On Alternative Models for Leaf Powers

Benjamin Bergougnoux
University of Bergen, Norway

Svein Høgemo
University of Bergen, Norway

Jan Arne Telle
University of Bergen, Norway

Martin Vatshelle
University of Bergen, Norway

Abstract

A fundamental problem in computational biology is the construction of phylogenetic trees, also called evolutionary trees, for a set of organisms. A graph-theoretic approach takes as input a similarity graph G on the set of organisms, with adjacency denoting evolutionary closeness, and asks for a tree T whose leaves are the set of organisms, with two vertices adjacent in G if and only if the distance between them in the tree is less than some specified distance bound. If this exists G is called a leaf power.

Over 20 years ago, [Nishimura et al., J. Algorithms, 2002] posed the question if leaf powers could be recognized in polynomial time. In this paper we explore this still unanswered question from the perspective of two alternative models of leaf powers that have been rather overlooked. These models do not rely on a variable distance bound and are therefore more apt for generalization. Our first result concerns leaf powers with a linear structure and uses a model where the edges of the tree T are weighted by rationals between 0 and 1, and the distance bound is fixed to 1. We show that the graphs having such a model with T a caterpillar are exactly the co-threshold tolerance graphs and can therefore be recognized in $O(n^2)$ time by an algorithm of [Golovach et al., Discret. Appl. Math., 2017].

Our second result concerns leaf powers with a star structure and concerns the geometric NeS model used by [Brandstädt et al., Discret. Math., 2010]. We show that the graphs having a NeS model where the main embedding tree is a star can be recognized in polynomial time. These results pave the way for an attack on the main question, to settle the issue if leaf powers can be recognized in polynomial time.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms

Keywords and phrases Leaf powers, Co-threshold tolerance graphs, Interval graphs, Neighborhood subtree

1 Introduction

Leaf powers were introduced by Nishimura et al. in [23], and have enjoyed a steady stream of research. Leaf powers are related to the problem of reconstructing *phylogenetic trees*. The introduction of [23] gives a nice overview of the connection of leaf powers to biology. For an integer k, a graph G is a k-leaf power if there exists a tree T – called a *leaf root* – with a one-to-one correspondence between $V(G)$ and the leaves of T, such that two vertices u and v are neighbors in G if and only if the distance between the two corresponding leaves in T is at most k. The class of graphs called leaf powers is formed by taking the union of the class of k-leaf powers for all distance values k. The outstanding open problem in the field is whether leaf powers can be recognized in polynomial time.

Simplifying somewhat we may say the work on leaf powers has followed two main lines, focusing either on the distance values k or on the relation of leaf powers to other graph classes, see e.g. the survey by Calamoneri et al. [7]. For the first approach, we know that

k-leaf powers for any $k \leqslant 6$ is recognizable in polytime [4, 5, 8, 11, 12, 23], but for any value of $k \geqslant 7$ it is an open problem if a polytime algorithm exists. Moreover, the recognition of k-leaf powers is known to be FPT parameterized by k and the degeneracy of the graph [13]. For the second approach, we can mention that interval graphs [5] and rooted directed path graphs [3] are leaf powers, and also that leaf powers have mim-width one [18] and are strongly chordal. The latter result follows since trees are strongly chordal and powers of strongly chordal graphs remain strongly chordal [9]. Moreover, an infinite family of strongly chordal graphs that are not leaf powers has been identified [19], see also [22].

To decide if leaf powers are recognizable in polynomial time, it may be better not to focus on the distance values k, since the recognition algorithms for small values of k do not seem to generalize, and it may even be that recognizing k-leaf powers for some fixed k is NP-complete while leaf powers could be recognized in polynomial time. In this paper we therefore take a different approach, and consider two alternative models for leaf powers: using weighted leaf roots, and using NeS (Neighborhood Subtree) models. Both models are probably known to researchers in the field but have so far not been a focus for the work on leaf powers. We show that basic structural restrictions of these models guarantee polynomial-time recognition, with a better prospect for generalizing, as the models do not depend on a distance value.

The first model uses rational edge weights between 0 and 1 in the tree T which allows to fix a bound of 1 for the tree distance. It is not hard to see that this coincides with the standard definition of leaf powers using an unweighted tree T and a bound k on distance. Given a solution of the latter type we simply set all edge weights to $1/k$, while in the other direction we let k be the least common denominator of all edge weights and then subdivide each edge a number of times equal to its weight times k.

The second model arises by combining the result of Brandstädt et al. that leaf powers are exactly the fixed tolerance NeST graphs [3, Theorem 4] with the result of Bibelnieks et al. [1, Theorem 3.3] that these latter graphs are exactly those that admit what they call a "neighborhood subtree intersection representation", that we choose to call a NeS model. Let us note right away that using NeS models the leaf powers are seen as a generalization of interval graphs to graphs having a tree structure. The standard way of generalizing interval graphs is from subpaths of a path to subtrees of a tree, to arrive at chordal graphs. If we view intervals of the line as having a center that stretches uniformly in both directions, we can generalize the line to a tree embedded in the plane, and the intervals to an embedded subtree with a center that stretches uniformly in all directions along tree edges. Thus a NeS model of a graph G consists of an embedded tree and one such subtree for each vertex, such that two vertices are adjacent in G iff their subtrees have non-empty intersection. Precise definitions are given later. The leaf powers are exactly the graphs having a NeS model. Certain results are much easier to prove using NeS models, for example that leaf powers are closed under addition of a universal vertex; just add a subtree that spans the entire embedding tree.

In this paper we show that fundamental restrictions on these models do allow polynomial-time recognition. Using the first model, we restrict to edge-weighted trees T having a path containing all nodes of degree 2 or more, i.e. T should be a caterpillar. If such a tree exists we call G a linear leaf power. Let us note that Brandstädt et al. [2] also considered leaf roots restricted to caterpillars (see also [6]), but they did this in the unweighted setting. They showed that the unit interval graphs are exactly the class of graphs that are k-leaf powers for some value of k with the leaf root being an unweighted caterpillar. Formulating our result in their setting, the linear leaf powers are graphs whose leaf roots are a *subdivision* of an unweighted caterpillar, in particular since we allow weights also on edges incident to a leaf of the caterpillar. We show that linear leaf powers are exactly the *co-threshold tolerance* *graphs*, and combined with the algorithm of Golovach et al. [15] this implies that we can recognize linear leaf powers in $O(n^2)$ time. Our proof goes via the equivalent concept of blue-red interval graphs, see Figure 2.

For NeS models, we restrict to graphs having a NeS model where the embedding tree is a star, and show that they can be recognized in polynomial time. Note that allowing the embedding tree to be a subdivided star will result in the same class of graphs. Our algorithm uses the fact that the input graph must be a chordal graph, and for each maximal clique X we check if G admits a star NeS model where the set of vertices having a subtree containing the central vertex of the star is X. To check this we use a combinatorial characterization, that we call a *good partition*, of a star NeS model. In the Conclusion section we discuss how the techniques introduced here define a roadmap towards possibly settling the complexity of leaf powers recognition.

2 Preliminaries

For a non-negative integer k, we denote by $[k]$ the set $\{1, 2, \ldots, k\}$. A partition of a set S is a collection of non-empty disjoint subsets $B_1, \ldots, B_t$ of S – called *blocks* – such that $S = B_1 \cup \cdots \cup B_t$. Given two partitions $\mathcal{A}, \mathcal{B}$ of a set S, we say that $\mathcal{A} \sqsubseteq \mathcal{B}$ if every block of $\mathcal{A}$ is included in a block of $\mathcal{B}$.

Graph. Our graph terminology is standard and we refer to [10]. The vertex set of a graph G is denoted by $V(G)$ and its edge set by $E(G)$. An edge between two vertices x and y is denoted by xy or yx. The set of vertices that is adjacent to x is denoted by $N(x)$. A vertex x is simplicial if $N(x)$ is a clique. Two vertices x, y are true twins if they are adjacent and $N(x) \cup \{x\} = N(y) \cup \{y\}$. The subgraph of G induced by a subset X of its vertex set is denoted by $G[X]$. Given a tree T and an edge-weight function $\mathsf{w} : E(T) \to \mathbb{Q}$, the distance between two vertices x and y denoted by $d_T(x, y)$ is $\sum_{e \in E(P)} \mathsf{w}(e)$ where P is the unique path between x and y.

Leaf power. In the Introduction we have already given the standard definition of leaf powers and leaf roots, and also we argued the equivalence with the following. Given a graph G, a leaf root of G is a pair (T, w) of a tree T and a rational-valued weight function $\mathsf{w} : E(T) \to [0, 1]$ such that the vertices of G are the leaves of T and for every $u, v \in V(G)$, u and v are adjacent if and only if $d_T(u, v) \leqslant 1$. Moreover, if T is a caterpillar we call (T, w) a *linear leaf root*. A graph is a *leaf power* if it admits a leaf root and it is a linear leaf power if it admits a linear leaf root. Since we manipulate at the same time graphs and trees representing them, the vertices of trees will be called *nodes*.

Interval graphs. Given a graph G, an *interval representation* $\mathcal{I} = (I_v)_{v \in V(G)}$ of G is a collection of intervals in $\mathbb{Q}$ associated with the vertices of G. G is an *interval graph* if it admits an interval representation $(I_v)_{v \in V(G)}$ such that for every pair of vertices $u, v \in V(G)$, the intervals I_v and I_u intersect if and only if $uv \in E(G)$. We call $(I_v)_{v \in V(G)}$ an interval model of G. For an interval $I = [\ell, r]$, we define the midpoint of I as $(\ell + r)/2$ and its length as $r - \ell$.

Clique tree. For a chordal graph G, a clique tree CT of G is a tree whose vertices are the maximal cliques of $V(G)$ and for every vertex $v \in V(G)$, the set of maximal cliques of G containing v induces a subtree of CT. Figure 4 gives an example of clique tree. Every chordal graph admits $O(n)$ maximal cliques and given a graph G, in time $O(n + m)$ we can construct a clique tree of G or confirms that G is not chordal [17, 24]. When a clique tree is a path, we call it clique path. We denote by $(K_1, \ldots, K_k)$ the clique path whose vertices are $K_1, \ldots, K_k$ and where K_i is adjacent to K_{i+1} for every $i \in [k - 1]$.

3 Linear leaf powers

In this section we show that linear leaf powers are exactly the *co-threshold tolerance graphs* (co-TT graphs), defined in [20]. Combined with the algorithm in [15], this implies that we can recognize linear leaf powers in $O(n^2)$ time.

An alternative characterization of the co-TT graphs is that of a *blue-red interval graph* [16, Proposition 3.3].

▶ **Definition 1** (Blue-red interval graph). *A graph G is a* blue-red interval *graph if there exists a bipartition (B, R) of $V(G)$ and an interval representation $\mathcal{I} = (I_v)_{v \in V(G)}$ such that*

$$E(G) = \{b_1 b_2 \,:\, b_1, b_2 \in B \text{ and } I_{b_1} \cap I_{b_2} \neq \varnothing\} \cup \{rb \,:\, r \in R, b \in B \text{ and } I_r \subseteq I_B\}.$$

We call $(B, R, \mathcal{I})$ a blue-red interval model of G.

The red vertices induce an independent set, $(I_b)_{b \in B}$ is an interval model of $G[B]$, and we have a blue-red edge for each red interval contained in a blue interval. See Figure 2 for an example of blue-red interval model of a graph. The following fact can be easily deduced from Figure 1.

▶ **Fact 2.** *Let two intervals I_1, I_2 with lengths ℓ_1, ℓ_2 and midpoints m_1, m_2 respectively. We have $I_1 \cap I_2 \neq \varnothing$ if and only if $|m_1 - m_2| \leqslant \frac{\ell_1 + \ell_2}{2}$. Moreover, we have $I_2 \subseteq I_1$ if and only if $|m_1 - m_2| \leqslant \frac{\ell_1 - \ell_2}{2}$.*

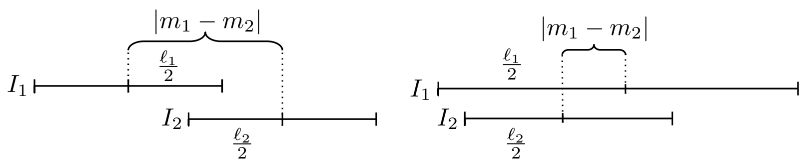

Figure 1 Example of two intervals overlapping and one interval containing another one.

To prove that linear leaf powers are exactly blue-red interval graphs, we use a similar construction as the one used in [2, Theorem 6] to prove that every interval graph is a leaf power, but in our setting, we have to deal with blue vertices and this complicates things quite a bit.

▶ **Theorem 3.** *A graph is a blue-red interval graph if and only if it is a linear leaf power.*

Proof. The constructions described in this proof are illustrated in Figure 2.

