Inf. Process. Lett. 195 (2027) 106664

Contents lists available at ScienceDirect

Information Processing Letters

journal homepage: www.elsevier.com/locate/ipl

Efficient reconstruction of arboreal networks

Katharina T. Huber [a,*], Katherine St. John [b,c]

[a] School of Computing Sciences, University of East Anglia, Norwich, UK
[b] Department of Computer Science, Hunter College and Graduate Center, City University of New York, New York, NY, USA
[c] Division of Invertebrate Zoology, American Museum of Natural History, New York, NY, USA

ARTICLE INFO

Keywords:
Arboreal phylogenetic network
Triplet & duet systems
Reconstruction algorithms
Metrics

ABSTRACT

Arboreal networks are multi-rooted phylogenetic networks whose underlying graph is a tree. We give an encoding of stack-free arboreal networks in terms of triplets and the novel concept of a duet. This yields a polynomial time algorithm to construct these networks from complete triplet and duet systems. The classification results show correctness and lead to a natural metric on these multi-rooted networks.

1. Introduction

An improved understanding of the complex processes that drive molecular evolution has led to extensive interest in phylogenetic networks to represent them. Recently introduced classes of multi-rooted (phylogenetic) networks include forest-based networks [1] and arboreal networks [2]. Both extend the notion of the popular tree-based networks [3] and may be useful for studying lateral gene transfer between bacteria living in ecologically distinct niches such as the hand and the human gut [4]. Both are multi-rooted directed acyclic graphs whose leaf set is a fixed set of taxa, and the study of their combinatorial structure has revealed intriguing links with, for example, Ptolemaic graphs [2]. The computational complexity of recognizing forest-based networks was studied in [5] under various scenarios. Their class encompasses the class of Overlaid Species Forests which have been used to shed light into introgression in Heliconius butterflies in [6].

We focus on arboreal networks, an intriguing class of multi-rooted networks whose underlying structure is a tree. We give a new combinatorial classification for arboreal networks which yields a natural metric on the duets (rooted 2-leaf subtrees) and triplets (rooted 3-leaf binary subtrees) compatible with arboreal networks. Using this characterization, we give a polynomial-time algorithm to reconstruct an arboreal network, if it exists, from the complete triplet and duet systems induced by the network. Allowing multiple roots in a network adds power to capture biological relationships, but, as we shall see, it also adds complexity. Our approach relies on the BUILD algorithm [7]. While the BUILD algorithm cannot directly reconstruct multi-rooted networks, a pre-processing step, based on duets, and additional clustering yields an efficient algorithm that matches the running time bounds of the BUILD algorithm.

2. Background

We follow the notation of [1,6], unless otherwise noted. Throughout, we assume that the leaf-label set is a finite set X with $|X| \geq 3$. Let G be a directed simple graph with vertex set $V(G)$ and arc (directed edge) set $E(G)$. For any multi-rooted directed acyclic graph (**mDAG**) G, we call a vertex of G with in-degree 0 and out-degree at least 2 a **root** of G and a vertex with in-degree 1 and out-degree zero a **leaf** of G. We denote the set of leaves of G by $L(G)$ and call every vertex of G that is neither a root nor a leaf an **interior vertex** of G. If all of the outward arcs of an interior vertex of G are adjacent to leaves, we call that set of leaves a **generalized cherry**. If a generalized cherry has size two, then we call it a **cherry**. We say that a vertex v of G is **above** a vertex u of G if there exists a root r of G such that a path from r to u crosses v, and we say that u is **strictly above** v if u is above v and $u \neq v$. We say that u is **below** v if v is above u. We let $U(G)$ denote the graph that results from G by ignoring the directions of its arcs. We call the graph, $U(G)^-$, obtained from $U(G)$ by suppressing all vertices of degree 2, the **underlying graph** of G (see Fig. 1(e,f)). An unrooted tree T is an **unrooted phylogenetic tree** if T contains no vertices of degree 2.

An m-**network** N (on X) is an mDAG with $m \geq 1$ roots such that $U(N)$ is connected, $L(N) = X$, a root has in-degree zero and out-degree 2, and an interior vertex has either out-degree 1 and in-degree at least 2 (**hybrid**) or in-degree 1 and out-degree at least 2 (**tree vertex**). We call an ordered pair (a, b) with $a \neq b \in X$ a **reticulated cherry** of N if b is the child of a hybrid h of N and h and a share a parent. For example, $(3, 4)$ is a reticulated cherry in the network in Fig. 1(c). Note that if $m = 1$, and the out-degree of the root is allowed to be 2 or more, then N is also called a **rooted phylogenetic network** (on X). Two m-networks N and N' on X are **isomorphic**, denoted by $N \simeq N'$,

* Corresponding author.
E-mail address: k.huber@uea.ac.uk (K.T. Huber).

https://doi.org/10.1016/j.ipl.2026.106664
Received 9 December 2025; Received in revised form 13 August 2026; Accepted 13 August 2026
Available online 15 August 2026
0020-0190/© 2026 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

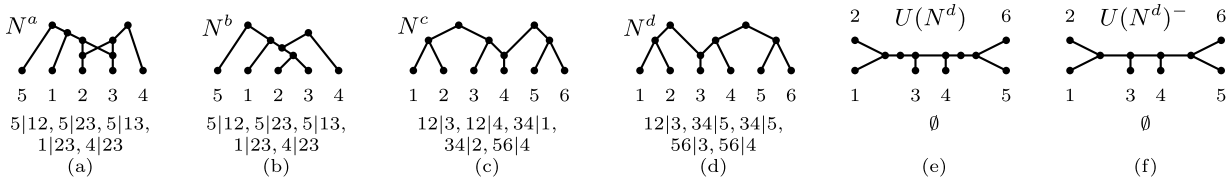

Fig. 1. Encoding networks and their induced triplet systems: the 2-network N^a is not encoded by its induced triplets since $\mathcal{R}(N^a) = \{5|12, 5|23, 5|13, 1|23, 4|23\} = \mathcal{R}(N^b)$ holds for the 2-network N^b. N^a is not arboreal, but N^b is. Two arboreal networks N^c and N^d that have the same underlying phylogenetic tree given in (f) and obtained from (e).

