Ranking top-k trees in tree-based phylogenetic networks

Momoko Hayamizu (✉ hayamizu@ism.ac.jp)

The Institute of Statistical Mathematics <https://orcid.org/0000-0001-8825-6331>

Kazuhisa Makino

Research Institute for Mathematical Sciences, Kyoto University

Research article

Keywords: phylogenetic tree, tree-based phylogenetic network, support tree, top-k ranking problem, maximum likelihood estimation; enumeration, algorithm

Posted Date: November 8th, 2019

DOI: <https://doi.org/10.21203/rs.2.15349/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

RESEARCH

Ranking top- k trees in tree-based phylogenetic networks

Momoko Hayamizu^{1,2*} and Kazuhisa Makino³

*Correspondence:

hayamizu@ism.ac.jp

¹The Institute of Statistical Mathematics, 190-8562 Tokyo, Japan

Full list of author information is available at the end of the article

Abstract

‘Tree-based’ phylogenetic networks provide a mathematically-tractable model for representing reticulate evolution in biology. Such networks consist of an underlying ‘support tree’ together with arcs between the edges of this tree. However, a tree-based network can have several such support trees, and this leads to a variety of algorithmic problems that are relevant to the analysis of biological data. Recently, Hayamizu (arXiv:1811.05849 [math.CO]) proved a structure theorem for tree-based phylogenetic networks and obtained linear-time and linear-delay algorithms for many basic problems on support trees, such as counting, optimisation, and enumeration. In the present paper, we consider the following fundamental problem in statistical data analysis: given a tree-based phylogenetic network N whose arcs are associated with probability, create the top- k support tree ranking for N by their likelihood values. We provide a linear-delay (and hence optimal) algorithm for the problem and thus reveal the interesting property of tree-based phylogenetic networks that ranking top- k support trees is as computationally easy as picking k arbitrary support trees.

Keywords: phylogenetic tree; tree-based phylogenetic network; support tree; top-k ranking problem; maximum likelihood estimation; enumeration; algorithm

AMS Subject Classification: Primary 05C85; secondary 62F07, 68W40, 05C05, 05C20, 05C30, 92D15

1 Introduction

Although phylogenetic trees have been used as the standard model of evolution, phylogenetic networks have become popular amongst biologists as a tool to describe conflicting signals in data or uncertainty in evolutionary histories [1, 2, 3]. In particular, ‘tree-based’ phylogenetic networks, which Francis and Steel proposed in [2] based on the observation by van Iersel in [4], are a natural generalisation of phylogenetic trees and have attracted much attention in the last few years (*e.g.*, [5, 6, 1, 7, 8, 9, 10, 11]).

Tree-based networks can be intuitively interpreted as a phylogenetic tree with independent noise; to be more precise, they consist of an underlying ‘support tree’ together with extra arcs that do not touch each other. Importantly, a tree-based network can have several support trees, and this leads to various interesting discussion relevant to the analysis of biological data. For instance, Francis and Steel [2] described the problems of counting and listing all support trees of tree-based networks. In addition, Hayamizu [9] also formulated the problem of finding an optimal support tree of ‘weighted’ tree-based networks as illustrated in Figure 1, which has direct implications for estimating the most probable tree from non-treelike data.

Figure 1 An example of a tree-based phylogenetic network where each arc is weighted by a probability. If we regard the dotted arc as noise, then we can find a support tree shown in solid lines, which is the most probable one in this case. Note that we only show the weights of uncertain arcs for visual clarity.

In [9], Hayamizu provided simple and fast algorithms for a series of problems including the above three by proving a powerful theorem that characterises the mathematical structure of support trees, and this opened up vast opportunities for the use of tree-based phylogenetic networks in data analysis. From a biological standpoint, however, there are still many challenges, such as the question of how to quickly list and prioritise highly probable support trees. This problem is indeed an important generalisation of the above optimisation problem, which was proven to be solvable in linear time [9], because retrieving suboptimal support trees besides the statistically best one can help gain more biological insights.