($\Rightarrow$) Let G be a blue-red interval graph with blue-red interval model $(B, R, (I_v)_{v \in V(G)})$. We assume w.l.o.g. that G is connected as otherwise we can obtain a leaf root of G from leaf roots of its connected components by creating a new node adjacent to an internal node of each leaf root via edges of weight 1. For every $v \in V(G)$, we denote by $\ell(v)$ and $m(v)$ the length and the midpoint of the interval I_v. We suppose w.l.o.g. that, for every $v \in V(G)$, we have $0 < \ell(v) \leqslant 1$ as we can always divide the endpoints of all intervals by $\mathsf{max}\{\ell(v) \,:\, v \in V(G)\}$ and add some $\epsilon > 0$ to the right endpoints of the intervals of length 0. Let $V(G) = \{v_1, \ldots, v_n\}$ such that $\ell(v_i) \leqslant \ell(v_j)$ for every $i < j$.

We define T a caterpillar and its edges $f_1, \ldots, f_n, e_1, \ldots, e_{n-1}$ as depicted in Figure 3. Let $\mathsf{w} : E(T) \to [0, 1]$ such that for every $i \in [n-1]$ we have $\mathsf{w}(e_i) = m(v_{i+1}) - m(v_i)$ and for every $i \in [n]$ the weight of f_i is $\frac{1-\ell(v_i)}{2}$ if $v_i \in R$ and $\frac{1+\ell(v_i)}{2}$ if $v_i \in B$. Since we assume

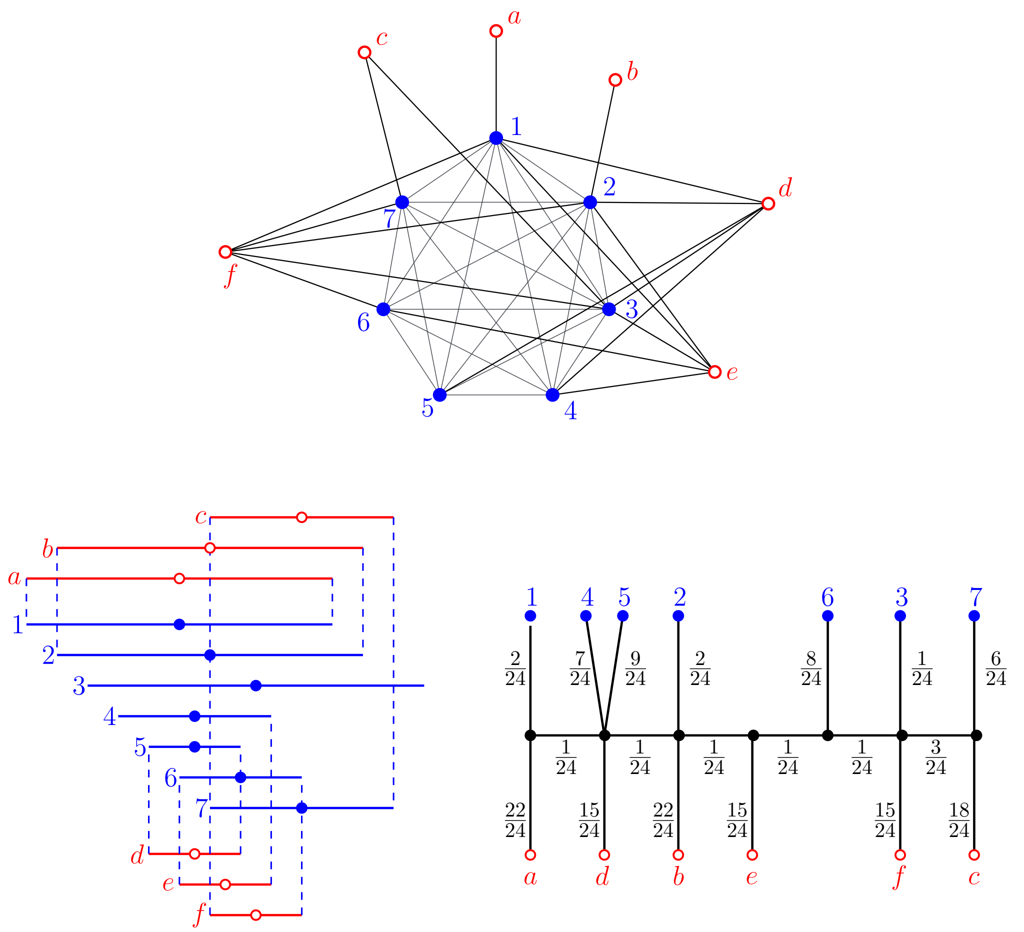

Figure 2 A split graph G with a blue-red interval model of G and a linear leaf root of G obtained from the blue-red interval model by using the construction described in Theorem 3. Note the midpoint of each interval has been marked, to better understand where each vertex is placed in T. Weights on edges is calculated as explained in the proof of Theorem 3 and note every edge has a weight that is a multiple of $\frac{1}{24}$ because the length of I_3 is 12.

that G is connected and the length of the intervals are at most 1, for every $i \in [n-1]$, we have $\mathsf{w}(e_i) = m(v_{i+1}) - m(v_i) \leqslant 1$ so the weights are well defined.

We claim that (T, w) is a linear leaf root of G. Let $i, j \in [n]$ such that $i < j$. We have to prove that $v_i v_j \in E(G)$ if and only if $d_T(v_i, v_j) \leqslant 1$. The edges of the path between v_i and v_j in T are $f_i, e_i, e_{i+1}, \dots, e_{j-1}, f_j$. By construction, we have

$$\mathsf{w}(e_i) + \cdots + \mathsf{w}(e_{j-1}) = m_{i+1} - m_i + m_{i+2} - m_{i+1} + \cdots + m_{j-1} - m_{j-2} + m_j - m_{j-1} = m_j - m_i.$$

Hence $d_T(v_i, v_j) = \mathsf{w}(f_i) + |m_j - m_i| + \mathsf{w}(f_j)$. If v_i and v_j are red vertices, then $\mathsf{w}(f_i), \mathsf{w}(f_j) > \frac{1}{2}$ because we suppose that $\ell(x) > 0$ for every vertex x. Thus, $d_T(v_i, v_j) > 1$ for every pair of red vertices (v_i, v_j). So at least one vertex among v_i, v_j is blue. Suppose w.l.o.g. that v_i is blue. As $d_T(v_i, v_j) = \mathsf{w}(f_i) + |m_j - m_i| + \mathsf{w}(f_j)$, we deduce that

$$d_T(v_i, v_j) = \begin{cases} 1 + |m(v_i) - m(v_j)| - \left(\frac{\ell(v_i)+\ell(v_j)}{2}\right) & \text{if } v_j \in B \\ 1 + |m(v_i) - m(u_i)| - \left(\frac{\ell(v_i)-\ell(v_j)}{2}\right) & \text{if } v_j \in R. \end{cases} \tag{1}$$

If $v_j \in B$ then by Fact 2, we have $I_{v_i} \cap I_{v_j} \neq \varnothing$ if and only if $|m(v_i) - m(v_j)| \leqslant \frac{\ell(v_i)+\ell(v_j)}{2}$. We

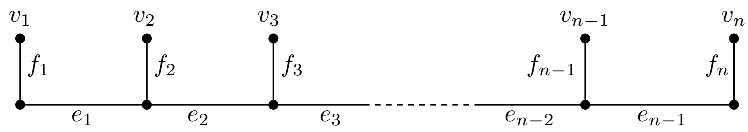

Figure 3 The linear branch-decomposition used to prove that every blue-red interval graph is a linear leaf power.

conclude that $v_iv_j \in E(G)$ if and only if $d_T(v_i, v_j) = |m(v_i) - m(v_j)| - \left(\frac{\ell(v_i)+\ell(v_j)}{2}\right) + 1 \leqslant 1$. On the other hand, if $v_j \in R$ then $I_{v_j} \subseteq I_{v_i}$ if and only if $|m(v_i) - m(v_j)| \leqslant \frac{\ell(v_i)-\ell(v_j)}{2}$. We conclude that $v_iv_j \in E(G)$ if and only if $d_T(v_i, v_j) = |m(v_i) - m(v_j)| - \left(\frac{\ell(v_i)-\ell(v_j)}{2}\right) + 1 \leqslant 1$. Hence, (T, w) is a linear leaf root of G. This proves that every blue-red interval graph is a linear leaf power.

($\Leftarrow$) Let G be a linear leaf power and (T, w) a linear leaf root of G. Let $(u_1, \ldots, u_t)$ be the path induced by the internal vertices of T. We suppose w.l.o.g. that G does not contain isolated vertices as we can easily deal with such vertices by associating each of them with an interval that does not intersect the other intervals. Consequently, for every leaf a of T adjacent to some u_i, we have $d_T(a, u_i) \leqslant 1$. For every vertex $v \in V(G)$ whose neighbor in T is u_i, we associate v with an interval I_v and a color such that the midpoint of I_v is $m(v) = d_T(u_1, u_i)$ and

- if $\mathsf{w}(u_iv) \leqslant \frac{1}{2}$, the length of I_v is $\ell(v) = 1 - 2\mathsf{w}(u_iv)$ and the color of v is blue,
- otherwise ($\frac{1}{2} < \mathsf{w}(u_iv) \leqslant 1$), the length of I_v is $\ell(v) = 2\mathsf{w}(u_iv) - 1$ and the color of v is red.

Let $B, R \subseteq V(G)$ be the sets of blue and red vertices respectively. Observe that the red vertices induce an independent set since their distance to the inner path is strictly more than 1/2. By construction, we have the following equation for every vertex v whose neighbor in T is u_i

$$w(u_iv) = \begin{cases} \frac{1-\ell(v)}{2} & \text{if } v \in B, \\ \frac{1+\ell(v)}{2} & \text{if } v \in R. \end{cases} \tag{2}$$

Moreover, from the definition of the midpoint's, we deduce that for every $v_i, v_j \in V(G)$ whose neighbors in T are respectively u_i and u_j, we have

$$d_T(v_i, v_j) = |m(v_i) - m(v_j)| + w(u_iv_i) + w(u_jv_j). \tag{3}$$

From Equations 2 and 3, we deduce that Equation 1 holds also for this direction for every $v_i \in B$ and $v_j \in V(G)$. By Fact 2 and with symmetrical arguments to the ones of the previous direction, we conclude that $(B, R, (I_v)_{v \in V(G)})$ is a blue-red interval model of G. ◀

4 NeS model

In this section, we present an alternative definition of leaf powers through the notion of NeS models. An *embedding tree* $\mathcal{T}$ of an edge-weighted tree T (with positive weights) is a set of points in the Euclidean plane so that each edge $e \in E(T)$ corresponds to a line in $\mathcal{T}$ whose length is its weight, the lines of $\mathcal{T}$ intersect one another only at their endpoints, and the vertices of T correspond (one-to-one) to the endpoints of the lines. The distance between two points x, y of $\mathcal{T}$ denoted by $d_{\mathcal{T}}(x, y)$ is the length of the unique path between x and y.

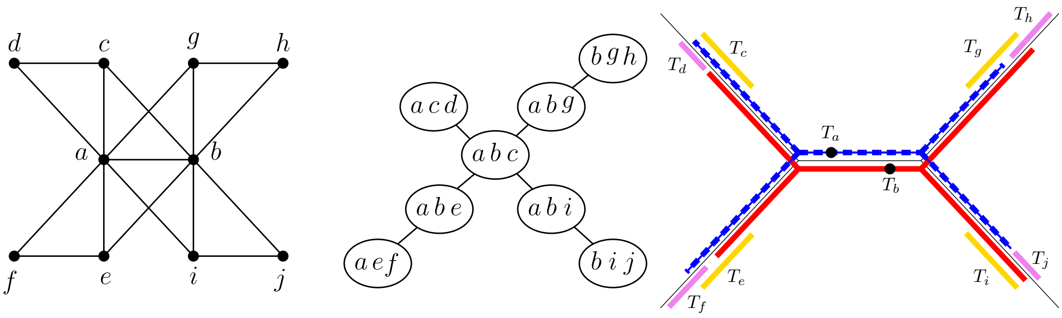

Figure 4 From left to right: a leaf power G, a clique tree of G and the representation of a NeS model of G. The dots are the centers of T_a and T_b. This graph does not admit a star NeS model.

▶ **Definition 4** (Neighborhood subtree, NeS-model). *Let $\mathcal{T}$ be an embedding tree. For each $p \in \mathcal{T}$ and non-negative rational w, we define the neigbhorhood subtree with center c and diameter w as the set of points $\{p \in \mathcal{T} : d_{\mathcal{T}}(p, c) \leqslant w\}$.*

A NeS model $(\mathcal{T}, (T_v)_{v \in V(G)})$ of a graph G is a pair of an embedding tree $\mathcal{T}$ and a collection of neigbhorhood subtrees of $\mathcal{T}$ associated with each vertex of G such that for every $u, v \in V(G)$, we have $uv \in E(G)$ if and only if $T_u \cap T_v \neq \varnothing$.