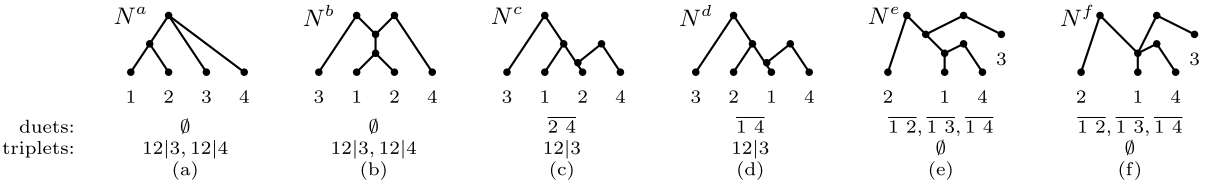

Fig. 2. Networks on $\{1, 2, 3, 4\}$: (a) An mDAG, N^a, that is a rooted phylogenetic tree, but not a 1-network, since the root has out-degree 3. (b) There are no triplets or duets that imply a relationship between leaves 3 and 4 in N^b. Thus, the only arboreal network that induces $\{12|3, 12|4\} = \mathcal{R}(N^a)$ is the 2-network N^b. This implies that N^b is encoded by $\mathcal{R}(N^b)$ within the class $\mathcal{A}(X)$ and illustrates the out-degree 2 requirement of a root of an m-network. (c) and (d) Arboreal networks N^c and N^d with $\mathcal{R}(N^d) = \mathcal{R}(N^c) \neq \emptyset \neq \mathcal{D}(N^c)$. (e) An arboreal network N^e that is not stack-free. (f) A stack-free arboreal network N^f such that $\mathcal{R}(N^e) \cup \mathcal{D}(N^e) = \mathcal{R}(N^f) \cup \mathcal{D}(N^f)$. Duet-triplet distance (see Section 5) examples: $d(N^b, N^c) = 2$ and $d(N^e, N^f) = 0$ since duets and triplets cannot account for stacked hybrids.

if they are isomorphic as mDAGs and this isomorphism is the identity on X.

A m-network N is called **binary** if every interior vertex has degree 3, and it is called **arboreal** if $U(N)^-$ is an unrooted phylogenetic tree. For $v \in V(N)$, we define $L(v)$ to be the set of all leaves of N that are below v. We define N_v to be the spanning tree of N with leaf set $L(v)$. There can be multiple arboreal networks for the same underlying phylogenetic tree (see Fig. 1). We call a rooted phylogenetic tree on $X = \{x, y, z\}$ with cherry $\{x, y\}$ a **(rooted) triplet (on X)** and denote it by $xy|z$ (or alternatively, by $z|xy$) where, the order of x and y does not matter. We refer to a set $\mathcal{R}$ of triplets as a **triplet system (on $L(\mathcal{R}) := \bigcup_{t\in\mathcal{R}} L(t)$)**. These arise naturally in the context of arboreal networks. Let N be an arboreal network on X and let $x, y, z \in X$ denote three pairwise distinct elements. Then we say that the triplet $xy|z$ is **induced** by N if there exists a root r of N such that $x, y, z \in L(r)$ and z is not below the least common ancestor of x or y. For example, the triplet 56|4 is induced by the arboreal network in Fig. 1(c). The set $\mathcal{R}(N)$ of all triplets induced by an arboreal network, N, can be empty as in Fig. 2(e).

3. Combinatorial classifications

A rooted phylogenetic tree T on X is **encoded** by $\mathcal{R}(T)$ (within the class $\mathcal{T}$ of rooted phylogenetic trees) if any other tree T' in $\mathcal{T}$ for which $\mathcal{R}(T) = \mathcal{R}(T')$ is isomorphic with T [8,9]. Extending the notion of an encoding to m-networks by replacing "rooted phylogenetic tree" by "m-network" in the definition of an encoding, then even if a 1-network N is binary and no two cycles in $U(N)^-$ share a vertex it is not, in general, encoded by $\mathcal{R}(N)$ [10]. Interestingly, such 1-networks are encoded by their induced set of trinets [11] where an encoding in terms of such structures is defined analogously to the triplet system case. Without going into details, a trinet can be thought of as a natural generalization of a triplet to a 1-network on 3 leaves (see [12,13] for more on encodings of 1-networks by trinets). Since trinets are, in particular, 1-networks it follows that they cannot directly be used as a starting point for finding an "encoding" for general arboreal networks.

Even if an m-network N is arboreal it is not, in general, encoded by $\mathcal{R}(N)$. The two arboreal networks N^c and N^d in Fig. 2 induce the same triplet system but are not isomorphic. To distinguish these networks, we define a combinatorial structure on pairs of leaves that is induced by an m-network: we call an unordered pair of two distinct elements $x, y \in X$ a **duet** and denote it by $\overline{x\ y}$ or, alternatively, $\overline{y\ x}$. Moreover, we refer to a set $\mathcal{D}$ of duets as a **duet system (on $L(D) := \bigcup_{\overline{a\ b}\in D}\{a, b\}$)**. Let N be an m-network on X. Then we say that a duet $\overline{x\ y}$ is **induced by** N if there exists no triplet $t \in \mathcal{R}(N)$ such that $x, y \in L(t)$ and the path in $U(N)$ joining x and y crosses precisely one degree two vertex of $U(N)$. For example, the duet $\overline{1\ 4}$ is induced by the m-network in the form of the arboreal network in Fig. 2(d). The arboreal networks in Figs. 1 and 2(b), demonstrate that there exist m-networks N for which $\mathcal{D}(N)$ is empty.

We extend the notion of an encoding of a rooted phylogenetic tree to m-networks N in a more restrictive way: in addition to making the same replacements as before we also replace "$\mathcal{R}(N)$" by "$\mathcal{R}(N) \cup \mathcal{D}(N)$". As the examples in Fig. 2(e,f) show even m-networks that are arboreal are, in general, not encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$. For this example, it is due to N^e containing a **stacked hybrid**, that is, an arc whose head and tail is a hybrid. We call an arboreal network N **stack-free** if N does not contain such an arc. Furthermore, we say that N is **banyan** if the parents of every hybrid of N are roots of N and every leaf of N is the child of a root or the child of a hybrid. For example, the arboreal network in Fig. 2(f) is banyan. Denoting the class of stack-free arboreal networks on X by $\mathcal{A}(X)$, we have:

Lemma 1. *Let N be a network in $\mathcal{A}(X)$. Then the following are equivalent.*

(i) N is banyan.
(ii) $\mathcal{R}(N) = \emptyset$.
(iii) N is encoded by $\mathcal{D}(N)$ within $\mathcal{A}(X)$.

