Abstract
We define treetopes, a generalization of the three-dimensional roofless polyhedra (Halin graphs) to arbitrary dimensions. Like roofless polyhedra, treetopes have a designated base facet which intersects every face of dimension greater than one in more than one point. We prove an equivalent characterization of the 4-treetopes using the concept of clustered planarity from graph drawing, and we use this characterization to recognize the graphs of 4-treetopes in polynomial time. This result provides one of the first classes of 4-polytopes, other than pyramids and stacked polytopes, that can be recognized efficiently from their graphs. Additionally we show that every d-dimensional treetope (with \(d \geq 3\)) has at most \(d + 1\) base facets, and that despite not having any forbidden minors the 4-treetopes obey a separator theorem like the one for planar graphs.

Keywords Polytope · Skeleton · Tree · Halin graph · Clustered planarity

Mathematics Subject Classification 52B12 · 05C05 · 05C62 · 68W40

1 Introduction

The three-dimensional convex polyhedra that have a special base facet that shares an edge with all other facets were studied already by Kirkman in [28]. They have
been called based polyhedra \cite{33}, roofless polyhedra \cite{8}, or domes \cite{11}. Their graphs are the Halin graphs, the graphs formed from a planar embedding of a tree without degree-two vertices by adding a cycle that connects the leaves of the tree in the order given by the embedding \cite{8,13,18,23}. In this paper we consider the question of how to generalize this class of polytopes to higher dimensions.

We answer this question by introducing a new class of convex polytopes, again defined in terms of a designated base facet, with the property that every face of dimension greater than one intersects the base in more than one point.\footnote{Throughout this paper we will only consider convex polyhedra and convex polytopes, so henceforth we omit the word “convex”.} We call these polytopes treetopes, because the edges of the polytope that do not lie within the base must form a tree. In three dimensions, the 3-treetopes are exactly the polyhedral realizations of Halin graphs.

We provide the following results:

\begin{itemize}
  \item We investigate the general properties of treetopes. We prove that the edges that do not lie within the base form a tree, we bound the number of faces of a treetope, and we bound the number of faces that can be chosen as the base of a single polytope.
  \item We provide a combinatorial characterization of treetopes by relating them clustered planarity. In this problem, a planar graph is given together with a hierarchical clustering on its vertices. It must be drawn representing the clusters as Jordan curves that surround their cluster, with no edge–edge or cluster–cluster crossings and no unnecessary edge–cluster crossings \cite{9,17}. After much previous work, a recent breakthrough has found a polynomial time algorithm for testing clustered planarity \cite{20}. We define a restricted type of clustering of planar graphs, which we call a well-connected clustering, and we show that the graphs formed by the vertices and edges of 4-treetopes are exactly the graphs that can be formed from a well-connected clustering by adding an additional vertex for each cluster.
  \item Based on the characterization in terms of clustered planarity, we also characterize the 4-treetope graphs in terms of certain contraction and expansion operations (replacing a cluster by a single vertex or the inverse operation), and we use these operations to build a realization of any given 4-treetope.
  \item We describe a polynomial-time recognition algorithm for the graphs of 4-treetopes, which uses only the graph structure and not its geometry to find a valid sequence of the same contraction operations.
  \item We investigate the sparsity properties of the graphs of 4-treetopes. We show that these graphs (as embedded in the geometry of a treetope) contain arbitrary knots and links in 3-dimensional space, and that they do not have any forbidden minors. Nevertheless, we show that they obey a separator theorem like the separator theorem for planar graphs.
\end{itemize}

\subsection*{1.1 Related Work}

Already in their 1969 survey on polytopes, Grünbaum and Shephard considered the problem of characterizing the graphs of polytopes, and noted that most of the partial characterizations of the face complexes of polytopes actually concerned their
graphs [22]. For instance, Steinitz’s theorem [38] characterizes the graphs of three-dimensional polyhedra as the 3-vertex-connected planar graphs. Based on this result, and a long line of research on algorithmic planarity testing [5–7,10,25,36], one can recognize the graphs of 3-polyhedra in linear time. However, recognizing the face lattice of a higher-dimensional polytope is complete for the existential theory of the reals, even for dimension four [34]. This puts the problem in a complexity class that, although solvable in polynomial space, is at least as hard as the NP-complete problems [35], and strongly suggests that recognition of the graphs of polytopes is also hard. In response, we can search for special classes of polytopes whose recognition problem is easier:

- The graphs of four-dimensional pyramids are *apex graphs*, graphs that can be made planar by deleting one vertex. They may be recognized by searching for a universal vertex (one whose degree is one less than the total number of vertices), deleting this vertex, and testing planarity and 3-connectivity of the remaining graph. Apex graphs in which the apex is not necessarily universal may also be recognized efficiently [27], although these graphs may not be polytopal.

