |V(N_2)| \cdot (d_1+1) \cdot (d_2+1))$ time, where d_1 and d_2 are the nesting depths of N_1 and N_2, respectively. (The algorithm given in [4] could be applied here but its running time is $O(|V(N_1)| \cdot |V(N_2)| \cdot 4^f)$, where $f \geq \max\{d_1, d_2\}$.) For the special case $d_1 = 1$, $d_2 = 1$, i.e., two level-1 networks, the running time of our new algorithm coincides with the running time of $O(n^2)$ of the algorithm in [4]. Next, in Section 4, we strengthen the NP-hardness result of [4] by proving that MASN is NP-hard already for *two* phylogenetic networks[3]. Finally, we discuss some open problems in Section 5. Proofs omitted due to space limitations will appear in the full-length version of this paper.

2 Preliminaries

We first state some basic properties of nested phylogenetic networks.

Lemma 1. *If N is a nested network then each split node in N is a split node of exactly one hybrid node, and each hybrid node in N has exactly one split node.*

Because of Lemma 1, each hybrid node in a nested phylogenetic network corresponds to a unique split node. For any such hybrid node h and split node s, s is called *the split node of h* and h is called *the hybrid node of s*.

Lemma 2. *Let h be a hybrid node in a nested phylogenetic network and let s be the split node of h. Then $d(h) = d(s)$.*

We now derive an upper bound on the total number of nodes in a nested phylogenetic network. The next two lemmas generalize Lemma 3.2 in [4].

Lemma 3. *If N is a nested phylogenetic network with n leaves and nesting depth d then the number of hybrid nodes in N is at most $(n-1) \cdot d$.*

Proof. Let $T_N(d)$ be the network N. For $i \in \{0, 1, \ldots, d-1\}$, define $T_N(i)$ as the directed graph constructed from $T_N(i+1)$ as follows. For every hybrid node h in $T_N(i+1)$ with $d(h) = i$, remove h's two incoming edges, contract the split node of h and all nodes on the two clipped merge paths of h to a single node s, and add

[3] The reduction in [1] for proving the NP-hardness of MAST restricted to three trees with unbounded degrees cannot be used directly for MASN with $k = 2$ because it constructs *three* trees and because here we require all nodes to have outdegree at most two. Interestingly, MAST for two binary trees is solvable in $O(n \log n)$ time [5, 14].

a directed edge from s to h. $T_N(0)$ is a tree because every node with indegree 2 in N has indegree 1 in $T_N(0)$ and no contraction increases the indegree of any node. $T_N(0)$ has n leaves, so the number of internal nodes in $T_N(0)$ with outdegree > 1 is at most $n-1$. Observe that at most d split nodes in N correspond to each internal node in $T_N(0)$ with outdegree > 1 and that the number of hybrid nodes in N equals the number of split nodes in N since N is nested. □

Lemma 4. *If N is a phylogenetic network with n leaves and H hybrid nodes then the total number of nodes in N is at most $2(n+H)-1$.*

Proof. Let z_{ij} denote the number of nodes in N which have i incoming edges and j outgoing edges. By the definition of a phylogenetic network, the total number of nodes in N is $z_{02}+z_{10}+z_{12}+z_{20}+z_{21}+z_{22}$. For every $u \in V(N)$, let $in(u)$ and $out(u)$ denote the number of incoming and outgoing edges incident to u. Since

$$\begin{cases} \sum\limits_{u \in V(N)} in(u) \;=\; z_{02} \cdot 0 \;+\; (z_{10}+z_{12}) \cdot 1 \;+\; (z_{20}+z_{21}+z_{22}) \cdot 2 \\ \sum\limits_{u \in V(N)} out(u) \;=\; (z_{10}+z_{20}) \cdot 0 \;+\; z_{21} \cdot 1 \;+\; (z_{02}+z_{12}+z_{22}) \cdot 2 \end{cases}$$

and $\sum_{u \in V(N)} in(u) = \sum_{u \in V(N)} out(u)$, we have $z_{12} = z_{10} + 2z_{20} + z_{21} - 2z_{02}$. Next, $H = z_{20}+z_{21}+z_{22}$, $n = z_{10}+z_{20}$, and $z_{02} = 1$ give us $z_{12} \le n+H-2$. Hence, $|V(N)| \le 1+n+(n+H-2)+H = 2n+2H-1$. □

