On Turning A Graph Into A Phylogenetic Network

Laurent Bulteau, Mathias Weller, Louxin Zhang

On Turning A Graph Into A Phylogenetic Network

Laurent Bulteau[1,2], Mathias Weller[1,2], and Louxin Zhang[3]

[1] Centre Nationale de Recherche Scientifique, FRANCE
[2] Université Gustave Eiffel, Paris, FRANCE
{firstname.lastname}@u-pem.fr
[3] Department of Mathematics, National University of Singapore
matzlx@nus.edu.sg

Abstract. Phylogenetic networks, both rooted and unrooted, are a versatile and invaluable tool for microbiological research. One of the many open theoretical questions sourrounding such networks is when it is possible to orient the edges of a given unrooted network into a valid rooted network, that is, a network in which all non-root nodes have in-degree one (corresponding to a speciation event) or out-degree one (corresponding to a hybridization event). We give a characterization based on the degree of the input graph, showing that graphs of maximum degree three can be oriented in linear time while orienting graphs that may contain degree-five nodes is NP-complete.

1 Introduction

Phylogenetic networks have frequently been used to model genomic evolution for different species and genetic variants of a population. These networks are divided into unrooted and rooted ones [10, 14]. Unrooted phylogenetic networks are simple graphs with labeled leaves (degree-one nodes), whereas rooted phylogenetic networks are rooted acyclic digraphs with labeled leaves in which the root is the unique source, all the leaves are sinks and all other nodes either have in-degree one or out-degree one. Different classes of unrooted and rooted phylogenetic networks and their connections have extensively been studied in the past decade [5, 6, 7, 8, 16].

Unrooted trees and rooted trees are closely related. It is well known that every unrooted tree can be converted into a unique rooted tree rooted at a specific non-leaf node by orienting each edge away from the node. It is also known that nearest neighbor exchange operation for unrooted trees [15] corresponds to rotation transformation for rooted trees [12, 13].

Far fewer connections between unrooted and rooted phylogenetic networks are known. Orienting graphs is a task that arises in many fields such as graph algorithm design, electrical networks in physics and flow transportation networks in operations research [1, 2, 3]. For instance, topological sorting is one of the most common orientation used for designing graph algorithms.

Here, we study the problem of orienting an unrooted graph into a phylogenetic network, that is, a (single-)rooted directed acyclic graph in which all leaves have degree one. Phylogenetic network orientation can be considered as a restricted version of the so-called bi-polar orientation problem. A bi-polar orientation is an acyclic orientations with exactly two poles. Even and Tarjan developed a linear-time algorithm to compute a bipolar orientation of a 2-connected graph [4]. Orientation of degree-3 graphs into rooted phylogenetic networks with special properties have been studied by different groups [8, 9, 11].

In this paper, we first prove that any unrooted binary phylogenetic network can always be oriented into a phylogenetic network rooted at any non-leaf node in linear time, provided that the root can be placed unambiguously. This is done by a reduction to the *st*-Numbering problem. We show that it is NP-complete to determine whether or not an unrooted phylogenetic network with the maximum degree five can be oriented into a rooted one.

2 Preliminaries

Let G be a simple graph with node set V and edge set E. The set of *neighbors* of a node u is $N_G(u) := \{v \mid uv \in E\}$ and $d_G(u) := |N_G(u)|$. If clear from context, we omit the subscript. If $d_G(u) = 1$, then u is called a *leaf* of G and we use $L(G)$ to denote the set of all leaves of G.

For a node set X, define $G - X$ to be the subgraph of G that has the node set $V \setminus X$ and the edge set $\{xy \in E \mid x, y \notin X\}$ and abbreviate $G - \{v\} =: G - v$ as well as $G - (V \setminus X) =: G[X]$. A *cut-node* of G is a node v such that $G - v$ has strictly more connected components than G. Each cut-node is associated with at least one tripartition $(X, \{v\}, Y)$ of $V(G)$ such that no edge of G has endpoints in both X and Y. We say that v *separates* X and Y in G.

