Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

Measuring how far a nonbinary phylogenetic network is from being tree-based

(Dutch title: Meten hoe ver een niet-binair fylogenetisch netwerk verwijderd is van boom-gebaseerd zijn)

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

BACHELOR OF SCIENCE
in
APPLIED MATHEMATICS

by

FRANK JACOB ADRIAAN JANISSE

Delft, the Netherlands
January 2018

BSc thesis APPLIED MATHEMATICS

"Measuring how far a nonbinary phylogenetic network is from being tree-based"

(Dutch title: "Meten hoe ver een niet-binair fylogenetisch netwerk verwijderd is van boom-gebaseerd zijn")

FRANK JACOB ADRIAAN JANISSE

Delft University of Technology

Supervisor

Dr.ir. L.J.J. van Iersel

Other thesis committee members

Dr. B. van den Dries		Dr. J.L.A. Dubbeldam

January, 2018		Delft

Abstract

In computational biology, phylogenetic trees are used to describe evolutionary history. This can be done more generally using phylogenetic networks, which can also describe nontreelike events such as hybridization. Some phylogenetic networks can be obtained from a base tree, a rooted spanning tree with the same leaf set, by adding linking edges. Such networks are called tree-based. In recent articles, characterizations of binary tree-based networks are given. They are linked to maximum-sized matchings in bipartite graphs, path partitions and antichains. However, in many real-life applications, phylogenetic networks are not binary. Therefore, we will prove that some characterizations are extendable to all (nonbinary) phylogenetic networks while some others are not.

We will discuss five proximity measures of how close an arbitrarily (nonbinary) phylogenetic network is to being tree-based. Three of the measures turn out to be equal and at least three of them are computable in polynomial time. We show that this is also true in the nonbinary case. Lastly, we prove two inequalities comparing the other measures.

Contents

1 Introduction

In biology, evolutionary relationships have been essential for most studies for a long time. Evolutionary processes are responsible for the biodiversity of today. All kinds of organisms, not only animals, have developed and changed all over the time to adjust to current circumstances. Furthermore, hybrid species play an important role in the research into evolution. Visualizing for example the origin of a species can clarify how parental species and other ancestors are related to each other.

Phylogenetic (Greek: phylé=tribe, genetikós=genetics) trees can give a useful representation for evolutionary processes, but turn out to be limited. All the taxa together in the tree have one common ancestor, the root of the tree. A phylogenetic tree can not display hybrids, because hybrids have at least two parent species; a tree has got only branching nodes, with a single parental species and two or more direct descendants. Therefore, for mathematicians and biologists it is obvious to use phylogenetic networks, where for example hybridization, introgression and horizontal gene transfer can be represented.

Mathematically, a phylogenetic network is a rooted directed acyclic graph with a root without incoming edges, tree vertices with one incoming edge and more than one outgoing edge, reticulations with more than one incoming edge and one outgoing edge and leaves without ougoing edges. A reticulation represents for example hybridization and leaves represent currently living species. A phylogenetic tree is a phylogenetic network without reticulations. Some phylogenetic networks can be obtained from a phylogenetic tree by attaching edges between the edges of the tree, such that for every attached edge a reticulation is obtained. This is an informal definition of tree-based phylogenetic networks.

For biologists, it is important to know whether a phylogenetic network is tree-based or not. If a network is tree-based, the branches, representing vertical descent, are dominant in the evolutionary process; if a network is not tree-based, reticulation processes are more important. In Figure 1(a) an example of a phylogenetic network on four plant taxa is shown; this network is tree-based. The phylogenetic network in Figure 1(b) is not tree-based. But if a network is not tree-based, it is interesting for biologists how close a network is to being tree-based. They can still decide whether the reticulation processes in a network are dominant processes and consider a network differently.

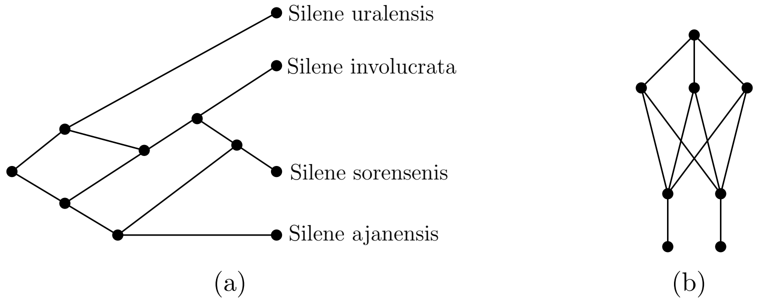


Figure 1: (a): A binary phylogenetic network on four plant taxa. (b): A nonbinary phylogenetic network that is not tree-based.

Recent articles characterize tree-based networks [3, 6]. In these articles, only binary tree-based networks are analyzed, just like Figure 1(a). In many evolutionary processes polytomies appear, where more than two branches descend from a single node. On the other hand, in many cases more than two edges can join in a reticulation. That is why we studied whether characterizations of tree-based networks still hold for all phylogenetic networks with polytomies. If this is true for some characterizations, we can extend the theorems stated in the articles. Furthermore, three proximity measures in the article of Francis, Semple and Steel [3] measure how close binary phylogenetic networks are to being tree-based. They give a polynomial-time algorithm to compute them, too, by using maximum-sized matching in bipartite graphs. We studied also whether the measures of deviation are extendable such that we can use them for nonbinary phylogenetic networks. In addition, we study two more proximity measures and we will compare the measures with each other in order to better understand their relationships.

We will principally research the possibility to extend characterizations of binary tree-based phylogenetic networks to all phylogenetic networks. First we will give the most important graph theoretical definitions. In Section 2, we research characterizations of tree-based networks, beginning in Subsection 2.1 where we use a bipartite graph obtained by considering reticulations and their parents that are tree vertices. In Subsection 2.2, we research other characterizations using vertex disjoint paths, path partitions and antichains. An antichain is a set of vertices in a phylogenetic network where for every pair of vertices in the antichain there exists no directed path from one to the other. Afterwards, we study a property related to Dilworth's theorem [2] by comparing the vertex set of a phylogenetic network and a partially ordered set. In the last part of Section 2, we discuss temporal phylogenetic networks that are for example used to research an evolutionary process during a period of time. A network is temporal if it is possible to assign a value which represents a moment in time to each node, such that every vertex represents a later moment in time than its parent, unless it is a reticulation, than the value remains the same.

In Section 3, we introduce five measures of deviation and extend known results to the nonbinary case. Furthermore, we study two measures that had not been studied before.

1.1 Preliminaries

First, we have to introduce some important concepts from graph theory used in this thesis. These concepts will help us to understand the characterizations of phylogenetic networks. A *network* is the same as a graph. V denotes the vertex set of a graph; E denotes the edge set. A graph $G = (V, E)$ is *bipartite* if V can be partitioned in two subsets such that every $e \in E$ has its ends in different subsets, so vertices in the same partition must not be adjacent. A *component* of a graph G is a maximal connected subgraph. A *directed graph* is a graph where the edges are directed from one vertex into another vertex. The notation of an undirected edge e (from vertex x to vertex y) is $e = \{x, y\}$ or $e = \{y, x\}$ and we notate a directed edge f as $f = (x, y)$. In this case is f the *outgoing* edge of x and the *incoming* edge of y. The number of incoming and outgoing edges of a vertex is called the *indegree* and *outdegree* of that vertex, respectively.

A *rooted* directed graph G is a directed graph where one vertex is distinguished as the *root* of G and has indegree 0 and outdegree 1 or more. The root is unique, so a rooted graph is connected. In a rooted directed tree, a *child* of a vertex u is the vertex that is connected to u by an outgoing edge of u. A *descendant* of a vertex $v \in V$ is a vertex that can be reached by a directed path out of v. A *parent* of a vertex w is the vertex of which w is a child. Parents and children are not unique. In this thesis all directed edges are directed into the lowest vertex and X indicates a finite non-empty set representing the studied objects (e.g. species).

Definition 1.1. *A* phylogenetic network $\mathcal{N} = (V, E)$ on X *is a rooted directed acyclic graph with the following properties:*

- *Vertices with indegree and outdegree 1 do not exist.*
- *All vertices except the root with outdegree 2 or more,* tree vertices, *have indegree 1.*
- *All vertices with indegree 2 or more,* reticulations, *have outdegree 1.*
- *The vertices with outdegree 0 have indegree 1, are called* leaves *and are labeled with an element of X.*
- *Every element of X is the label of exactly one leaf.*

In this paper we differentiate between binary and nonbinary phylogenetic networks. If none of these words is used, the definition is just as before. Binary networks have more restrictions: a *binary phylogenetic network on X* is a phylogenetic network on X where the indegree and outdegree of a vertex are at most 2; a *nonbinary phylogenetic network* has the same definition as a phylogenetic network. It is a not *necessarily* binary network where vertices can possibly have indegree or outdegree greater than 2. An example of a phylogenetic network is given in Figure 2 and Figure 3(b) is an example of a binary phylogenetic network.

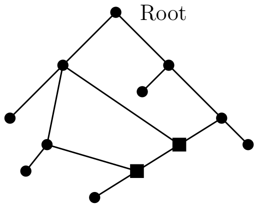


Figure 2: A phylogenetic network with five leaves and two reticulations. The reticulations are displayed by square nodes.

A *tree* is a connected graph with no cycles; a *spanning tree* of a graph G is a subgraph containing all vertices of G. A *phylogenetic tree* is a phylogenetic network without reticulations. All edges directed into a reticulation are called *reticulation edges*; all other edges are called *tree edges.*

Definition 1.2. *A phylogenetic network $\mathcal{N} = (V, E)$ is* tree-based *if $\mathcal{N}$ has a rooted spanning tree $T = (V, E')$ with the same leaf set as $\mathcal{N}$ and where $E' \subseteq E$. In this case we call T a* base tree *of $\mathcal{N}$.*

Figure 3(a) shows a tree-based phylogenetic network and (b) is a network that is not tree-based because both parents of vertex v are reticulations. Hence, one of these reticulations will be a leaf in any rooted spanning tree of $\mathcal{N}_2$, while it is not a leaf in the network. Therefore, no spanning tree can have the same leaf set as $\mathcal{N}_2$. When a rooted tree is a subgraph of a phylogenetic network $\mathcal{N}$ and has the same leafset X but it is not necessarily a spanning tree, than we call it a *display tree*. The root of a display tree does not have to be the same as the root of $\mathcal{N}$.

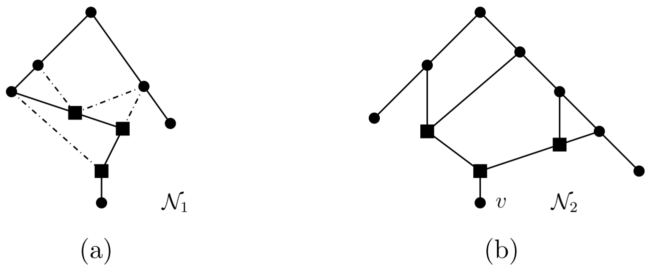


Figure 3: (a): A tree-based phylogenetic network $\mathcal{N}_1$ where the only possible base tree is indicated by the solid edges. (b): A binary non-tree-based phylogenetic network $\mathcal{N}_2$ that is binary because it contains no vertices with indegree or outdegree greater than 2.

In several proofs we make use of matchings in bipartite graphs. A *matching* in a graph $G = (V, E)$ is an edge set $M \subseteq E$ such that every pair of edges in M does not have a vertex in common. A vertex is *matched* if it is one of the ends of an edge in M. A *perfect matching* in G is a matching which matches all $v \in V$.