Theorem 1. *If N is a nested phylogenetic network with n leaves and nesting depth d then $|V(N)| = O(n(d+1))$.*

Theorem 2. *Let N be a phylogenetic network with n leaves and H hybrid nodes. We can test whether N is nested in $O(|V(N)| \cdot (H+1))$ time; if N is nested, the test takes only $O(|V(N)| \cdot (d(N)+1))$ time and its nesting depth can be determined in the same asymptotic time bound.*

Proof. Use the following method to construct a list $L(u)$ for every $u \in V(N)$ consisting of all hybrid nodes which have a clipped merge path passing through u, plus u itself if u is a hybrid node. Associate an initially empty list $L(u)$ to each $u \in V(N)$, and define $L(\emptyset) = \emptyset$. Do a postorder traversal of the nodes of N. Whenever a non-leaf node u is visited, examine $L(u_L)$ and $L(u_R)$, where u_L and u_R are the children of u (if u only has one child then let u_R equal $\emptyset$). If $L(u_L)$ is empty then let $L(u) := L(u_R)$; else if $L(u_R)$ is empty then let $L(u) := L(u_L)$. Otherwise, check whether $L(u_L)$ equals $L(u_R)$. If no then N is not nested, and the algorithm terminates; if yes then let $L(u) := L(u_L)$ and remove the last element ℓ from $L(u)$ (here, u is in fact the split node for the hybrid node ℓ). Finally, if u is a hybrid node then insert u at the end of $L(u)$. Note that a node may be both a split node and a hybrid node. No $|L(u)|$ can exceed the number of hybrid nodes in N. Moreover, when the algorithm is finished, if N is a nested phylogenetic network then its nesting depth $d(N)$ equals the maximum length of $L(u)$ over all $u \in V(N)$ since $d(u) = |L(u)|$ for each non-hybrid node u. □

3 An Algorithm for MASN for Two Nested Networks

In this section, we show how to solve MASN for two nested phylogenetic networks N_1, N_2 with n leaves in $O(|V(N_1)| \cdot |V(N_2)| \cdot (d_1 + 1) \cdot (d_2 + 1))$ time, where d_1 and d_2 are the nesting depths of N_1 and N_2, respectively.

Let N be any nested phylogenetic network. From this point onward, assume that some arbitrary left-to-right ordering of the children of every node has been fixed. If $u \in V(N)$ has two children then let u_L and u_R denote the left and right child of u, respectively, and if u only has one child c then set $u_L = c$ and $u_R = \emptyset$. For every $u \in V(N)$, $N[u]$ is the subnetwork of N rooted at u, i.e., the minimal subgraph of N which includes all nodes and directed edges of N reachable from u. $N[\emptyset]$ refers to the empty network with no nodes or edges.

Each $u \in V(N)$ belongs to $d(u)$ different clipped merge paths. Since N is nested, the $d(u)$ different hybrid nodes corresponding to these clipped merge paths have nesting depths 0, 1, …, $d(u) - 1$. For $i \in \{1, \ldots, d(u)\}$, we define $h^i(u)$ as the hybrid node h which has a clipped merge path passing through u and which satisfies $d(h) = i - 1$. Next, for $i \in \{1, \ldots, d(u)\}$, let $N^i[u]$ be the subgraph of $N[u]$ where $N[h^i(u)]$ and $h^i(u)$'s incoming edge have been removed, and let $N^0[u]$ be $N[u]$. Define $N^i[u]$ for $i > d(u)$ as $N^0[u]$ if u is not a hybrid node, and as $N[\emptyset]$ if u is a hybrid node. See Fig. 1 for an example. Intuitively, the parameter i informs us at which descendant hybrid node of u to cut $N[u]$ to obtain $N^i[u]$.