Each maximal $C \subseteq V(G)$ such that no node of $G[C]$ is a cut-node in $G[C]$ is called *block* (note that some nodes of $G[C]$ might be cut-nodes in G). Since the graph with a single node is considered to be connected, $|C| \geq 2$. If $|C| = 2$, then C is an edge which is called a *bridge* of G and C is called *trivial* in this case. Any node v of C is said to be *private* in G if all neighbors of v in G are in C. Further, two blocks C and C' may *overlap*, that is, $C \cap C' \neq \varnothing$. Observe that the nodes that appear in at least two blocks are exactly the cut-nodes of G. In other words, a node of G is a cut-node if and only if it is contained in two or more blocks.

Let D be a digraph with node set V and arc set A. Depending on context, we use uv to denote the node set/edge $\{u, v\}$ or the arc (u, v). For nodes u and v with $uv \in A$, u is called a *predecessor* of v and v is called a *successor* of u. The sets of the predecessors and successors of a node v are denoted by $\mathrm{N}^{\mathrm{in}}_D(v)$ and $\mathrm{N}^{\mathrm{out}}_D(v)$, respectively and their respective sizes are denoted by $\deg^{\mathrm{in}}_D(v)$ and $\deg^{\mathrm{out}}_D(v)$, respectively. Additionally, if there is a directed path from u to v, then u is said to be an *ancestor* of v and v a *descendant* of u, written as $v <_D u$.

Definition 1. *A digraph D is a* phylogenetic network *if (i) D has a unique node of in-degree 0 (called the* root*), (ii) all nodes have either in-degree at most*

one (called tree nodes*) or out-degree exactly one. Nodes with out-degree zero are called* leaves *and nodes that are not tree nodes are called* reticulations*.*

An *orientation* of a graph G is a digraph D obtained from G by assigning a direction to every edge of G. An orientation D of G is called *phylogenetic* (or *valid*) if D is a phylogenetic network. Although all undirected graphs can be oriented in an acyclic manner, not all acyclic digraphs are phylogenetic networks. Thus, it is interesting to identify and characterize which graphs can be oriented into phylogenetic networks. This leads to the following problem.

Phylogenetic Network Orientation
Input: an undirected graph G
Question: Does G admit a phylogenetic orientation?

Any acyclic digraph D admits a linear layout σ (called *topological order*) of its nodes such that all ancestors of a node v in D precede v in σ (implying that all descendants of v in D succeed v in σ). Likewise, each linear order of the nodes of G corresponds to a DAG-orientation of G and we call an order *valid* if it corresponds to a valid DAG-orientation. For a node or a set of nodes X, we let (X) denote any sequence on X. Note that paths in (di)graphs can be viewed as linear orderings of (a subset of) their nodes. Further, for each cut-node v in G separating the node set X from the node set Y, there is a topological order in which Y occurs consecutively.

3 Properties of Phylogenetic Network Orientations

In the following, we list some properties of network orientations for graphs. To this end, suppose that G can be oriented into a phylogenetic network D.

First, define the graph $\mathcal{B}(G)$ to be the graph whose nodes are the blocks of G and blocks C and C' are connected by an edge if and only if they overlap, that is $C \cap C' \neq \varnothing$. Then, by maximality of blocks, $\mathcal{B}(G)$ is acyclic. Thus, we will refer to $\mathcal{B}(G)$ as the *block tree* of G. As $\mathcal{B}(G)$ is a tree, each block C has at most one node with parents in another block and we call this node the *block-root* of C with respect to D.

Second, observe that all leaves of D have degree one in G and thus appear in distinct trivial blocks of G. Further, we have the following fact.