Proof. (i) ⇒ (ii): Suppose that N is banyan. Then, every leaf is the child of a root or of a hybrid. For N to induce a triplet $xy|z$ with $x \neq y \neq z$ in X, the shared parent p of x and y cannot be a root. So, p must be a hybrid, but this contradicts the fact that the child of a hybrid is a leaf. Thus, $\mathcal{R}(N) = \emptyset$.

(ii) ⇒ (i): Assume that $\mathcal{R}(N) = \emptyset$ and, for contradiction, that N is not banyan. Then there exists a leaf l of N that is neither the child of a hybrid nor of a root of N, or there exists a hybrid h of N that has a parent that is not a root of N. In the former, it follows that l's parent p must be a tree vertex of N. Choose a vertex v strictly above p and leaves l_v and l_p such that l_v is below v but not below p and l_p is below p but distinct from l. Then $l_p l|l_v \in \mathcal{R}(N)$; a contradiction as $\mathcal{R}(N) = \emptyset$. For the latter case, assume that p is a parent of h that is not a root of N. Then since N is stack-free, p must be a tree vertex. Let v be a root above p and let l_v be a leaf below v but not below p. Choosing l_p as before and denoting by l a leaf below h implies again $\mathcal{R}(N) \neq \emptyset$; a contradiction.

(i) & (ii) ⇒ (iii); Suppose that N is banyan and that there exists a network $N' \neq N$ in $\mathcal{A}(X)$ such that $\mathcal{R}(N') \cup \mathcal{D}(N') = \mathcal{D}(N) \cup \mathcal{R}(N)$.

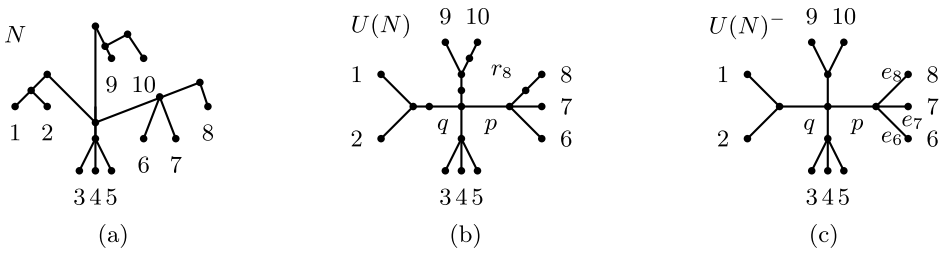

Fig. 3. (a) An arboreal network N on $X = \{1, \ldots, 10\}$ considered in the case of $|S| \geq 2$ in the proof of Theorem 1. (b) The tree $U(N)$ on X. (c) The phylogenetic tree $U(N)^-$ on X. For the vertex p of $U(N)^-$ as indicated, $Y = \{6, 7, 8\}$ and $S = \{6, 7\}$.

Then, by (ii), $\mathcal{R}(N) = \emptyset$. Since $\mathcal{R}(M) \cap \mathcal{D}(M) = \emptyset$ holds for any arboreal network M, it follows that $\mathcal{R}(N') = \emptyset$. By the equivalence of (i) and (ii), N' must also be banyan as it is stack-free. Since $\mathcal{D}(N') = \mathcal{R}(N') \cup \mathcal{D}(N') = \mathcal{R}(N) \cup \mathcal{D}(N) = \mathcal{D}(N)$, it follows that every element in X must be contained in the same number of duets in $\mathcal{D}(N)$ as in $\mathcal{D}(N')$. For all $x \in X$, we therefore have that x is either adjacent to a root in N if and only if x is adjacent to a root in N' or that x is adjacent to a hybrid in N if and only if x is adjacent to a hybrid in N'. Since the length of a path from a root to a leaf in a banyan network is one or two, it follows that there exists a bijection from $V(N)$ to $V(N')$ that induces an isomorphism between N and N'. Thus, N is encoded by $\mathcal{D}(N)$.

(iii) $\Rightarrow$ (ii): Assume that N is encoded by $\mathcal{D}(N)$. Then the definition of a duet combined with the fact that $X = L(\mathcal{D}(N))$ implies that $\mathcal{R}(N) = \emptyset$. □

Fig. 2(e,f) shows the necessity that the arboreal network be stack-free for Lemma 1 to hold.

Theorem 1. *A network N in the class $\mathcal{A}(X)$ is encoded by $\mathcal{D}(N) \cup \mathcal{R}(N)$ within that class.*

Proof. We use induction on $n = |X|$. The base case: $n = 3$. Then N has either one or two roots. If N has one root, then N is a rooted triplet, since a root has out-degree 2. Clearly, N is encoded by $\mathcal{R}(N)$ in this case. If N has two roots, then N is banyan. By Lemma 1, N is encoded by $\mathcal{D}(N)$.

Inductive Step: Assume true for n and show true for $n+1$. Let $|X| = n+1$ and assume that $N \in \mathcal{A}(X)$. In view of Lemma 1, we may assume that $\mathcal{R}(N) \neq \emptyset$ since otherwise N is banyan and is encoded by $\mathcal{D}(N) = \mathcal{D}(N) \cup \mathcal{R}(N)$. Since $\mathcal{R}(N) \neq \emptyset$, it follows that $U(N)^-$ cannot be a star tree. Choose an interior vertex p of $U(N)^-$ such that p is adjacent to every element in a generalized cherry Y and precisely one interior vertex in $U(N)^-$, denoted q. Note that q must exist since $U(N)^-$ cannot be a star tree. Let S denote the set of elements $y \in Y$ such that $\{y, p\}$ is also an edge in $U(N)$. We perform a case analysis on $|S|$ – see Fig. 3 for an illustration of the case $|S| \geq 2$. For all $y \in Y$, let e_y denote the edge $\{p, y\}$ in $U(N)^-$ and, for all $y \in Y - S$, let r_y denote the subdivision vertex of e_y in $U(N)$. Note that r_y is a root of N for all such y.