Lemma 5. *For any nested phylogenetic network N, $u \in V(N)$, and $0 \leq j < i \leq d(u)$, it holds that $N^i[u]$ is a proper subgraph of $N^j[u]$.*

Lemma 6. *Let N be a nested phylogenetic network. For any $u \in V(N)$ and $i \in \{0, 1, \ldots, d(u)\}$, we have: (1) $N^i[u_L]$ and $N^x[u_R]$ are disjoint, and (2) $N^x[u_L]$ and $N^i[u_R]$ are disjoint, where $x = d(u) + 1$ if u is a split node and $x = i$ otherwise.*

Proof. If u is a split node then let h be the hybrid node of s. By Lemma 2, $d(h) = d(u)$. Let c_1 be a child of u with $c_1 \neq h$ and let c_2 be the other child of u. We have $h^x(c_1) = h^{d(u)+1}(c_1) = h$, which means that $N^x[c_1]$ does not contain any nodes in $N[h]$; hence, $N^x[c_1]$ and $N^0[c_2]$ are disjoint, and Lemma 5 then implies that $N^x[c_1]$ and $N^i[c_2]$ are disjoint. Similarly, $N^i[c_1]$ and $N^x[c_2]$ are disjoint (if $c_2 \neq h$ then $h^x(c_2) = h^{d(u)+1}(c_2) = h$ so $N^x[c_2]$ contains no nodes in $N[h]$ and thus no nodes in $N^i[c_1]$; if $c_2 = h$ then $N^x[c_2] = N^{d(u)+1}[h] = N^{d(h)+1}[h] = N[\emptyset]$).

If u is not a split node then $N[u_L]$ ($= N^0[u_L]$) and $N[u_R]$ ($= N^0[u_R]$) are always disjoint. By Lemma 5, $N^i[u_L]$ and $N^i[u_R]$ are disjoint. □

For any two phylogenetic networks N_1, N_2, define $Masn(N_1, N_2)$ as the number of leaves in a maximum agreement subnetwork. If N_1 or N_2 is an empty network then $Masn(N_1, N_2)$ is 0. Otherwise, $Masn(N_1, N_2)$ for two nested networks can be expressed recursively using the following lemma which is a generalization of the main lemma in [20] for MAST. In the *Match* case, when trying to match two subnetworks $N_1^i[u_L]$ and $N_1^x[u_R]$ to two subnetworks $N_2^k[v_L]$ and $N_2^y[v_R]$, Lemma 6 ensures that the set of nodes in the intersection of $V(N_1[u_L])$ and $V(N_1[u_R])$ is matched to only one of $N_2^k[v_L]$ and $N_2^y[v_R]$, and vice versa.

Lemma 7. *Let N_1 and N_2 be two nested phylogenetic networks. For every $(u, v) \in V(N_1) \times V(N_2)$ and $0 \leq i \leq d(u)$, $0 \leq k \leq d(v)$,*

$$Masn(N_1^i[u], N_2^k[v]) = \begin{cases} |\Lambda(N_1^i[u]) \cap \Lambda(N_2^k[v])|, & \text{if at least one of } u \text{ and } v \\ & \text{is a leaf} \\ \max\{Diag(N_1^i[u], N_2^k[v]),\ Match(N_1^i[u], N_2^k[v])\}, & \\ & \text{otherwise} \end{cases}$$

where

$$Diag(N_1^i[u], N_2^k[v]) = \max\{Masn(N_1^i[u], N_2^k[v_L]),\ Masn(N_1^i[u], N_2^k[v_R]),\ Masn(N_1^i[u_L], N_2^k[v]),\ Masn(N_1^i[u_R], N_2^k[v])\}$$