In the present paper, with the aim of facilitating the practical application of tree-based networks, we therefore formulate the question mentioned above into the following ‘top- k support tree ranking problem’: given a tree-based phylogenetic network N where each arc a exists in the true evolutionary lineage with probability $w(a) > 0$, list top- k support trees of N in non-increasing order by their likelihood values. At first glance, ranking top- k support trees may seem more difficult than picking k arbitrary ones, the latter of which was treated in [9]. However, as a main result of this paper, we provide a linear-delay (*i.e.*, optimal) algorithm for the top- k support tree ranking problem. This implies that the above two problems have the same time complexity, which is an interesting and useful property of tree-based phylogenetic networks.

2 Preliminaries

Throughout this paper, X represents a non-empty finite set of present-day species. All graphs considered here are finite, simple, directed acyclic graphs. For a graph G , $V(G)$ and $A(G)$ denote the sets of vertices and arcs of G , respectively. A graph G is called a *subgraph* of a graph H if both $V(G) \subseteq V(H)$ and $A(G) \subseteq A(H)$ hold, in which case we write $G \subseteq H$. When $G \subseteq H$ but $G \neq H$, then G is called a *proper* subgraph of H . When $G \subseteq H$ and $V(G) = V(H)$, G is a *spanning* subgraph of H . Given a graph G and a non-empty subset A' of $A(G)$, A' is said to *induce the subgraph* $G[A']$ of G , that is, the one whose arc-set is A' and whose vertex-set consists of all ends of arcs in A' . For a graph G with $|A(G)| \geq 1$ and a partition $\{A_1, \dots, A_d\}$ of $A(G)$, the collection $\{G[A_1], \dots, G[A_d]\}$ of arc-induced subgraphs of G is called a *decomposition* of G . For an arc $a = (u, v) \in A(G)$, u and v are called the tail and head of a and are denoted by $\text{tail}(a)$ and $\text{head}(a)$, respectively. For a vertex v of a graph G , the *in-degree* of v in G , denoted by $\deg_G^-(v)$, is defined to be the cardinality of the set $\{a \in A(G) \mid \text{head}(a) = v\}$. The *out-degree* of v in G , denoted by $\deg_G^+(v)$, is defined in a similar manner. For any graph G , a vertex $v \in V(G)$ with $(\deg_G^-(v), \deg_G^+(v)) = (1, 0)$ is called a *leaf* of G .

Definition 1 A *rooted binary phylogenetic X -network* is defined to be a finite simple directed acyclic graph N with the following properties:

- 1 N has a unique vertex ρ with $\deg_N^-(\rho) = 0$ and $\deg_N^+(\rho) \in \{1, 2\}$;
- 2 X is the set of leaves of N ;
- 3 for any $v \in V(N) \setminus (X \cup \{\rho\})$, $\{\deg_N^-(v), \deg_N^+(v)\} = \{1, 2\}$ holds.

In Definition 1, the vertex ρ is called *the root* of N , and a vertex $v \in V(N)$ with $(\deg_N^-(v), \deg_N^+(v)) = (2, 1)$ is called a *reticulation vertex* of N . When N has no reticulation vertex, N is called a *rooted binary phylogenetic X -tree*.

Definition 2 ([2]) If a rooted binary phylogenetic X -network N that has a spanning tree τ that can be obtained by inserting zero or more vertices into each arc of a rooted binary phylogenetic X -tree T , then N is said to be *tree-based* and τ is called a *support tree* of N .

Theorem 3 ([2]) Let N be a rooted binary phylogenetic X -network and let S be a subset of $A(N)$. Then, the subgraph $N[S]$ of N is a support tree of N if and only if S satisfies the following three conditions, in which case S is called an ‘admissible’ arc-set of N . Moreover, there exists a one-to-one correspondence between support trees of N and admissible arc-sets of N .

- 1 S contains all $(u, v) \in A(N)$ with $\deg_N^-(v) = 1$ or $\deg_N^+(u) = 1$.
- 2 for any $a_1, a_2 \in A(N)$ with $\text{head}(a_1) = \text{head}(a_2)$, exactly one of $\{a_1, a_2\}$ is in S .
- 3 for any $a_1, a_2 \in A(N)$ with $\text{tail}(a_1) = \text{tail}(a_2)$, at least one of $\{a_1, a_2\}$ is in S .

In this paper, as the conditions in Theorem 3 still make sense for any subgraph of N , we consider admissible arc-sets of subgraphs of N .