In some proofs in this paper we make use of subdivisions. A *subdivision* of a graph $G = (V, E)$ is a graph obtained from G by subdividing one or more edges in G. A graph is also a subdivision of itself. To *subdivide* an edge $e = \{u, v\} \in E$ means deleting e, adding an extra vertex x to V and connect x to the two endpoints, u and v, of e by two new edges $\{u, x\}$ and $\{x, v\}$. If G is directed, connect x to the begin and end vertices of e and orient the two new edges in the same direction as e.

Antichains have an important role in characterizing phylogenetic networks. An *antichain* in a directed graph $G = (V, E)$ is a subset of vertices $W \subseteq V$ where for all $u, v \in W$ and $u \neq v$ a directed path from u to v does not exist. For example, the leaf set X of a phylogenetic network is an antichain. Every tree-based phylogenetic network $\mathcal{N}$ satisfies an important property: the antichain-to-leaf property. The *antichain-to-leaf property* says that for every antichain of k vertices, there exist k vertex disjoint paths from the vertices in the antichain to (not necessarily all) leaves of $\mathcal{N}$. However, this does not mean that every network with this property is tree-based. In Figure 4, an interesting example from Francis, Semple and Steel [3] is given: it is a non-tree-based phylogenetic network that satisfies the antichain-to-leaf property.

2 Characterizing tree-based phylogenetic networks

2.1 Tree-basedness and bipartite graphs

In this subsection we will relate tree-basedness of phylogenetic networks to bipartite graphs that can be constructed out of the considered network. In these corresponding bipartite graphs we will be looking for matchings, determine the tree-basedness of the phylogenetic networks and distinguish between binary networks and the general case.

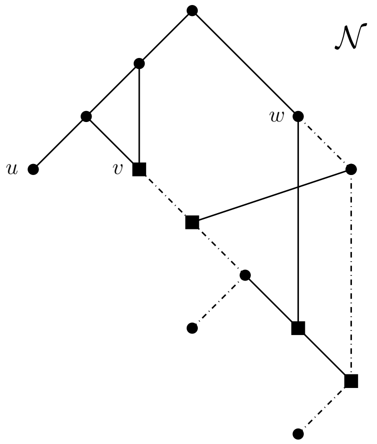


Figure 4: A phylogenetic network that is not tree-based, but it does satisfy the antichain-to-leaf property. An example of an antichain is $\{u, v, w\}$ and the three vertex disjoint paths to leaves are dash-dotted, where one path is trivial.

Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Zhang [6] defines a bipartite graph that will lead us to later used equivalences concerning tree-based networks. Let T be the set of tree vertices in $\mathcal{N}$ that are parents of reticulations and let R be the set of all reticulations in $\mathcal{N}$. Let $\mathcal{Z}_{\mathcal{N}}$ be the bipartite graph with vertex set $T \cup R$ and edge set

$$\{\{t, r\} : t \in T, r \in R \text{ and } (t, r) \text{ is an edge in } \mathcal{N}\}$$

An example is given in Figure 5. Note that $T \cap R = \emptyset$. Zhang proved the following equivalences for the binary case:

Theorem 2.1. [6] *Let $\mathcal{N} = (V, E)$ be a binary phylogenetic network. The following equivalences hold:*

(*i*) *$\mathcal{N}$ is tree-based.*

(*ii*) *The bipartite graph $\mathcal{Z}_{\mathcal{N}}$ has a matching such that each reticulation is matched.*

(*iii*) *The bipartite graph $\mathcal{Z}_{\mathcal{N}}$ has no maximal path that has reticulations as starting and ending vertices.*

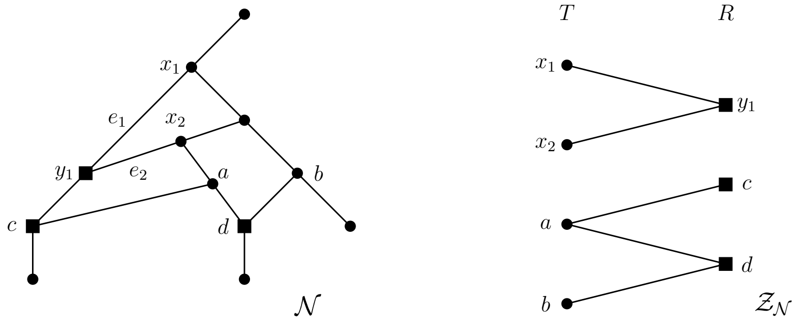
Proof. First we have got two facts:

1. Let $e = (x, y) \in E$. If x is a reticulation, then it has outdegree 1. x becomes a vertex with outdegree 0 in $\mathcal{N} - \{e\}$.

2. Let $e_1 = (x_1, y_1) \in E$ *and* $e_2 = (x_2, y_2) \in E$ such that $x_1, x_2 \in T$ and $y_1, y_2 \in R$. If $x_1 = x_2$, then x_1 becomes a leaf in $\mathcal{N} - \{e_1, e_2\}$ and if $y_1 = y_2$, than y_1 has indegree 0 in $\mathcal{N} - \{e_1, e_2\}$. For an example where $y_1 = y_2$, see Figure 5.

Every edge (t, r) where $t \in T$ and $r \in R$ corresponds an edge in $\mathcal{Z}_{\mathcal{N}}$. We define the set $\mathcal{E}(\mathcal{Z}_{\mathcal{N}})$ as the subset of edges in $\mathcal{Z}_{\mathcal{N}}$ that correspond one to one to the edges in the subset $\mathcal{E} \subseteq E \cap (T \times R)$ of edges in $\mathcal{N}$.

From the two facs above it turns out that removing the adjacent edges of a reticulation to obtain a tree with the same leaves as $\mathcal{N}$ is possible in only one way. Namely by deleting one of the two incoming reticulation edges. Indeed, from fact 1 follows that deleting the outgoing edge of a reticulation gives a different leaf set. According to fact two, we would obtain a different leaf set or a nonphylogenetic network because we would create another root. The two facts also imply that $\mathcal{N} - \mathcal{E}$ is a tree with the same leaf set as $\mathcal{N}$ if and only if $\mathcal{E}$ is a matching that covers all reticulations in $\mathcal{N}$ and therefore if and only if $\mathcal{E}(\mathcal{Z}_{\mathcal{N}})$ is a matching that covers R. So $\mathcal{N}$ is tree-based if and only if $\mathcal{Z}_{\mathcal{N}}$ has a matching that matches every reticulation. Thereby, we proved (i) $\Leftrightarrow$ (ii).



(a)

(b)

Figure 5: (a): A binary phylogenetic network $\mathcal{N}$. From fact 2 in the proof it follows that deleting edges e_1 and e_2 will make vertex y_1 a second root in $\mathcal{N}$. This is not allowed in a base tree. (b): The corresponding bipartite graph $\mathcal{Z}_{\mathcal{N}}$

$\mathcal{Z}_{\mathcal{N}}$ is bipartite, so we can use Hall's marriage theorem [4]. The theorem applied to $\mathcal{Z}_{\mathcal{N}}$ states that a matching that covers R exists if and only if $|R'| \leq |N(R')|$ for every $R' \subseteq R$, where $N(R')$ is the set of vertices in T adjacent to the vertices in R'. Moreover, there exists a matching that covers R if and only if for every component C in $\mathcal{Z}_{\mathcal{N}}$ there is a matching that covers $C \cap R$. We will make use of components to finish the proof.

A vertex $r_i \in V(\mathcal{Z}_{\mathcal{N}})$ corresponding a reticulation in $\mathcal{N}$ has degree 1 if it has as parents a tree vertex and a reticulation in $\mathcal{N}$; r_i has degree 2 when it has two tree vertices as parents in $\mathcal{N}$; a vertex t_i in $\mathcal{Z}_{\mathcal{N}}$ corresponding a tree vertex in $\mathcal{N}$ has degree 1 or 2 when it has one or two reticulations as children in $\mathcal{N}$, respectively. It follows that every component in $\mathcal{Z}_{\mathcal{N}}$ is either a cycle or a path.

Let C be a component in $\mathcal{Z}_{\mathcal{N}}$. If C is a nontrivial cycle, it has a perfect matching from $C \cap R$ to $C \cap T$ (and that matching covers $C \cup R$). If C is a nontrivial path, it contains exactly two vertices v_1 and v_2 with degree 1. There exists a matching covering whole $C \cap R$ if and only if v_1 or v_2 is not contained in R. In the case where v_1 and v_2 would be reticulations, $|C \cup T| > |C \cup R|$ and one of the vertices with degree 1 could not be matched. $\mathcal{Z}_{\mathcal{N}}$ could possibly have trivial paths: vertices with degree 0 corresponding reticulations with two reticulations as parents in $\mathcal{N}$. This vertex could not be matched and a network with one or more vertices like this is clearly not tree-based.

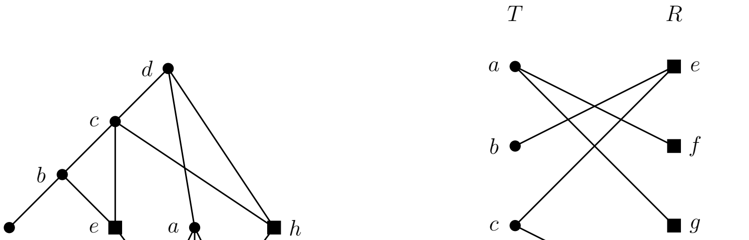
It follows that $\mathcal{Z}_{\mathcal{N}}$ has a matching matching every reticulation if and only if $\mathcal{Z}_{\mathcal{N}}$ has no maximal path that has reticulations as starting and ending vertices. The equivalence (ii) $\Leftrightarrow$ (iii) has now been proved and so the whole theorem. □

We wonder whether Zhang's theorem [6] still holds for all phylogenetic networks, so we will now take a look at the case where $\mathcal{N}$ is not necessarily binary. At first, $\mathcal{Z}_{\mathcal{N}}$

will be constructed in the same way. Second, we notice that it is not always possible to create a base tree by deleting just one incoming edge of a reticulation. For example, when a reticulation has more than two incoming edges we should delete more than one incoming edge to create a base tree. We can not refer to matchings in $\mathcal{Z}_{\mathcal{N}}$ anymore which gives a problem. Two counterexamples will show that Theorem 2.1 will not hold for all phylogenetic networks.

In Figure 6(a), a phylogenetic network is shown with three tree vertices that have outdegree 3. This network is not binary, but it is tree-based. The corresponding bipartite graph $\mathcal{Z}_{\mathcal{N}}$ is shown in Figure 6(b) that does not have a matching such that each reticulation is matched. This is because it is not possible to match both f and g. Furthermore, this graph has a maximal path that starts and ends with reticulations f and g, respectively. These two corresponding graphs have now shown that (i) $\not\Rightarrow$ (ii) and (i) $\not\Rightarrow$ (iii) for all phylogenetic networks.

In Figure 7(a), a phylogenetic network with two tree vertices having outdegree 3 is shown and two reticulations with indegree 3, so this network is not binary. This network is not tree-based, because it can be checked that any rooted spanning tree of $\mathcal{N}$ has a different leaf set. This leaf set must contain x_1, x_2 and x_3, but also one vertex of c, f and h. If no vertex of c, f and h is a leaf in a rooted spanning tree, than g or i has to be a reticulation, which contradicts the spanning tree being a tree. The corresponding bipartite graph $\mathcal{Z}_{\mathcal{N}}$ is shown in Figure 7(b) and has a matching M such that each reticulation is matched. Furthermore, it has no maximal path that has reticulations as starting and ending vertices. Therefore, these two corresponding graphs have shown that (ii) $\not\Rightarrow$ (i) and (iii) $\not\Rightarrow$ (i) for nonbinary phylogenetic networks. We can conclude from the two counterexamples that Zhang's theorem only holds for binary phylogenetic networks.