and

$$\begin{aligned} Match(N_1^i[u], N_2^k[v]) = \\ \max\{ & Masn(N_1^i[u_L],\, N_2^k[v_L]) + Masn(N_1^x[u_R],\, N_2^y[v_R]), \\ & Masn(N_1^i[u_L],\, N_2^y[v_L]) + Masn(N_1^x[u_R],\, N_2^k[v_R]), \\ & Masn(N_1^i[u_L],\, N_2^k[v_R]) + Masn(N_1^x[u_R],\, N_2^y[v_L]), \\ & Masn(N_1^i[u_L],\, N_2^y[v_R]) + Masn(N_1^x[u_R],\, N_2^k[v_L]), \\ & Masn(N_1^x[u_L],\, N_2^k[v_L]) + Masn(N_1^i[u_R],\, N_2^y[v_R]), \\ & Masn(N_1^x[u_L],\, N_2^y[v_L]) + Masn(N_1^i[u_R],\, N_2^k[v_R]), \\ & Masn(N_1^x[u_L],\, N_2^k[v_R]) + Masn(N_1^i[u_R],\, N_2^y[v_L]), \\ & Masn(N_1^x[u_L],\, N_2^y[v_R]) + Masn(N_1^i[u_R],\, N_2^k[v_L])\}, \end{aligned}$$

$$\textit{where } x = \begin{cases} d(u) + 1, & \text{if } u \text{ is a split node} \\ i, & \text{otherwise} \end{cases}$$

$$y = \begin{cases} d(v) + 1, & \text{if } v \text{ is a split node} \\ k, & \text{otherwise} \end{cases}$$

Now, given two nested phylogenetic networks N_1 and N_2, we can use Lemma 7 to compute $Masn(N_1^i[u], N_2^k[v])$ for all $0 \leq i \leq d(u)$ and $0 \leq k \leq d(v)$ by applying dynamic programming in a bottom-up manner. The resulting algorithm (Algorithm *NestedMasn*) is listed in Fig. 2.

Lemma 8. *NestedMasn runs in $O\big(|V(N_1)| \cdot |V(N_2)| \cdot (d(N_1)+1) \cdot (d(N_2)+1)\big)$ time.*

Algorithm *NestedMasn* can be modified to compute the set of leaves in a maximum agreement subnetwork without increasing the asymptotic running time by also recording information about how each *Masn*-value is attained as it is computed, e.g., by saving pointers. To construct an actual maximum agreement subnetwork from such a set L', we may use a standard traceback technique to obtain a tree with leaf set L' which is an agreement subnetwork. This yields:

Theorem 3. *Given two nested phylogenetic networks N_1 and N_2 with nesting depths d_1 and d_2, respectively, a maximum agreement subnetwork can be computed in $O(|V(N_1)| \cdot |V(N_2)| \cdot (d_1 + 1) \cdot (d_2 + 1))$ time.*

```
Algorithm  $NestedMasn$
Input:   Two nested phylogenetic networks $N_1$ and $N_2$.
Output:  The number of leaves in a maximum agreement subnetwork of $\{N_1, N_2\}$.
1  Compute and store $d(u)$ and $h^i(u)$ for all $u \in V(N_1) \cup V(N_2)$, $i \in \{1, \dots, d(u)\}$.
2  Let $\mathcal{O}$ be the lexicographic ordering of $V(N_1) \times V(N_2)$, where the nodes in each
   $V(N_i)$ are ordered according to postorder.
3  for each $(u, v) \in V(N_1) \times V(N_2)$ in increasing order in $\mathcal{O}$ do
       Compute $Masn(N_1^i[u], N_2^k[v])$ for all $0 \le i \le d(u),\ 0 \le k \le d(v)$ by using
       the expression in Lemma 7.
   endfor
4  return $Masn(N_1^0[r_1], N_2^0[r_2])$, where $r_i$ is the root of $N_i$ for $i \in \{1, 2\}$.
End  $NestedMasn$
```

Fig. 2. A dynamic programming algorithm for computing all values of $Masn$

4 MASN with $k = 2$ Is NP-Hard

To prove the NP-hardness of MASN for every fixed $k \geq 2$, we provide a polynomial-time reduction from the following problem.