3 Known results: the structure of support trees

Here, we summarise without proofs the relevant material in [9]. A connected subgraph Z of a tree-based phylogenetic X -network N with $|A(Z)| \geq 1$ is called a *zig-zag trail* (in N) if there exists a permutation $(a_1, \dots, a_m)$ of $A(Z)$ such that for each $i \in [1, m-1]$, either $\text{head}(a_i) = \text{head}(a_{i+1})$ or $\text{tail}(a_i) = \text{tail}(a_{i+1})$ holds. Then, any zig-zag trail Z in N is specified by an alternating sequence of (not necessarily distinct) vertices and distinct arcs of N , such as $(v_0, (v_0, v_1), v_1, (v_2, v_1), v_2, (v_2, v_3), \dots, (v_m, v_{m-1}), v_m)$, which can be more concisely expressed as $v_0 > v_1 < v_2 > v_3 < \dots > v_{m-1} < v_m$ or in reverse order. A zig-zag trail Z in N is said to be *maximal* if N contains no zig-zag trail Z' such that Z is a proper subgraph of Z' . A maximal zig-zag trail Z with even $m := |A(Z)| \geq 4$ is called a *crown* if Z can be written in the cyclic form $v_0 < v_1 > v_2 < v_3 > \dots > v_{m-2} < v_{m-1} > v_m = v_0$ and is called a *fence* otherwise. Furthermore, a fence Z with odd $|A(Z)|$ is called an *N -fence*, in which case Z can be expressed as $v_0 > v_1 < v_2 > v_3 < \dots > v_{m-2} < v_{m-1} > v_m$. A fence Z with even $|A(Z)|$ is called an *M -fence* if it can be written in the form $v_0 < v_1 > v_2 < v_3 > \dots > v_{m-2} < v_{m-1} > v_m$, rather than $v_0 > v_1 < v_2 > v_3 < \dots < v_{m-2} > v_{m-1} < v_m$.

From now on, we represent a maximal zig-zag trail Z by a sequence $\langle a_1, \dots, a_{|A(Z)|} \rangle$ of the elements of $A(Z)$ that form the zig-zag trail in this order, assuming that no

confusion arises. Then, we can encode an arbitrary arc-induced subgraph of Z by an $|A(Z)|$ -dimensional vector. For example, for an N-fence $Z = \langle a_1, a_2, a_3, a_4, a_5 \rangle$, the subgraph of Z induced by the subset $\{a_1, a_3, a_5\} \subseteq A(Z)$ is specified by the vector $(1\ 0\ 1\ 0\ 1) = (1(01)^2)$. With this notation, we can state Hayamizu's structure theorem for tree-based phylogenetic networks, which gives an explicit characterisation of the family Ω of all admissible arc-sets of N as follows.

Theorem 4 ([9]) *Any tree-based phylogenetic X -network N is uniquely decomposed into maximal zig-zag trails $Z_1, \dots, Z_d$, each of which is a crown, M -fence or N -fence. Moreover, a subgraph G of N is a support tree of N if and only if $A(G) \cap A(Z_i)$ is an admissible arc-set of Z_i for any $i \in [1, d]$. Furthermore, the collection Ω of support trees of N is characterised by a direct product of families $\Omega_1, \dots, \Omega_d$ of the admissible arc-sets of $Z_1, \dots, Z_d$, namely, we have $\Omega = \prod_{i=1}^d \Omega_i$ with*

$$\Omega_i := \begin{cases} \{((01)^{|A(Z_i)|/2}), ((10)^{|A(Z_i)|/2})\} & \text{if } Z_i \text{ is a crown;} \\ \{(1(01)^{(|A(Z_i)|-1)/2})\} & \text{if } Z_i \text{ is an } N\text{-fence;} \\ \{(1(01)^p(10)^q1) \mid p, q \in \mathbb{Z}_{\geq 0}, p+q = (|A(Z_i)|-2)/2\} & \text{if } Z_i \text{ is an } M\text{-fence.} \end{cases}$$

4 Top- k support tree ranking problem

Given a tree-based phylogenetic X -network N where each arc a is chosen with probability $w(a) \in (0, 1]$, we can assign a ranking number to each support tree $\tau \in \Omega$ of N by the likelihood value $f(\tau) := \prod_{a \in A(\tau)} w(a)$. In principle, the top- k support tree ranking problem for N asks for an ordered set $\langle \tau^{(1)}, \dots, \tau^{(k)} \rangle$ of k support trees of N such that $f(\tau^{(1)}) \geq \dots \geq f(\tau^{(k)}) \geq f(\tau)$ holds for any support tree τ of N other than $\tau^{(i)}$ ($i = 1, \dots, k$). However, such a ranking is not unique in general, since there can be 'ties' in the collection Ω of support trees of N as well as in the family Ω_i of admissible arc-sets of each maximal zig-zag trail Z_i in N . For convenience, we ensure the uniqueness of the ranking by using the lexicographical order $\leq_{\text{lex}}$ on vectors as follows.