(a) (b)

Figure 6: (a): A tree-based phylogenetic network $\mathcal{N}$, showing that (i) $\not\Rightarrow$ (ii) and (i) $\not\Rightarrow$ (iii) in Zhang's theorem for nonbinary phylogenetic networks. (b): The corresponding bipartite graph $\mathcal{Z}_{\mathcal{N}}$.

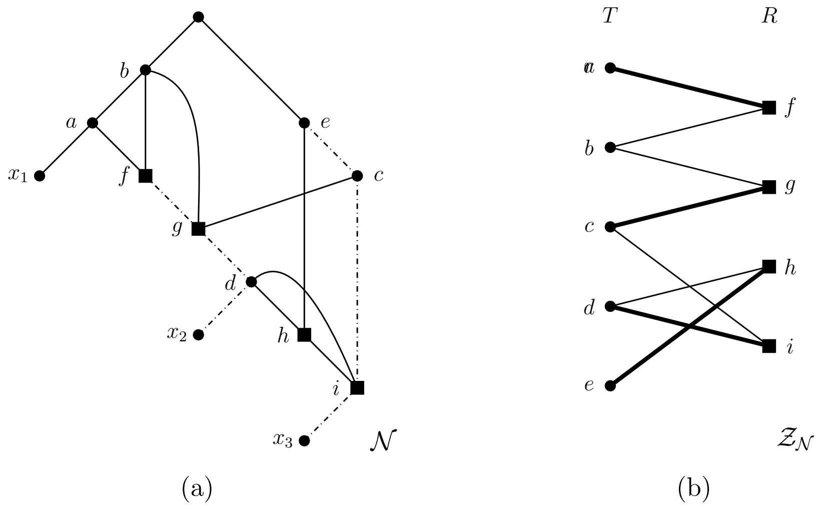


Figure 7: (a): A phylogenetic network $\mathcal{N}$ that is not tree-based, but satisfies the antichain-to-leaf property. An example of an antichain is $\{x_1, e, f\}$ and the three vertex disjoint paths from the antichain to the leaves are dash-dotted. For nonbinary phylogenetic networks, it shows also that (ii) $\not\Rightarrow$ (i) and (iii) $\not\Rightarrow$ (i) in Zhang's theorem. (b): The corresponding bipartite graph $\mathcal{Z}_\mathcal{N}$ with matching M that is displayed by bold edges.

2.2 Path partitions and antichains

In this section we consider four properties of a binary phylogenetic network that are equivalent to being tree-based. In the previous subsection we found out that Zhang's characterizations for tree-based networks do not hold for all phylogenetic networks. Some new characterizations for tree-based networks, based on path partitions and earlier defined antichains, are established in the article of Francis, Semple and Steel [3]. We will take a deeper look at the proofs and find out whether or not these characterizations will hold for all phylogenetic networks. In the following theorem, the first three characterizations of Francis, Semple and Steel are generalized.

Theorem 2.2. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. The following are equivalent:*

- (i) *$\mathcal{N}$ is tree-based.*
- (ii) *$\mathcal{N}$ has an antichain $\mathcal{A} \subseteq V$ and V can be partitioned into $|\mathcal{A}|$ chains where every chain forms a path in $\mathcal{N}$ ending at a leaf.*
- (iii) *For each subset $U \subseteq V$ there exists a set of vertex disjoint paths in $\mathcal{N}$ each of which ends at a leaf in X such that each element of U lies on exactly one path.*
- (iv) *V can be partitioned into a set of vertex disjoint paths, each of which ends at a leaf of $\mathcal{N}$.*

To prove this theorem we begin with a lemma. This lemma will help us to shorten the proofs of the different implications. We will prove the lemma for all phylogenetic networks; we will use this for the proof of Theorem 2.2. This lemma was proved by Francis, Semple and Steel [3] for the binary case. We will take a deeper look at this proof, after which we prove the lemma for the nonbinary case. Let T be a subdivision of a rooted directed binary tree with vertex set V_T. The following lemma has been used for the binary case.

Lemma 2.3. [3] *For every $U \subseteq V_T$ there exists a set of vertex disjoint directed paths in T each of which ends at a leaf of T and every vertex in U lies on exactly one of these paths.*

Proof. We make use of a proof by induction on the number of vertices $n = |V_T|$ of T. In the case where $n = 1$ the lemma holds because there is only one vertex that results in a trivial path that is a leaf at the same time.

If $n \geq 2$, suppose that the lemma holds for every T with $n - 1$ vertices (induction hypothesis). An example of a subdivision like this is given in Figure 8. Note that not every leaf has to be used and trivial paths can appear. We show that the lemma holds for every T with n vertices.

Let $U \subseteq V_T$ be an arbitrary subset. We distinguish two cases concerning the leaves of T.

(i) T has a leaf v of which the parent u has indegree and outdegree 1.

(ii) T has a vertex w which is a parent of the two leaves x and y.

Indeed, this subdivision is a binary rooted tree (and therefore without reticulations) with more than one vertex and therefore at least one of the cases will always hold in T. In Figure 8, both cases appear. First we make sure that the induction hypothesis holds in both cases.

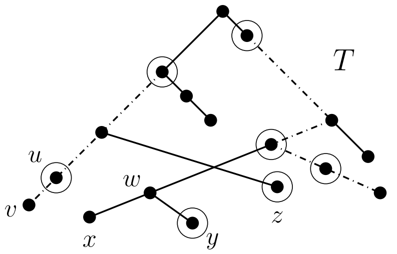


Figure 8: A subdivision T of a rooted directed binary tree. The set of the encircled vertices is an example of U and the dash-dotted edges form a set of vertex disjoint paths in T all ending at a leaf and every vertex in U lies on exactly one path. The vertices $y, z \in U$ are both trivial paths (ending at a leaf). The vertices u and v form an example of case (i) and w, x and y form an example of case (ii).

Case (i): Remove v and its adjacent edge from T. Name the resulting rooted binary tree $T' = (V'_T, E')$ where u is now a leaf of T'. Let

$$U' = \begin{cases} U, & v \notin U; \\ U - \{v\}, & u, v \in U; \\ (U - \{v\}) \cup \{u\}, & v \in U \wedge u \notin U. \end{cases}$$

Now v is not contained in $U' \subseteq V'_T$ and T' has $n - 1$ vertices. The induction hypothesis holds for T': there is a set of vertex disjoint paths in T' and one of each path ends at a leaf of T' and every vertex of U' lies on exactly one path. There exists a path in T' ending at u; in the case where $U' = U$ it is not necessary for a vertex from U' to lie on this path. We extend this path with an edge and a vertex x such that the path ends at x.

This gives a set of vertex disjoint paths in T, each of which ends at a leaf of T and every vertex in U still lies on one path. Now, the lemma holds for case (i).

Case (ii): Remove y and its adjacent edge from T. The rooted binary tree $T' = (V', E')$ has now been obtained. Let

$$U' = \begin{cases} U, & y \notin U; \\ U - \{y\}, & y \in U. \end{cases}$$

In that case, y will never be contained in $U' \subseteq V'_T$ and T' contains $n-1$ vertices. In the case where $U = U'$ the induction hypothesis holds. There might be a path in T ending at leaf y. This path can be adjusted to end at leaf x of T'. This set of paths works for T too.

For every path it must still be possible to end at a leaf. Therefore, when $U' = U - \{y\}$, there is a set of at most $|U| - 1$ vertex disjoint paths in T', all ending at a leaf of T' and every vertex in U' lies on exactly one path (induction hypothesis). When we add the trivial path, which consists only of the vertex y, to the set of paths, we obtain a set of paths where we were looking for. The paths are vertex disjoint in T and are all ending at a leaf of T and every vertex in U lies on exactly one path after all. Case (ii) has now also been proved. □

We will now extend Lemma 2.3 for all phylogenetic networks and prove it using a simplified proof, so we can use it to prove Theorem 2.2. Let T be a subdivision of a rooted directed tree with vertex set V_T.

Lemma 2.4. *For every $U \subseteq V$ there exists a set of vertex disjoint directed paths in T each of which ends at a leaf of T and every vertex in U lies on exactly one of these paths.*

Proof. Let X be the set of $|X| = n$ leaves of T. Let $P = \{p_1, p_2, \ldots, p_n\}$ be the set of n directed paths where every path $p_i, i = \{1, \ldots, n\}$ has starting vertex $x_i \in X$. For now, every path is a trivial path containing one vertex. Note, all n elements of X form an antichain of T. We will describe an algorithm for extending the paths to get the set of vertex disjoint paths we want to have. First, note that every path in P is a directed path. Furthermore, T is a subdivision of a tree, so there are no reticulations. When we extend a path $p_i \in P$ in the direction of the root, there are exactly two possibilities: the starting vertex has indegree 0 or indegree 1. Otherwise, the starting vertex would be a reticulation which contradicts T being a tree. The algorithm for extending one path is as follows:

For path $p_i \in P$ take the starting vertex s.

1. If the indegree of s is 0, stop extending the path. s is the root of T and the path is definite.
2. If the indegree of s is 1 and:
   (a) If the parent of s is not an element of a path $p_j, j = 1, \ldots, n$, extend the path by adding the parent of s to the path. This vertex becomes the new starting vertex of the path.
   (b) If the parent of s is contained in a path $p_j, j = 1, \ldots, n$, stop the extension, the path is definite.

Repeat the algorithm until p_i is definite. Repeat the algorithm for every path in P.

Now we have constructed $|X|$ vertex disjoint paths in T all ending at a leaf. Furthermore, these paths form a partition of V. Every $v \in V$ lies on one path, so for every $U \subseteq V$ there exists a set of vertex disjoint paths in T, all ending at a leaf of T and every vertex in U lies on exactly one path. □

Proof theorem 2.2. $(i) \Rightarrow (ii)$: Suppose $\mathcal{N}$ is tree-based and T is a base tree for $\mathcal{N}$. Take $U = V$. According to Lemma 2.4, there exists a set of vertex disjoint directed paths in $\mathcal{N}$ each of which ends at a leaf in X and every vertex in U lies on exactly one path. Take $\mathcal{A} = X$. The $|\mathcal{A}|$ vertex disjoint paths all end at a leaf in X. Furthermore, each of the vertex sets of these paths form a block of the partition Π of V into $|\mathcal{A}|$ chains.

$(ii) \Rightarrow (iii)$: Suppose that there exists a partition Π of V of which every block is the set of vertices of a path in $\mathcal{N}$ ending at a leaf. Let $U \subseteq V$ arbitrary. Π gives us a set of vertex disjoint paths each of which ends at a leaf of $\mathcal{N}$. Every element of U lies on exactly one path; this completes the implication.

$(iii) \Rightarrow (iv)$: Take $U = V$. There exists a set of vertex disjoint paths in $\mathcal{N}$ each of which ends at a leaf in X such that each element of V is on exactly one path. This means that the paths partition V. This is what we needed for property (iv).

$(iv) \Rightarrow (i)$: Let Π be a partition of V into a set of vertex disjoint paths, each of which ends at a leaf of $\mathcal{N}$. Every path is a block of Π and contains one leaf. Therefore, Π contains $|X|$ blocks. We can extend every vertex disjoint path of Π in the direction of the root $b \in V$ in the following way.

Let $|X| = n$ and let the path with starting vertex b be p_1. For every path $p_i \in \Pi$, $i = 2, \ldots, n$, let the starting vertex be s_i. For $i = 2, \ldots, n$, if s_i is a tree vertex, add the incoming edge of s_i to p_i and if s_i is a reticulation, add an arbitrarily incoming edge to p_i. Now we have obtained a new set of (nondisjoint) paths T which form a base tree of $\mathcal{N}$, so $\mathcal{N}$ is tree-based. □

The next property was already stated by Francis, Semple and Steel [3]. This is a property for tree-based binary phylogenetic networks. After the proof we will analyze an interesting example and investigate whether the theorem holds for all phylogenetic networks.