Lemma 1. *Let G be orientable into a phylogenetic network D. Let z be a cut-node of G separating the node set X from the node set Y such that $L(G) \subseteq X$. Then, Y contains the root of D.*

Proof. Since z is a cut-node in G, it is easy to see that D admits a topological order σ in which the nodes in Y appear consecutively. Let u and v be the respective first and last nodes of Y in σ and recall that u and v have degree at least two in G. Now, if $z <_\sigma u$, then all neighbors of v in G precede v in σ, implying that v has out-degree zero but in-degree at least two in D, contradicting

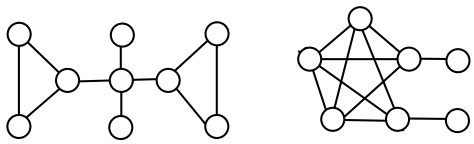

Fig. 1. Two graphs that cannot be oriented into phylogenetic networks. The left graph contains two or more non-trivial blocks. The right graph is a pseudo d-regular graph with two leaves.

Definition 1. Thus, $z >_\sigma v$, implying that all neighbors of u in G succeed u in σ, that is, u is the root of D. □

By Lemma 1, non-trivial blocks with at most one cut-node must contain the root of any valid phylogenetic orientation. Therefore, we call such blocks *root-forcing*. In particular, if a graph has multiple root-forcing blocks, then it cannot be oriented into a phylogenetic network (see Figure 1 (left)). Thus, if G can be oriented into a phylogenetic network, then each leaf C of $\mathcal{B}(G)$ is either trivial or root-forcing.

Third, we observe that each cut-node of G has parents in at most one of its blocks.

Lemma 2. *Let G be orientable into a phylogenetic network D and let v be a node of G. Then, all parents of v in D are in the same block of G.*

Proof. First, suppose v is a cut-node as, otherwise, the lemma trivially holds. Let u_1 and u_2 be parents of v in D. Since both u_1 and u_2 appear before v in any topological order σ of D, both are reachable from the root in $D - v$. Hence, u_1 and u_2 are connected in $G - v$ and biconnected in G. □

Lemma 2 implies that the block-root of any non-trivial block is a tree node. Lemma 2 further implies that, if a graph G can be oriented into a phylogenetic network D such that the root r of D is a leaf of G, then r is the only parent of its child u and, thus, reversing the arc ru results in a valid orientation whose root is u. Thus, if G contains more than two nodes and we can orient G into a phylogenetic network D, then we can assume that D is rooted at a non-leaf.

Finally, observe that every node v that is neither a leaf nor the root of D must have at least one parent and at least one child. However, by Definition 1(ii), it cannot have more than one parent and more than one child at the same time.

Observation 1 *Let D be a valid DAG, let v be a node of D with a parent u and a child w. Then, either u is the only parent of v or w is the only child of v.*

In the following, a graph or digraph is called *pseudo d-regular* if each node has either degree one or degree d. Then, we can observe restrictions on the number of reticulations, tree nodes, and leaves of such graphs.

Proposition 1. *Let G be a graph with ℓ leaves in which each non-leaf node is of degree $d \geq 3$. Let G be orientable into a phylogenetic network D with r reticulations and t tree nodes. Then, $(d-2)(t-r) = \ell - 2$.*

Proof. Since the root has out-degree d and each non-root tree node has out-degree $d-1$, the total out-degree is

$$d + (t-1)(d-1) + r = (d-1)t + r + 1 = td + r - t + 1.$$

Since each leaf has in-degree one and each reticulation has in-degree $d-1$, the total in-degree is

$$r(d-1) + t - 1 + \ell = rd + t - r + \ell - 1.$$

Hence, $td + r - t + 1 = rd + t - r + \ell - 1$, that is, $(d-2)(t-r) = \ell - 2$. □

We can combine Proposition 1 with the fact that the sum of degrees in G is twice the number of edges in G, that is, $\ell + d(t+r)$ is even.

Corollary 1. *Let $d \geq 3$ and let G be a pseudo d-regular graph with ℓ leaves that can be oriented into a phylogenetic network. Then (a) $\ell \geq 2$ and (b) ℓ is odd $\Rightarrow$ d and $t - r$ are odd (that is, $|V(G)|$ is even). Further, if $\ell = 2$, then $t = r$ and $t + r + \ell = |V(G)|$ is even.*

Corollary 1 implies that no pseudo d-regular graph with two leaves can be oriented into a rooted phylogenetic network if there is an odd number of nodes (see Figure 1 (right)).