Three-Dimensional Matching (3DM): Given a set $M \subseteq X \times Y \times Z$, where X, Y, and Z are disjoint sets and $X = \{x_1, ..., x_q\}$, $Y = \{y_1, ..., y_q\}$, $Z = \{z_1, ..., z_q\}$, is there a subset M' of M with $|M'| = q$ such that M' is a matching, i.e., such that for every pair $e, f \in M'$ it holds that e and f differ in all coordinates?

3DM is NP-complete (see, e.g., [8]). Given an arbitrary instance of 3DM, construct an instance of MASN with two phylogenetic networks N_1 and N_2 with a leaf set L as described below. The elements of M are encoded in subtrees called S_{x_i,z_k} in N_1 and in subtrees called U_{y_j} in N_2. The purpose of the subtrees named A_{x_i}, B_{x_i,z_k}, and W_{z_k} is to make sure that for any two triples e and f in M, a maximum agreement subnetwork of N_1 and N_2 can contain both of the two leaves representing e and f if and only if e and f differ in all coordinates.

Take $L = M \cup A \cup B$, where A is a set of $q^6 \cdot (q + 2)$ elements not in M and B is a set of q^6 elements not in M or A. Let $A_{x_0}, \dots, A_{x_q}, A_{x_{q+1}}$ be $q + 2$ binary trees with q^6 leaves each, distinctly labeled by A. For every $(x_i, z_k) \in X \times Z$, let B_{x_i,z_k} be a binary tree with q^4 leaves, distinctly labeled by B. For every $(x_i, z_k) \in X \times Z$, define: (1) M_{x_i,z_k} as the subset of M containing all triples of the form (x_i, y, z_k) where $y \in Y$; and (2) S_{x_i,z_k} to be a tree obtained from a binary caterpillar tree with $|M_{x_i,z_k}| + 1$ leaves distinctly labeled by M_{x_i,z_k} and where one of the bottommost leaves has been replaced by the root of B_{x_i,z_k}. See Fig. 3. For every $y_j \in Y$, define: (1) M_{y_j} as the subset of M containing all triples of the form (x, y_j, z) where $x \in X$ and $z \in Z$; and (2) U_{y_j} to be a binary caterpillar tree with $|M_{y_j}| + q$ leaves in which the $|M_{y_j}|$ leaves closest to the root are distinctly labeled by M_{y_j} and the rest are unlabeled nodes referred to as v_{y_j,z_k} for $1 \leq k \leq q$. Then, for every $z_k \in Z$, define W_{z_k} to be a tree obtained from the binary caterpillar tree with q leaves by replacing the leaves with the roots of B_{x_1,z_k}, …, B_{x_q,z_k}.

Next, let P be any sorting network (see, e.g., [6]) for q elements with a polynomial number p of comparator stages. Build a directed acyclic graph Q from P with $(p+1) \cdot q$ nodes $\{Q_{i,j} \mid 1 \leq i \leq p{+}1,\ 1 \leq j \leq q\}$ such that there is a directed edge $(Q_{i,j}, Q_{i+1,j})$ for every $1 \leq i \leq p$ and $1 \leq j \leq q$, and two directed edges $(Q_{i,j}, Q_{i+1,k})$ and $(Q_{i,k}, Q_{i+1,j})$ for every comparator (j,k) at stage i in P for $1 \leq i \leq p$, as illustrated in Fig. 4. Furthermore, construct q directed paths $\{G_1, \ldots, G_q\}$ where each $G_k = (G_{1,k}, \ldots, G_{q,k})$.