Theorem 2.5. [3] *Let $\mathcal{N}$ be a binary phylogenetic network. $\mathcal{N}$ is tree-based if and only if there does not exist a pair of subsets $U_1, U_2 \subseteq V$ such that*

1. *$|U_1| > |U_2|$, and*

2. *every path from a vertex in U_1 to a leaf of $\mathcal{N}$ passes a vertex in U_2, and*

3. *for $\{i, j\} = \{1, 2\}$, if there is a path from a vertex in U_i to a vertex in U_i, then this path passes a vertex in U_2.*

Proof. Let $\mathcal{N} = (V, E)$ be a tree-based binary phylogenetic network. We will use a proof by contradiction to show that if $\mathcal{N}$ is tree-based, then the property holds. First, suppose that the property is false. Then there exists a pair of subsets $U_1, U_2 \subseteq V$ satisfying all the three conditions. An example is given in Figure 9(a).

$\mathcal{N}$ satisfies property (iii) in Theorem 2.2. We take $U = U_1$ in property (iii) and show that a contradiction will appear. Observe a path P in $\mathcal{N}$ from a vertex in U_1 to a vertex in X. Because of condition 2 this path contains a vertex in U_2 and condition 3 ensures that if this path traverses a vertex in $U_i \cup U_j$ for $\{i,j\} = \{1,2\}$, then the elements in P will alternate between vertices in U_1 and U_2. So if P contains a second vertex of U_1, again a vertex of U_2 has to follow on the path. In Figure 9(b) is a path from vertex a to a leaf an example of an alternating path. Therefore, there are at least as many vertices of U_2 as U_1 in P. Property *(iii)* is true, so every element of U_1 is contained in exactly one path of the set of vertex disjoint paths each of which ends at a leaf of $\mathcal{N}$. However, it is not possible for the set of paths to contain all the vertices of U_2 because condition 1, $|U_1| > |U_2|$, is true when there are at least as many vertices of U_2 as U_1 in every path. This contradicts property *(iii)* in Theorem 2.2 for $U = U_1$ and the tree-basedness of $\mathcal{N}$. In Figure 9(a) it is also impossible for $U = U_1$ to have all its vertices lying on one path because U_1 is an antichain, but $\mathcal{N}_1$ has got only two leaves. Thus if $\mathcal{N}$ is tree-based, then there is no pair of subsets $U_1, U_2 \subseteq V$ satisfying conditions 1, 2 and 3.

Next, we will proof the conversion by using again a proof by contradiction. First, suppose that $\mathcal{N}$ is not tree-based. By Theorem 2.1, the bipartite graph $\mathcal{Z}_{\mathcal{N}}$ has a maximal path that starts and ends with a reticulation. We write this path as $(r_1 t_1 r_2 \cdots t_{k-1} r_k)$. The parents of r_1 and r_k are obviously not both tree vertices. Otherwise, the path would not be maximal. So the parents $q, q' \notin T$ of r_1 and r_k respectively are reticulations. To show that the property is wrong we make two subsets of $\mathcal{N}$: Let $U_1 = \{q, t_1, \cdots, t_{k-1}, q'\}$ and $U_2 = \{r_1, r_2, \cdots, r_k\}$. An example is given in Figure 9(b). The two subsets defined as here satisfy the following three conditions:

- $|U_1| > |U_2|$.
- Every path from a vertex in U_1 to a leaf of $\mathcal{N}$ traverses a vertex in U_2 because U_1 is the set of all parents of the vertices in U_2.
- For $\{i,j\} = \{1,2\}$, if there is a path from a vertex in U_i to a vertex in U_i, then this path traverses a vertex in U_j. This holds because U_1 is the set of all parents of all vertices in U_2 and U_2 is the set of all children of the vertices in U_1.

The vertex sets $U_1, U_2 \subseteq V$ form now a pair of subsets with the same conditions as in Theorem 2.5. So it contradicts with the nonexistence of a pair of subsets satisfying the three conditions. Furthermore, in Figure 9(b) we can see a path from vertex a to a leaf that alternates between vertices of U_1 and U_2. □

Another interesting example is given in Figure 10 where a non-tree-based phylogenetic network is given and two subsets, obtained from the bipartite graph $\mathcal{Z}_{\mathcal{N}}$, that satisfy the three conditions of Theorem 2.5. If we follow the proof of Theorem 2.5, we can make two sets U_1, U_2 out of $\mathcal{Z}_{\mathcal{N}}$ by adding the reticulations from the maximum path that starts and ends with reticulations to U_2; we add the tree vertices and the parents of the starting and ending vertices that are reticulations to U_1. This gives us $U_1 = \{r_2, t_3, t_4, r_4\}$ and $U_2 = \{r_3, r_4, r_5\}$.

A remarkable fact is that U_1 and U_2 are not disjoint. It is clear that $|U_1| > |U_2|$. Furthermore, every path from a vertex in U_1 to a leaf of $\mathcal{N}$ traverses a vertex in U_2, namely r_3 or r_5. We allow that $U_1 \cap U_2 \neq \emptyset$. U_1 is still the set of all parents of all the vertices in U_2; U_2 is still the set of all children of all the vertices in U_1. The third condition

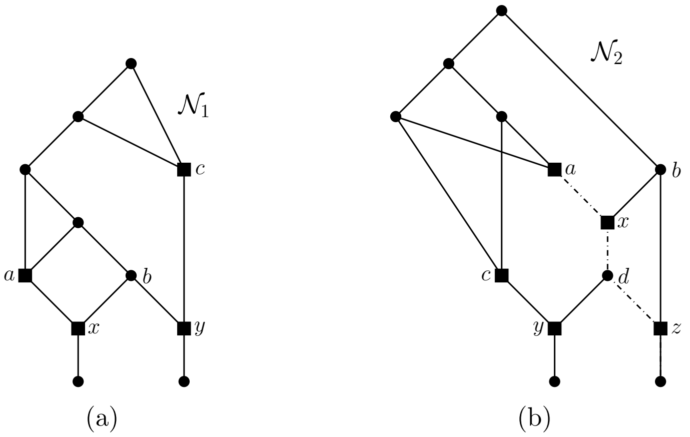


Figure 9: (a): A phylogenetic network $\mathcal{N}_1$ that is not tree-based with $U_1 = \{a, b, c\}$ and $U_2 = \{x, y\}$ satisfying all the three conditions of Theorem 2.5. (b): A phylogenetic network $\mathcal{N}_2$ that is not tree-based and where $U_1 = \{a, b, c, d\}$ and $U_2 = \{x, y, z\}$ satisfy the three conditions of Theorem 2.5 too. The alternating path is dash-dotted.

holds because we consider r_4 as a child and therefore as a vertex in U_2 in the paths (t_4, r_4) and (t_3, r_4). We consider r_4 as a parent and therefore as a vertex in U_1 in the path (r_4, r_5).

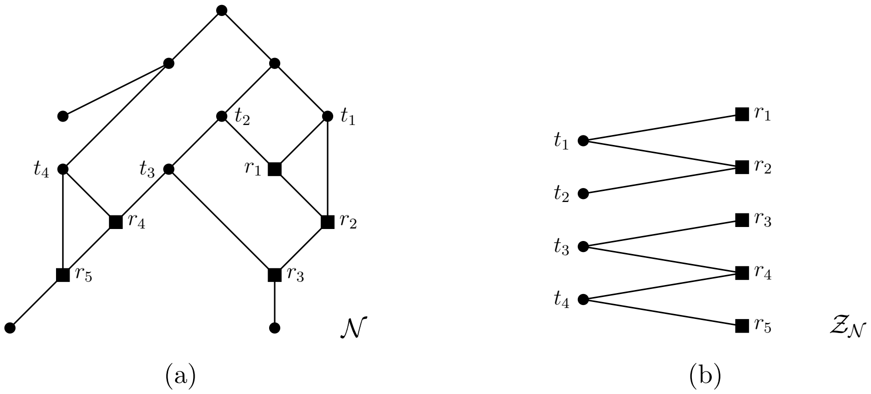


Figure 10: (a): A binary phylogenetic network $\mathcal{N}$ that is not tree based. It contains two subsets satisfying the three conditions of Theorem 2.5. (b): The bipartite graph $\mathcal{Z}_{\mathcal{N}}$ obtained by $\mathcal{N}$ with a maximum path starting and ending with reticulations.

Wondering whether Theorem 2.5 holds for all phylogenetic networks, we can fast see that one direction of the proof still holds for nonbinary tree-based phylogenetic networks. We can just follow the first part of the proof, because Theorem 2.2 holds for all phylogenetic networks. Now we have found out the following:

Theorem 2.6. *Let $\mathcal{N}$ be a phylogenetic network. If $\mathcal{N}$ is tree-based, then there does not exist a pair of subsets $U_1, U_2 \subseteq V$ such that*

1. *$|U_1| > |U_2|$, and*

2. *every path from a vertex in U_1 to a leaf of $\mathcal{N}$ passes a vertex in U_2, and*

3. *for $\{i, j\} = \{1, 2\}$, if there is a path from a vertex in U_i to a vertex in U_i, then this path passes a vertex in U_2.*

Proof. Identical to the first part of the proof of Theorem 2.5. □

Is a phylogenetic network $\mathcal{N} = (V, E)$ tree-based if a pair of subsets U_1, U_2 satisfying the three properties of Theorem 2.6 does not exist? We leave this question for the general case unanswered.

We can conclude that the first three properties for binary phylogenetic networks, established by Francis, Semple and Steel [3], still hold for all (nonbinary) phylogenetic networks, so these can be used for many more evolutionary histories. The extension was possible because we could extend the needed lemma. Lastly, the fourth property holds if a (nonbinary) phylogenetic network is tree-based, but we do not know whether the other direction is true.

2.3 Partially ordered sets and Dilworth's theorem

We can relate property *(ii)* in Theorem 2.2 to a version of Dilworth's theorem [2]. In this subsection we will describe how we can consider the vertex set of a phylogenetic network as a partially ordered set, how Dilworth's theorem can be applied to tree-based networks and explore the counterexample in Figure 7(a) in more detail.

A *partially ordered set* (or *poset*) is a set P and a binary relation $\leq$ such that for all $a, b, c \in P$

1. $a \leq a$ (reflexivity), and
2. $a \leq b$ and $b \leq a$ implies $a = b$ (anti-symmetry), and
3. $a \leq b$ and $b \leq c$ implies $a \leq c$ (transitivity).

We write $a < b$ if $a \leq b$ and $a \neq b$. A *chain* of length n is a sequence $a_1 < a_2 < \ldots < a_n$ and an *antichain* in a poset is a set $\mathcal{A} \subseteq P$ of which every pair of elements $a, b \in \mathcal{A}$ does not satisfy $a \leq b$ or $b \leq a$. Now let $\mathcal{N} = (V, E)$ be a phylogenetic network and let $\leq$ be the relation 'is a descendant of' where each vertex is also a descendant of itself. V is a poset and each vertex is a descendant of the root of $\mathcal{N}$. An antichain $\mathcal{A}$ in V is the same as an antichain defined as in Section 1: a subset of vertices where for every pair of vertices $u, v \in \mathcal{A}$ there exists no directed path from u to v.

A version of Dilworth's theorem [2] states that for a poset P there exists an antichain $\mathcal{A}$ and a partition Π of P into $|P|$ chains, such that $|\mathcal{A}| = |\Pi|$. We can consider V in Theorem 2.2 as a poset; property *(ii)* is related to Dilworth's theorem if we require that the $|\mathcal{A}|$ chains each must form one path. Otherwise, there can be chains consisting of more than one path.