Figure 4 represents a graph and one of its NeS models.

▶ **Theorem 5.** *A graph is a leaf power if and only if it admits a NeS model.*

Proof. Brandstädt et al. prove that leaf powers correspond to the graph class called fixed tolerance NeST graph [3, Theorem 4]. Moreover, Bidelnieks and Dearing showed that a graph is a fixed tolerance NeST graph if and only if it has a NeS model [1, Theorem 3.3]. ◀

Observe that every interval graph has a NeS model where the embedding tree is a single edge. Moreover, if a graph G admits a NeS model $(\mathcal{T}, (T_v)_{v \in V(G)})$, then for every embedding path $\mathcal{L}$ of $\mathcal{T}$, $(\mathcal{L}, (T_v \cap \mathcal{L})_{v \in X})$ is an interval model of $G[X]$ with X the set of vertices v such that T_v intersects $\mathcal{L}$. NeS models provide a practical characterization of leaf powers from which many results can be easily deduced. For example, it is straightforward to prove that every leaf power is chordal and the following fact.

▶ **Fact 6.** *For every leaf power graph G, if G admits a NeS model $(\mathcal{T}, (T_v)_{v \in V(G)})$ where $\mathcal{T}$ is the embedding of a tree T, then G admits a clique tree that is a subdivision of T.*

The converse of Fact 6 is not true. For example, the graph represented in Figure 4 admits a clique tree that is a subdivided star but as we explain later in the text it does not admit a NeS model whose embedding tree is a star.

5 Star NeS model

In this section we show that we can recognize in polynomial time graphs with a *star NeS model*: a NeS model whose embedding tree is a star (considering subdivided stars instead of stars does not make a difference). Our result is based on the purely combinatorial definition of *good partition*, we show that a graph admits a star NeS model if and only if it admits a good partition. We also show that, given a good partition, we can compute a star NeS model in polynomial time. Finally, we prove that Algorithm 1 in polynomial time constructs a good partition of the input graph or confirms that it does not admit one.

Consider a star NeS model $(\mathcal{T}, (T_v)_{v\in V(G)})$ of a graph G. Observe that $\mathcal{T}$ is the union of lines $L_1, \ldots, L_\beta$ with a common endpoint c that is the center of $\mathcal{T}$. The good partition associated with this star NeS model is the pair $(X, \mathcal{B})$ where X is the set of vertices whose neighborhood subtrees contain c and $\mathcal{B}$ is a partition of $V(G) \setminus X$ containing, for every $i \in [\beta]$, the set B_i of vertices in $V(G) \setminus X$ whose neighborhood subtrees intersect L_i. The following facts are immediate.

▶ **Fact 7.**
- *There is no edge between B_i and B_j for $i \neq j$ and thus $\mathsf{CC}(G - X) \sqsubseteq \mathcal{B}$.*
- *For every $i \in [\beta]$ the NeS model $(L_i, (T_v \cap L_i)_{v\in B_i\cup X})$ is an interval model of $G[X \cup B_i]$.*
- *For each $x \in X$ the neighborhood subtree T_x is the union of the β intervals $L_1 \cap T_x, \ldots, L_\beta \cap T_x$ and there exist positive rationals ℓ_x and h_x with $\ell_x \leqslant h_x$ such that one interval among these intervals have length h_x and the other $\beta - 1$ intervals have length ℓ_x.*

If $\ell_x = h_x$, then the center of T_x is c.

Let us also observe that some intervals can always be stretched so that X corresponds to a maximal clique. The following observation is important for our algorithm.

▷ Claim 8. If a graph has a star NeS model, it has a star NeS model $(\mathcal{T}, (T_v)_{v\in V(G)})$ where the vertices whose neighborhood subtrees contain the center of $\mathcal{T}$ is a maximal clique.

Proof. Let $(\mathcal{T}, (T_v)_{v\in V(G)})$ be the star NeS model of a graph G with c the center of $\mathcal{T}$. Suppose that the set X of vertices x such that $c \in T_x$ is not a maximal clique. We deduce that there exists at least one maximal clique Y containing X. For every such clique Y, $\cap_{y\in Y} T_y$ does not contain c since $X \subset Y$. We take a maximal clique Y such that $X \subset Y$ and $\min\{d_T(c, p) : p \in \cap_{y\in Y} T_y\}$ is minimum. Let L be the line of $\mathcal{T}$ containing $\cap_{y\in Y} T_y$. By Fact 7, the neighborhood subtrees of the vertices in $Y \setminus X$ are intervals of L. We modify the NeS model by stretching the neighborhood subtrees of the vertices in $Y \setminus X$ so that they admit c as an endpoint. After this operation, these subtrees remain intervals of L and consequently, they still are neighborhood subtrees. Moreover, the choice of Y implies that the only neighborhood subtrees that were intersecting the interval between c and $\cap_{y\in Y} T_y$ are those associated with the vertices of Y. Thus, after this operation, we obtain a star NeS model where the set of vertices x such that $c \in T_x$ is now the maximal clique Y. ◁

So far we have described a good partition as it arises from a star NeS model. Now we introduce the properties of a good partition that will allow to abstract away from geometrical aspects while still being equivalent, i.e. so that a graph has a good partition $(X, \mathcal{B})$ iff it has a star NeS model. The first property is $\mathsf{CC}(G - X) \sqsubseteq \mathcal{B}$ and the second is that for every $B \in \mathcal{B}$ the graph $G[X \cup B]$ is an interval graph having a model where the intervals of X are the leftmost ones.

▶ **Definition 9** (X-interval graph). *Let G be a graph and X a maximal clique of G. We say that G is an X-interval graph if G admits a clique path $(K_1, \ldots, K_k, X)$.*

The third property is the existence of an elimination order for the vertices of X based on the lengths ℓ_x in the last item of Fact 7, namely the permutation $(x_1, \ldots, x_t)$ of X such that $\ell_{x_1} \leqslant \ell_{x_2} \leqslant \ldots \leqslant \ell_{x_t}$. This permutation has the property that for any $i \in [t]$, among the vertices $x_i, x_{i+1}, ..., x_t$ the vertex x_i must have the minimal neighborhood in at least $\beta - 1$ of the blocks of $\mathcal{B}$.

▶ **Definition 10** (Removable vertex). *Let $X, X' \subseteq V(G)$ such that $X' \subseteq X$ and let $\mathcal{B}$ be a partition of $V(G) \setminus X$. Given a block B of $\mathcal{B}$ and $x \in X'$, we say $N(x)$ is minimal in B for X' if $N(x) \cap B \subseteq N(y)$ for every $y \in X'$. We say that a vertex $x \in X'$ is removable from X' for $\mathcal{B}$ if $N(x)$ is minimal in at least $|\mathcal{B}| - 1$ blocks of $\mathcal{B}$ for X'.*

▶ **Definition 11** (Good partition). *A good partition of a graph G is a pair $(X, \mathcal{B})$ where X is a maximal clique of G and $\mathcal{B}$ a partition of $V(G) \setminus X$ satisfying the following properties:*

(A) $\mathsf{CC}(G - X) \sqsubseteq \mathcal{B}$, *i.e. every component $C \in \mathsf{CC}(G - X)$is contained in a block of $\mathcal{B}$.*

(B) *For each block $B \in \mathcal{B}$, $G[X \cup B]$ is an X-interval graph.*

(C) *There exists an elimination order $(x_1, \dots, x_t)$ on X such that for every $i \in [t]$, x_i is removable from $\{x_i, \dots, x_t\}$ for $\mathcal{B}$.*

Call X the central clique of $(X, \mathcal{B})$ and $(x_1, \dots, x_t)$ a good permutation of X or of $(X, \mathcal{B})$.

▶ **Theorem 12.** *A graph G admits a good partition if and only if it admits a star NeS model. Moreover, given the former we can compute the latter in polynomial time.*

Proof. ($\Leftarrow$) Let G be a graph with a star NeS model $(\mathcal{T}, (T_v)_{v \in V(G)})$. Let X be the set of vertices x such that T_x contains the center c of the star $\mathcal{T}$. By Claim 8, we can assume that X is a maximal clique. As $\mathcal{T}$ is a star, $\mathcal{T}$ is an union of lines $L_1, \dots, L_\beta$ with one common endpoint that is c. Let $\mathcal{B} = \{B_1, \dots, B_\beta\}$ such that, for every $i \in [\beta]$, B_i is the set of vertices $v \in V(G) \setminus X$ such that T_v intersects L_i. We claim that $(X, \mathcal{B})$ is a good partition of G. Fact 7 implies that Property (A) and Property (B) are satisfied. For every $x \in X$, let ℓ_x be the rational defined in Fact 7. Take the permutation $(x_1, \dots, x_t)$ of X such that $\ell_{x_1} \leqslant \ell_{x_2} \leqslant \dots \leqslant \ell_{x_t}$. Let $f(i) \in [\beta]$ such that the center of x_i lies in $L_{f(i)}$. From Fact 7, we have $|T_{x_i} \cap L_j| = \ell_{x_i}$ for every $j \neq f(i)$. Hence, for every $j \neq f(i)$ the interval $T_{x_i} \cap L_j$ are contained in the neighborhood subtree T_y for every $y \in \{x_i, \dots, x_t\}$. Consequently $N(x_i)$ is minimal in B_j for $\{x_i, \dots, x_t\}$ for every $j \neq f(i)$. We conclude that $(x_1, \dots, x_t)$ is a good permutation of $(X, \mathcal{B})$, i.e. Property (C) is satisfied.

($\Rightarrow$) Let $(X, \mathcal{B})$ be a good partition of a graph G with $\mathcal{B} = \{B_1, \dots, B_\beta\}$ and $(x_1, \dots, x_t)$ be a permutation of $(X, \mathcal{B})$. For every $i \in [t]$, we define $X_i = \{x_i, \dots, x_t\}$. Since x_i is removable from X_i, there exists an integer $f(i) \in [\beta]$ such that $N(x_i)$ is minimal for X_i in every block of $\mathcal{B}$ different from $B_{f(i)}$. [1]

Take $\mathcal{T}$, the embedding of a star with center c that is the union of β lines $L_1, \dots, L_\beta$ of length $2t + 1$ whose intersection is $\{c\}$. We start by constructing the neighborhood subtree of the vertices in X. For doing so, we associate each $x_i \in X$ and each line L_j with a rational $\ell(x_i, L_j)$ and define T_{x_i} as the union of over $j \in [\beta]$ of the the points on L_j at distance at most $\ell(x_i, L_j)$ from c .

For every $i \in [t]$ and $j \in [\beta]$ such that $j \neq f(i)$, we define $\ell(x_i, L_j) = i$. We define $\ell(x_t, L_{f(t)}) = t$ and for every i from $t - 1$ to 1 we define $\ell(x_i, L_{f(i)})$ as follows:

- If $N(x_i)$ is minimal in $B_{f(i)}$ for X_i then we define $\ell(x_i, L_{f(i)}) = i$.
- If $N(x_j) \cap B_{f(i)} \subset N(x_i)$ for every $x_j \in X_i$, then we define $\ell(x_i, L_{f(i)}) = 1 + \max\{\ell(y, L_{f(i)}) : y \in X_{i+1}\}$.
- Otherwise, we take $x_{\min}$ and $x_{\max} \in X_{i+1}$ such that

$$N(x_{\min}) \cap B_{f(i)} \subseteq N(x_i) \cap B_{f(i)} \subseteq N(x_{\max}) \cap B_{f(i)}$$

  and $N(x_{\min}) \cap B_{f(i)}$ is maximal and $N(x_{\max}) \cap B_{f(i)}$ is minimal. We define $\ell(x_i, L_{f(i)}) = (\ell(z^{\min}, L_{f(i)}) + \ell(z^{\max}, L_{f(i)}))/2$. Observe that the vertices $x_{\min}$ and $x_{\max}$ exist because $G[X \cup B_{f(i)}]$ is an X-interval graph and thus the neighborhoods of the vertices in X in $B_{f(i)}$ are pairwise comparable for the inclusion.

By construction, we deduce the following properties on the lengths $\ell(x_i, L_j)$.

[1] This integer is not unique only when x_i is minimal in every block of $\mathcal{B}$ for X_i.

▷ Claim 13. The following conditions holds for every $j \in [\beta]$:

1. For every $i \in [t]$, we have $\ell(x_i, L_j) \geqslant i$.
2. For every $x, y \in X$, if $N(x) \cap B_j \subset N(y)$ then $\ell(x, L_j) < \ell(y, L_j)$.