Case $|S| = 0$: Then, for all $y \in Y$, the edge e_y of $U(N)^-$ is subdivided by the vertex r_y in $U(N)$. Hence, r_y is a root of N for all such y. Thus, p must be a hybrid in N. Since the out-degree of a hybrid of N is 1 it follows that q must be the child of p in N. Furthermore, the fact that N is in $\mathcal{A}(X)$ combined with the fact that q is an interior vertex of $U(N)^-$, implies that q is a tree vertex of N. Choose some element $x \in Y$. Note that $x \neq q$ because $q \notin Y$. Let N' denote the mDAG with leaf set $X' := X - \{x\}$ obtained from N by deleting the vertices x and r_x and the arcs (r_x, x) and (r_x, p) from N (and suppressing p if this has rendered it a vertex with in-degree 1 and out-degree 1). Since $|Y| \geq 2$ it follows that $N' \in \mathcal{A}(X')$ since $N \in \mathcal{A}(X)$. By induction hypothesis, N' is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Since q is an interior vertex of $U(N)^-$, we have $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$. Since N can be recovered from N' by adding the vertices x and r_x and the arcs (r_x, x) and (r_x, p) to N' in case p was not suppressed in the construction of N' from N and this only adds triplets in $\mathcal{R}(N)$ whose leaf set contains x, it follows that N is also encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$. If p was suppressed in the construction of N' from N, we first subdivide the incoming arc of q in N' by a new vertex p' and add the vertices x and r_x and the arcs (r_x, x) and (r_x, p') to N'. Similar arguments imply that N is encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$.

Case $|S| = 1$: Let $S = \{s\}$. If p is a non-hybrid, we can remove s, the arc (p, s) and suppress p to obtain a network N' in $\mathcal{A}(X - \{s\})$. By induction, N' is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Since $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : s \in L(t)\}$, similar arguments as before imply that N is encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$. If p is a hybrid, choose an element $x \in Y - S$ which must exist since $|Y| \geq 2$. Constructing N' from N as before implies that N' is a network in $\mathcal{A}(X - \{x\})$ that is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Note that contrary to the previous case, $\mathcal{D}(N') = \mathcal{D}(N) - \{\overline{x\ s}\}$ and $\mathcal{R}(N') = \mathcal{R}(N)$. Clearly, N can be recovered from N' by adding the vertices x and r_x and the arcs (r_x, x) and (r_x, p) in case p was not suppressed in the construction of N' and by subdividing the incoming arc of s in N' by a vertex p' and adding the vertices x and r_x and the arcs (r_x, x) and (r_x, p') otherwise. Thus, N is again encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$ since this only adds duets from $\mathcal{D}(N)$ to $\mathcal{D}(N')$ that contain x.

Case $|S| \geq 2$: Then p must be a tree vertex of N. Hence, one of the following two cases must hold: (i) $S = Y$ and so the arc (q, p) or the arcs $a_p := (r, p)$ and $a_q := (r, q)$ are contained in N where r is a root of N or (ii) $|S| = |Y| - 1$ and (p, q) is an arc in N.

Case (i): Choose some $x \in S$. Then $x \neq q$ since $S \subseteq Y$ and $q \notin Y$. If $|S| \geq 3$, then let N' denote the mDAG with leaf set $X' := X - \{x\}$ obtained from N by deleting x and the arc (p, x). Note that the out-degree of p in N' is still at least 2. Then similar arguments as in the previous cases imply that N' is a network in $\mathcal{A}(X')$ that is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Furthermore and independent of whether $(q, p) \in E(N)$ or the arcs a_p, a_q are in $E(N)$, we have $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$. Since N can be recovered from N' by adding x and the arc (p, x), similar arguments as before imply that N is encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$.

So assume that $|S| = 2$. Let N' denote the mDAG with leaf set X' obtained from N by deleting x and the arc (p, x) and suppressing p. Then, similar arguments as in the previous cases imply that N' is a network in $\mathcal{A}(X')$ that is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Let y denote the other element in S. Since q is an interior vertex of $U(N)^-$ it cannot be a root of N.

Assume $a_p, a_q \in E(N)$. If q has at least 2 leaves below it, then $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$. Otherwise, let l be the sole leaf below q. Then $\mathcal{D}(N') = \mathcal{D}(N) \cup \{\overline{l\ y}\}$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{l|xy\}$. Similar arguments as before imply that N is encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$.

Assume that $(q, p) \in E(N)$. If q is a tree vertex, then we have $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$. Similar arguments imply that N is also encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$. If q is a hybrid of N, then $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$ holds again. Since N is stack-free, a parent of q in N must either be a tree vertex or a root of N. Let Q denote the set of parents of q that are roots in N and whose other child is a leaf of N. If $Q = \emptyset$ then $\mathcal{D}(N') = \mathcal{D}(N)$. Otherwise $\mathcal{D}(N') = \mathcal{D}(N) \cup \{\overline{d\ y} \mid d \in Q\}$. Similarly, the result holds.

ARBOREAL RECONSTRUCTION ALGORITHM (ARA):
INPUT: A triplet system, $\mathcal{R}$, and a duet system, $\mathcal{D}$, on $X := L(\mathcal{R}) \cup L(\mathcal{D})$ with $n := |X|$.
OUTPUT: A network $N \in \mathcal{A}(X)$ with $\mathcal{R} = \mathcal{R}(N)$ and $\mathcal{D} = \mathcal{D}(N)$ or None, if no network exists.

1.	Partition X corresponding to disjoint network regions (see Sec. 4.1).	$O(\lvert\mathcal{R}\rvert + \lvert\mathcal{D}\rvert) = O(n^3)$
2.	Build a scaffold linking the roots to the elements in X (see Sec. 4.2 & Fig. 6).	$O(n^3)$
3.	For each block in the partition, iteratively refine the tree under each root in that block, merging common subtrees (see Sec. 4.3 and Fig. 7).	$O(n^3)$
4.	Return the network, or None, if no network is possible (see Sec. 4.4).	$O(1)$

Fig. 4. Outline of the ARBOREAL RECONSTRUCTION ALGORITHM (ARA) with running time complexity in the right margin.