4 Orientation Into a Binary Phylogenetic Network

In the following, we assume that the maximum degree in the input graph G is three. As proven in Section 3, if G has more than one root-forcing block, we correctly conclude that G cannot be oriented into a phylogenetic network. In the following, we show that a valid orientation D can be computed in all other cases. To this end, we select the root of D as follows: if G contains a root-forcing block C, then select a private node of C, otherwise select any node of G. This fixes an orientation of the block-tree $\mathcal{B}(G)$ and, thus, an orientation of all bridges of G. Now, the problem reduces to finding a valid orientation in a graph G that has a single non-trivial block and in which a single node r is annotated to be the root.

Algorithm 1 produces a topological ordering σ that corresponds to a valid orientation of G (and this orientation can trivially be obtained from σ and G). Herein, the nodes x and y returned in line 3 can be found using breadth-first-search (BFS) or depth-first search (DFS) from the root r and the path p returned in line 4 can be found using BFS/DFS from y. Note that p always exists since G is biconnected and, by minimality of p, only its last node is in $V(\sigma)$ in this iteration. Thus, no node is added twice into σ, implying that σ is a linear order of $V(G)$.

Algorithm 1: Compute a valid topological ordering σ for graph G containing at most one non-trivial block such that σ starts with r.

1 $\sigma \leftarrow$ any path in G from r to a node in $L(G)$;
2 **while** $V(\sigma) \setminus L(G) \neq V(G) \setminus L(G)$ **do**
3 compute a minimal (wrt. σ) node x in σ with a neighbor $y \notin V(\sigma) \cup L(G)$;
4 $p \leftarrow$ any minimal y-$V(\sigma)$-path in $G - x$;
5 remove the last node of p;
6 insert (p) right behind x in σ;
7 append all nodes of $L(G) \setminus V(\sigma)$ to σ in any order;

Lemma 3. *The digraph corresponding to the result σ of Algorithm 1 is a valid orientation of G with root r.*

Proof. Let D denote the digraph corresponding to σ and let $z \in L(G)$ be the last node of the r-$L(G)$-path that σ is initialized with in line 1 of Algorithm 1.

We show that $r \leq_D y \leq_D z$ for all $y \in V(\sigma)$ at all times in the execution of Algorithm 1, implying that all nodes except r, z, and $L(G)$ have in- and out-degree at least one in D which implies validity since the maximum degree in G is three. The proof is by induction on the iteration of the `while`-loop in Algorithm 1. Since σ is initialized with an r-z-path, this is true before the first iteration. For later iterations, each node z added to σ is part of a path originating in a node $x \in V(\sigma)$ and terminating in a node $x' \in V(\sigma)$ with $x \neq x'$. By induction hypothesis, $r \leq_D x <_D y <_D x' \leq_D z$, implying the claim. □

Regarding the running time, the nodes x for each iteration can be found in linear amortized time by employing one global DFS while finding the paths p requires a linear-time DFS on each iteration.

Corollary 2. *Given a graph G with maximum degree three, one can find a valid network orientation of G in $O(|V(G)|^2)$ time, or correctly conclude that such an orientation does not exist.*

To solve Phylogenetic Network Orientation in linear time on pseudo 3-regular graphs, we reduce to computing *st*-numberings, which can be done in linear time [4]. Herein, an *st-numbering* of a biconnected graph H containing nodes s and t is a permutation π of $V(G)$ starting with s and ending with t such that each $v \in V(H) \setminus st$ has a predecessor and a successor (wrt. π) in $N(v)$.

Construction 1 *Let G be a connected graph with at most one root-forcing block. Let r be a node of G such that, if G has a root-forcing block C, then r is private in C. Then, add new nodes s and t as well as edges sr, st, and vt for each degree-one node v of G.*

Lemma 4. *Let H be a graph with nodes s and t resulting from the application of Construction 1 to a connected graph G. Let π be an st-numbering of H and let σ be the result of removing s and t from π. Then, H is biconnected and σ is a valid ordering for G.*

Proof. First, assume that H has a cut-node u separating the node set X from the node set Y. Clearly, $u \notin st \cup L(G)$ since G is connected. Hence, $st \cup L(G) \subseteq X$ without loss of generality (implying that u is also a cut-node in $H - \{s,t\} = G$). But then, there is a non-trivial block C of $H - X = G - X$ that is a leaf in $\mathcal{B}(G)$ and C contains a single cut-node of G. However, since $L(G) \subseteq X$, C is root-forcing in G. Thus, r is a private node of C in G and, thus, we know that the edge sr in H does not exist in $H - u$ and, thus, $r = u$, contradicting u being a cut-node in G.