Proof. We prove by induction on i from t to 1 that Condition (1) holds for x_i and that Condition (2) holds for every $x, y \in X_i$. That is obviously the case when $i = t$. Let $i \in [t-1]$ and suppose that Condition (1) holds for every $x_{i+1}, \dots, x_t$ and Condition (2) holds for every $x, y \in X_{i+1}$. Let $j \in [\beta]$ such that $B_j \neq B_{f(i)}$. By construction, we have $\ell(x_i, L_j) = i$ so Condition (1) holds for x_i and B_j. Moreover, we have $N(x_i) \setminus B_{x_i} \subseteq N(y)$ for every $y \in X_i$. As $B_j \neq B_{f(i)}$, it follows that $N(x_i) \cap B_j \subseteq N(y)$ for every $y \in X_i$. The induction hypothesis implies that $\ell(y, L_j) \geqslant i + 1$ for every $y \in X_i$. Hence, $\ell(x_i, L_j) < \ell(y, L_j)$ for every $y \in X_i$ and Condition (2) holds for X_i and every $B_j \neq B_{f(i)}$.

It remains to prove that both Conditions holds for $B_{f(i)}$. We define $\ell(x_i, L_{f(i)})$ either to i, $a + 1$ or $(b + c)/2$ where a, b and c belong to $\{\ell(y, L_{f(i)}) : y \in X_{i+1}\}$. The induction assumption implies that $\ell(y, L_{f(i)}) \geqslant i + 1$ for every $y \in X_{i+1}$. Thus a, b and c greater than or equal to $i + 1$. We deduce that $\ell(x_i, L_{f(i)}) \geqslant i$. This proves that Condition (1) holds for x_i.

If $N(x_i)$ is minimal in $B_{f(i)}$ for X_i, then $\ell(x_i, L_j) < \ell(y, L_j)$ for every $y \in X_i$ and Condition (2) is satisfied. Suppose that $N(x_i)$ is maximal in $B_{f(i)}$ for X_i, i.e. $N(y) \cap B_{f(i)} \subseteq N(x_i)$ for every $y \in X_i$. In this case, we set $\ell(x_i, L_{f(i)})$ is define as $1 + \mathsf{max}\{\ell(y, L_{f(i)}) : y \in X_{i+1}\}$. Thus, we have $\ell(x_i, L_{f(i)}) > \ell(y, L_{f(i)})$ for every $y \in X_{i+1}$ and Condition (2) holds for $\leqslant_{B_{f(i)}}$.

Finally, assume that $N(x_i)$ is neither minimal nor maximal in $B_{f(i)}$ for X_i. If $z^{\mathsf{max}} = z^{\mathsf{min}}$, then we have $\ell(x_i, L_{f(i)}) = \ell(z^{\mathsf{max}}, L_{f(i)})$ and also $N(x_i) \cap B_{f(i)} = N(z^{\mathsf{min}}) \cap B_{f(i)}$. We deduce that Condition (2) holds for every $x, y \in X_i$ because by induction hypothesis we have for every $x, y \in X_{i+1}$ and $z^{\mathsf{min}} \in X_{i+1}$.

Now, assume that $z^{\mathsf{min}} \neq z^{\mathsf{max}}$. The way we choose z^{min} and z^{max} implies that

$$N(z^{\mathsf{min}}) \cap B_{f(i)} \subset N(x_i) \cap B_{f(i)} \subset N(z^{\mathsf{max}}) \cap B_{f(i)}.$$

As $N(z^{\mathsf{min}}) \cap B_{f(i)} \subset N(z^{\mathsf{max}}) \cap B_{f(i)}$, by induction hypothesis, we deduce that $\ell(z^{\mathsf{min}}, L_{f(i)}) < \ell(z^{\mathsf{max}}, L_{f(i)})$ and thus we have

$$\ell(z^{\mathsf{min}}, L_{f(i)}) < \ell(x_i, L_{f(i)}) < \ell(z^{\mathsf{max}}, L_{f(i)})$$

So Condition (2) holds for every $x, y \in \{x_i, z^{\mathsf{min}}, z^{\mathsf{max}}\}$.

Let $y \in X_{i+1}$ such that $N(x_i) \cap B_{f(i)} \subseteq N(y)$. The minimality of $N(z^{\mathsf{max}}) \cap B_{f(i)}$ implies that $N(z^{\mathsf{max}}) \cap B_{f(i)} \subseteq N(y)$. By induction hypothesis, we have $\ell(z^{\mathsf{max}}, L_{f(i)}) \leqslant \ell(y, L_{f(i)})$. Thus for every $y \in X_{i+1}$ such that $N(x_i) \cap B_{f(i)} \subseteq N(y)$, we have $\ell(x_i, L_{f(i)}) < \ell(y, L_{f(i)})$.

Symmetrically, the maximality of z^{min} implies that for every $y \in X_{i+1}$ such that $N(y) \cap B_{f(i)} \subseteq N(x_i)$, we have $\ell(y, L_{f(i)}) < \ell(x_i, L_{f(i)})$. We conclude that Condition (2) holds for every $x, y \in X_i$. By induction, we conclude that Condition (1) holds for every $i \in [t]$ and Condition (2) holds for every $x, y \in X$. This concludes the proof of Claim 13. ◀

Observe that each T_{x_i} is a neighborhood subtree as by construction and Condition (1) of Claim 13 the lengths $\ell(x_i, L_j)$ satisfy the last item of Fact 7. See Figure 5 for an illustration of these neighborhood subtrees.

It remains to construct the neighborhood subtrees of the vertices in $V(G) \setminus X$. Let us explain how we do it for the vertices in B_j for some $j \in [\beta]$. For every $b \in B_j$, we define T_b as an interval of the line L_j. As $(X, \mathcal{B})$ is a good partition, $G[X \cup B_j]$ admits a clique path

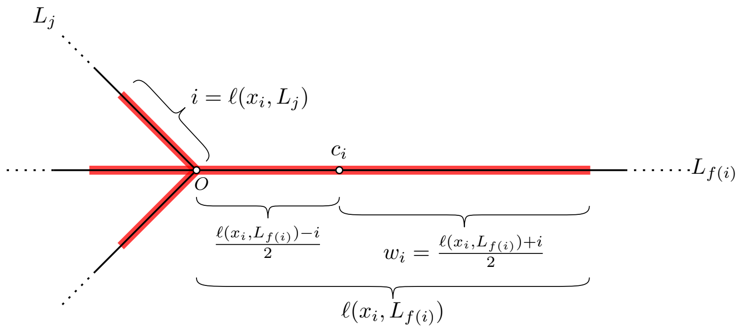

Figure 5 Illustration of the neighborhood subtree T_{x_i} where the number β of edges of the star NeS model is 4.

$(X, K_1, \ldots, K_k)$. Clique path properties implies that $X \supseteq K_1 \cap X \supseteq \cdots \supseteq K_k \cap X$. For every $i \in [k]$, $x \in X \setminus K_i$ and $y \in X \cap K_i$, we have $N(x) \cap B_j \subset N(y)$ and thus $\ell(x, L_j) < \ell(y, L_j)$ thanks to Condition (2) of Claim 13. As $\ell(x, L_j)$ corresponds to the length of the interval $T_x \cap L_j$ for every $x \in X$, we conclude that there exist $p_1, \ldots, p_k$ points on L_j such that

- for every $i \in [k]$, $K_i \cap X$ is exactly the set of vertices x in X such that $p_i \in T_x$ and
- we have $0 < d_{\mathcal{T}}(p_1, c) < d_{\mathcal{T}}(p_2, c) < \cdots < d_{\mathcal{T}}(p_k, c)$.

For every $b \in B_j$ such that b is contained in every clique K_i for i between s and t, we define T_b as the interval of L_j between p_s and p_t.

By construction, $(\mathcal{T}, (T_v)_{v \in V(G)})$ fulfill every property of Fact 7. We deduce that $(\mathcal{T}, (T_v)_{v \in V(G)})$ is a NeS model of G. Obviously, the construction of $(\mathcal{T}, (T_v)_{v \in V(G)})$ can be done in polynomial time. ◀

As mentioned in Fact 6, every graph that admits a star NeS model has a clique-tree that is a subdivided star. The converse is not true. In fact, for every graph G with a clique tree that is a subdivided star with center X, the pair $(X, \mathsf{CC}(G - X))$ satisfies Properties (A) and (B) of Definition 11 but Property (C) might not be satisfied. See for example the graph in Figure 4 and note that the pair $(\{a, b, c\}, \mathsf{CC}(G - \{a, b, c\}))$ does not satisfy Property (C), as after removing the vertex c neither a nor b is removable from $\{a, b\}$.

We now describe Algorithm 1 that decides whether a graph G admits a good partition. Clearly G must be chordal, so we start by checking this. A chordal graph has $O(n)$ maximal cliques, and for each maximal clique X we try to construct a good partition $(X, \mathcal{A})$ of G. We start with $\mathcal{A} \leftarrow \mathsf{CC}(G - X)$ and note that $(X, \mathcal{A})$ trivially satisfies Property (A) of Definition 11. Moreover, if G admits a good partition with central clique X, then $(X, \mathcal{A})$ must satisfy Property (B) of Definition 11, and we check this in Line 3.

Then, Algorithm 1 iteratively in a while loop tries to construct a good permutation of X, while possibly merging some blocks of $\mathcal{A}$ along the way so that it satisfies Property (C), or discover that there is no good partition with central clique X. For the graph in Figure 4, with $X = \{a, b, c\}$ the first iteration of the while loop will succeed and set $w_1 = c$, but in the second iteration neither a nor b satisfy the condition of Line 6. At the start of the i-th iteration of the while loop the algorithm has already removed the vertices $w_1, w_2, ..., w_{i-1}$ and it remains to remove the vertices of $W \subseteq X$. According to Definition 10 the next vertex w_i to be removed should have $N(w_i)$ non-minimal for W in at most one block of the final good partition. However, the neighborhood $N(w_i)$ may be non-minimal for W in several blocks of the current partition $\mathcal{A}$, since these blocks may be (unions of) separate components of $G \setminus X$ that should live on the same line of a star NeS model and thus actually be a single

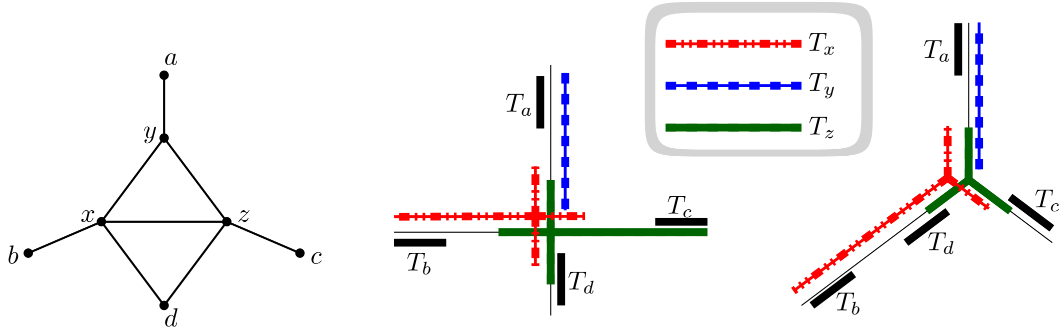

Figure 6 The leftmost figure shows a graph G, with a maximal clique $X = \{x, y, z\}$ and four singleton components in $G - X$. There are two distinct star NeS models of G, the one in the middle (with four arms) corresponding to the permutations (y, x, z) or (y, z, x) and the one on the right (with three arms) corresponding to the permutation (x, y, z). Note if Algorithm 1 chooses $w_1 = x$, then the components $\{b\}$ and $\{d\}$ will be merged in Line 8. There is also a third model, corresponding to the permutation (z, y, x), which is similar to the one on the right. The last two permutations of X are not good permutations.

block which together with X induces an X-interval graph. An example of this merging is given in Figure 6. The following definition captures the non-minimality.

▶ **Definition 14** (notmin). *For $W \subseteq X$, $x \in W$ and partition $\mathcal{A}$ of $V(G) \setminus X$, we denote by* $\mathsf{notmin}(x, W, \mathcal{A})$ *the union of the blocks $A \in \mathcal{A}$ where $N(x)$ is not minimal in A for W.*

Algorithm 1

Input: A graph G.
Output: A good partition of G or "no".

1 Check if G is chordal and if so, compute its set of maximal cliques $\mathcal{X}$, otherwise **return** no;
2 **for** *every* $X \in \mathcal{X}$ **do**
3   **if** *there exists* $C \in \mathsf{CC}(G - X)$ *such that* $G[X \cup C]$ *is not an* X*-interval graph* **then continue**;
4   $\mathcal{A} \leftarrow \mathsf{CC}(G - X)$, $W \leftarrow X$ and $c \leftarrow 1$;
5   **while** $W \neq \varnothing$ **do**
6   Take $w_c \in W$ such that $G[X \cup \mathsf{notmin}(w_c, W, \mathcal{A})]$ is an X-interval graph;
7   **if** *there is no such vertex* **then break**;
8   Replace the blocks of $\mathcal{A}$ contained in $\mathsf{notmin}(w_c, W, \mathcal{A})$ by $\mathsf{notmin}(w_c, W, \mathcal{A})$;
9     $W \leftarrow W \setminus \{w_c\}$ and $c \leftarrow c + 1$;
10   **end**
11   **if** $W = \varnothing$ **then return** $(X, \mathcal{A})$;
12 **end**
13 **return** no;

As we already argued, when Algorithm 1 starts a while loop, $(X, \mathcal{A})$ satisfies Properties (A) and (B), and it is not hard to argue that in each iteration, for every $i \in [c-1]$, w_i is removable from $X \setminus \{w_1, \ldots, w_{i-1}\}$ for $\mathcal{A}$. Hence when $W = \varnothing$, $(w_1, \ldots, w_{|X|})$ is a good permutation of $(X, \mathcal{A})$ and Property (C) is satisfied as well.