Let N_1 be a phylogenetic network (in fact, a leaf-labeled binary tree) obtained by attaching to a directed path $(m_1, m_2, \ldots, m_{q^2+q+2})$, in order of non-decreasing distance from m_1, the roots of A_{x_0}, S_{x_1,z_1}, S_{x_1,z_2}, …, S_{x_1,z_q}, A_{x_1}, S_{x_2,z_1}, …, S_{x_q,z_q}, A_{x_q}, and $A_{x_{q+1}}$, and letting m_1 be the root of N_1. See Fig. 5. The phylogenetic network N_2 is obtained by first attaching to a directed path $(n_1, n_2, \ldots, n_{2q+2})$, in order of non-decreasing distance from n_1, the root of A_{x_0}, the node $Q_{1,1}$, the root of A_{x_1}, the node $Q_{1,2}$, the root of A_{x_2}, …, the root of A_{x_q}, and the root of $A_{x_{q+1}}$, and letting n_1 be the root of N_2. Then, for $j \in \{1, \ldots, q\}$, let $Q_{p+1,j}$ coincide with the root of U_{y_j}, and for every $1 \leq j \leq q$ and $1 \leq k \leq q$ add a directed edge $(v_{y_j,z_k}, G_{j,k})$. Next, for every $1 \leq k \leq q$ add a directed edge from $G_{q,k}$ to the root of W_{z_k}. Finally, for every node in N_1 and N_2 having indegree 1 and outdegree 1, contract its outgoing edge.

Lemma 9. *There exists an agreement subnetwork of* (N_1, N_2) *with* $q^7 + 2q^6 + q^5 + q$ *leaves if and only if* M *has a matching of size* q.

Theorem 4. *MASN is NP-hard even if restricted to* $k = 2$.

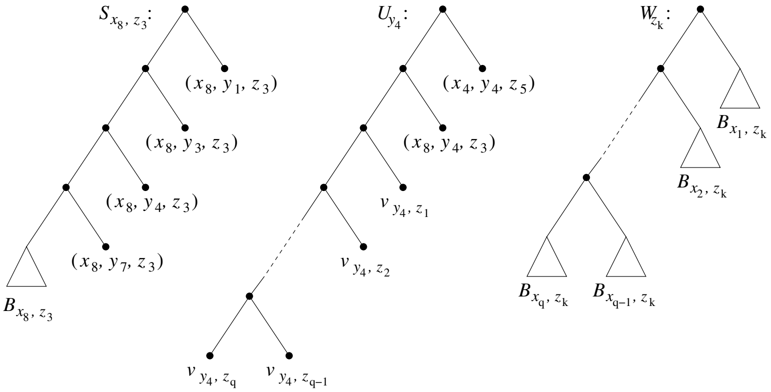

Fig. 3. Assume $M_{x_8,z_3} = \{(x_8, y_1, z_3), (x_8, y_3, z_3), (x_8, y_4, z_3), (x_8, y_7, z_3)\}$ and $M_{y_4} = \{(x_4, y_4, z_5), (x_8, y_4, z_3)\}$. S_{x_8,z_3} and U_{y_4} are shown on the left and in the center, respectively. The structure of each W_{z_k} is shown on the right