Assume that N is a tree-based phylogenetic X -network with $\Omega = \prod_{i=1}^d \Omega_i$ as in Theorem 4 and that Z_i is any maximal zig-zag trail in N . We define the *local ranking for Z_i* to be a totally ordered set $(\Omega_i, \leq^*)$ such that for any $x, y \in \Omega_i$, $x \leq^* y$ holds if either $f(x) > f(y)$ or $(f(x) = f(y) \wedge x \leq_{\text{lex}} y)$ holds. Note that the elements of Ω_i are $|A(Z_i)|$ -dimensional vectors and any two of them are comparable lexicographically. From now, we identify the j -th element of $(\Omega_i, \leq^*)$ with its local ranking number $j \in \{1, \dots, |\Omega_i|\}$ in order to write $\Omega = \prod_{i=1}^d \{1, \dots, |\Omega_i|\}$. Then, the elements of Ω are vectors having the same dimension again and so we can break ties by using $\leq_{\text{lex}}$ as before. Abusing the notation $\leq^*$ slightly, we call the totally ordered set $(\Omega, \leq^*)$ the *support tree ranking (for N)*. For any $k \in \mathbb{N}$ with $k \leq |\Omega|$, the *top- k support tree ranking (for N)* is defined to be a unique subsequence of the first k elements of $(\Omega, \leq^*)$. Note that for any $k \in \mathbb{N}$, one can determine in $O(|A(N)|)$ time whether or not $k \leq |\Omega|$ holds [9].

Top- k support tree ranking problem

Input: A tree-based phylogenetic X -network N with associated probability $w : A(N) \rightarrow (0, 1]$ and $k \in \mathbb{N}$ not exceeding the number $|\Omega|$ of support trees of N .

Output: The top- k support tree ranking $\langle \tau^{(1)}, \dots, \tau^{(k)} \rangle$ for N .

5 Results

As a preliminary step, we prove the following proposition about the local ranking.

Proposition 5 *For any maximal zig-zag trail Z_i in a tree-based phylogenetic X -network N with associated probability $w : A(N) \rightarrow (0, 1]$, the first element in the local ranking $(\Omega_i, \leq^*)$ can be found in $O(|A(Z_i)|)$ time. Moreover, given the j -th element in $(\Omega_i, \leq^*)$, one can find the $(j + 1)$ -th element in $O(|A(Z_i)|)$ time.*

Proof One can check in $O(|A(Z_i)|)$ time whether Z_i is a crown, N-fence or M-fence. In the case when Z_i is a crown or N-fence, the local ranking for Z_i is trivial to compute as $|\Omega_i| \leq 2$ holds by Theorem 4. Assume that Z_i is an M-fence $\langle a_1, \dots, a_{2m} \rangle$ with $|A(Z_i)| = 2m$. Also, let $x_{p+1} := (101)^p(10)^q1$ with $p + q = m - 1$ for each $p \in [0, m - 1]$ and let $\Delta_p := w(a_{2p+1}) - w(a_{2p})$ for each $p \in [1, m - 1]$. Then, $f(x_{p+1}) = f(x_p) + \Delta_p$ holds for each $p \in [1, m - 1]$. As one can obtain both $f(x_1)$ and $\langle \Delta_1, \dots, \Delta_{m-1} \rangle$ in $O(|A(Z_i)|)$ time, computing the likelihood values $f(x)$ for all $x \in \Omega_i$ requires $O(|A(Z_i)|)$ time. This completes the proof. $\square$

We define $I_0 := \emptyset$ and $I_j := \{\tau^{(0)}, \dots, \tau^{(j)}\}$ for each $j \in [1, k]$. Recalling $\Omega = \prod_{i=1}^d \{1, \dots, |\Omega_i|\}$, we see that $(\Omega, \leq^*)$ is a linear extension of the partially ordered set $(\Omega, \leq)$ (i.e., $x \leq y$ implies $x \leq^* y$), where $\leq$ is the usual component-wise order on vectors (e.g., $(x_1 \ x_2) \leq (y_1 \ y_2)$ if and only if $x_1 \leq y_1$ and $x_2 \leq y_2$). We also note that this requires each I_j to be an order ideal of $(\Omega, \leq)$ (i.e., for any $x \leq y$, $y \in I_j$ implies $x \in I_j$). These arguments lead to the following proposition.