▶ **Lemma 15.** *If Algorithm 1 returns a pair $(X, \mathcal{B})$, then $(X, \mathcal{B})$ is a good partition.*

Proof. Assume that Algorithm 1 returns a pair $(X, \mathcal{B})$. We show that $(X, \mathcal{B})$ satisfies Definition 11. As the algorithm starts with $\mathcal{A} = \mathsf{CC}(G - X)$ and only merge blocks, Property (A) holds. Line 7 guarantees that every block of $\mathsf{CC}(G - X)$ induces with X an X-interval graph. Moreover, when merging blocks then Line 6 checks that the merged block induces with X an X-interval graph. Thus Property (B) holds.

Observe that at every step of the while loop, we have $\mathcal{A} \sqsubseteq \mathcal{B}$ since we only merge blocks of $\mathcal{A}$ and $\mathcal{B}$ is the last value of $\mathcal{A}$. We show that $(w_1, \ldots, w_t)$ is a good permutation of $(X, \mathcal{B})$ where, for each $i \in [t]$, w_i is the vertex chosen by the algorithm at the i-th iteration. Let $i \in [t]$. At the start of the i-th iteration of the while loop, the value of W is $X_i = \{w_i, \ldots, w_t\}$. During this iteration, we merge the blocks of $\mathcal{A}$ contained in $\mathsf{notmin}(w_i, X_i, \mathcal{A})$. Consequently, after the merging, there exists a block A_{w_i} of $\mathcal{A}$ such that $N(w_i)$ is minimum for X_i in every block of $\mathcal{A}$ different from A_{w_i}. As $\mathcal{A} \sqsubseteq \mathcal{B}$, there exists a block B_{w_i} of $\mathcal{B}$ containing A_{w_i}. We deduce that, $N(w_i)$ is minimal for X_i in every block of $\mathcal{B}$ different from B_{w_i}, i.e. w_i is removable from X_i for $\mathcal{B}$. Thus, $(w_1, \ldots, w_t)$ is a good permutation of $(X, \mathcal{B})$ and we conclude that $(X, \mathcal{B})$ is a good partition of G. ◀

To prove the opposite direction, namely that if G has a good partition $(X, \mathcal{B})$ associated with a good permutation $(x_1, \ldots, x_t)$, then Algorithm 1 finds a good partition, we need two lemmata. The easy case is when Algorithm 1 chooses consecutively $w_1 = x_1, \ldots, w_t = x_t$, and we can use Lemma 16 to prove that it will not return no. However, Algorithm 1 does not have this permutation as input and at some step with $W = \{x_i, \ldots, x_t\}$, the algorithm might stop to follow the permutation $(x_1, \ldots, x_t)$ and choose a vertex $w_i = x_j$ with $j > i$ because x_i may not be the only vertex such that $G[X \cup \mathsf{notmin}(x_i, W, \mathcal{A})]$ is an X-interval graph. In Lemma 20 we show that choosing $w_i = x_j$ is then not a mistake as it implies the existence of another good partition and another good permutation that starts with $(x_1, \ldots, x_{i-1}, w_i = x_j)$. See Figure 6 for an example of a very simple graph with several good permutations leading to quite distinct star NeS models.

But first we need some definitions. Given a permutation $P = (x_1, \ldots, x_\ell)$ of a subset of X and $i \in [\ell]$, we define $\mathcal{A}_0^P = \mathsf{CC}(G - X)$ and $\mathcal{A}_i^P$ the partition of $V(G) \setminus X$ obtained from $\mathcal{A}_{i-1}^P$ by merging the blocks contained in $\mathsf{notmin}(x_i, X \setminus \{x_1, \ldots, x_{i-1}\}, \mathcal{A}_{i-1}^P)$. Observe that when Algorithm 1 treats X and we have $w_1 = x_1, \ldots, w_\ell = x_\ell$, then the values of $\mathcal{A}$ are successively $\mathcal{A}_0^P, \ldots, \mathcal{A}_\ell^P$. The following lemma proves that if there exists a good permutation starting with $(x_1, \ldots, x_\ell)$, then there exists a vertex $w_{\ell+1}$ satisfying the condition of Line 6 and Algorithm 1 will not return no at the $\ell + 1$-st iteration. Thus, as long as Algorithm 1 follows a good permutation, it will not return no.

▶ **Lemma 16.** *Let G be a graph with good partition $(X, \mathcal{B})$ and $P = (x_1, \ldots, x_t)$ be a good permutation of $(X, \mathcal{B})$. For every $i \in [t]$, we have $\mathcal{A}_i^P \sqsubseteq \mathcal{B}$ and the graph $G[X \cup \mathsf{notmin}(x_i, \{x_i, \ldots, x_t\}, \mathcal{A}_{i-1}^P)]$ is an X-interval graph.*

Proof. For every $i \in [t]$, we denote $\{x_i, \ldots, x_t\}$ by X_i. We start by proving by induction that, for every $i \in [t]$, we have $\mathcal{A}_i^P \sqsubseteq \mathcal{B}$. This is true for $\mathcal{A}_0^P = \mathsf{CC}(G - X)$ by Property (A) of Definition 11. Let $i \in [t]$ and suppose that $\mathcal{A}_{i-1}^P \sqsubseteq \mathcal{B}$. Every block of $\mathcal{A}_i^P$ different from $\mathsf{notmin}(x_i, X_i, \mathcal{A}_{i-1}^P)$ is a block of $\mathcal{A}_{i-1}^P$ and is included in a block of $\mathcal{B}$ by the induction hypothesis. As x_i is removable from X_i for $\mathcal{B}$, there exists $B_{x_i} \in \mathcal{B}$ such that $N(x_i)$ is minimal for X_i in every block of $\mathcal{B}$ different from B_{x_i}. Since $\mathcal{A}_{i-1}^P \sqsubseteq \mathcal{B}$, for every block $\widehat{A} \in \mathcal{A}_{i-1}^P$ such that $N(x_i)$ is not minimal in $\widehat{A}$ for X_i, we have $\widehat{A} \subseteq B_{x_i}$. As the union of these blocks $\widehat{A}$'s is $\mathsf{notmin}(x_i, X_i, \mathcal{A}_{i-1}^P)$, we deduce that $\mathsf{notmin}(x_i, X_i, \mathcal{A}_{i-1}^P) \subseteq B_{x_i}$. Hence, every block of $\mathcal{A}_i^P$ is included in a block of $\mathcal{B}$, thus by induction $\mathcal{A}_i^P \sqsubseteq \mathcal{B}$ for every $i \in [t]$.

For every $i \in [t]$, we have $\mathsf{notmin}(x_i, X_i, \mathcal{A}^P_{i-1}) \subseteq B_{x_i}$. From Property (B) of Definition 11, $G[X \cup B_{x_i}]$ is an X-interval graph. We conclude that $G[X \cup \mathsf{notmin}(x_i, X_i, \mathcal{A}^P_{i-1})]$ is also an X-interval graph. ◀

To prove the following lemma, we rely on the following relation that will be applied to vertex subsets corresponding to unions of connected components of $G - X$.

▶ **Definition 17.** *For every $A \subseteq V(G) \setminus X$, we define $X\text{-}\mathsf{max}_\cap(A) = N(A) \cap X$ and $X\text{-}\mathsf{min}_\cap(A) = \cap_{a \in A} N(a) \cap X$. Given $A_1, A_2 \subseteq V(G) \setminus X$, we say that $A_1 \leqslant_X A_2$ if $X\text{-}\mathsf{max}_\cap(A_1) \subseteq X\text{-}\mathsf{min}_\cap(A_2)$.*

As an example of the use of this, note that in Figure 6, we have $\{b\} \leqslant_{\{x,y,z\}} \{d\}$ and in the rightmost star NeS model we see this ordering reflected by d being placed closer to the center of the star than b.

For every $A \subseteq V(G) \setminus X$, we have $X\text{-}\mathsf{min}_\cap(A) \subseteq X\text{-}\mathsf{max}_\cap(A)$ and thus $\leqslant_X$ is transitive. The following claim reveals the connection between Definition 17 and Property (B) of Definition 11.

▷ Claim 18. Let $P = (x_1, \dots, x_t)$ be a good permutation of X and $\ell \in \{0, \dots, t\}$. For all distinct blocks $A, A^\star \in \mathcal{A}^P_\ell$, if $G[X \cup A \cup A^\star]$ is an X-interval graph, then A and $A^\star$ are comparable for $\leqslant_X$.

Proof. We prove this claim by induction on ℓ. By definition, we have $\mathcal{A}^P_0 = \mathsf{CC}(G - X)$. Let $C, C^\star$ be two distinct components of $G - X$ such that $G[X \cup C \cup C^\star]$ is an X-interval graph. Let $(K_1, \dots, K_k; X)$ be a clique path of $G[X \cup C \cup C^\star]$. Since C and $C^\star$ are distinct connected components, for every $i \in [k]$, K_i is either a maximal clique of $G[X \cup C]$ or $G[X \cup C^\star]$. Assume w.l.o.g. that K_1 is a maximal clique of $G[X \cup C]$. Moreover, there exists $i \in [k-1]$ such that $X, K_1, K_2, \dots, K_i$ are all the maximal cliques of $G[X \cup C]$ and $X, K_{i+1}, \dots, K_k$ are all the maximal cliques of $G[X \cup C^\star]$. Since $(K_1, \dots, K_k, X)$ is a clique path, we have $K_1 \cap X \subseteq \dots \subseteq K_k \cap X \subseteq X$. From Definition 17, we deduce that $X\text{-}\mathsf{max}_\cap(C) = K_i \cap X$ and $X\text{-}\mathsf{min}_\cap(C^\star) = K_{i+1} \cap X$. As $K_i \cap X \subseteq K_{i+1} \cap X$, we have $X\text{-}\mathsf{max}_\cap(C) \subseteq X\text{-}\mathsf{min}_\cap(C^\star)$ and thus $C \leqslant_X C^\star$ and the claim holds for $\mathcal{A}^P_0$.

Let $\ell \in [t]$ and suppose that the claim holds for $\mathcal{A}^P_{\ell-1}$. In the rest of this proof, we use the shorthand $\mathsf{notmin}(x_\ell)$ to denote $\mathsf{notmin}(x_\ell, \{x_\ell, \dots, x_t\}, \mathcal{A}^P_{\ell-1})$. We obtain $\mathcal{A}^P_\ell$ from $\mathcal{A}^P_{\ell-1}$ by merging the blocks of $\mathcal{A}^P_{\ell-1}$ contained in $\mathsf{notmin}(x_\ell)$. So every block of $\mathcal{A}^P_\ell$ different from $\mathsf{notmin}(x_\ell)$ is also a block of $\mathcal{A}^P_{\ell-1}$. By assumption, the claim holds for every pair of distinct blocks of $\mathcal{A}^P_\ell \setminus \{\mathsf{notmin}(x_\ell)\}$. To prove that it holds for $\mathcal{A}^P_\ell$, it is enough to prove that it holds for every pair $(A, \mathsf{notmin}(x_\ell))$ with $A \neq \mathsf{notmin}(x_\ell)$.

Let $A \in \mathcal{A}^P_\ell$ such that $A \neq \mathsf{notmin}(x_\ell)$ and $G[X \cup A \cup \mathsf{notmin}(x_\ell)]$ is an X-interval graph. We need to prove that $A \leqslant_X \mathsf{notmin}(x_\ell)$ or $\mathsf{notmin}(x_\ell) \leqslant_X A$. Observe that $\mathsf{notmin}(x_\ell)$ is a union of blocks $A_1, \dots, A_k$ of $\mathcal{A}^P_{\ell-1}$. As $G[X \cup A \cup \mathsf{notmin}(x_\ell)]$ is an X-interval graph, also for any two blocks B, C out of $A, A_1, \dots, A_k$ the graph $G[X \cup B \cup C]$ is an X-interval graph, and thus by induction hypothesis, we deduce that every pair of blocks among $A, A_1, \dots, A_k$ is comparable for $\leqslant_X$. Suppose w.l.o.g. that $A_1 \leqslant_X \dots \leqslant_X A_k$.