We look back at Figure 7(a) where an example of a phylogenetic network $\mathcal{N} = (V, E)$ is given that is not tree-based, but it satisfies the antichain-to-leaf property. The maximum size of an antichain in $\mathcal{N}$ is three; an example is $\{x_1, f, e\}$. In Figure 7(a), three vertex disjoint paths from the antichain to the leaves of $\mathcal{N}$ give the antichain-to-leaf property: $(x_1), (f, g, d, x_2)$ and (e, c, i, x_3). However, property *(ii)* does not hold because $\mathcal{N}$ is not tree-based. According to Dilworth's theorem, the minimum number of chains that partition V in the example has to be three, because that is the size of a maximum antichain in $\mathcal{N}$. However, the minimum size of a partition into paths is four.

This is very remarkable. First, the supposed phylogenetic network is not tree-based, but is satisfies the antichain-to-leaf property. Second, it may first look like Dilworth's theorem does not hold for a found poset. However, there is a difference between a path in a network and a chain in a poset. In a phylogenetic network, where the vertex set V with the binary relation 'is a descendant of' is a poset, a chain of V can consist of $k \geq 1$ paths $\pi_1, \ldots, \pi_k$. For every path $\pi_i, i = 1, \ldots, k$, let the starting and ending vertices be a_i and b_i, respectively. A requirement for such a chain is that for every path $\pi_j, j = 1, \ldots, k-1$ the property $a_{j+1} \leq b_j$ holds. Thus a chain in a poset can consist of more paths and a path in a network is just individual and has to be connected.

However, the network we consider can be partitioned into three chains. We will construct four paths in $\mathcal{N}$, such that they form a partition of V and two of them will form a chain, satisfying the requirement in the previous paragraph. A copy $\mathcal{N}$ in Figure 7(a) is given in Figure 11, but now with a partition of V where the three chains are displayed by four dash-dotted paths. One chain (e, c, h, i, x_3) is formed by the two paths (e, c) and (h, i, x_3), where h is a descendant of c, so $x_3 < i < h < c < e$.

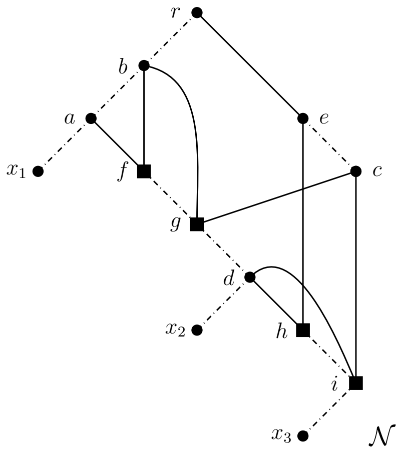


Figure 11: The same phylogenetic network as in Figure 7(a) that is not tree-based, but satisfies the antichain-to-leaf property. The paths that partition V are displayed by dash-dotted edges.

We can conclude that Dilworth's theorem holds for all phylogenetic networks where the vertex set can be chosen as poset where the relation is the ancestor-descendant relation. For tree-based networks it is clear by property (*ii*) in Theorem 2.2. For phylogenetic networks that are not tree-based, not every chain will be a path ending at a leaf or there are chains consisting of more than one path.

2.4 Temporal networks

In different kinds of research fields relations between objects (e.g. evolutionary relationships) can be found out. *Temporal networks* were introduced to describe evolutionary relationships changing over time. We assign a real number to every vertex in the network that stands for a point in time. The number is basically increasing as the distance between the root and the concerned vertex becomes bigger, except for reticulation edges, where it remains the same.

Formally, a phylogenetic network $\mathcal{N} = (V, E)$ is *temporal* if there is a map $\lambda : V \to \mathbb{R}$ such that $\lambda(u) < \lambda(v)$ for every tree edge (u, v); for every reticulation edge (u, v) holds $\lambda(u) = \lambda(v)$. We call λ a *temporal map* for $\mathcal{N}$.

First, we will take a look at the tree-basedness of binary temporal networks compared to the earlier discussed phylogenetic networks. The antichain-to-leaf property gives us an important difference. It turns out that the antichain-to-leaf property for a temporal network is a sufficient property to be tree-based. However, for phylogenetic networks in general it is not a sufficient property to be tree-based, like in the example in Figure 4.

Theorem 2.7. [3] *Let $\mathcal{N}$ be a binary temporal phylogenetic network. $\mathcal{N}$ is tree-based if and only if $\mathcal{N}$ satisfies the antichain-to-leaf property.*

Proof. Let $\mathcal{A}$ be an arbitrary antichain in $\mathcal{N}$. Take $U = \mathcal{A}$ in property (iii) in Theorem 2.2. From this property follows the antichain-to-leaf property, because for any antichain of k vertices, there exists a set of vertex disjoint paths in $\mathcal{N}$ each of which ends at a leaf such that each element of $\mathcal{A}$ is on exactly one path. $\mathcal{A}$ is an antichain, so the set has to have exactly k paths. Thus if $\mathcal{N}$ is tree-based, than $\mathcal{N}$ satisfies the antichain-to-leaf property.

We will prove the other direction using a proof by contradiction. Suppose $\mathcal{N}$ is temporal but not tree-based. By Theorem 2.1 it follows that the bipartite graph $\mathcal{Z}_{\mathcal{N}}$ contains a maximal path that starts and ends with a reticulation. We write this path as $(r_1\ t_1\ r_2\ \ldots t_{k-1}\ r_k)$.

First we take a look at the case where $k = 1$. Then the parents of r_1 are both not tree vertices and therefore have to be reticulations. The antichain $\mathcal{A}$ that consists of these two parents q_1 and q_2 violates the antichain-to-leaf property. Thus for $k = 1$, the case has lead us to a contradiction.

From now, suppose $k \geq 2$. q_1 is now the parent of r_1; q_1 is not a tree vertex and therefore must be a reticulation. q_2 is the parent of r_k that is not t_{k-1} so q_2 is a reticulation too. We define two new sets: $U = \{q_1, t_1, t_2, \ldots, t_{k-1}, q_2\}$ and $R = \{r_1, r_2, \ldots, r_k\}$. All outgoing edges of U are reticulation edges, because all the children of all the elements in U are reticulations. $\mathcal{N}$ is temporal, so there is a temporal map λ for $\mathcal{N}$ which gives

$$\lambda(q_1) = \lambda(r_1) = \lambda(t_1) = \lambda(r_2) = \ldots = \lambda(t_{k-1}) = \lambda(r_k) = \lambda(q_2). \tag{1}$$

Thus for all $x, y \in U : \lambda(x) = \lambda(y)$.

If U is an antichain, then the $k + 1$ paths from the vertices in U to the leaves in $\mathcal{N}$ con not be disjoint because all these paths have to contain one of the k reticulations of R whereby at least two paths contain the same reticulation. Thus the antichain-to-leaf property is not valid when U is an antichain.

From now on, we will consider the case where U is not an antichain. An element $u' \in U$ can be reached by a directed path p in $\mathcal{N}$ from $u \in U$. For all $x, y \in U : \lambda(x) = \lambda(y)$, so every edge in p must be a reticulation edge. There is only one vertex in p that is possibly a tree vertex, namely u, the first one. The other vertices in p are reticulations. Remember, a reticulation edge is by definition a vertex directed into a reticulation. $u' \in U$ is the last vertex in p and is not a tree vertex. So $u' = q_1$ or $u' = q_2$. The second vertex in p is a reticulation out of R because it is a child of $u \in U$. q_1 or q_2 can be reached by a directed path from a vertex in R. We name this vertex r_{q_1} or r_{q_2}, respectively.

The example in Figure 12(a) shows a path from u to u' and it shows that r_{q_1} and r_{q_2} can be equal. However, we found a way to let them never be equal. If q_1 and q_2 are both reachable by a directed path from the same reticulation $r \in R$, then

1. q_2 is also reachable by a directed path from r_1, or

2. q_1 is also reachable by a directed path from r_k.

When 1 occurs, take $r_{q_2} = r_1$; when 2 occurs, take $r_{q_1} = r_k$. These two possibilities always occur in this case because the only children of q_1 and q_2 are r_1 and r_k, respectively, and there is always one of these children that is between q_1 and q_2 on the path. Note that this is a path consisting of only reticulations.

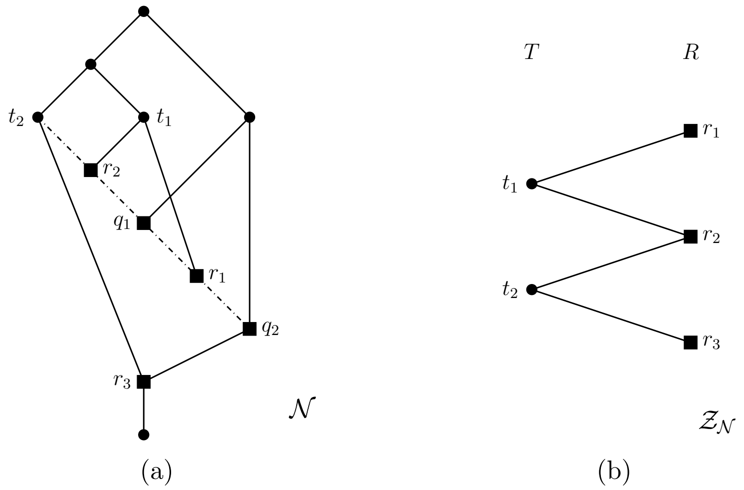


Figure 12: (a): This binary temporal phylogenetic network $\mathcal{N}$ that is not tree-based shows a dash-dotted directed path from $u = t_2$ to $u' = q_2$. $\mathcal{N}$ does not satisfy the antichain-to-leaf property. $U = \{q_1, t_1, t_2, q_2\}$, $r_{q_1} = r_2$ and $r_{q_2} = r_1$. (b): The corresponding bipartite graph $\mathcal{Z}_\mathcal{N}$ with a maximal path that has reticulations as begin and end vertices.

Next, we adjust U to get an antichain U' of k or $k-1$ vertices: Let

$$U' = \begin{cases} U - \{q_1\}, & \text{if } r_{q_1} \text{ exists;} \\ U - \{q_2\}, & \text{if } r_{q_2} \text{ exists;} \\ U - \{q_1, q_2\}, & \text{if } r_{q_1} \text{ and } r_{q_2} \text{ exist.} \end{cases}$$

If $|U'| = k$, r_{q_1} or r_{q_2} exists; any of the k paths from a vertex in U' to a leaf of $\mathcal{N}$ must traverse one of the k vertices in R. To be vertex disjoint, no path is allowed to traverse a vertex in R again. However, one of these traversed vertices is r_{q_1} or r_{q_2}. The path containing one of these is a path as described above that contains only reticulations and must traverse q_1 or q_2, respectively. The only children of q_1 and q_2 are r_1 and r_k, respectively. Now there exists a path from an element of the antichain U' to a leaf of $\mathcal{N}$ containing two elements of R. The paths can not be vertex disjoint anymore which violates the antichain-to-leaf property.