Second, we show that σ is a valid ordering for G. To this end, let D denote the DAG corresponding to σ and let v be a node of G. Since π is an st-numbering of H, we know that v has a predecessor u and a successor w (wrt. π) in H. Now, if $u = s$, then $v = r$ by construction, implying that r is the only root of D. Further, if $w = t$, then v is a leaf of G, implying that u is the only neighbor of v in G and, thus, v has in-degree one in D. In all other cases, u and w exist in D and, since G has maximum degree three, v has at most one additional neighbor x in G. If $x <_\pi v$, then v has out-degree one in D and if $x >_\pi v$, then v has in-degree one in D. In either case, Definition 1 holds.

Corollary 3. *Given a graph G with maximum degree three, one can find a valid network orientation of G in $O(|V(G)|)$ time, or correctly conclude that such an orientation does not exist.*

5 NP-completeness on Arbitrary Graphs

In this subsection, we prove the NP-completeness of the Phylogenetic Network Orientation problem for graphs that may contain nodes of degree five by a reduction from the Not-All-Equal 3-SAT (NAE3SAT). Recall that given a collection of 3-literal clauses over a set of Boolean variables, the NAE3SAT asks a truth assignment to the variable such that the three values in each clause are not all equal to each other.

To this end, we call a node r in G *pseudo-root* with parent u if all valid DAG orientations of G contain the arcs $u \to r$ and $r \to x$ for all $x \in N(r) - u$.

Construction 2 (See Figure 2) *Let φ be an instance of* Not-All-Equal 3-SAT *with variables $\mathcal{X} := \{x_1, x_2, \ldots, x_n\}$ and clauses $\mathcal{C} := \{C_1, C_2, \ldots, C_m\}$. Then, construct the incidence graph (V, E) of φ, add a triangle (r_0, r_1, r_2), add a node r connected to r_0, and make all x_i adjacent to r. Finally, add a new leaf y_i to each x_i.*

Lemma 5. *Let G be the graph constructed from an instance formula φ in Construction 2. Then,*
(a) r is a block-root for all valid DAG-orientations of G and
(b) φ has a NAE-satisfying assignment if and only if G has a valid DAG-orientation.

Proof. (a) & (b $\Leftarrow$): Let D be any valid DAG-orientation of G and let σ be a topological order of D. By Lemma 1, σ starts with $(\{r_2, r_1\}, r_0, r)$, implying (a).

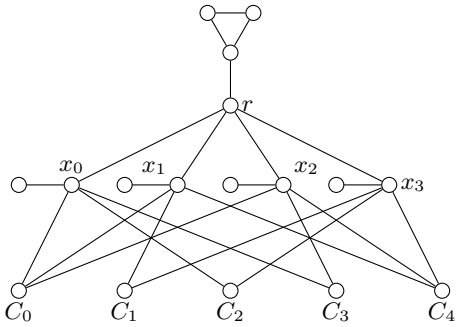

Fig. 2. Illustration of Construction 2 for $\varphi = \{x_0x_1x_2, x_1x_3, x_0x_3, x_0x_2, x_1x_2x_3\}$.

Then, by Observation 1, each x_i either precedes or succeeds *all* its adjacent clause nodes in σ. We define an assignment β by setting $\beta(x_i) := 0$ in the former case and $\beta(x_i) := 1$ in the latter case. Towards a contradiction, assume that β does not NAE-satisfy a clause C_j. If $\beta(x_i) = 0$ for all variables $x_i \in C_j$, then C_j precedes all its incident variable nodes in σ. If $\beta(x_i) = 1$ for all variables $x_i \in C_j$, then C_j succeeds all its incident variable nodes in σ. In both cases, C_j is invalid.