Assume towards a contradiction that there exists $i \in [k-1]$ such that $A_i \leqslant_X A \leqslant_X A_{i+1}$. By Definition 14, we have

$$X\text{-}\mathsf{max}_\cap(A_i) \subseteq X\text{-}\mathsf{min}_\cap(A) \subseteq X\text{-}\mathsf{max}_\cap(A) \subseteq X\text{-}\mathsf{min}_\cap(A_{i+1}).$$

Since $N(w)$ is not minimal in A_i for X_ℓ, we have $N(w) \cap A_i \neq \varnothing$ and thus $w \in X\text{-}\mathsf{max}_\cap(A_i)$. We deduce that w belongs to $X\text{-}\mathsf{min}_\cap(A)$ and $X\text{-}\mathsf{min}_\cap(A_{i+1})$ and thus $A, A_{i+1} \subseteq N(w)$. As

$N(w)$ is not minimal in A_{i+1} and $A_{i+1} \subseteq N(w)$, we deduce that there exists $y \in X_\ell$ such that $A_{i+1} \not\subseteq N(y)$ which means that $y \notin X\text{-}\mathsf{min}_\cap(A_{i+1})$. Since $X\text{-}\mathsf{max}_\cap(A) \subseteq X\text{-}\mathsf{min}_\cap(A_{i+1})$, we have $y \notin X\text{-}\mathsf{max}_\cap(A)$. Hence, $A \subseteq N(w)$ and $N(y) \cap A = \varnothing$. We conclude that $N(w)$ is not minimal in A for X_ℓ, a contradiction with $A \neq \mathsf{notmin}(x_\ell)$ because $N(w)$ is minimal for X_ℓ in every block of $\mathcal{A}_\ell^P$ different from $\mathsf{notmin}(x_\ell)$.

It follows that $A \leqslant_X A_1$ or $A_k \leqslant_X A$. Since $\mathsf{notmin}(x_\ell) = A_1 \cup \cdots \cup A_k$, we deduce that $A \leqslant_X \mathsf{notmin}(x_\ell)$ or $\mathsf{notmin}(x_\ell) \leqslant_X A$, that is the claim holds for $\mathcal{A}_\ell^P$. By induction, we conclude that the claim is true for every $\ell \in \{0, \ldots, t\}$. ◁

▷ Claim 19. Let $X \subseteq V(G)$ and $\mathcal{A}$ a partition of $V(G) \setminus X$ such that $\mathsf{CC}(G - X) \sqsubseteq \mathcal{A}$. For any $A_1, \ldots, A_k \in \mathcal{A}$ that are pairwise comparable for $\leqslant_X$, if for each $i \in [k]$ $G[X \cup A_i]$ is an X-interval graph, then $G[X \cup A_1 \cup \cdots \cup A_k]$ is an X-interval graph.

Proof. Let $A_1, \ldots, A_k$ be k blocks of $\mathcal{A}$ such that $A_1 \leqslant_X A_2 \leqslant_X \ldots \leqslant_X A_k$. Assume that $G[X \cup A_i]$ is an X-interval graph for each $i \in [k]$.

Since $G[X \cup A_i]$ is an X-interval graph for every $i \in [k]$, $G[X \cup A_i]$ admits a clique path $(K_1^i, \ldots, K_{f(i)}^i, X)$, where $f(i)$ is the number of maximal cliques minus one in $G[X \cup A_i]$. For every $i \in [k]$, we have $K_1^i \cap X \subseteq K_2^i \cap X \subseteq \cdots \subseteq K_{f(i)}^i \cap X$. Moreover, by definition, we have $X\text{-}\mathsf{min}_\cap(A_i) = K_1^i$ and $X\text{-}\mathsf{max}_\cap(A_i) = K_{f(i)}^i$. For every $i \in [k-1]$, since $A_i \leqslant_X A_{i+1}$, we have $X\text{-}\mathsf{max}_\cap(A_i) \subseteq X\text{-}\mathsf{min}_\cap(A_{i+1})$, that is $K_{f(i)}^i \cap X \subseteq K_1^{i+1} \cap X$. Because $\mathsf{CC}(G - X) \sqsubseteq \mathcal{A}$, every maximal clique of $G[X \cup A_1 \cup \cdots \cup A_k]$ different from X is a maximal clique of a unique $G[X \cup A_i]$. We conclude that $(K_1^1, \ldots, K_{f(1)}^1, \ldots, K_1^k, \ldots, K_{f(k)}^k, X)$ is a clique path of $G[X \cup A_1 \cup \cdots \cup A_k]$. ◁

▶ **Lemma 20.** *Let $P = (x_1, \ldots, x_t)$ be a good permutation of X and $\ell \in [t]$. For every $w \in \{x_\ell, \ldots, x_t\}$ such that $G[X \cup \mathsf{notmin}(w, \{x_\ell, \ldots, x_t\}, \mathcal{A}_{\ell-1}^P)]$ is an X-interval graph, there exists a good permutation of X starting with $(x_1, \ldots, x_{\ell-1}, w)$.*

Proof. For every $i \in [t]$, we denote by X_i the set $\{x_i, \ldots, x_t\}$. Let $w \in X_\ell$ such that $G[X \cup \mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)]$ is an X-interval graph. Let $(X, \mathcal{B})$ a good partition of G such that $(x_1, \ldots, x_t)$ is a good permutation of $(X, \mathcal{B})$. If $w = x_p$ is removable from $\mathcal{B}$ for X_ℓ, then $(x_1, \ldots, x_{\ell-1}, w, x_\ell, \ldots, x_{p-1}, x_{p+1}, \ldots, x_t)$ is a good permutation of $(X, \mathcal{B})$ and we are done. In particular, w is removable from $\mathcal{B}$ if $|\mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)| \leqslant 1$. In the following, we assume that $|\mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)| \geqslant 2$.

We construct a good partition $(X, \mathcal{B}_{\mathsf{new}})$ that admits good permutation starting with $(x_1, \ldots, x_{\ell-1}, w)$. Let $A_1, \ldots, A_k$ be the blocks of $\mathcal{A}_{\ell-1}^P$ such that $\mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P) = A_1 \cup A_2 \cup \cdots \cup A_k$. As $G[X \cup \mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)]$ is an X-interval graph, by Claim 18, the blocks $A_1, \ldots, A_k$ are pairwise comparable for $\leqslant_X$. Suppose w.l.o.g. that $A_1 \leqslant_X \ldots \leqslant_X A_k$. In Figure 6 with $\mathsf{notmin}(x, \{x, y, z\}, \{\{a\}, \{b\}, \{c\}, \{d\}\}) = \{b\} \cup \{d\}$ we have $A_1 = \{b\}$ and $A_2 = \{d\}$.

By Lemma 16, we have $\mathcal{A}_{\ell-1}^P \sqsubseteq \mathcal{B}$. Thus, there exists a block B_1 of $\mathcal{B}$ containing A_1 and B_1 is a union of blocks of $\mathcal{A}_{\ell-1}^P$. Let B_1^{max} be the union of all the blocks A of $\mathcal{A}_{\ell-1}^P$ included in B_1 such that A is not contained in $\mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)$ and $A_1 \leqslant_X A$. Note that for every A_i and $A \in B_1^{\mathsf{max}}$, we have $A_i \leqslant_X A$ because otherwise we have $A_1 \leqslant_X A \leqslant_X A_i$ and that implies $A \subseteq \mathsf{notmin}(w, X_\ell, \mathcal{A}_{\ell-1}^P)$ (see the arguments used in the proof of Claim 18).

▷ Claim 21. We can assume that B_1^{max} is empty.

Proof. Suppose that $B_1^{\mathsf{max}} \neq \varnothing$. Let $B_1^\star = B_1 \setminus B_1^{\mathsf{max}}$ and $\mathcal{B}^\star$ the partition obtained from $\mathcal{B}$ by replacing the block B_1 with the blocks B_1^{max} and $B_1^\star$. As B_1^{max} is an union of blocks of

$\mathcal{A}^P_{\ell-1}$ and $\mathcal{A}^P_{\ell-1} \sqsubseteq \mathcal{B}$, by construction, we have $\mathcal{A}^P_{\ell-1} \sqsubseteq \mathcal{B}^\star$. We claim that $(X, \mathcal{B}^\star)$ is a good partition that admits $(x_1, \ldots, x_t)$ as a good permutation, this is sufficient to prove the claim.

As $\mathsf{CC}(G - X) \sqsubseteq \mathcal{A}^P_{\ell-1}$ by definition, we have $\mathsf{CC}(G - X) \sqsubseteq \mathcal{B}^\star$. So Property (A) of Definition 11 is satisfied. As $B_1^\star$ and B_1^{max} are subsets of B_1, we deduce that $G[X \cup B_1^\star]$ and $G[X \cup B_1^{\mathsf{max}}]$ are both X-interval graphs. Thus, Property (B) is satisfied.

It remains to prove that $(x_1, \ldots, x_t)$ is a good permutation of $(X, \mathcal{B}^\star)$. First observe that for every $i \in [\ell - 1]$, by definition, x_i is removable from X_i for $\mathcal{A}^P_i$. Since $\mathcal{A}^P_i \sqsubseteq \mathcal{A}^P_{\ell-1} \sqsubseteq \mathcal{B}^\star$, we deduce that x_i is removable from X_i for $\mathcal{B}^\star$.

Let $i \in \{\ell, \ldots, t\}$. The vertex x_i is not minimal in at most one block of $\mathcal{B}$ for X_i. We claim that $N(x_i)$ is minimal in B_1^{max} for X_i. Since the partition $\mathcal{B}^\star$ is obtained from $\mathcal{B}$ by splitting the block B_1 into $B_1^\star$ and B_1^{max}, this implies that x_i is removable from X_i for $\mathcal{B}^\star$. We prove that $N(x_i)$ is minimal in B_1^{max} for X_i by showing that, for every $x_j \in X_\ell$, we have $B_1^{\mathsf{max}} \subseteq N(x_j)$.

Since $A_1 \subseteq \mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$, we have $N(w) \cap A_1 \neq \varnothing$ and thus $w \in X\text{-}\mathsf{max}_\cap(A_1)$. Let A be a block of $\mathcal{A}^P_{\ell-1}$ contained in B_1^{max}. As $A \leqslant_X A_1$, we deduce that $w \in X\text{-}\mathsf{min}_\cap(A)$, that is $A \subseteq N(w)$. By definition, A is not contained in $\mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$ and thus $N(w)$ is minimal in A for X_ℓ. Consequently, for every $x_j \in X_\ell$, we have $A \subseteq N(x_j)$. As this holds for every $A \subseteq B_1^{\mathsf{max}}$, we deduce that $B_1^{\mathsf{max}} \subseteq N(x_i)$ for every $x_i \in X_\ell$. This ends the proof of Claim 21. ◁

From now, based on Claim 21, we assume that $B_1^{\mathsf{max}} = \varnothing$. We construct $\mathcal{B}_{\mathsf{new}}$ as follows. Recall that $A_1 \subseteq B_1$. We create a new block $\widehat{B}_1 = B_1 \cup A_2 \cup \cdots \cup A_k$, and for every block $B \in \mathcal{B}$ such that $B \neq B_1$, we create a new block $\widehat{B} = B \setminus (A_2 \cup \cdots \cup A_k)$. We define $\mathcal{B}_{\mathsf{new}} = \{\widehat{B}_1\} \cup \{\widehat{B} : B \in \mathcal{B} \setminus \{B_1\} \text{ and } \widehat{B} \neq \varnothing\}$. The construction of $\mathcal{B}_{\mathsf{new}}$ is illustrated in Figure 7.

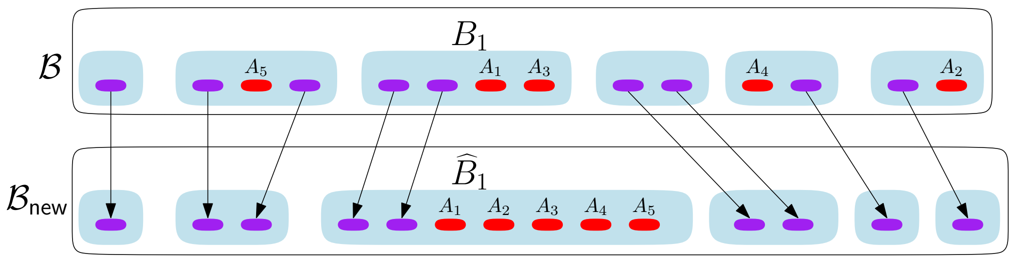

Figure 7 Construction of $\mathcal{B}_{\mathsf{new}}$ from $\mathcal{B}$ with $k = 5$. The blocks of $\mathcal{B}$ and $\mathcal{B}_{\mathsf{new}}$ are in blue, the blocks $A_1, \ldots, A_5$ of $\mathcal{A}^P_{\ell-1}$ contained in $\mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$ are in red, the other blocks of $\mathcal{A}^P_{\ell-1}$ are in purple. In each block of $\mathcal{B}$ or $\mathcal{B}_{\mathsf{new}}$, the blocks of $\mathcal{A}^P_{\ell-1}$ are ordered w.r.t. $\leqslant_X$ from left to right.