Lastly, we contradict the antichain-to-leaf property when $|U'| = k-1$. In the example in Figure 12 is $U' = \{t_1, t_2\}$ and $|U'| = 2$. In general, r_{q_1} and r_{q_2} both exist, so the path or paths traversing r_{q_1} or r_{q_2} or both will traverse a second or third element of R. In the example, the path from t_1 to the leaf of $\mathcal{N}$ traverses $r_1 = r_{q_2}$ and r_3, as the path from t_2 does. The paths can not be vertex disjoint anymore which violates the antichain-to-leaf property in the same way as when $|U'| = k$. □

The property of being temporal has clearly important consequences for binary phylogenetic networks to be tree-based. But does this hold for all phylogenetic networks? It is clear that if a temporal phylogenetic network $\mathcal{N}$ is tree-based, then $\mathcal{N}$ satisfies the antichain-to-leaf property, because Theorem 2.2 holds for all phylogenetic networks. For the proof in the other direction, we can not use Zhang's theorem [6], because this theorem does only hold for binary phylogenetic networks. However, we still feel that this direction could hold, as we have not been able to find a counterexample.

Conjecture 2.8. *Let $\mathcal{N}$ be a temporal phylogenetic network. $\mathcal{N}$ is tree-based if and only if $\mathcal{N}$ satisfies the antichain-to-leaf property.*

3 Proximity measures for being tree-based

Besides the properties of tree-based networks we have some measures to see how close to being tree-based a network is if it is not tree-based. We will research in total five different proximity measures. We will relate one to a matching in a different bipartite graph than $\mathcal{Z}_{\mathcal{N}}$ in Section 2. Furthermore, we will show that three of the measures, that are natural numbers, are always equal and we will compare two others.

First, remember how we can subdivide an edge of a directed graph. We use this to 'attach a new leaf' to a network. Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Let $(u, v) \in E$; we subdivide this edge, name the new vertex w and we add the edge (w, x) to E where x is the new *attached leaf* of $\mathcal{N}$.

The five proximity measures we consider are as follows:

(i) The minimum number $l(\mathcal{N})$ of leaves in $V \setminus X$ that must be present as leaves in a rooted spanning tree of $\mathcal{N}$. These leaves are called *dummy leaves.*

(ii) $p(\mathcal{N}) = d(\mathcal{N}) - |X|$, where $d(\mathcal{N})$ is the minimum number of vertex disjoint paths that partition the vertices of $\mathcal{N}$.

(iii) The minimum number $t(\mathcal{N})$ of leaves that need to be attached to $\mathcal{N}$ so the resulting network is tree-based.

(iv) The minimum number $a(\mathcal{N})$ of vertices in $\mathcal{N}$ that are required to be absent from any display tree of $\mathcal{N}$.

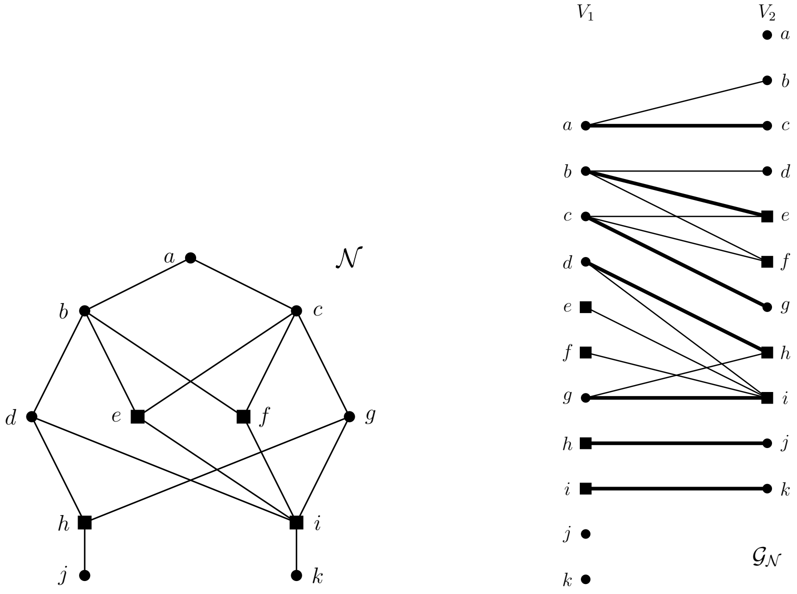
(v) The minimum number $b(\mathcal{N})$ of display trees of $\mathcal{N}$ such that every vertex of V is present in at least one of the trees.

Each of these measures is unambiguous and non-negative. If $\mathcal{N}$ is tree-based, then $l(\mathcal{N}) = p(\mathcal{N}) = t(\mathcal{N}) = a(\mathcal{N}) = 0$ and $b(\mathcal{N}) = 1$. For $l(\mathcal{N})$, there is in that case a base tree and thus there are no dummy leaves; $p(\mathcal{N})$ relies on (iv) in Theorem 2.2; for $t(\mathcal{N})$ it is clear; $a(\mathcal{N}) = 0$ because a display tree can be a base tree, such as the only display tree that is needed to obtain the value of $b(\mathcal{N})$.

3.1 Matchings in bipartite graphs

One proximity measure will be used to prove an equivalence that gives us a fifth property of tree-based networks. First, we define a new bipartite graph $\mathcal{G}_{\mathcal{N}}$ that can be obtained for every phylogenetic network. In $\mathcal{G}_{\mathcal{N}}$ we will make use of matchings to prove the equivalence. Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X and let V_1 and V_2 be copies of V. The bipartite graph $\mathcal{G}_{\mathcal{N}}$ has vertex set $V_1 \cup V_2$. An edge e' in $\mathcal{G}_{\mathcal{N}}$ connects $u \in V_1$ with $v \in V_2$ if and only if $e = (u, v)$ is an edge in $\mathcal{N}$. An example of a phylogenetic network with its corresponding bipartite graph $\mathcal{G}_{\mathcal{N}}$ is given in Figure 13.

We want to have a maximum-sized matching of $\mathcal{G}_{\mathcal{N}}$. Let $u(\mathcal{G}_{\mathcal{N}})$ be the number of unmatched vertices in V_1 when a maximum-sized matching has been created. The vertices that will never be matched in V_1 are the elements of X, because these have no outgoing edges in $\mathcal{N}$ and thus are isolated vertices in $\mathcal{G}_{\mathcal{N}}$. A maximum-sized matching is displayed by bold edges in Figure 13(b) where $u(\mathcal{G}_{\mathcal{N}}) = 4$; the vertices j and k are unmatched because they are leaves in $\mathcal{N}$ in Figure 13(a). The following lemma will help us to prove the theorem that gives the fifth property of tree-based phylogenetic networks. Both were proved for binary networks by Francis, Semple and Steel [3]. We prove them for general networks.



(a) (b)

Figure 13: (a): A phylogenetic network that is not tree-based. (b): The bipartite graph $\mathcal{G}_{\mathcal{N}}$ obtained by $\mathcal{N}$ with a maximum-sized matching M in bold edges.

Lemma 3.1. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Then*

$$p(\mathcal{N}) = u(\mathcal{G}_{\mathcal{N}}) - |X|. \tag{2}$$

Proof. First we will show that $p(\mathcal{N}) \leq u(\mathcal{G}_{\mathcal{N}}) - |X|$ by constructing a collection of directed paths in $\mathcal{N}$ that are vertex disjoint and form a partition of V. Let M be a maximum matching of $\mathcal{G}_{\mathcal{N}}$. Every path is constructed in the same way, beginning with a vertex from U_2, the set of unmatched vertices in V_2. So for each vertex $u \in U_2$, set $u = u_0$ and set the path $P_u = (u_0)$. We extend each path in the following way.

- 1. If u_0 is unmatched in V_1, the extension stops and set $P_u = (u_0)$.
  2. If u_0 is matched in V_1, extend P_u and the path becomes $P_u = (u_0 u_1)$ where $\{u_0, u_1\} \in M$.
- 1. If u_1 is unmatched in V_1, the extension stops and set $P_u = (u_0 u_1)$.
  2. If u_1 is matched in V_1, extend P_u and the path becomes $P_u = (u_0 u_1 u_2)$ where $\{u_1, u_2\} \in M$.

This construction of each path P_u terminates with a last vertex u_k, because the network is acyclic. Indeed, u_k is the first vertex in P_u that is unmatched in V_1. Note that the paths we have constructed are directed paths in $\mathcal{N}$. They form a collection $\mathcal{P} = \{P_u : u \in U_2\}$.

The paths are vertex disjoint because M is a matching and therefore our construction makes sure that every used vertex can not be contained in a second path. In Figure 13 is $|\mathcal{P}| = 4$ so $p(\mathcal{N}) = 2$ and the paths are $P_a = (a, c, g, i, k)$, $P_b = (b, e)$, $P_d = (d, h, j)$ and $P_f = (f)$.

Furthermore, the paths partition V, because every unmatched vertex in V_2 is the starting vertex of a path and every matched vertex in V_2 can be referred to a first (unmatched) vertex in V_2 by reversing the above construction. Therefore every vertex in V_2 is contained in a path P_u and $\mathcal{P}$ is a partition of V.

No vertex in X has outgoing edges and therefore these vertices are unmatched in V_1. The number of unmatched vertices in V_1 is $u(\mathcal{G}_\mathcal{N})$ which is equal to the number of unmatched vertices in V_2, so $|\mathcal{P}| = u(\mathcal{G}_\mathcal{N})$. Indeed, M is a matching. If we choose M of maximum size, we have got the result

$$p(\mathcal{N}) \leq |\mathcal{P}| - |X| = u(\mathcal{G}_\mathcal{N}) - |X|. \tag{3}$$

In this second part of the proof we will show that $p(\mathcal{N}) \geq u(\mathcal{G}_\mathcal{N}) - |X|$. Let $\mathcal{P}$ be an arbitrary collection of vertex disjoint paths that is a partition of V. M is the matching of $\mathcal{G}_\mathcal{N}$ obtained by $\mathcal{P}$ with the property that $\{u, v\} \in M$ if and only if u and v are consecutive vertices in some path in $\mathcal{P}$. We get a matching because $\mathcal{P}$ is a partition. The number of paths in $\mathcal{P}$ equals the number of unmatched vertices in V_1. Indeed, each unmatched vertex in V_1 is the last vertex of a path. By choosing $\mathcal{P}$ of minimum size we got the result we wanted:

$$p(\mathcal{N}) = d(\mathcal{N}) - |X| = |\mathcal{P}| - |X| \geq u(\mathcal{G}_\mathcal{N}) - |X|. \tag{4}$$

□

Theorem 3.2. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. $\mathcal{N}$ is tree-based if and only if $\mathcal{G}_\mathcal{N}$ has a matching of size $|V| - |X|$.*

Proof. $\mathcal{G}_\mathcal{N}$ has a maximum-sized matching of size $|V| - |X|$ if and only if a matching in $\mathcal{G}_\mathcal{N}$ of size $|V| - |X|$ does exist. A matching in $\mathcal{G}_\mathcal{N}$ of that size exists if and only if $u(\mathcal{G}_\mathcal{N}) = |X|$. Indeed, vertices in X are always unmatched in V_1 because of the outdegree. In that case $u(\mathcal{G}_\mathcal{N}) - |X| = 0$ and by Lemma 3.1 $p(\mathcal{N}) = 0$. $p(\mathcal{N}) = 0$ if and only if $\mathcal{N}$ is tree-based. Therefore $\mathcal{N}$ is tree-based if and only if $\mathcal{G}_\mathcal{N}$ has a matching of size $|V| = |X|$. □

3.2 Comparing measures of deviation

In this subsection we will take a further look at the five measures and compare them, beginning with the equality of the first three measures. This was proved by Francis, Semple and Steel [3] for binary networks. We generalized this to nonbinary networks. Our result is the following theorem.