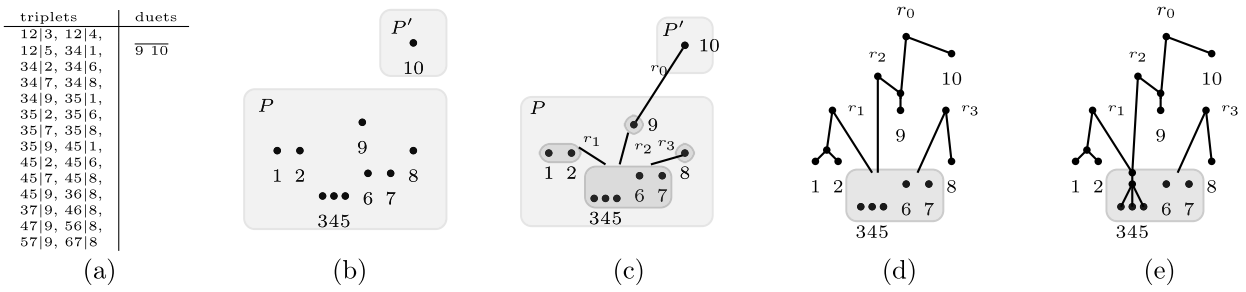

Fig. 5. AN ARA EXAMPLE: (a) The systems $\mathcal{R}(N)$ and $\mathcal{D}(N)$ for the network N from Fig. 3(a). After Line 1: (b) The $(\mathcal{D}(N), \mathcal{R}(N))$-induced partition of the leaves with blocks P and P' with $P = \{1, 2, \ldots, 9\}$ and $P' = \{10\}$. After Line 2: (c) we have edges for each root induced by the duet (r_0) and the triplets (r_1, r_2, r_3). P' has a single component while there are three components of P. After Line 3: (d) The components with one or two leaves refined, and (e) refining the component with leaf set $\{3, 4, 5, 6, 7\}$ is done by reconstructing the structure below each root. First, the tree on the leaf set below r_1, $L_{r_1} = \{3, 4, 5\}$, is constructed. Since $L_{r_1} = L_{r_2}$, their subtrees are identical. Merging results in the subtree below a hybrid. Next, the tree under r_3 on $L_{r_3} = \{3, 4, 5, 6, 7\}$ is reconstructed and merged, also resulting in a hybrid. Since all blocks have now been processed, the generated network (which is the network in Fig. 3(a)) is returned in Line 4.

Case (ii): Choose some $x \in S$. As before, $x \neq q$. Construct an mDAG N' from N as in the previous cases by removing x. Note that the out-degree of p in N' is still at least 2 because q is a child of p. Using similar arguments as before, it follows that N' is a network in $\mathcal{A}(X - \{x\})$ that is encoded by $\mathcal{R}(N') \cup \mathcal{D}(N')$. Furthermore, $\mathcal{D}(N') = \mathcal{D}(N)$ and $\mathcal{R}(N') = \mathcal{R}(N) - \{t \in \mathcal{R}(N) : x \in L(t)\}$. Similar arguments imply that N is encoded by $\mathcal{R}(N) \cup \mathcal{D}(N)$. □

4. Reconstructing arboreal networks

We present an algorithm, ARA, that efficiently reconstructs an arboreal network from them, if one exists, when given the basic building blocks of duets and triplets. It relies on our combinatorial classification (Theorem 1) and the work of Bryant [8]. Bryant [8] used the BUILD algorithm of Aho et al. [7] to determine, in polynomial time, if a triplet system is compatible with a rooted phylogenetic tree. Bryant further showed that if the triplet system is not compatible, finding a maximal set of compatible triplets is computationally hard. The BUILD algorithm first constructs a graph whose components correspond to the subtrees of the root of an envisaged tree. If there is only one component, then the root would have only one child which is not possible, and the algorithm halts. Otherwise, the process is recursively applied to the subcomponents until the tree is resolved. If the triplets are not compatible, no tree is built and the algorithm halts. A tree can have **polytomies** i. e. vertices with out-degree larger than 3, if information about the relationships in the triplets is limited.

We can show similar results for arboreal networks and duets and triplets, with a subtle difference. In the case of arboreal networks, the absence of triplets and duets is itself a statement about the possible network. For the triplet system $\mathcal{C} = \{12|3, 12|4\}$ in Fig. 2(a), any binary rooted phylogenetic tree that induces $\mathcal{C}$ also induces one of $t_1 = 24|3$ or $t_2 = 23|4$. Thus, there exists no such tree T such that $\mathcal{R}(T) = \mathcal{C}$. With multiple roots allowed, there is an arboreal network N^b (Fig. 2(b)) such that $\mathcal{R}(N^b) = \mathcal{C}$. Fig. 4 gives an overview of ARA with the details of each step provided in the subsequent sections and figures. We illustrate its inner workings in Fig. 5.

4.1. A partition induced by duets and triplets

Given a duet system $\mathcal{D}$ and a triplet system $\mathcal{R}$, we first find a partition of $X := L(\mathcal{D}) \cup L(\mathcal{R})$ whose blocks (see the shaded areas in Fig. 5(b)) can be processed independently. To this end, we call a partition $\mathcal{P}$ of X a $(\mathcal{D}, \mathcal{R})$-*induced partition* if, for all duets $\overline{x\ y} \in \mathcal{D}$, there is no block in $\mathcal{P}$ that contains $\{x, y\}$, and for all triplets $t \in \mathcal{R}$, there exists a block in $\mathcal{P}$ that contains $L(t)$. Note that $\mathcal{P}$ is well-defined since if $\overline{x\ y} \in \mathcal{D}$ then, by definition of a duet, there is no $z \in X$ such that $xy|z \in \mathcal{R}$. We use this type of partition as a framework to the later subroutines that make up ARA and compute it in Proposition 2. The blocks in the partition we then refine further into components connected by edges that correspond to the roots in the envisaged network.

The next result is a consequence of Theorem 1.

Proposition 1. *Let N be an arboreal network. Then each duet in $\mathcal{D}(N)$ gives rise to a root of N.*

Continuing with the notation in Fig. 5, a $(\mathcal{D}, \mathcal{R})$-partition can be found by an algorithm whose running time is dominated by $|\mathcal{R}|$ which is at most $O(n^3)$:

Proposition 2. *Let $\mathcal{R}$ be a triplet system and let $\mathcal{D}$ be a duet system such that $X = L(\mathcal{R}) \cup L(\mathcal{D})$. Then a $(\mathcal{D}, \mathcal{R})$-induced partition can be computed in $O(|\mathcal{R} \cup \mathcal{D}|) = O(|X|^3)$ time.*

Proof. We first initialize several look-up tables for more efficient access. We set up a table of triplets with values, *unused*, to keep track of which triplets we have used and a table, *belong*, of triplets in terms of leaves: for each $xy|z \in \mathcal{R}$, append $xy|z$ to *belong*$[x]$, to *belong*$[y]$, and to *belong*$[z]$. The initialization takes $O(|\mathcal{R}|) = O(|X|^3)$ and allows us to find the triplets in constant time in the steps below. We also initialize to 0 a table *which* of the leaves.