Proposition 6 *Let $\langle \tau^{(1)}, \dots, \tau^{(k)} \rangle$ be the top- k support tree ranking for a tree-based phylogenetic X -network N with associated probability $w : A(N) \rightarrow (0, 1]$ and let I_j be as defined above. Then, $I_1 = \{(1 \ \dots \ 1)\}$ holds, and for each $j \in [1, k - 1]$, there exists $\tau \in I_j$ with $\|\tau^{(j+1)} - \tau\|_1 = 1$.*

Let e_i be the unit vector such that i -th component is one and the others are all zeros. Also, for each $\tau \in \Omega \setminus \{\tau^{(1)}\}$, let $\text{id}(\tau)$ be the first index such that the i -th component of τ is strictly greater than one and let $e(\tau) := e_{\text{id}(\tau)}$. For example, $\tau = (1 \ 1 \ 1 \ 5 \ 8)$ gives $e(\tau) = (0 \ 0 \ 0 \ 1 \ 0)$. Then, we have the next lemma, which is illustrated in Figure 2.

Lemma 7 *Let $(\Omega, \leq^*)$ be the support tree ranking for a tree-based phylogenetic X -network N with associated probability $w : A(N) \rightarrow (0, 1]$ and let Γ be a graph with $V(\Gamma) = \Omega$ and $A(\Gamma) = \{(\tau, \tau') \in \Omega \times (\Omega \setminus \{\tau^{(1)}\}) \mid \tau = \tau' - e(\tau')\}$. Then, Γ is a spanning tree of the Hasse diagram of $(\Omega, \leq)$ such that $\tau^{(1)}$ is the root of Γ and $(\tau, \tau') \in A(\Gamma)$ implies $\tau \leq^* \tau'$.*

Proof It is clear that $(\tau, \tau') \in A(\Gamma)$ implies $\tau \leq \tau'$ (and hence $\tau \leq^* \tau'$). By construction, Γ is a tree rooted at $\tau^{(1)}$ because $\deg_{\Gamma}^{-}(\tau^{(1)}) = 0$ holds and for each $\tau' \in V(\Gamma) \setminus \{\tau^{(1)}\}$, there exists a unique element $\tau \in V(\Gamma)$ with $(\tau, \tau') \in A(\Gamma)$. This completes the proof. $\square$

In what follows, for any $\Omega' \subseteq \Omega$, we write $\text{least}(\Omega')$ to mean the least element of $(\Omega', \leq^*)$. For any $\tau \in \Omega$, let $\text{Child}(\tau) := \{\tau' \in V(\Gamma) \mid (\tau, \tau') \in A(\Gamma)\}$ and $\text{child}^*(\tau) := \text{least}(\text{Child}(\tau))$. Also, for any $\tau' \in \Omega \setminus \{\tau^{(1)}\}$, let $\text{sibling}^*(\tau') := \text{least}(\{\tau \in \text{Child}(\text{parent}(\tau')) \mid \tau' \leq^* \tau \wedge \tau \neq \tau'\})$, where $\text{parent}(\tau')$ represents a unique element $\tau \in V(\Gamma)$ with $(\tau, \tau') \in A(\Gamma)$. We note that both $\text{child}^*(\tau') = \emptyset$ and $\text{sibling}^*(\tau') = \emptyset$ are possible to occur.

Figure 2 An illustration of Lemma 7. The top left is a tree-based phylogenetic X -network N whose arcs are associated with probability. The top right shows the maximal zig-zag trails Z_i in N with $|\Omega_i| \geq 2$ and the likelihood of each element of $(\Omega_i, \leq^*)$ ($i = 1, 2, 3$). On the bottom is the spanning tree Γ (shown in bold) of the Hasse diagram of $(\Omega, \leq)$.