- The *d*-dimensional *stacked polytopes* are formed by gluing simplices together on shared faces. Their graphs are \((d + 1)\)-trees in which each \(d\)-clique is a subgraph of at most two \((d + 1)\)-cliques [29]. This characterization allows these polytopes to be recognized in polynomial time regardless of dimension.

- It is also possible to recognize the graphs of a class of generalized prisms, formed as the Cartesian products of any number of line segments, polygons, and three-dimensional polyhedra, in polynomial time [16,26].

Beyond these special cases and their combinations, very little is known. Friedman [19] showed that, given the graph of a simple polytope (a polytope with vertex degree equal to its dimension), one can in polynomial time reconstruct the entire face structure of the polytope; however, even this result does not allow us to distinguish the graphs of simple polytopes from other \(d\)-regular \(d\)-connected graphs. Set in this context, the 4-treetopes studied here are unusual in having a polynomial-time recognition algorithm.

Recently, we made a more general study of the polytopes in which the faces disjoint from a base facet all have bounded dimension [15]. As we observed, when the dimension bound is one, the faces that are disjoint from the base form a tree. However, there exist polytopes that meet this definition but are not treetopes (Fig. 1).

### 1.2 Organization

The rest of this paper is organized as follows. In Sect. 2 we define treetopes, and in Sects. 3 and 4 we prove some structural properties of their face lattices and their graphs that can be stated independently of their dimension. In Sect. 5 we investigate the uniqueness of the choice of base facet of a treetope. In Sect. 6 we turn to clustered planar graphs. We define the cluster graph of a hierarchically clustered graph (a graph augmented by adding a new vertex for each cluster). We also define well-connected clusterings (clusterings that obey certain graph-theoretic properties analogous to the properties of treetopes), and we prove that these clusterings may be obtained by a sequence of operations in which we replace a single vertex of a graph by a new cluster. In Sect. 7 we show how to realize each such expansion operation geometrically,
proving that the cluster graphs of well-connected clusterings are exactly the graphs of 4-treetopes. In Sect. 8 we use the clustering-based characterization of these graphs to develop an algorithm for recognizing these graphs in polynomial time. Finally, in Sect. 9 we discuss the sparsity and minor-containment properties of the graphs of 4-treetopes and of certain related clustered planar drawings.

2 Definitions

The following definitions are standard.

Definition 2.1 A polytope is the convex hull of a finite set $S$ of points in a Euclidean space. The faces of a polytope are its intersections with halfspaces whose boundaries are disjoint from the relative interior of the polytope. They form a lattice by inclusion, in which the bottom element is the empty set and the top element is the polytope itself. The dimension $\dim F$ of a face $F$ is one less than the minimum number of points of $S$ whose affine hull contains the face. The empty face has dimension $-1$, and the faces of dimension zero (vertices) are a subset of $S$. The edges of a polytope, its faces of dimension one, are line segments, and the edges and vertices together form an undirected graph, the graph or 1-skeleton of the polytope. The facets of a $d$-dimensional polytope are its faces of dimension $d-1$, and the ridges are the faces of dimension $d-2$.

The next definition is new, and our main object of study.

Definition 2.2 We define a treetope to be a polytope in which every face of dimension greater than one intersects a distinguished base facet $B$ in more than one point. A $k$-treetope is a treetope of dimension $k$.