SCAFFOLD SUBROUTINE:
INPUT: Lists, $\mathcal{D}$, of duets and $\mathcal{R}$ of triplets, and a $(\mathcal{D},\mathcal{R})$-induced partition, $\mathcal{P}$, of $X := L(\mathcal{D}) \cup L(\mathcal{R})$.
OUTPUT: A list, *comps*, of triplet systems, a list, re, of root edges and an ancestor (look-up) table, *anc*, for X.

1.	Initialize $comps = \emptyset$, $E = \emptyset$, $re = \emptyset$, $\forall l \in X$, $anc[l] = \emptyset$.			$O(n)$
2.	For each $ab\|c \in \mathcal{R}$: Append $\{a,b\}$ to E.			$O(\|\mathcal{R}\|)$
3.	For each $\overline{a\ b} \in \mathcal{D}$: Append $\{a,b\}$ to E, re, $anc(a)$, & $anc(b)$.			$O(\|\mathcal{D}\|)$
4.	For each block $P \in \mathcal{P}$:			$O(n^2)$
5.	Construct graph, $G_P = (P, \{\{u,v\} \in E : u,v \in P\})$.		$O(\|P\|^2)$	
6.	If G_P is connected and $\|P\| > 1$, the construction halts, as in [1].		$O(\|P\|)$	
7.	Else: Let $C_P = \{C_{P,1}, \ldots, C_{P,k}\}$ be the component leaf sets of G_P.		$O(\|P\|)$	
8.	Append the set C_P to *comps*.		$O(\|P\|)$	
9.	For each distinct $C, C' \in C_P$:		$O(\|P\|^2)$	
10.	If $ab\|c \in \mathcal{R}$, $a \in C$, and $c \in C'$,	$O(1)$		
11.	Add edge $e = \{C, C'\}$ to re.	$O(1)$		
12.	Append e to $anc[a]$ and $anc[b]$.	$O(1)$		
13.	Return re, *comps*, and *anc*.			$O(1)$

Fig. 6. Outline of the SCAFFOLD subroutine with running time complexity $O(|\mathcal{D}| + |\mathcal{R}|) = O(n^3)$. Bounds on the running time of each line is in the margin with loops containing the complexity of all nested operations.

For each duet, $\overline{a\ b} \in \mathcal{D}$, we set up queues $q_a = (a)$ and $q_b = (b)$. We remove the first item, x, from q_a. We set $which[x] = a$ and add the leaves of all unused triplets involving x to q_a, marking them as used. We repeat these steps until q_a is empty. We similarly work through q_b with an extra check to see if a leaf has already been assigned to a. If so, we return None.

Note that after working through all duets, every leaf must have been assigned, since every leaf must be a part of some triplet or duet. We process each duet and triplet once and the resulting $(\mathcal{D},\mathcal{R})$-partition of X has $|\mathcal{D}| + 1$ blocks and can be determined in linear time in the size of $\mathcal{R}$. □

4.2. Initializing the scaffold

As suggested by Fig. 5(b), a necessary condition for a $(\mathcal{D},\mathcal{R})$-induced partition, $\mathcal{P}$, to be induced by an arboreal network is that $\mathcal{D}$ induces a tree on the blocks of $\mathcal{P}$. We therefore build up the envisaged network using a tree as a scaffold. This scaffold starts with the edges, representing the roots induced by the duets in $\mathcal{D}$ and triplets in $\mathcal{R}$. To do this efficiently, we compute a look-up table of the roots that will be above each element in X, an augmented list of edges corresponding to the roots induced by the duets and by the triplets, and the induced components.

We show for a network, $N \in \mathcal{A}(X)$, that the SCAFFOLD subroutine returns the correct structure when given $\mathcal{D}(N)$, $\mathcal{R}(N)$ and the $(\mathcal{D}(N),\mathcal{R}(N))$-induced partition of X obtained in Proposition 2. More precisely, we define the **component graph**, $G(N)$, of an arboreal network N as follows: Let $C(N)$ be the partition of X when all degree two vertices, and their adjacent edges are removed from N. Let $EC(N)$ be the set of removed edges, that is: $\{C, C'\} \subseteq C(N)$ is an edge precisely if there is a degree 2 vertex that is adjacent to C and C'. Let $G(N) = (C(N), EC(N))$. Note that we will use the ancestor table *anc* built in Line 12 in the REFINE subroutine and show the correctness there.

Proposition 3. *Let N be an arboreal network on X. Let re, comps and anc result from running the* SCAFFOLD *subroutine on the $(\mathcal{D}(N),\mathcal{R}(N))$-induced partition. Let $G' = (comps, re)$. Then, $G(N) \simeq G'$.*

Proof. We use induction on the number c of components of $G(N)$. Let $\mathcal{P}(N)$ be the $(\mathcal{D}(N),\mathcal{R}(N))$-induced partition of X. The base case: $c = 2$. Then N has a single root, and we have two cases. The first case is that N induces only a single duet, $\overline{a\ b}$ and no triplets. The edge $r = \{a,b\}$ is added to E, re, $anc(a)$ and $anc(b)$ in Line 3 of Fig. 6, while *comps* is updated to $\{\{a\},\{b\}\}$ in Line 8. The algorithm returns $comps = \{\{a\},\{b\}\}$, $re = \{r\}$, and the ancestor tables $anc[a] = r$ and $anc[b] = r$. The resulting graph $G' = (comps, re) = (\{\{a\},\{b\}\},\{r\})$ is exactly $G(N)$. The second case is $|\mathcal{D}(N)| = 0$. Then $\mathcal{P}(N)$ has a single block, and $G(N)$ has two vertices C and C' and a single edge joining them. Then, N is a rooted phylogenetic tree, and Line 2 of Fig. 6 yields two maximal cliques on X. Thus, G_P has two component leaf sets $C_{P,1}$ and $C_{P,2}$, and $\{C_{P,1}, C_{P,2}\}$ is appended to *comps* in Line 8. Since $|X| \geq 3$, there is a triplet that crosses the root of N. Thus, in Line 11, the edge $\{C_{P,1}, C_{P,2}\}$ is added to re. Since both $G(N)$ and the graph $G = (comps, re)$ have two vertices and one edge, they are isomorphic.