Lemma 8 *Let Γ be the graph as in Lemma 7 and let Q_j be a subset of $V(\Gamma)$ that is recursively defined by*

$$Q_j := \begin{cases} (Q_{j-1} \setminus \{\tau^{(j-1)}\}) \cup \{\text{child}^*(\tau^{(j-1)}), \text{sibling}^*(\tau^{(j-1)})\} & j \in [2, k] \\ \{\tau^{(1)}\} & j = 1. \end{cases} \quad (1)$$

Then, for each $j \in [1, k]$, we have $\tau^{(j)} \in Q_j$ and $\tau^{(\ell)} \notin Q_j$ for all $\ell < j$.

Proof Let $P_1 = Q_1$ and $P_j = \{\text{least}(\text{Child}(\tau) \setminus I_{j-1}) \mid \tau \in I_{j-1}\}$ for $j \in [2, k]$. We will show that $P_j = Q_j$ holds for any $j \in [1, k]$, which completes the proof, since $\tau^{(j)} \in P_j$ and $\tau^{(\ell)} \notin P_j$ for all $\ell < j$.

For $j \in [2, k]$, we have

$$\begin{aligned} P_j &= \{\text{least}(\text{Child}(\tau) \setminus I_{j-1}) \mid \tau \in I_{j-2}\} \cup \{\text{least}(\text{Child}(\tau^{(j-1)}) \setminus I_{j-1})\} \\ &= \{\text{least}(\text{Child}(\tau) \setminus I_{j-1}) \mid \tau \in I_{j-2}\} \cup \{\text{child}^*(\tau^{(j-1)})\}, \end{aligned}$$

where we assume that $I_0 = \emptyset$. Note that I_{j-2} contains $p := \text{parent}(\tau^{(j-1)})$. This implies

$$\begin{aligned} P_j &= \{\text{least}(\text{Child}(\tau) \setminus I_{j-1}) \mid \tau \in I_{j-2} \setminus \{p\}\} \cup \{\text{child}^*(\tau^{(j-1)}), \text{least}(\text{Child}(p) \setminus I_{j-1})\} \\ &= \{\text{least}(\text{Child}(\tau) \setminus I_{j-1}) \mid \tau \in I_{j-2} \setminus \{p\}\} \cup \{\text{child}^*(\tau^{(j-1)}), \text{sibling}^*(\tau^{(j-1)})\}. \end{aligned}$$

For any $\tau \in I_{j-2} \setminus \{p\}$, we have $\text{least}(\text{Child}(\tau) \setminus I_{j-1}) = \text{least}(\text{Child}(\tau) \setminus I_{j-2})$ because $\tau^{(j-1)} \notin \text{Child}(\tau)$ holds. We thus obtain

$$\begin{aligned} P_j &= (\{\text{least}(\text{Child}(\tau) \setminus I_{j-2}) \mid \tau \in I_{j-2}\} \setminus \{\text{least}(\text{Child}(p) \setminus I_{j-2})\}) \cup \{\text{child}^*(\tau^{(j-1)}), \text{sibling}^*(\tau^{(j-1)})\} \\ &= (P_{j-1} \setminus \{\tau^{(j-1)}\}) \cup \{\text{child}^*(\tau^{(j-1)}), \text{sibling}^*(\tau^{(j-1)})\}. \end{aligned}$$

From Equation (1) and $P_1 = Q_1$, the desired conclusion follows. $\square$

We are in a position to give an algorithm for Problem 4. As illustrated in Table 1, the algorithm starts by setting $j := 1$ and $Q_1 := \{\tau^{(1)}\}$ and then returns $\tau^{(j)} = \text{least}(Q_j)$ for each $j \in [1, k]$, where Q_j is iteratively updated using Equation (1).