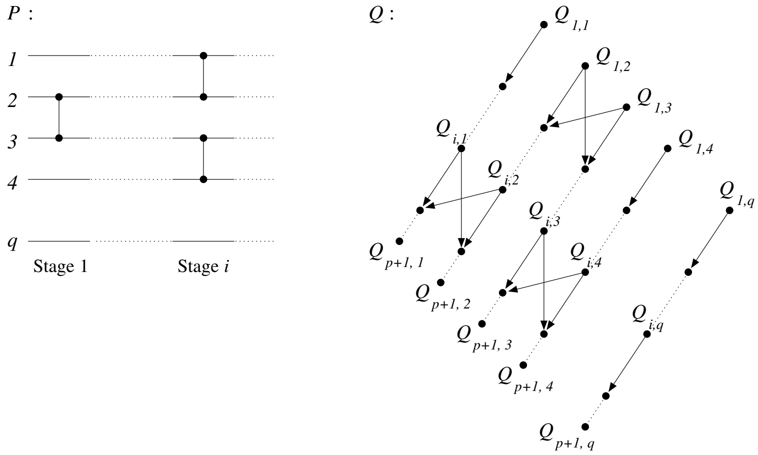

Fig. 4. The sorting network P on the left yields a directed acyclic graph Q

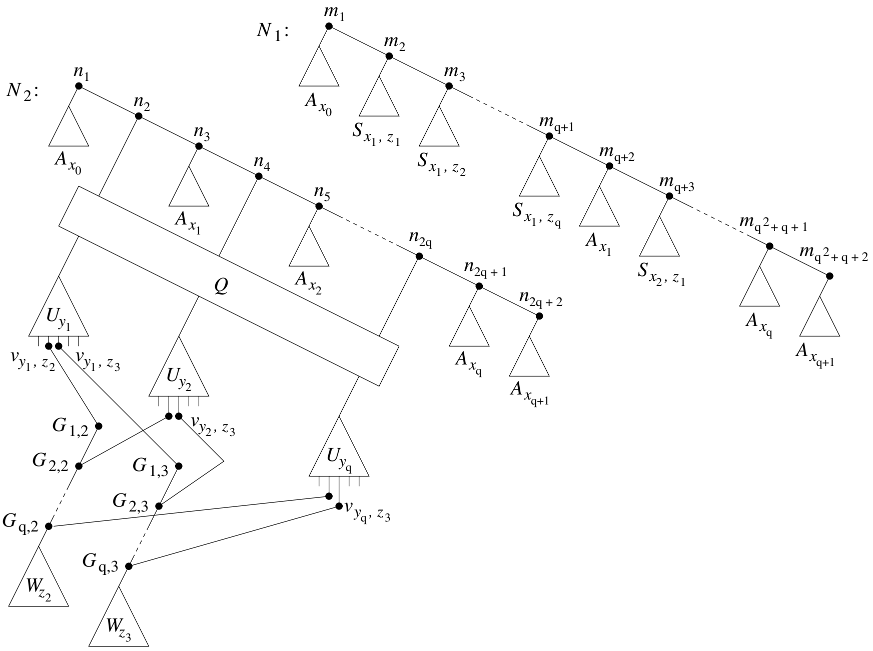

Fig. 5. The phylogenetic networks N_1 and N_2

5 Open Problems

Does MASN for other types of structurally restricted phylogenetic networks admit efficient algorithms? In particular, is it possible to extend our method in

Section 3 to two networks in which every hybrid node has exactly one split node? An example of such a network is shown in Fig. 6. It would also be interesting to investigate if any other problems which are hard to solve for unrestricted phylogenetic networks but solvable in polynomial time for galled-trees (i.e., networks with nesting depth 1)[4] can be solved efficiently for nested phylogenetic networks.

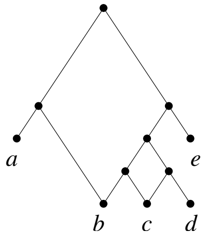

Fig. 6. This network is not nested, yet every hybrid node has exactly one split node and every split node has exactly one hybrid node, i.e., the converse of Lemma 1 is not true

We believe MASN for more than two nested phylogenetic networks can be solved in polynomial time when $k = O(1)$. On the other hand, if the outdegree 2 constraint in the definition of phylogenetic networks is removed, MASN seems to be NP-hard already for two networks with nesting depth 1. The final open question is: can the running time of our algorithm for two nested phylogenetic networks be improved, e.g., by applying sparsification techniques?