Inductive Step: Assume true for c and show true for $c + 1$. Again, we have two cases: The first case is $|\mathcal{D}(N)| > 0$. Then, $|\mathcal{P}(N)| > 1$. Let $\overline{a\ b} \in \mathcal{D}(N)$ and let N' and N'' be the restrictions of N to each side of the duet. The number of components in N and in N' is at most c. By the inductive hypothesis, $G(N')$ and $G(N'')$ are isomorphic to the graphs returned by the SCAFFOLD subroutine. Since the blocks are processed independently of each other in Fig. 6, the resulting graph is $G(N')$ and $G(N'')$ with the additional edge $\{a,b\}$ added (Line 3) which is isomorphic with $G(N)$.

The second case is $|\mathcal{D}(N)| = 0$. Then, $\mathcal{P}(N)$ has a single block, P. Since N is an arboreal network, there exists a subtree, T, of $U(N)$ which is adjacent to a vertex of degree 2. Since T and the arboreal network $N|_{X-L(T)}$ obtained by restricting N to $X - L(T)$ both have at most c components and the inductive hypothesis applies, the respective component graphs and the graphs resulting from Fig. 6 are isomorphic. By choice of T, there is a single edge connecting T and $N|_{X-L(T)}$ in $G(N)$ as well as constructed in Line 10, yielding the desired isomorphism. □

4.3. Refining the partition

In the previous step, we decomposed each block of a $(\mathcal{D},\mathcal{R})$-induced partition of X into components. By calling the REFINE subroutine on each component, we can reconstruct an arboreal network (if one exists). For each component, C, we iterate through the roots above C, constructing the tree of descendants. Given that for an arboreal network N the graph $U(N)^-$ is a phylogenetic tree, if a set of leaves is below two different roots, then their overlap forms a subtree with identical structure under each root:

Lemma 2. *Let N be an arboreal network and let r and r' be roots of N such that $L(r) \cap L(r') \neq \emptyset$. Then there exists a vertex, v, on the (undirected) path from r to r' that is a hybrid below both r and r'. Further, when viewed as phylogenetic trees, the subtree of N_r rooted at the child c of v is isomorphic with the subtree of $N_{r'}$ rooted at c.*

Proof. Choose some $x \in L := L(r) \cap L(r')$. Let P_{rx} and $P_{r'x}$ denote the paths in N from r to x and r' to x, respectively. Since x is a vertex in both paths, there must exist a vertex v of N that is the first vertex on P_{rx} that also lies on $P_{r'x}$. By construction, v has in-degree at least 2 and so must be a hybrid of N. Let K be the subtree of N_r rooted at the

REFINE SUBROUTINE:
INPUT: A triplet system $\mathcal{C}$ with $m = |L(\mathcal{C})| \leq n$ and an ancestor (look-up) table anc.
OUTPUT: The arc set, $arcs$, of an arboreal network N, on $\mathcal{L} := L(\mathcal{C}) \cup values(anc)$ such that $\mathcal{C} = \mathcal{R}(N)$, or None if no network exists.

0.	Initialize table $arcs$ to $\emptyset$.		$O(1)$
1.	For each r in the values of anc table:		$O(\|\mathcal{C}\|)$
2.	Let $L_r = \{l \in \mathcal{L} : r \in anc[l]\}$	$O(m)$	
3.	Run BUILD on $\mathcal{C}$ restricted to L_r resulting in a rooted phylogenetic tree $T(r)$.	$O(\|\mathcal{C}\|)$	
4.	If the root of $T(r)$ does not have out-degree 2, return None.	$O(1)$	
5.	Else: Add the arcs of $T(r)$ to $arcs$, merging common arcs.	$O(\|L_r\|)$	
6.	Add the arc (r, r') to $arcs$ where r' is the root of $T(r)$.	$O(1)$	
7.	Return $arcs$.		$O(1)$

Fig. 7. Outline of the REFINE subroutine with running time complexity $O(|\mathcal{C}|) = O(m^3)$. Bounds on the running time of each line is in the margin with loops containing the complexity of all nested operations.

child c of v and let K' be the subtree of $N_{r'}$ rooted at c, with degree two vertices suppressed in each case to yield phylogenetic trees. Then $L(K) = L = L(K')$. Since $\mathcal{R}(K)$ equals the restriction of $\mathcal{R}(N)$ to L and similarly for $\mathcal{R}(K')$, K and K' must be isomorphic phylogenetic trees [9, Theorem 6.4.1]. □

4.4. Correctness & running time

Using the analysis above, we show that ARA is correct and runs in $\Theta(|\mathcal{R}|) = \Theta(n^3)$.

Lemma 3. *Let $N \in \mathcal{A}(X)$. Let N' be the network built by* ARA *from $\mathcal{R}(N)$ and $\mathcal{D}(N)$. Then $N \simeq N'$.*

Proof. We rely on Theorem 1 that if $\mathcal{D}(N) = \mathcal{D}(N')$ and $\mathcal{R}(N) = \mathcal{R}(N')$, then N and N' are isomorphic. By Proposition 1, every duet corresponds to a root in N. By construction, a root is added to N' for each duet, yielding, $\mathcal{D}(N) = \mathcal{D}(N')$.

To show that $\mathcal{R}(N) = \mathcal{R}(N')$, let $ab|c \in \mathcal{R}(N)$. We have two cases. The first case that $ab|c$ is *underneath* a root – that is the undirected paths between the three leaves do not cross a root. Let r be that root. Then $a, b, c \in L(r)$. Let C be the leaf set of a connected component of the graph G_P (see Line 7 of the SCAFFOLD subroutine) such that $a, b, c \in C$. To construct N', the REFINE subroutine is run on all components. In Line 3, the BUILD algorithm [7] is run on $\mathcal{R}(N)$ restricted to the set L_r defined in that line, which generates a subtree $T(r)$ also defined in that line which induces all triplets $t \in \mathcal{R}(N)$ with $L(t) \subseteq C$. Thus, $ab|c$ is induced by the generated subtree. By Lemma 2, $ab|c \in \mathcal{R}(N')$. To show the reverse direction for triplets underneath a root, we note that if $ab|c \in \mathcal{R}(N')$ and below all roots, then it was added to N' in Line 6 of the REFINE subroutine. For that to occur, there exists a root r in N such that $a, b, c \in L(r) \cap C$ for C the leaf set of a component (see Line 5 of the SCAFFOLD subroutine). The BUILD algorithm on the triplets in $\mathcal{R}(N)$ underneath r yields $ab|c$. By the correctness of the BUILD algorithm [8], $ab|c \in \mathcal{R}(N)$. Thus, $\mathcal{R}(N) = \mathcal{R}(N')$ in this case.