We claim that $(X, \mathcal{B}_{\mathsf{new}})$ is a good partition that admits a good permutation starting with $(x_1, \ldots, x_{\ell-1}, w)$. By construction, $\mathcal{B}_{\mathsf{new}}$ is a partition of $V(G) \setminus X$ and $\mathcal{A}^P_{\ell-1} \sqsubseteq \mathcal{B}_{\mathsf{new}}$. As $\mathsf{CC}(G - X) \sqsubseteq \mathcal{A}^P_{\ell-1}$ by definition, we have $\mathsf{CC}(G - X) \sqsubseteq \mathcal{B}_{\mathsf{new}}$. So $(X, \mathcal{B}_{\mathsf{new}})$ satisfies Property (A) of Definition 11.

Observe that for every block $\widehat{B}$ of $\mathcal{B}_{\mathsf{new}}$ such that $\widehat{B} \neq \widehat{B}_1$, $\widehat{B}$ is a subset of the block B of $\mathcal{B}$. Since $(X, \mathcal{B})$ is a good partition, $G[X \cup B]$ is an X-interval graph and hence $G[X \cup \widehat{B}]$ is also an X-interval graph. To prove that Property (B) is satisfied, it remains to show that $G[X \cup \widehat{B}_1]$ is an X-interval graph, for which we will use Claim 19.

Since $G[X \cup B_1]$ is an X-interval graph we know by Claim 18 that the blocks of $\mathcal{A}^P_{\ell-1}$ contained in B_1 are pairwise comparable for $\leqslant_X$. Since $B_1^{\mathsf{max}} = \varnothing$, for every block $A \in \mathcal{A}^P_{\ell-1}$ such that $A \subseteq B_1$ and A is not contained in $\mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$, we have $A \leqslant_X A_1$.

Moreover, we have $A_1 \leqslant_X \ldots \leqslant_X A_k$. We deduce that the blocks of $\mathcal{A}^P_{i-1}$ contained in $\widehat{B}_1$ are pairwise comparable for $\leqslant_X$ because $\leqslant_X$ is transitive. By Claim 19, this implies that $G[X \cup \widehat{B}_1]$ is an X-interval graph. We conclude that Property (B) is satisfied.

It remains to prove that $(X, \mathcal{B}_{\mathsf{new}})$ admits a good permutation starting with $(x_1, \ldots, x_{\ell-1}, w)$. We prove it with the following four claims.

$\triangleright$ Claim 22. For every $i \in [\ell - 1]$, x_i is removable from X_i for $\mathcal{B}_{\mathsf{new}}$ and w is removable from X_ℓ for $\mathcal{B}_{\mathsf{new}}$. Moreover, $N(w)$ is minimal for X_ℓ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_1$.

Proof. For every $i \in [\ell - 1]$, we have $\mathcal{A}^P_i \sqsubseteq \mathcal{A}^P_{\ell-1}$ and x_i is removable from X_i for $\mathcal{A}^P_i$. By construction of $\mathcal{B}_{\mathsf{new}}$, we have $\mathcal{A}^P_{\ell-1} \sqsubseteq \mathcal{B}_{\mathsf{new}}$. We deduce that x_i is removable from X_i for $\mathcal{B}_{\mathsf{new}}$ for every $i \in [\ell - 1]$. By construction of $\mathcal{B}_{\mathsf{new}}$, we have $\mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1}) \subseteq \widehat{B}_1$. Hence, $N(w)$ is minimal for X_ℓ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_1$. It follows that w is removable from X_ℓ for $\mathcal{B}_{\mathsf{new}}$. $\triangleleft$

Let $p \in \{\ell, \ldots, t\}$ such that $x_p = w$.

$\triangleright$ Claim 23. For every $i, j \in [t]$ such that $j > p$ and $\ell \leqslant i < j$, if $N(x_i)$ is minimal in B_1 for X_i, then $N(x_i) \cap \widehat{B}_1 \subseteq N(x_j)$.

Proof. Let $j > p$ and $\ell \leqslant i < j$ such that $N(x_i)$ is minimal in B_1 for X_i. Assume towards a contradiction that $N(x_i) \cap \widehat{B}_1 \not\subseteq N(x_j)$. It follows that $N(x_j) \cap \widehat{B}_1 \subset N(x_j)$ because $G[X \cup \widehat{B}_1]$ is an X-interval graph (these neighborhoods in $\widehat{B}_1$ are comparable). As $N(x_i)$ is minimal in B_1 for X_i and $j \in X_i$, we have[2] $N(x_i) \cap B_1 \subseteq N(x_j)$. So by construction of $\widehat{B}_1$, there exists a block $A \in \mathcal{A}^P_{\ell-1}$ such that $A \subseteq \widehat{B}_1 \setminus B_1$ and $N(x_j) \cap A \subset N(x_j) \cap A$. This implies that $x_j \notin X\text{-}\mathsf{min}_\cap(A)$. Since $A \subseteq \mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$ and $A \not\subseteq B_1$, we have $A_1 \leqslant_X A$ and thus $X\text{-}\mathsf{max}_\cap(A_1) \subseteq X\text{-}\mathsf{min}_\cap(A)$. We deduce that $x_j \notin X\text{-}\mathsf{max}_\cap(A_1)$.

By definition $A_1 \subseteq \mathsf{notmin}(w, X_\ell, \mathcal{A}^P_{\ell-1})$ and thus $w \in X\text{-}\mathsf{max}_\cap(A_1) \subseteq X\text{-}\mathsf{min}_\cap(A)$. Hence, we have $N(x_j) \cap A \subset N(w) \cap A$ and $N(x_j) \cap A_1 \subset N(w) \cap A_1$. Since $j > p$ and $w = x_p$, we have $x_j \in X_p$. Consequently, $N(w)$ is not minimal in A and A_1 for X_p. Let B_A be the block of $\mathcal{B}$ containing A. Since $A_1 \subseteq B_1$ and $A \subseteq B_A$, $N(w)$ is not minimal in B_1 and B_A for X_p. But B_1 and B_A are two distinct blocks of $\mathcal{B}$ because $A \not\subseteq B_1$, this is a contradiction with w being removable from X_p for $\mathcal{B}$. $\triangleleft$

$\triangleright$ Claim 24. For every $i \in \{p + 1, \ldots, t\}$, the vertex x_i is removable from X_i for $\mathcal{B}_{\mathsf{new}}$.

Proof. Let $i \in \{p + 1, \ldots, t\}$. The vertex x_i is removable from X_i from $\mathcal{B}$. So there exists a block $B_{x_i} \in \mathcal{B}$ such that $N(x_i)$ is minimal for X_i in every block of $\mathcal{B}$ different from B_{x_i}. By construction, every block $\widehat{B}$ of $\mathcal{B}_{\mathsf{new}}$ is associated with a block B of $\mathcal{B}$ and if $B \neq B_1$, we have $B \subseteq \widehat{B}$. Thus, for every block $\widehat{B} \in \mathcal{B}_{\mathsf{new}}$, such that $B \neq B_1$ and $B \neq B_{x_i}$, $N(x_i)$ is minimal in $\widehat{B}$ for X_i because $\widehat{B} \subseteq B$ and $N(x_i)$ is minimal in B for X_i. If $B_{x_i} = B_1$, then x_i is removable from X_i for $\mathcal{B}_{\mathsf{new}}$.

Now suppose that $B_{x_i} \neq B_1$. It follows that $N(x_i)$ is minimal for X_i in B_1. As $i > p$, by Claim 23, for every $j > i$, $N(x_i) \cap \widehat{B}_1 \subseteq N(x_j)$. Thus, $N(x_i)$ is minimal in $\widehat{B}_1$ for X_i. Consequently, $N(x_i)$ is minimal in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_{x_i}$. In both cases, x_i is removable from X_i for $\mathcal{B}_{\mathsf{new}}$. $\triangleleft$

It remains to deal with the vertices of $(x_1, \ldots, x_t)$ between x_ℓ and x_{p-1}.

[2] In fact, we have $N(x_i) \cap B_1 = N(x_j) \cap B_1$ since $B_1 \subseteq \widehat{B}_1$ and $N(x_j) \cap \widehat{B}_1 \subset N(x_j)$.

▷ Claim 25. There exists a permutation $(x^\star_{\ell+1}, \ldots, x^\star_p)$ of $\{x_\ell, \ldots, x_{p-1}\}$ such that, for every $i \in \{\ell+1, \ldots, p\}$, $x^\star_i$ is removable from $\{x^\star_i, \ldots, x^\star_p\} \cup X_{p+1}$ for $\mathcal{B}_{\mathsf{new}}$.

Proof. Since $(x_1, \ldots, x_t)$ is a good permutation of $(X, \mathcal{B})$, for every $i \in \{\ell, \ldots, p-1\}$, there exists a block B_{x_i} of $\mathcal{B}$ such that $N(x_i)$ is minimal for X_i in every block of $\mathcal{B}$ different from B_{x_i}. Let $i \in \{\ell, \ldots, p-1\}$. We start by proving that $N(x_i)$ is minimal for X_ℓ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_{x_i}$ and $\widehat{B}_1$. Take $\widehat{B} \in \mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_{x_i}$ and $\widehat{B}_1$. By construction, $\widehat{B}$ is associated with a block B of $\mathcal{B}$. Since $\widehat{B} \neq \widehat{B}_{x_i}$, $N(x_i)$ is minimal in B for X_i. As $i < p$ and $w = x_p$, we have $w \in X_i$ and thus $N(x_i) \cap B \subseteq N(w)$. By Claim 22, $N(w)$ is minimal for X_ℓ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_1$. In particular, $N(w)$ is minimal in $\widehat{B}$ for X_ℓ. As $N(x_i) \cap B \subseteq N(w)$ and $\widehat{B} \subseteq B$, we deduce that $N(x_i)$ is also minimal in $\widehat{B}$ for X_ℓ.

Since $G[X \cup \widehat{B}_1]$ is an X-interval graph, the neighborhoods of X in $\widehat{B}_1$ are pairwise comparable for the inclusion. Thus, there exists a permutation $(x^\star_{\ell+1}, \ldots, x^\star_p)$ of $\{x_\ell, \ldots, x_{p-1}\}$ such that $N(x^\star_{\ell+1}) \cap \widehat{B}_1 \subseteq N(x^\star_{\ell+2}) \cap \widehat{B}_1 \subseteq \cdots \subseteq N(x^\star_p) \cap \widehat{B}_1$.

Let $i \in \{\ell+1, \ldots, p\}$ and $X^\star_i = \{x^\star_i, \ldots, x^\star_p\} \cup X_{p+1}$. We need to show that $x^\star_i$ is removable from $X^\star_i$ for $\mathcal{B}_{\mathsf{new}}$. As $X^\star_i$ is a subset of X_ℓ, from what we have proved above, we know that $N(x^\star_i)$ is minimal for $X^\star_i$ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_1$ and $\widehat{B}_{x^\star_i}$. Thus, if $\widehat{B}_1 = \widehat{B}_{x^\star_i}$, then $x^\star_i$ is removable from $X^\star_i$.

Suppose that $\widehat{B}_1 \neq \widehat{B}_{x^\star_i}$. Then $N(x^\star_i)$ is minimal in $\widehat{B}_{x^\star_i}$ for X_q with $q \in [t]$ such that $x^\star_i = x_q$. By Claim 23, for every $x_j \in X_{p+1}$, we have $N(x^\star_i) \cap \widehat{B}_1 \subseteq N(x_j)$. Since $N(x^\star_i) \cap \widehat{B}_1 \subseteq N(x^\star_{i+1}) \cap \widehat{B}_1 \subseteq \cdots \subseteq N(x^\star_p) \cap \widehat{B}_1$ and $X^\star_i = \{x^\star_i, \ldots, x^\star_p\} \cup X_{p+1}$, $N(x_i)$ is minimal for $X^\star_i$ in $\widehat{B}_1$. We deduce that $N(x^\star_i)$ is minimal for $X^\star_i$ in every block of $\mathcal{B}_{\mathsf{new}}$ different from $\widehat{B}_{x^\star_i}$. Thus, $x^\star_i$ is removable from $X^\star_i$ for $\mathcal{B}_{\mathsf{new}}$. ◁

Let $(x^\star_{\ell+1}, \ldots, x^\star_p)$ be the permutation of $\{x_\ell, \ldots, x_{p-1}\}$ satisfying the condition of Claim 25. Take $P_{\mathsf{new}} = (x_1, \ldots, x_{\ell-1}, w, x^\star_{\ell+1}, \ldots, x^\star_p, x_{p+1}, \ldots, x_t)$, from Claims 22, 24 and 25, we deduce that P_{new} is a good permutation of $(X, \mathcal{B}_{\mathsf{new}})$. Hence, $(X, \mathcal{B}_{\mathsf{new}})$ satisfies Property (C) of Definition 11 and $(X, \mathcal{B}_{\mathsf{new}})$ is a good partition of G. As P_{new} starts with $(x_1, \ldots, x_{\ell-1}, w)$, this concludes the proof of Lemma 20. ◀

▶ **Lemma 26.** *If G has a good partition then Algorithm 1 returns one.*

Proof. Suppose that G admits a good partition with central clique X. We prove that the following invariant holds after the i-th iteration of the while loop for X.