We will be particularly interested in 4-treetopes. Every convex polygon is a 2-treetope. The 3-treetopes are the polyhedra whose graphs are the Halin graphs. Treetopes with arbitrary dimension $d$ include the pyramids over $(d-1)$-dimensional bases, polytopes formed by the convex hull of the base with one more vertex (the apex) that is affinely independent of the base. Figure 2, left, shows a pyramid whose base is a cube, projected into a three-dimensional Schlegel diagram in which the apex is shown as the point in the center of the cube. In a pyramid, every face of dimension greater than one contains two or more base vertices, because there is only one non-base vertex to include, so these shapes necessarily meet the definition of a treetope.
If $P$ is a pyramid with base $B$, then the prism over $P$ (the Cartesian product $P \times [0, 1]$) is a treetope with base $B \times [0, 1]$. However, prisms over treetopes may not be treetopes. For instance, like every polygon, a square is a treetope but its prism, the cube, is not.

**Definition 2.3** We say that a treetope is in *general position* if no two vertices have equal nonzero distance from its base.

For the purposes of understanding the combinatorial structure of treetopes we may assume without loss of generality that any treetope is in general position. For if not, it can be perturbed into general position by a projective transformation that does not change its combinatorial structure.

### 3 Face Structure

In this section we examine the face structure of treetopes. As we will show, the edges that do not lie in the base form a tree, justifying the name. This tree also has a close connection to the rest of the faces.

**Definition 3.1** If $v$ is any vertex of a treetope $P$, we define a *parent* of $v$ to be a vertex adjacent to $v$ in the graph of $P$ and farther than $v$ from the base hyperplane of $P$, and we define a *root* of $P$ to be a vertex with no parent. (Shown for a 3-treetope in Fig. 3.)

**Definition 3.2** Let $v$ be a vertex of a $d$-dimensional polytope. Then the *link* of $v$ is the $(d-1)$-dimensional polytope formed by intersecting $P$ with any hyperplane that is disjoint from $v$ but passes through the interior of $P$ near enough to $v$ so that all other vertices of $P$ are on the other side of the hyperplane. The precise geometry of the link depends on the choice of hyperplane, but its combinatorial structure does not.

**Lemma 3.3** For a treetope in general position, every vertex has at most one parent and there is exactly one root.
Proof Let $P$ be a given treetope with base $B$. The fact that there is only one root follows from the simplex method in linear programming, which can be used to find the maximum of any linear function (such as the function mapping each point in $P$ to its distance from the hyperplane of $B$) by following a path in the graph of $P$ along which the function is monotonically increasing. The only vertex that can be a root is the maximum of $f$ (unique by the assumption of general position), because for any other vertex the simplex method will find a parent edge as the first edge of its path.

To see that each vertex $v$ can have at most one parent, consider the link of $v$. Each vertex of the link corresponds to an edge incident to $v$, and for each such edge all points on the edge lie above $v$ or below $v$ with respect to $f$. Therefore, each vertex of the link may be seen as being above $v$ or below $v$, without regard to the specific hyperplane near $v$ that was used to form the link. Similarly, there are three cases for each face of the link: a face may be entirely above $v$, it may be entirely below $v$, or it may contain vertices of the link that are both above and below $v$. By the same simplex-algorithm argument the faces of the link that are above $v$ with respect to $f$ form a connected complex. If $v$ could have more than one increasing edge, this complex would have more than one vertex, and hence would have at least one edge. This edge in the link would necessarily correspond to a two-dimensional face $F$ of $P$ within which $v$ is the
unique minimum point of $f$. But then $F \cap B$ would either equal $v$ (if $v$ belongs to $B$) or be empty (otherwise), contradicting the assumption that no faces of dimension two or more intersect $B$ in at most one point.

Corollary 3.4 Let $P$ be a treetope of dimension $d$ with base $B$, and let $T$ be the set of faces of $P$ that intersect $B$ in at most one point. Then $T$ is an unrooted tree, each leaf of $T$ lies in $B$, and each non-leaf of $T$ has degree at least $d$.

Proof The fact that $T$ is a tree follows from Lemma 3.3. The claim about the degrees of the non-leaf vertices of $T$ follows from the fact that the vertex figure of any vertex in a $d$-polytope is a $(d-1)$-polytope, which necessarily has at least $d$ vertices.

Definition 3.5 For a treetope $P$ with base $B$ and tree $T$ defined as above, we call $T$ the canopy of $P$.