(b $\Rightarrow$): Let β be an assignment for φ such that all clauses are covered by both $\beta^{-1}(0)$ and $\beta^{-1}(1)$. We show that the DAG-orientation D corresponding to the order $\sigma = (r_2, r_1, r_0, r, \beta^{-1}(1), \mathcal{C}, \beta^{-1}(0), L(G))$ is valid. We prove the properties of Definition 1: For (i), note that all nodes except r_2 have a predecessor before them in σ since $\beta^{-1}(1)$ covers $\mathcal{C}$. For (ii), suppose there is some node v in D with both in-degree and out-degree at least two. Clearly, $v \notin \{r_2, r_1, r_0, r\}$ by construction and $v \notin \mathcal{C} \cup L(G)$ since v has degree at least four. Thus, $v = x_i$ for some i. However, if $x_i \in \beta^{-1}(1)$, then r is the unique parent of x_i in D and if $x_i \in \beta^{-1}(0)$, then y_i is the unique child of x_i in D, contradicting x_i being invalid. □

Degree Reduction. In order to produce graphs of maximum degree five, we can first use a version of Not-All-Equal 3-SAT where each variable occurs at most thrice (this can be achieved using clauses of the form $x_i\overline{x}_j$ which is equivalent to $x_i \iff x_j$ so x_j can be used as an alias for x_i). This leaves the node r with a high degree of $n + 1$ in the result of Construction 2 and we reduce its degree using the following rule.

Rule 1 *Let $r \in V(G)$ be a pseudo root with parent u and let $v, w \in N(r) - u$. Then, add a new node x with two leaves ℓ and ℓ' and replace the edges rv and rw by rx, xv and xw.*

Lemma 6. *Let G' be the result of applying Rule 1 to G. Then, (a) r and x are pseudo roots with parents u and r, respectively, in G' and (b) G has a valid DAG orientation if and only if G' has.*

Proof. (a) & (b $\Leftarrow$): Let D' be any DAG-orientation of G and let σ be a topological order of D. By Lemma 1, σ starts with $(\{r_2, r_1\}, r_0, r)$, implying (a) for r. Since ℓ and ℓ' are leaves, we know that D' contains the arcs $x\ell$ and $x\ell'$. By Observation 1, D' also contains the arcs xv and xw, implying (a) for x. Let D be the result of removing x, ℓ and ℓ' and adding the arcs rv and rw and note that D is an orientation of G. Further, since $r <_{D'} v$ and $r <_{D'} w$, we know that D is a DAG and D is valid since r is a pseudo root by Lemma 5(a) and all other nodes have the same in- and out-degree as in D'.

(b $\Rightarrow$): Let D be a valid DAG for G and let σ be a topological order of D. Since r is a pseudo root, we have $u <_\sigma r <_\sigma \{v, w\}$. Let D' be the result of replacing the arcs rv and rw by the arcs rx, xv, xw, $x\ell$, $x\ell'$ and note the D' is an orientation for G'. Further, D' admits a topological order π which is the result of adding x right after r and adding ℓ and ℓ' in the end of σ. To show that D' is valid, note that r is a pseudo root by (a), x is valid since r is its unique predecessor, ℓ and ℓ' are valid leaves, and the in- and out-degrees of no other node change compared to D. □

With Rule 1, we can reduce the maximum degree in the graph produced by Construction 2 to five.

Corollary 4. *It is* NP*-hard to decide if a graph G with maximum-degree five has a valid ordering.*

Noting that in the proof of Lemma 5, when G has a valid DAG-orientation, the resulting DAG is actually a tree-child network if the part above r is ignored. Since each variable x_i is a tree node if and only if the corresponding variable is true, thus, r has at least one child that is a tree node. Since each clause contains at least one true variable, each clause vertex C_i has at least one child that is a tree node. Finally, every variable vertex x_i has a leaf child. Therefore, we have the following fact.