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$
Q_j	$\{(1\ 1\ 1)\}$	$\{(1\ 1\ 2)\}$	$\{(2\ 1\ 1), (2\ 1\ 2)\}$	$\{(1\ 2\ 1), (2\ 1\ 2)\}$	$\{(1\ 2\ 1), (1\ 2\ 2)\}$	$\{(1\ 2\ 2), (2\ 2\ 1)\}$	$\{(2\ 2\ 1), (2\ 2\ 2)\}$	$\{(2\ 2\ 2)\}$
$\tau^{(j)}$	$(1\ 1\ 1)$	$(1\ 1\ 2)$	$(2\ 1\ 1)$	$(2\ 1\ 2)$	$(1\ 2\ 1)$	$(1\ 2\ 2)$	$(2\ 2\ 1)$	$(2\ 2\ 2)$
$child^*(\tau^{(j)})$	$(1\ 1\ 2)$	$(2\ 1\ 2)$	$\emptyset$	$\emptyset$	$(2\ 2\ 1)$	$(2\ 2\ 2)$	$\emptyset$	$\emptyset$
$sibling^*(\tau^{(j)})$	$\emptyset$	$(2\ 1\ 1)$	$(1\ 2\ 1)$	$(1\ 2\ 2)$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 1: Application of the proposed algorithm to the input N in Figure 2 ($k = 8$).

In order to analyse the running time of the above algorithm, let us review some basics of a *priority queue*, which is a data structure for maintaining objects that are prioritised by their associated values. In its most basic form, a priority queue supports the operations called INSERT and DELETE-MIN, where the former refers to adding a new object, and the latter to detecting and deleting the one with the highest-priority [12]. Implemented with a binary heap, each of these operations can be performed in $O(\log n)$ time, where n denotes the number of the elements in the priority queue [12].

Theorem 9 *The top- k support tree ranking problem (Problem 4) can be solved with linear delay, and hence in $O(k|A(N)|)$ time.*

Proof As Equation (1) implies that $|Q_{j+1} - Q_j| \leq 1$ holds for any $j \in [1, k-1]$, $|Q_j| \leq k$ holds for any $j \in [1, k]$. Then, if we keep the elements of each Q_j in a priority queue, $O(\log k)$ time suffices to return $\tau^{(j)}$ and to delete $\tau^{(j)}$ from Q_j . Also, once $child^*(\tau^{(j)})$ and $sibling^*(\tau^{(j)})$ have been obtained, inserting the two elements requires $O(\log k)$ time. We note that $O(\log k) \leq O(|A(N)|)$ follows from $k \leq 2^{|A(N)|}$. By Proposition 5, for each $j \in [1, k-1]$, one can compute $\{child^*(\tau^{(j)}), sibling^*(\tau^{(j)})\}$ in $\sum_{i=1}^d O(|A(Z_i)|)$ time, which equals $O(|A(N)|)$ time as $\{Z_1, \dots, Z_d\}$ is a decomposition of N . Hence, our algorithm can return $\tau^{(1)}, \dots, \tau^{(k)}$ one after the other in such a way that the delay between two consecutive outputs is $O(|A(N)|)$ time. This completes the proof. $\square$

Finally, we make two remarks. First, $\Omega(k|A(N)|)$ time is required to output k distinct support trees of N as each support tree has size $\Omega(|A(N)|)$. Therefore, the running time of our algorithm (as well as that of the enumeration algorithm in [9]) is $\Theta(k|A(N)|)$, which guarantees the optimality of those algorithms. Second, as commonly in the literature (*e.g.*, [13, 14]), it would be natural to wonder about the time complexity of an analogue of Problem 4 that only asks for outputting a sequence of the differences between $\tau^{(j-1)}$ and $\tau^{(j)}$; however, we note that this problem still requires $\Omega(k|A(N)|)$ time because the size of each difference is $\Omega(|A(N)|)$. To illustrate this, consider a tree-based phylogenetic X -network N that is decomposed into maximal fences, each of which has only one admissible arc-set, and c crowns, each of which has size $\Omega(|A(N)|/c)$. The difference between any two support trees has size $\Omega(|A(N)|/c)$, which equals $\Omega(|A(N)|)$ if c is a constant.

6 Conclusion and discussion

We have considered the top- k support tree ranking problem (Problem 4) and provided a linear-delay (and hence optimal) algorithm for solving it. While the results in this paper hold for any tree-based networks whose arcs are weighted by non-negative real numbers, we expect that our algorithm is particularly useful to discover and

prioritise highly probable trees from non-tree-like data as described in Figure 1. Of course, the input network can be far larger and can also have a greater number of support trees in practical situations. However, as we have seen, when we are interested only in a small fraction of the candidates such as top 10, the proposed method can quickly find them and create their ranking.