Lemma 3.6 Let $P$ be a treetope with base $B$. Then for every face $F$ of $P$ of dimension greater than one, either $F \subset B$ or $F$ is a treetope with base $F \cap B$. In particular, $\dim(F \cap B) \geq \dim F - 1$.

Proof We prove first the claim about the dimension. By the definition of treetopes, $F \cap B$ contains at least two vertices; let $v$ be one such vertex. Then $F$ lies within the positive hull of the edges of $F$ incident to $v$. By Lemma 3.3 all but at most one of those edges lies within $F \cap B$, so the dimension of the positive hull (and therefore of $F$) is at most one more than the dimension of $F \cap B$.

Now suppose that $F$ is not a subset of $B$. Then, by the dimension claim, $F \cap B$ is a facet of $F$. Let $F'$ be a face of $F$ such that $F' \cap (F \cap B)$ is a single vertex. Then by associativity $(F' \cap F) \cap B = F' \cap B$ is the same single vertex, and by the assumption that $P$ is a treetope $F'$ has dimension at most one. Thus, the faces of $F$ have the defining property of treetopes.

For instance, every non-base facet of a 4-treetope must be a roofless polyhedron.

Lemma 3.7 Let $P$ be a treetope with base $B$, and let $F$ be a nonempty face of $B$. Then there is exactly one face $F'$ of $P$ such that $F' \neq F$ and $F' \cap B = F$.

Proof A face $F'$ meeting the description of the property may be found by starting with $F' = P$ and then, as long as the intersection of $F'$ with $B$ is of too large a dimension, replacing $F'$ by one of its facets, choosing the replacement at each step to be any facet of $F'$ that contains $F$ and is not contained in $B$. Prior to each step, $F$ is not a facet of $F'$, so it is necessarily contained in at least two facets of $F'$, at least one of which is not contained in $B$. Therefore, each step is possible. Each such step reduces the dimension of the intersection with $B$ by one unit, by Lemma 3.6, so it is not possible for this sequence of steps to skip over $F$.

Let $F'$ be any such face, and let $v$ be any vertex of $F$. Then $F'$ must lie in the affine hull of $F$ and the parent edge of $v$. The dimension of this affine hull equals the dimension of $F'$, so it equals the affine hull of $F'$ itself. However, any two different faces of a polytope must have different affine hulls, so $F'$ is the unique face satisfying the properties required by the lemma.
Definition 3.8 For the faces $F$ and $F'$ of Lemma 3.7, we say that $F$ is the base of $F'$ and that $F'$ is the lift of $F$.

Lemma 3.9 Let $F'$ be the lift of a face $F$ in a treetope $P$ with canopy $T$, such that the dimension of $F'$ is greater than one. Choose a root for $T$ at a leaf vertex that does not belong to $F$, and let $a$ be the lowest common ancestor in $T$ of the vertices of $F$. Then the canopy of $F'$ is the union of the paths in $T$ between $a$ and the vertices of $F$.

Proof By Lemma 3.6, $F'$ is itself a treetope. Every edge of the canopy of $F'$ belongs to a 2-face of $F'$, so the canopy of $F'$ equals the union of the canopies of the 2-faces of $F'$. By Lemma 3.7, every 2-face is the lift of an edge $uv$ of $F$. The canopy of this 2-face consists of the unique path in $T$ from $u$ to $v$. This path belongs to the union of paths described in the lemma, so the canopy of $F'$ is a subset of the union of paths.

To show that it equals the union of paths, let $v$ be any vertex of $F$ and $e$ be any edge on the path from $v$ to $a$. Thus, $e$ is an arbitrary edge in the union of paths, and we must show that $e$ belongs to the canopy of $F'$. Let $w$ be an arbitrary descendant of $a$ in $F$ through a different child than the one leading to $v$. Then the path in $T$ from $v$ to $w$ passes through $e$. However, $vw$ might not be an edge of $F$ and this path might not be the boundary face of a 2-face in $F'$. Nevertheless, because $F$ is connected, there exists a path $\pi$ in $F$ from $v$ to $w$. Choose such a path arbitrarily and let $vu$ be the first edge on this path. If $u$ is connected to $a$ through a path that does not contain $e$, then the path from $u$ to $v$ does contain $e$, and we have found a 2-face (the lift of $uv$) that contains $e$. If the path from $u$ to $a$ does contain $e$, then the pair of vertices $u$ and $w$ are connected in $T$ by a path containing $e$, and are connected in $F$ by a shorter path than $\pi$. In this case the result follows by induction on the length of $\pi$. $\square$