Theorem 3.3. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Then*

$$l(\mathcal{N}) = p(\mathcal{N}) = t(\mathcal{N}). \tag{5}$$

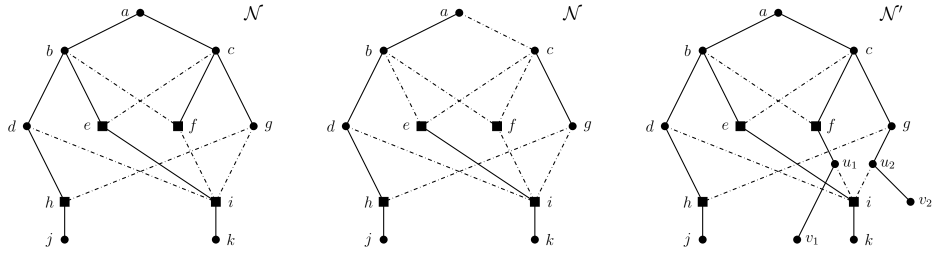
Proof. First, we show that $l(\mathcal{N}) \leq p(\mathcal{N})$. To construct a rooted spanning tree (to obtain $l(\mathcal{N})$) we consider $d(\mathcal{N})$ vertex disjoint paths of $\mathcal{N}$ such that these form a partition Π of

V. Due to the minimality of $d(\mathcal{N})$, $|X|$ of the paths are ending at a leaf of $\mathcal{N}$; $p(\mathcal{N})$ of these are not. Every $v \in V$ is contained in a path $P \in \Pi$. An example of these paths is given in Figure 14(b). We construct a rooted spanning tree T by adding one edge directed into the starting vertex of each path except the path that contains the root. The leaves of T not in X are the last vertices of the $|p(\mathcal{N})|$ paths not ending at a leaf of $\mathcal{N}$. We can link every leaf of T not in X to one of these $|p(\mathcal{N})|$ paths, so $l(\mathcal{N}) \leq p(\mathcal{N})$. For example, in Figure 14(b) we can do this by adding (b, e), (a, c) and (c, f) to the set of paths to obtain a rooted spanning tree.

Next, we will prove that $p(\mathcal{N}) \leq t(\mathcal{N})$. First we will obtain a tree-based network $\mathcal{N}'$ out of $\mathcal{N}$ by attaching $t(\mathcal{N})$ leaves. In Figure 14(c) two leaves have been attached to get $\mathcal{N}'$ and a base tree is displayed. To apply Lemma 2.4, we need a rooted tree, but all the leaves of $\mathcal{N}'$ too, so let T be the base tree of $\mathcal{N}'$. We will now apply Lemma 2.4 to the same tree T. Indeed, T is a subdivision of itself. For U in the lemma, take the leaf set of T. This set includes also the $t(\mathcal{N})$ attached leaves. The lemma implies that there exists a set of vertex disjoint paths each of which ends at a leaf of T and each vertex in U lies on exactly one path. The maximum number of paths is $|U|$ because there can not be a path not ending at a leaf. $|U| = t(\mathcal{N}) + |X|$ is a maximum and we got $d(\mathcal{N})$ paths that partition V. $d(\mathcal{N}) \leq t(\mathcal{N}) + |X|$, so $p(\mathcal{N}) \leq t(\mathcal{N})$.

To prove the equality in the theorem, the last inequality we have to prove is $t(\mathcal{N}) \leq l(\mathcal{N})$. First, we observe a rooted spanning tree and by attaching leaves to $\mathcal{N}$ and extending the spanning tree it will become a base tree for the extended network.

Let T be a rooted spanning tree of $\mathcal{N}$. If T has the same leaf set as $\mathcal{N}$ it is a base tree and it realises $l(\mathcal{N})$. Let the leaf set of k leaves in $V \setminus X$ be $L = \{l_1, l_2, \ldots, l_k\}$. An example of L, the set of dummy leaves, is $\{f, g\}$ in Figure 14(a). We attach one leaf v_i to an outgoing egde of a leaf $l_i \in L$ by subdividing that edge. In the case where l_i has more outgoing edges, choose one out of them to subdivide. Let u_i be the extra vertex we get by subdividing the chosen outgoing edge. We call the obtained network $\mathcal{N}'$ that is still phylogenetic. We want $\mathcal{N}'$ to be tree-based. It is, because we can extend T with $\{u_1, v_1, u_2, v_2, \ldots, u_k, v_k\}$ and its adjacent edges. Furthermore, the obtained tree is a rooted spanning tree of $\mathcal{N}'$ with the same leaf set as $\mathcal{N}'$. $t(\mathcal{N})$ is defined as a minimum and we've got a realization of these leaves that need to be attached to $\mathcal{N}$ to get a tree-based resulting network, so $t(\mathcal{N}) \leq l(\mathcal{N})$. □



(a) (b) (c)

Figure 14: (a): A rooted spanning tree of $\mathcal{N}$ is displayed by solid edges. (b): Four vertex disjoint paths in $\mathcal{N}$ displayed by solid edges. One of them is a trivial path: the vertex f. (c): $\mathcal{N}'$ obtained by attaching two leaves v_1, v_2 to $\mathcal{N}$ and a base tree displayed by solid edges.

The measures $a(\mathcal{N})$ and $b(\mathcal{N})$ were suggested by Francis, Semple and Steel [3], but have not been studied before. In the last two theorems we compare $b(\mathcal{N})$ to the first three measures and compare $a(\mathcal{N})$ and $b(\mathcal{N})$ with each other. These theorems are completely new.

Theorem 3.4. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Then*

$$b(\mathcal{N}) \leq p(\mathcal{N}) + 1. \tag{6}$$

Proof. First note that if $\mathcal{N}$ is tree-based, $b(\mathcal{N}) = p(\mathcal{N}) + 1 = 0$. Let $\mathcal{G}_{\mathcal{N}}$ be the corresponding bipartite graph defined as before. Let M be a maximum-sized matching in $\mathcal{G}_{\mathcal{N}}$. We construct $\mathcal{P} = \{P_u : u \in U_2\}$ in exactly the same way as in the proof of Lemma 3.1; $|\mathcal{P}|$, the number of vertex disjoint paths that partition V, is now of minimum size. In the example in Figure 15(a), $|\mathcal{P}| = 4$ and two of these paths end at a leaf. The paths have been obtained by an arbitrary maximum-sized matching of $\mathcal{G}_{\mathcal{N}}$ in Figure 15(c).

Let $|X| = n$ and let $\pi_1, \ldots, \pi_n$ be the paths in $\mathcal{P}$ ending at a leaf. For each $\pi_i, i = 1, \ldots, n$, extend the path by adding an (arbitrarily) incoming edge to the starting vertex of π_i. Repeat this until a display tree T_1 has been obtained. This can be done because there exists always a path from the root to the starting vertex of π_i. In Figure 15(a), T_1 can be obtained by adding $(a, c), (c, f)$ and (f, j) to the two paths ending at x_1 and x_2.

The $p(\mathcal{N})$ paths in $\mathcal{P}$ are not ending at a leaf of $\mathcal{N}$. Let $p(\mathcal{N}) = k$ and let these paths be $\phi_1, \ldots, \phi_k$. For each $\phi_i, i = 1, \ldots, k$, extend the path by adding arbitrarily outgoing edges to the ending vertex of ϕ_i until it ends at a leaf x_i. Remove the path π_i that ends at x_i too from $\mathcal{P}$ and remove $\{\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_k\}$ from $\mathcal{P}$. Now, extend $\{\pi_1, \ldots, \pi_{i-1}, \phi_i, \pi_{i+1}, \ldots, \pi_n\}$ by adding arbitrarily incoming edges to the starting vertices of each path until a new display tree has been formed. In Figure 15(a), add $(g, k), (k, o)$ and (o, x_1) to the path ending at g; remove the path from f to x_1 from $\mathcal{P}$; add (f, j) and (c, f) to the path ending at x_2; a new display tree has been formed that contains three other vertices than T_1.

In a phylogenetic network, there exists always a path from the root of $\mathcal{N}$ to the starting vertex of $\pi_i, i = 1, \ldots, n$ and $\phi_j, j = 1, \ldots, k$ and there exists always a path from the ending vertex of a path ϕ_j. Therefore, for each path $\phi_j, j = 1, \ldots, k$, we can always construct one extra display tree by this algorithm and the minimum is also one. $b(\mathcal{N})$ is defined as minimal. Now $b(\mathcal{N}) \not> p(\mathcal{N}) + 1$ and so $b(\mathcal{N}) \leq p(\mathcal{N}) + 1$. □

From Theorem 3.3 and 3.4 follows that $b(\mathcal{N}) \leq l(\mathcal{N}) + 1$ and $b(\mathcal{N}) \leq t(\mathcal{N}) + 1$. Intuitively, we can construct a rooted spanning tree out of $\mathcal{P}$ by adding incoming edges to the paths that do not start with the root of $\mathcal{N}$. The number of dummy leaves $l(\mathcal{N})$ is minimal and equals k in the proof. Attaching a new leaf to an edge directed out of a dummy leaf will lead to $t(\mathcal{N}) = l(\mathcal{N})$. We can find $l(\mathcal{N})$ and $t(\mathcal{N})$ after constructing $\mathcal{P}$ out of $\mathcal{G}_{\mathcal{N}}$. This can be done as described in the proof of Theorem 3.4. It follows that the first three measures can be computed in polynomial time. The explicit algorithms to compute these will not be given in this thesis.

Comparing the fourth and fifth measures, $a(\mathcal{N})$ and $b(\mathcal{N})$, we found out that $b(\mathcal{N}) \leq a(\mathcal{N}) + 1$ for all phylogenetic networks. If $\mathcal{N}$ is tree-based, then $a(\mathcal{N}) + 1 = b(\mathcal{N}) = 1$; for networks far away from being tree-based, $a(\mathcal{N})$ can be much greater than $b(\mathcal{N})$. The idea of the proof is that one extra display tree can ensure that more than one absent vertex from the display tree with the most vertices will be contained in it.

Theorem 3.5. *Let $\mathcal{N} = (V, E)$ be a phylogenetic network on X. Then*

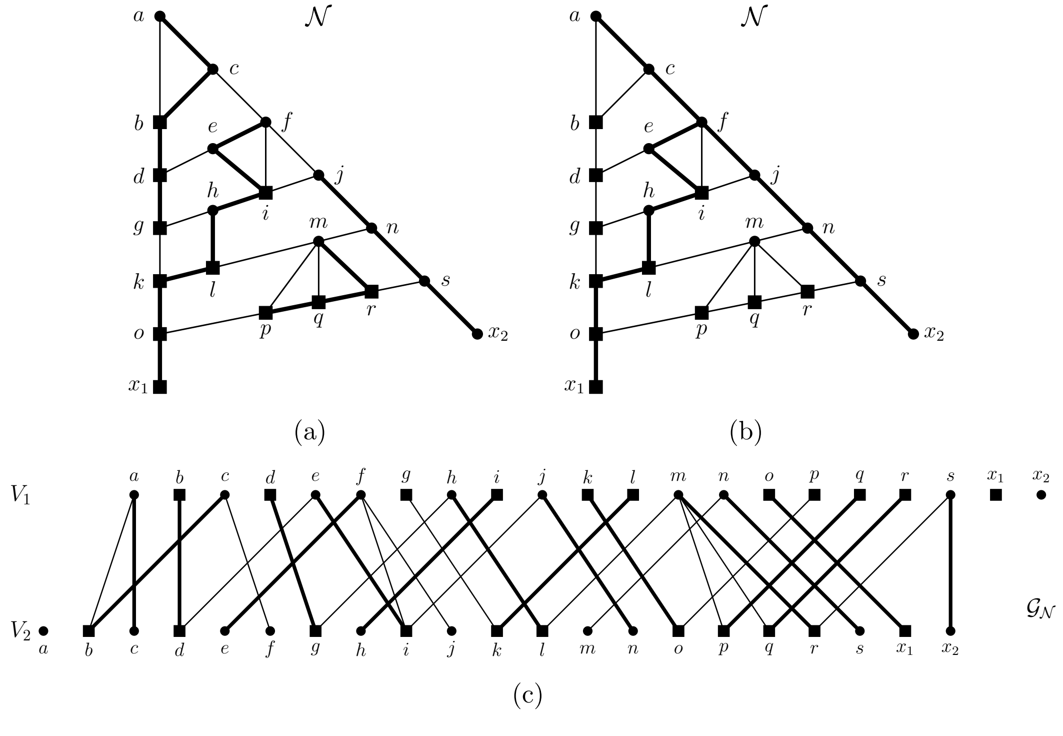
$$b(\mathcal{N}) \leq a(\mathcal{N}) + 1. \tag{7}$$

Proof. If $\mathcal{N}$ is tree-based, then $a(\mathcal{N}) + 1 = b(\mathcal{N}) = 1$ and the inequality holds. If $\mathcal{N}$ is not tree-based, then $\mathcal{N}$ has at least one reticulation. Let $T_1 = (V_1, E_1)$ be a display tree of $\mathcal{N}$ covering a maximum number of vertices. Since $a(\mathcal{N})$ is minimal, $|V \setminus V_1| = a(\mathcal{N})$. In Figure 15(b), there is one possible maximal display tree with $a(\mathcal{N}) = 7$.