Because biological data are almost always not exactly tree-like and a statistically optimal solution is not necessarily most meaningful, our fast algorithm for the top- k support tree ranking problem can become a powerful tool for analysing phylogenetic data. Potential applications include the removal of noise from data and the detection of reticulation events such as horizontal gene transfer amongst bacterias.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Availability of data and materials
This article is present on a university repository website and can be accessed on <http://export.arxiv.org/pdf/1904.12432>. This article is not published nor is under publication elsewhere.

Competing interests
The authors declare that they have no competing interests.

Funding
MH acknowledges support from JST PRESTO Grant Number JPMJPR16EB. The funding bodies did not play any role in the design of the study, data collection and analysis, or preparation of the manuscript.

Authors' contributions
MH and KM designed the research and conceptualized the algorithm together. MH wrote the article by adding some numerical examples and describing biological implications. Both authors have read and approved the final manuscript.

Acknowledgements
The authors thank Mike Steel and Mats Gyllenberg for helpful suggestions on an earlier version of this paper, Takeaki Uno and Kunihiro Wasa for useful discussion on the literature of enumeration algorithms, and Miho Suzuki for her help in the production of Figure 1.

Author details

¹The Institute of Statistical Mathematics, 190-8562 Tokyo, Japan. ²JST PRESTO, 190-8562 Tokyo, Japan.

³Research Institute for Mathematical Sciences, Kyoto University, 606-8502 Kyoto, Japan.

References

- Francis, A., Huber, K.T., Moulton, V.: Tree-based unrooted phylogenetic networks. *Bulletin of mathematical biology* **80**(2), 404–416 (2018)
- Francis, A.R., Steel, M.: Which phylogenetic networks are merely trees with additional arcs? *Systematic Biology* **64**(5), 768–777 (2015)
- Huson, D.H., Rupp, R., Scornavacca, C.: *Phylogenetic Networks: Concepts, Algorithms and Applications*. Cambridge University Press, ??? (2010)
- van Iersel, L.: Different topological restrictions of rooted phylogenetic networks. Which make biological sense? <http://phylonetworks.blogspot.nl/2013/03/different-topological-restrictions-of.html>. Accessed: 2019-03-16 (2013)
- Anaya, M., Anipchenko-Ulaj, O., Ashfaq, A., Chiu, J., Kaiser, M., Ohsawa, M.S., Owen, M., Pavlechko, E., St. John, K., Suleria, S., Thompson, K., Yap, C.: On determining if tree-based networks contain fixed trees. *Bulletin of Mathematical Biology* **78**(5), 961–969 (2016)
- Fischer, M., Galla, M., Herbst, L., Long, Y., Wicke, K.: Non-binary treebased unrooted phylogenetic networks and their relations to binary and rooted ones. *arXiv:1810.06853 [q-bio.PE]* (2018)
- Francis, A., Semple, C., Steel, M.: New characterisations of tree-based networks and proximity measures. *Advances in Applied Mathematics* **93**, 93–107 (2018)
- Hayamizu, M.: On the existence of infinitely many universal tree-based networks. *Journal of Theoretical Biology* **396**, 204–206 (2016)
- Hayamizu, M.: A structure theorem for tree-based phylogenetic networks. *arXiv:1811.05849 [math.CO]* (2018)

10. Pons, J.C., Semple, C., Steel, M.: Tree-based networks: characterisations, metrics, and support trees. *Journal of Mathematical Biology* **78**(4), 899–918 (2019)
11. Zhang, L.: On tree-based phylogenetic networks. *Journal of Computational Biology* **23**(7), 553–565 (2016)
12. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*. MIT press, ??? (2009)
13. Kapoor, S., Ramesh, H.: Algorithms for enumerating all spanning trees of undirected and weighted graphs. *SIAM Journal on Computing* **24**(2), 247–265 (1995)
14. Savage, C.: A survey of combinatorial gray codes. *SIAM review* **39**(4), 605–629 (1997)

Figures

Figure 1

An example of a tree-based phylogenetic network where each arc is weighted by a probability. If we regard the dotted arc as noise, then we can find a support tree shown in solid lines, which is the most probable one in this case. Note that we only show the weights of uncertain arcs for visual clarity.

Figure 2

See supplementary files for legend

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Figure2legend.jpg](#)