Invariant. G admits a good permutation starting with $w_1, ..., w_i$.

By assumption, X admits a good permutation and thus the invariant holds before the algorithm starts the first iteration of the while loop. By induction, assume the invariant holds when Algorithm 1 starts the i-th iteration of the while loop. Let $L = (w_1, \ldots, w_{i-1})$ be the consecutive vertices chosen at Line 6 before the start of iteration i (observe that L is empty when $i = 1$). The invariant implies that there exists a good permutation $P = (w_1, \ldots, w_{i-1}, x_i, x_{i+1}, ..., x_t)$ of G starting with L. By Lemma 16, the graph $G[X \cup \mathsf{notmin}(x_i, \{x_i, \ldots, x_t\}, \mathcal{A}^P_{i-1})]$ is an X-interval graph. Observe that $\mathcal{A}^P_{i-1}$ and $\{x_i, \ldots, x_t\}$ are the values of the variables $\mathcal{A}$ and W when Algorithm 1 starts the i-th iteration. Thus, at the start of the i-th iteration, there exists a vertex w_i such that $G[X \cup \mathsf{notmin}(w_i, W, \mathcal{A})]$ is an X-interval graph. Consequently, the algorithm does not return no at the i-th iteration and chooses a vertex $w_i \in \{x_i, \ldots, x_t\}$ such that $G[X \cup \mathsf{notmin}(w_i, \{x_i, \ldots, x_t\}, \mathcal{A}^P_i)]$ is an X-interval graph. By Lemma 20, G admits a good permutation starting with $(w_1, \ldots, w_{i-1}, w_i)$. Thus, the invariant holds at the end of the i-th iteration. If at the end of the i-th iteration

W is empty, then the while loop stops and the algorithm returns a pair $(X, \mathcal{A})$. Otherwise, the algorithm starts an $i+1$-st iteration and the invariant holds at the start of this new iteration. By induction, we conclude that the invariant holds at every step and Algorithm 1 returns a pair $(X, \mathcal{A})$ which is a good partition by Lemma 15. ◀

▶ **Theorem 27.** *Algorithm 1 decides in polynomial time whether G admits a star NeS model.*

Proof. Theorem 12 shows that G admits a star NeS model if and only if G admits a good partition, and Lemmata 15 and 26 show that Algorithm 1 finds a good partition if and only the input graph has a good partition. It remains to argue for the polynomial time. Checking that G is chordal and finding the $O(n)$ maximal cliques can be done in polynomial time [17, 24]. Given $X, Y \subseteq V(G)$ we check in polynomial time whether $G[X \cup Y]$ is an X-interval graph, as follows. Take G' the graph obtained from $G[X \cup Y]$ by adding two new vertices u and v such that $N(u) = \{v\}$ and $N(v) = \{u\} \cup X$. It is easy to see that $G[X \cup Y]$ is an X-interval graph if and only if G' is an interval graph, which can be checked in polynomial time. ◀

6 Conclusion

The question if leaf powers can be recognized in polynomial time was raised over 20 years ago [23], and this problem related to phylogenetic trees still remains wide open. We have shown that polynomial-time recognition can be achieved if the weighted leaf root is required to be a caterpillar (using the connection to blue-red interval graphs and co-TT graphs) or if the NeS model has a single large-degree branching.

We strongly believe our results could be combined to recognize in polynomial time the graphs with a leaf root where the internal vertices induce a subdivided star. For space reasons we do not elaborate on this. Let us instead sketch how the techniques in this paper could be generalized to handle the general case of leaf powers. Given a NeS model $\mathcal{N} = (\mathcal{T}, (T_v)_{v \in V(G)})$ of a graph G with $\mathcal{T}$ the embedding of a tree T, we define the *topology* of $\mathcal{N}$ as the pair $(T', (X_u)_{u \in V(T')})$ where T' is the subtree induced by the internal nodes of T and for each node u of T', we define $X_u = \{v \in V(G) : u \in T_v\}$, which will be a clique of G. For a star NeS model, we have T' being a single node and we proved that (1) if a graph admits a star NeS model, then it admits one where the only clique in the topology is maximal and (2) given a topology with T' a single node, we can decide in polynomial time whether the graph admits a star NeS model with this topology. For the general case it would suffice to solve the following problems:

- Problem 1: given a graph G compute in polynomial time a polynomial-sized family $\mathcal{F}$ of topologies such that G is a leaf power if and only if G admits a NeS model with topology in $\mathcal{F}$. A first step towards solving this problem would be to generalize Claim 8 to show that we can always modify a NeS model so that all the cliques of its topology become maximal cliques (or minimal separators, as these are also manageable in chordal graphs). A second step is to study the notion of *clique arrangements* $\mathcal{A}(G)$, introduced by Nevries and Rosenke [21], which describes the intersections between the maximal cliques of a chordal graph G more precisely than the clique trees, with the aim of showing a connection between $\mathcal{A}(G)$ and a family $\mathcal{F}$ of topologies of NeS models. For chordal graphs $\mathcal{A}(G)$ can have exponential size but for strongly chordal graphs, and hence leaf powers, Nevries and Rosenke show a polynomial upper bound. We can in polynomial time decide if a graph G is strongly chordal [14] (if not then G is not a leaf power) and compute $\mathcal{A}(G)$ [21].

- Problem 2: given a graph and a topology, construct a NeS model with this topology or confirm that no such NeS model exists, in polynomial time. For star NeS models, this is handled inside the for loop of Algorithm 1 and relies on Definition 11. A step towards solving this problem would be to generalize the combinatorial definition of a good partition to handle any topology.

Inspired by blue-red interval graphs, we can define blue-red NeS models where the red vertices induce an independent set and for every red vertex v its neighbors are the blue vertices u such that $T_v \subseteq T_u$. We can adapt the proof of Theorem 3 to show that G admits a NeS model iff G has a blue-red NeS model where the blue vertices are simplicial vertices without true twins, so these have the same modeling power. However, for blue-red NeS models the necessary topologies can be significantly simpler. For example, NeS models of linear leaf powers are caterpillars, whose topologies are paths, while its blue-red NeS models can be restricted to an edge, whose topology is empty. Can we generalize the concept of blue-red NeS models so that we can recursively color more vertices blue (by allowing some of them to be adjacent under some conditions) and thereby simplify the topologies we need to consider to solve Problems 1 and 2?

References

1 Eric Bibelnieks and Perino M. Dearing. Neighborhood subtree tolerance graphs. *Discret. Appl. Math.*, 43(1):13–26, 1993. doi:10.1016/0166-218X(93)90165-K.

2 Andreas Brandstädt and Christian Hundt. Ptolemaic graphs and interval graphs are leaf powers. In *LATIN 2008: Theoretical Informatics, 8th Latin American Symposium, Búzios, Brazil, April 7-11, 2008, Proceedings*, volume 4957 of *Lecture Notes in Computer Science*, pages 479–491. Springer, 2008. doi:10.1007/978-3-540-78773-0_42.

3 Andreas Brandstädt, Christian Hundt, Federico Mancini, and Peter Wagner. Rooted directed path graphs are leaf powers. *Discret. Math.*, 310(4):897–910, 2010. doi:10.1016/j.disc.2009.10.006.

4 Andreas Brandstädt and Van Bang Le. Structure and linear time recognition of 3-leaf powers. *Inf. Process. Lett.*, 98(4):133–138, 2006. doi:10.1016/j.ipl.2006.01.004.

5 Andreas Brandstädt, Van Bang Le, and R. Sritharan. Structure and linear-time recognition of 4-leaf powers. *ACM Trans. Algorithms*, 5(1):11:1–11:22, 2008. doi:10.1145/1435375.1435386.

6 Tiziana Calamoneri, Antonio Frangioni, and Blerina Sinaimeri. Pairwise compatibility graphs of caterpillars. *Comput. J.*, 57(11):1616–1623, 2014. doi:10.1093/comjnl/bxt068.

7 Tiziana Calamoneri and Blerina Sinaimeri. Pairwise compatibility graphs: A survey. *SIAM Rev.*, 58(3):445–460, 2016. doi:10.1137/140978053.

8 Maw-Shang Chang and Ming-Tat Ko. The 3-steiner root problem. In Andreas Brandstädt, Dieter Kratsch, and Haiko Müller, editors, *Graph-Theoretic Concepts in Computer Science, 33rd International Workshop, WG 2007, Dornburg, Germany, June 21-23, 2007. Revised Papers*, volume 4769 of *Lecture Notes in Computer Science*, pages 109–120. Springer, 2007. doi:10.1007/978-3-540-74839-7_11.

9 Elias Dahlhaus. On strongly chordal graphs. *Ars Combin., B*, 24:23–30, 1987.

10 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

11 Michael Dom, Jiong Guo, Falk Hüffner, and Rolf Niedermeier. Extending the tractability border for closest leaf powers. In Dieter Kratsch, editor, *Graph-Theoretic Concepts in Computer Science, 31st International Workshop, WG 2005, Metz, France, June 23-25, 2005, Revised Selected Papers*, volume 3787 of *Lecture Notes in Computer Science*, pages 397–408. Springer, 2005. doi:10.1007/11604686_35.

12 Guillaume Ducoffe. The 4-steiner root problem. In Ignasi Sau and Dimitrios M. Thilikos, editors, *Graph-Theoretic Concepts in Computer Science - 45th International Workshop, WG*

2019, Vall de Núria, Spain, June 19-21, 2019, Revised Papers, volume 11789 of *Lecture Notes in Computer Science*, pages 14–26. Springer, 2019. doi:10.1007/978-3-030-30786-8_2.

13 David Eppstein and Elham Havvaei. Parameterized leaf power recognition via embedding into graph products. *Algorithmica*, 82(8):2337–2359, 2020. doi:10.1007/s00453-020-00720-8.

14 Martin Farber. Characterizations of strongly chordal graphs. *Discret. Math.*, 43(2-3):173–189, 1983. doi:10.1016/0012-365X(83)90154-1.

15 Petr A. Golovach, Pinar Heggernes, Nathan Lindzey, Ross M. McConnell, Vinícius Fernandes dos Santos, Jeremy P. Spinrad, and Jayme Luiz Szwarcfiter. On recognition of threshold tolerance graphs and their complements. *Discret. Appl. Math.*, 216:171–180, 2017. doi:10.1016/j.dam.2015.01.034.

16 Martin Charles Golumbic, Nirit Lefel Weingarten, and Vincent Limouzy. Co-tt graphs and a characterization of split co-tt graphs. *Discret. Appl. Math.*, 165:168–174, 2014. doi:10.1016/j.dam.2012.11.014.

17 Michel Habib, Ross M. McConnell, Christophe Paul, and Laurent Viennot. Lex-bfs and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theor. Comput. Sci.*, 234(1-2):59–84, 2000. doi:10.1016/S0304-3975(97)00241-7.

18 Lars Jaffke, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. Mim-width III. graph powers and generalized distance domination problems. *Theor. Comput. Sci.*, 796:216–236, 2019. doi:10.1016/j.tcs.2019.09.012.

19 Manuel Lafond. On strongly chordal graphs that are not leaf powers. In *Graph-Theoretic Concepts in Computer Science - 43rd International Workshop, WG 2017, Eindhoven, The Netherlands, June 21-23, 2017, Revised Selected Papers*, pages 386–398, 2017. doi:10.1007/978-3-319-68705-6_29.

20 Clyde L. Monma, Bruce A. Reed, and William T. Trotter. Threshold tolerance graphs. *J. Graph Theory*, 12(3):343–362, 1988. doi:10.1002/jgt.3190120307.

21 Ragnar Nevries and Christian Rosenke. Characterizing and computing the structure of clique intersections in strongly chordal graphs. *Discrete Applied Mathematics*, 181:221–234, 2015. URL: https://www.sciencedirect.com/science/article/pii/S0166218X14003941, doi:https://doi.org/10.1016/j.dam.2014.09.003.

22 Ragnar Nevries and Christian Rosenke. Towards a characterization of leaf powers by clique arrangements. *Graphs Comb.*, 32(5):2053–2077, 2016. doi:10.1007/s00373-016-1707-x.

23 Naomi Nishimura, Prabhakar Ragde, and Dimitrios M. Thilikos. On graph powers for leaf-labeled trees. *J. Algorithms*, 42(1):69–108, 2002. doi:10.1006/jagm.2001.1195.

24 Donald J. Rose, Robert Endre Tarjan, and George S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM J. Comput.*, 5(2):266–283, 1976. doi:10.1137/0205021.