Corollary 5. *It is NP-hard to decide if a graph with maximum-degree five with a designated node r can be oriented into a tree-child network rooted at r.*

6 Conclusion

In this work, we considered the problem of orienting the edges of a given graph G such that the result is a valid phylogenetic network. It turns out that, if the root is not forced into different components, then this is always possible if G contains no node of degree at least four and a valid orientation can be computed in linear time in this case. We further showed that the problem becomes NP-hard if G may contain nodes of degree five, leaving open the case that the maximum degree in G is four. By providing some general observations about positive instances of the orientation problem, we lay the foundation for future work in this direction. In particular, for graphs of degree larger than three, it is no longer sufficient to be able to unambiguously place the root, as shown in Figure 1. Is there maybe a

finite list of forbidden substructures characterizing degree-four graphs that can be oriented into phylogenetic networks?

Finally, we are interested in orientations of G into certain classes of networks. Note that, even for graphs of maximum degree three, it may not be easy to orient G into a network that is tree-child, tree-based, reticulation-visible, etc.

Acknowledgements

This work was supported by Singapore MOE ARC grant R-146-000-318-114.

References

[1] Arulselvan A, Groß M, Skutella M. Graph orientation and flows over time. Networks. 2015 Oct;66(3):196-209.
[2] Asahiro Y, Miyano E, Ono H. Graph classes and the complexity of the graph orientation minimizing the maximum weighted out-degree. Discrete applied mathematics. 2011 Apr 6;159(7):498-508.
[3] Berglin E, Brodal GS. A simple greedy algorithm for dynamic graph orientation. Algorithmica. 2020 Feb;82(2):245-59.
[4] Even S, Tarjan RE. Computing an st-numbering. Theoretical Computer Science. 1976 Sep 1;2(3):339-44.
[5] Huber KT, Moulton V, Wu T. Transforming phylogenetic networks: Moving beyond tree space. Journal of theoretical biology. 2016 Sep 7;404:30-9.
[6] Fischer M, Galla M, Herbst L, Long Y, Wicke K. Non-binary treebased unrooted phylogenetic networks and their relations to binary and rooted ones. arXiv preprint arXiv:1810.06853. 2018 Oct 16.
[7] Francis A, Huber KT, Moulton V. Tree-based unrooted phylogenetic networks. Bulletin of mathematical biology. 2018 Feb 1;80(2):404-16.
[8] Gambette P, Huber KT, Scholz GE. Uprooted Phylogenetic Networks. Bulletin of mathematical biology. 2017 Sep 1;79(9):2022-48.
[9] Huber KT, van Iersel L, Janssen R, Jones M, Moulton V, Murakami Y, Semple C. Rooting for phylogenetic networks. arXiv preprint arXiv:1906.07430. 2019 Jun 18.
[10] Huson DH, Rupp R, Scornavacca C. Phylogenetic networks: concepts, algorithms and applications. Cambridge University Press; 2010 Dec 2.
[11] Van Iersel L, Kelk S, Stamoulis G, Stougie L, Boes O. On unrooted and root-uncertain variants of several well-known phylogenetic network problems. Algorithmica. 2018 Nov 1;80(11):2993-3022.
[12] Moore GW, Goodman M, Barnabas J. An iterative approach from the standpoint of the additive hypothesis to the dendrogram problem posed by molecular data sets. Journal of theoretical biology. 1973 Mar 1;38(3):423-57.
[13] Sleator DD, Trajan RE, Thurston WP. Short encodings of evolving structures. SIAM Journal on Discrete Mathematics. 1992 Aug;5(3):428-50.
[14] Steel M. Phylogeny: discrete and random processes in evolution. Society for Industrial and Applied Mathematics; 2016 Sep 19.
[15] Waterman MS, Smith TF. On the similarity of dendrograms. Journal of theoretical biology. 1978 Aug 21;73(4):789-800.

[16] Zhang LX. Clusters, trees and phylogenetic network classes. Bioinformatics and Phylogenetics—Seminal Contributions of Bernard Moret (in Press), Springer Nature, Switzerland. 2019.