The second case is that $ab|c \in \mathcal{R}(N)$ crosses a root, r, of N'. Then there exist components with leaf sets C and C', with $C \neq C'$ and $a, b \in C$ and $c \in C'$ (Line 10 of the SCAFFOLD subroutine), and C and C' are both below r. After applying the REFINE subroutine, each component below a root is a subtree of $U(N')^-$, $a, b \in C$ and $c \in C'$. Thus, $ab|c \in \mathcal{R}(N')$. Lastly, consider the case that $ab|c \in \mathcal{R}(N')$ and crosses a root, r, in N'. As before, there exist components with leaf sets C and C', with $C \neq C'$ and $a, b \in C$ and $c \in C'$ (see Line 10 of the SCAFFOLD subroutine). Similar arguments as before imply $ab|c \in \mathcal{R}(N)$. Thus, $\mathcal{R}(N) = \mathcal{R}(N')$ holds in this case. □

Theorem 2. *Let $\mathcal{D}$ be a duet system and $\mathcal{R}$ a triplet system. For $X = L(\mathcal{D}) \cup L(\mathcal{R})$, we can decide in polynomial time in $|X|$, if there exists an arboreal network N on X such that $\mathcal{R}(N) = \mathcal{R}$ and $\mathcal{D}(N) = \mathcal{D}$.*

Proof. The running time complexity is given line-by-line in Figs. 4, 6, and 7 and is polynomial in $|X|$. □

5. A metric for stack-free arboreal networks

Since triples are not enough to encode a network in $\mathcal{A}(X)$ but triples and duets are (Theorem 1), we obtain a metric on $\mathcal{A}(X)$ that canonically extends the popular triplet distance [14] for rooted phylogenetic trees. More precisely, we define the **duet-triplet distance** $d(N, N')$ between networks $N, N' \in \mathcal{A}(X)$ as $|(\mathcal{D}(N) \cup \mathcal{R}(N))\Delta(\mathcal{D}(N') \cup \mathcal{R}(N'))|$ where Δ is the symmetric distance (see Fig. 2 for examples).

To see that our distance is a metric, note that non-negativity, symmetry, and the triangle inequality follow directly from the symmetric difference. For the identity condition, note that by the symmetric difference, $d(N, N') = 0$ if and only if $\mathcal{D}(N) = \mathcal{D}(N')$ and $\mathcal{R}(N) = \mathcal{R}(N')$. Since, by Theorem 1, a network $M \in \mathcal{A}(X)$ is encoded by $\mathcal{R}(M) \cup \mathcal{D}(M)$ it follows that $d(N, N') = 0$ if and only if $N \simeq N'$. We leave as an open question: what is the diameter of $\mathcal{A}(X)$ under the duet-triplet distance?

6. Conclusion

Arboreal networks are multi-rooted structures whose underlying graph is a tree. We give an elegant characterization of stack-free arboreal networks in terms of the rooted 2-sets and 3-sets that they induce. This characterization allows for an algorithm for reconstructing such networks with running time linear in the number of inputted duets and triplets, as well as an intuitive metric to compare the networks. We leave as a further open question: can we bound the number of the triplet/duets needed to reconstruct a stack-free arboreal network up to isomorphism?

CRediT authorship contribution statement

Katharina T. Huber: Writing – original draft, Methodology, Conceptualization; **Katherine St. John:** Writing – original draft, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors thank the referees for their helpful comments. They also thank the Institute for Computational and Experimental Research in Mathematics (ICERM) in Providence, RI (Fall 2024), the Center for Interdisciplinary Research (ZiF), University Bielefeld, Germany (Summer 2025), and the Banff International Research Station (BIRS) (August 2025) for hosting them.

Data availability

No data was used for the research described in the article.

References

[1] K.T. Huber, V. Moulton, G.E. Scholz, Forest-based networks, Bull. Math. Biol. 84 (10) (2022) 119.

[2] K.T. Huber, V. Moulton, G.E. Scholz, Shared ancestry graphs and symbolic arboreal maps, SIAM J. Disc. Math. 38 (4) (2024) 2553–2577.

[3] A.R. Francis, M. Steel, Which phylogenetic networks are merely trees with additional arcs?, Syst. Biol. 64 (5) (2015) 768–777.

[4] H. Jeong, B. Arif, G. Caetano-Anolles, K.M. Kim, A. Nasir., Horizontal gene transfer in human-associated microorganisms inferred by phylogenetic reconstruction and reconciliation, Sci. Rep. 9 (1) (2019) 1–18.

[5] G.E. Scholz, A.-A. Popescu, M.I. Taylor, V. Moulton, K.T. Huber, OSF-builder: a new tool for constructing and representing evolutionary histories involving introgression, Syst. Biol. 68 (5) (2019) 717–729.

[6] K.T. Huber, L. van Iersel, V. Moulton, G.E. Scholz, Is this network proper forest-based?, Inf. Process. Lett. 187 (2025) 106500.

[7] A.V. Aho, Y. Sagiv, T.G. Szymanski, J.D. Ullman, Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions, SIAM J. Comput. 10 (3) (1981) 405–421.

[8] D. Bryant, Building Trees, Hunting for Trees, and Comparing Trees, Ph.D. thesis, University of Canterbury, Christchurch, New Zealand, 1997.

[9] C. Semple, M. Steel, Phylogenetics, Oxford University Press, 2003.

[10] P. Gambette, K.T. Huber, On encodings of phylogenetic networks of bounded level, J. Math. Biol. 65 (1) (2012) 157–180.

[11] K.T. Huber, V. Moulton, Encoding and constructing 1-nested phylogenetic networks with trinets, Algorithmica 66 (3) (2013) 714–738–180.

[12] J. Oldman, T. Wu, L. van Iersel, V. Moulton, TriLoNet: piecing together small networks to reconstruct reticulate evolutionary histories, Mol. Biol. Evol. 33 (8) (2016) 2151–2162.

[13] L. van Iersel, V. Moulton, Trinets encode tree-child and level-2 phylogenetic networks, J. Math. Biol. 68 (2014) 1707–1729.

[14] D.E. Critchlow, D.K. Pearl, C. Qian, The triples distance for rooted bifurcating phylogenetic trees, Syst. Biol. 45 (3) (1996) 323–334.