Delete $|X| - 1$ edges from T_1 to obtain $|X| = n$ vertex disjoint paths each of which ends at a leaf of $\mathcal{N}$. Let this set of paths be $P = \{\pi_1, \ldots, \pi_n\}$. Let $Q = \phi_1, \ldots, \phi_k$ be the set of vertex disjoint paths not ending at a leaf of $\mathcal{N}$ such that k is minimal and $\mathcal{P} = P \cup Q$ forms a partition of V. In Figure 15(b), (f, j) can be deleted; $\pi_1 = (a, c, f, e, i, h, l, k, o, x_1)$; $\pi_2 = (j, n, s, x_2)$; $\phi_1 = (b, d, g)$; $\phi_2 = (m, r, p, q)$.

Now we will follow the algorithm in the proof of Theorem 3.4. So we will construct extra display trees by extending the k paths in Q such that they end at a leaf and there is a path from the root of $\mathcal{N}$ to the starting vertex of the path. For each path $\phi_j, j = 1, \ldots, k$, we can always construct one extra display tree by this algorithm, too.

Since $\mathcal{N}$ is not tree-based, T_1 is maximal and $\mathcal{P}$ is a partition and every required absent vertex from T_1 must be contained in one path in Q. Since $b(\mathcal{N})$ is defined to be minimal, by the algorithm, for every path in Q we need at most one extra display tree and it is always possible to construct one. Therefore $b(\mathcal{N}) \leq a(\mathcal{N}) + 1$. □



(a)

(b)

(c)

Figure 15: (a): A phylogenetic network $\mathcal{N}$ that is not tree-based. A path partition $\mathcal{P}$ based on $\mathcal{G}_{\mathcal{N}}$ in (c) is displayed in bold edges. (b): A maximum display tree V_1 is displayed in bold edges. The absent vertices are a, b, c, d, f, g and h, which is the minimum number, so $a(\mathcal{N}) = 7$. (c): The bipartite graph $\mathcal{G}_{\mathcal{N}}$ belonging to $\mathcal{N}$ with a maximum-sized matching M in bold edges.

Finally, we discuss $\mathcal{N} = (V, E)$ in Figure 15(a) and (b) in more detail to clarify the discussed algorithms, beginning with Figure 15(a). In Figure 15(c), M is an arbitrary maximum matching and $u(\mathcal{G}_{\mathcal{N}}) = 4$. The number of leaves $|X| = 2$, so $p(\mathcal{N}) = 4 - 2 = 2$. Two paths in $\mathcal{P}$ are ending at a leaf: $\pi_1 = (f, e, i, h, l, k, o, x_1)$ and $\pi_2 = (j, n, s, x_2)$. Two paths in $\mathcal{P}$ are not ending at a leaf: $\phi_1 = (a, c, b, d, g)$ and $\phi_2 = (m, r, q, p)$. We are going to obtain a display tree T_1 out of π_1 and π_2 by adding arbitrarily incoming edges to them. However, in this case there is only one possibility: adding $(a, c), (c, f)$ and (f, j) to them.

To construct a second display tree, add $(g, k), (k, o)$ and (o, x_1) to ϕ_1 and remove π_1 and ϕ_2. We do not have to add incoming edges to ϕ_1, but we add (c, f) and (f, j) to obtain a display tree T_2. Note that it is impossible to let ϕ_1 or ϕ_2 end at x_2. Therefore, a third display tree has to be made out of ϕ_2 and π_2. Add (p, o) and (o, x_1) to ϕ_2, remove π_1 and ϕ_1 and add $(a, c), (c, f)$ and (f, j) to π_2 and display tree T_3 has been constructed. Every vertex of V is present in at least one of the display trees T_1, T_2 and T_3, so $b(\mathcal{N}) = 3$ and this is minimal; $b(\mathcal{N}) \leq p(\mathcal{N}) + 1$.

In Figure 15(b) $T_1 = (V_1, E_1)$ is a display tree of $\mathcal{N}$, covering a maximum number of vertices. T_1 is basically arbitrary, but this example has only one possibility for a maximum display tree. Therefore, we can see that $\{b, d, g, m, p, q, r\}$ is the set of a minimum number of required absent vertices from any display tree of $\mathcal{N}$, because T_1 is maximal. In that case, $a(\mathcal{N}) = 7$. We have to delete $|X| - 1 = 1$ edge from T_1 to obtain two vertex disjoint paths each of which ends at a leaf of $\mathcal{N}$. This edge must be (f, e) or (f, j).

To form a minimum path partition, we have to obtain a minimum number of vertex disjoint paths out of $\{b, d, g, m, p, q, r\}$ and E which forms Q. By deleting (f, j), $P = \{(a, c, f, e, i, h, , l, k, o, x_1), (j, n, s, x_2)\}$. The partition of V is $\mathcal{P} = P \cup Q$ where $Q = \{(b, d, g), (m, r, q, p)\}$. Following the algorithm to construct a minimum of display trees such that every vertex of V is present in at least one of the trees gives us three display trees, so $b(\mathcal{N}) = 3$ and $b(\mathcal{N}) \leq a(\mathcal{N})$. The reason of the low value of $b(\mathcal{N})$ is that the two paths in Q consist of more than one required absent vertex. They consist of three and four vertices, respectively.

After the introduction of three measures of deviation, established before by Francis, Semple and Steel, we have introduced two new measures after which we proved the fifth property of tree-based networks. We could do the latter because Lemma 3.1 is a generalization. We could also define all the measures for nonbinary networks and prove the equality of the first three measures for the general case, using three inequalities and Lemma 2.4. The last two theorems and their proofs make clear that $b(\mathcal{N})$ is for many non-tree-based networks a smaller integer, but less precise. On the other hand, the construction of different display trees to obtain $b(\mathcal{N})$ has been made clear.

4 Conclusion and further questions

In this thesis we researched characterizations of tree-based networks and proximity measures. We studied several known results for binary networks and investigated whether they still hold for all (nonbinary) phylogenetic networks. It turned out that Zhang's theorem [6] does not hold for all phylogenetic networks. First, we proved the theorem for the binary case using Hall's marriage theorem [4]. Afterwards, we disproved both the first and the second equivalence by giving two counterexamples of nonbinary phylogenetic networks with the corresponding graphs $\mathcal{Z}_{\mathcal{N}}$.

Francis, Semple and Steel [3] established five characterizations of binary tree-based phylogenetic networks. We proved three of these equivalences for all phylogenetic networks using a lemma. We simplified the proof of this lemma for binary networks and have proved it for the general case afterwards. We proved a fourth characterization established by Francis, Semple and Steel by adding interesting counterexamples, where non-tree-based binary phylogenetic networks do have subsets satisfying the conditions of Theorem 2.5. Furthermore, we found another example showing that it is possible to have nondisjoint subsets of the vertex set of $\mathcal{N}$. It seems that we can extend this fourth property for all phylogenetic networks, but in this thesis we did this in only one direction.

We can consider the vertex set of any phylogenetic network as a partially ordered set. For this poset, Dilworth's theorem [2] holds for all phylogenetic networks, but for non-tree-based networks there could be chains consisting of more than one path, like the ones that satisfy the antichain-to-leaf property. Temporal networks behave different too. For binary temporal phylogenetic networks the antichain-to-leaf property is a sufficient propery to being tree-based, contrary to nontemporal binary networks.

In Section 3 we have considered five indices, non-negative integers, that measure the deviation for non-tree-based networks to being tree-based. First, we can construct a bipartite graph $\mathcal{G}_{\mathcal{N}}$ to characterize tree-based phylogenetic networks with a fifh property: $\mathcal{G}_{\mathcal{N}}$ has a matching of size $|V|-|X|$. First, we succeed to prove a lemma for all phylogenetic networks. This lemma compares $p(\mathcal{N})$ with the number of unmatched vertices in $\mathcal{G}_{\mathcal{N}}$ when a maximum-sized matching has been created. Afterwards, we could fast complete the proof of the theorem, thus the fifth characterization does hold for all phylogenetic networks.

We have shown that $l(\mathcal{N}) = p(\mathcal{N}) = t(\mathcal{N})$ for all phylogenetic networks, and they equal zero if $\mathcal{N}$ is tree-based, as $a(\mathcal{N})$ does. In contrast, $b(\mathcal{N}) = 1$ if $\mathcal{N}$ is tree-based. We described an algorithm to construct an extra display tree using path partitions and maximum-sized matchings when a value of $p(\mathcal{N})$ is given. The conclusion is that $b(\mathcal{N}) \leq p(\mathcal{N}) + 1$. We also showed that $b(\mathcal{N})$ can be much smaller than $l(\mathcal{N}), p(\mathcal{N})$ and $t(\mathcal{N})$ as in the example in Figure 15(a). We get a similar inequality if we compare $b(\mathcal{N})$ with $a(\mathcal{N})$: $b(\mathcal{N}) \leq a(\mathcal{N}) + 1$. Comparing $a(\mathcal{N})$ with $p(\mathcal{N})$ could be a topic for future work.

Theorem 2.6 is a generalization of Theorem 2.5, but only in one direction. Proving or giving a counterexample for the 'if'-direction can be future work: Is a phylogenetic network $\mathcal{N} = (V, E)$ tree-based if there is no pair of subsets U_1, U_2 satisfying the three properties of Theorem 2.6?

The generalization of Theorem 2.7 has not been proved yet. Future work can be proving Conjecture 2.8: A temporal phylogenetic network is tree-based if and only if it satisfies the antichain-to-leaf property.

References

[1] Diestel, R., *Graph Theory*, Springer, 2010.

[2] Dilworth, R.P., "A decomposition theorem for partially ordered sets", *Annals of Mathematics*, vol. 51, no. 1, pp. 161-166, 1950.

[3] Francis, A., Semple, C., Steel, M., "New characterisations of tree-based networks and proximity measures", *Advances in Applied Mathematics*, vol. 93, pp. 93-107, 2018.

[4] Hall, P., "On representatives of subsets", *J. London Math. Soc*, vol. 10, no. 1, pp. 26-30, 1935.

[5] Jetten, L., van Iersel, L.J.J., "Nonbinary tree-based phylogenetic networks", *IEEE/ACM Transactions on Computational Biology and Bioinformatics.* DOI:10.1109/TCBB.2016.2615918, 2016.

[6] Zhang, L., "On tree based phylogenetic networks", *Journal of Computational Biology*, vol. 63, no. 7, pp. 553-565, 2016.