References

1. A. Amir and D. Keselman. Maximum agreement subtree in a set of evolutionary trees: Metrics and efficient algorithms. *SIAM J. on Computing*, 26:1656–1669, 1997.
2. D. Bryant. *Building trees, hunting for trees, and comparing trees: theory and methods in phylogenetic analysis.* PhD thesis, Univ. of Canterbury, New Zealand, 1997.
3. D. Bryant and V. Moulton. NeighborNet: an agglomerative method for the construction of planar phylogenetic networks. In *Proc. of the 2^{nd} Workshop on Algorithms in Bioinformatics* (WABI 2002), volume 2452 of *LNCS*, pages 375–391. Springer, 2002.
4. C. Choy, J. Jansson, K. Sadakane, and W.-K. Sung. Computing the maximum agreement of phylogenetic networks. In *Proc. of Computing: the 10^{th} Australasian Theory Symposium* (CATS 2004), volume 91 of *ENTCS*, pages 134–147. Elsevier, 2004.
5. R. Cole, M. Farach-Colton, R. Hariharan, T. Przytycka, and M. Thorup. An $O(n \log n)$ algorithm for the maximum agreement subtree problem for binary trees. *SIAM J. on Computing*, 30(5):1385–1404, 2000.
6. T. Cormen, C. Leiserson, and R. Rivest. *Introduction to algorithms.* MIT Press, 1990.
7. M. Farach, T. Przytycka, and M. Thorup. On the agreement of many trees. *Information Processing Letters*, 55:297–301, 1995.

[4] For example, the perfect phylogenetic network with recombination problem is NP-hard for unrestricted networks [21], but polynomial-time solvable for galled-trees [10].

8. M. Garey and D. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, New York, 1979.
9. L. Gąsieniec, J. Jansson, A. Lingas, and A. Östlin. On the complexity of constructing evolutionary trees. *Journal of Combinatorial Optimization*, 3:183–197, 1999.
10. D. Gusfield, S. Eddhu, and C. Langley. Efficient reconstruction of phylogenetic networks with constrained recombination. In *Proc. of the Computational Systems Bioinformatics Conference* (CSB2003), pages 363–374, 2003.
11. J. Hein. Reconstructing evolution of sequences subject to recombination using parsimony. *Mathematical Biosciences*, 98(2):185–200, 1990.
12. D. H. Huson, T. Dezulian, T. Klöpper, and M. Steel. Phylogenetic super-networks from partial trees. In *Proc. of the 4th Workshop on Algorithms in Bioinformatics* (WABI 2004), to appear.
13. J. Jansson and W.-K. Sung. Inferring a level-1 phylogenetic network from a dense set of rooted triplets. In *Proc. of the 10th International Computing and Combinatorics Conference* (COCOON 2004), to appear.
14. M.-Y. Kao, T.-W. Lam, W.-K. Sung, and H.-F. Ting. An even faster and more unifying algorithm for comparing trees via unbalanced bipartite matchings. *Journal of Algorithms*, 40(2):212–233, 2001.
15. W.-H. Li. *Molecular Evolution.* Sinauer Associates, Inc., Sunderland, 1997.
16. L. Nakhleh, J. Sun, T. Warnow, C. R. Linder, B. M. E. Moret, and A. Tholse. Towards the development of computational tools for evaluating phylogenetic reconstruction methods. In *Proc. of the 8th Pacific Symposium on Biocomputing* (PSB 2003), pages 315–326, 2003.
17. L. Nakhleh, T. Warnow, and C. R. Linder. Reconstructing reticulate evolution in species – theory and practice. In *Proc. of the 8th Annual International Conf. on Research in Computational Molecular Biology* (RECOMB 2004), pages 337–346, 2004.
18. D. Posada and K. A. Crandall. Intraspecific gene genealogies: trees grafting into networks. *TRENDS in Ecology & Evolution*, 16(1):37–45, 2001.
19. J. Setubal and J. Meidanis. *Introduction to Comp. Molecular Biology.* PWS, 1997.
20. M. Steel and T. Warnow. Kaikoura tree theorems: Computing the maximum agreement subtree. *Information Processing Letters*, 48:77–82, 1993.
21. L. Wang, K. Zhang, and L. Zhang. Perfect phylogenetic networks with recombination. *Journal of Computational Biology*, 8(1):69–78, 2001.