We may summarize the results in this section as a theorem:

Theorem 3.10 Let $P$ be a treetope with base $B$. Then the edges of $P$ that do not lie on $B$ form a tree, the canopy $T$ of $P$. The faces of $P$ may be partitioned into three classes:

1. the faces of $B$,
2. the vertices and edges of $T$ that are disjoint from $B$, and
3. one face $F'$ of dimension $i + 1$ for each $i$-dimensional face $F$ of $B$, called the lift of $F$.

For each face $F$ of $B$ with lift $F'$, $F'$ is a treetope with base $F$, and the canopy of $F'$ is the minimal subtree of $T$ that connects all the vertices in $F$.

Corollary 3.11 Every 4-treetope has a linear number of faces. More precisely, a 4-treetope with $n$ vertices has at most $4n - 10$ edges, $5n - 15$ 2-faces, and $2n - 5$ facets. A 4-treetope with $f$ facets has at most $3f - 10$ vertices, $6f - 20$ edges, and $4f - 10$ 2-faces.

Proof In any 4-treetope, the facets of the base correspond one-to-one with the non-base facets of the treetope, so if the treetope has $f$ facets then the base has $f - 1$ facets. We can count the faces of the treetope by the following case analysis:
• The base is a 3-polytope with at most \( n - 1 \) vertices. Therefore, it has at most \( 3n - 9 \) edges and at most \( 2n - 6 \) 2-faces. Dually, it has at most \( 2f - 6 \) vertices, at most \( 3f - 9 \) edges, and at most \( f - 1 \) 2-faces. It is also itself one of the facets of the treetope.
• The canopy is a tree spanning the vertices, so it has at most \( n - 1 \) edges. It has at most \( 2f - 6 \) leaves, and all interior vertices have degree at least four, so it has at most \( 3f - 10 \) vertices and at most \( 3f - 11 \) edges.
• Each remaining face is the lift of an edge or 2-face of the base. There are at most \( 3n - 9 \) or \( 3f - 9 \) lifts of edges (forming 2-faces of the treetope), and at most \( 2n - 6 \) or \( f - 1 \) lifts of 2-faces (forming facets of the polytope).

Summing the cases gives the numbers of faces stated in the corollary. □

The bounds of the corollary are tight for a pyramid over a simplicial 3-polytope with \( n \) vertices, or for a simple treetope with \( f \) facets. Corollary 3.11 does not generalize to a linear bound on the complexity of higher-dimensional treetopes, because treetopes of dimension five or more include the pyramids over arbitrary polytopes of dimension four or more, which can have quadratic or higher complexity. It also does not generalize to 4-polytopes that are not treetopes, which can have a number of faces that is quadratic in \( n \) or in \( f \). (It is open whether the number of faces of an arbitrary 4-polytope can be nonlinear relative to both \( n \) and \( f \) [14].)

### 4 Branches and Slices

Any tree can be partitioned into two subtrees by removing any of its edges. In this section we examine structures within a treetope that correspond to these partitions of its canopy.

**Definition 4.1** Suppose that \( P \) is a treetope with base \( B \), and \( uv \) is an edge of \( P \) such that neither \( u \) nor \( v \) belongs to \( B \). Then we can partition the canopy into two subtrees by deleting edge \( uv \). Let \( U \) be the subset of the vertices of \( P \) in the subtree containing \( u \), and \( V \) be the subset of the vertices of \( P \) in the other subtree containing \( v \). Then we call \( U \) and \( V \) branches of \( P \), and we call the partition \((U, V)\) of the vertices of \( P \) into two complementary branches a slice of \( P \).

An example of a slice and its two branches is shown in Fig. 4.

**Observation 4.2** Each branch must have at least \( \dim P \) vertices.

**Proof** The branch contains one vertex for the endpoint \( u \) of edge \( uv \) defining the branch, and another vertex for each neighbor of \( u \) other than \( v \). By Corollary 3.4 the number of these other neighbors is at least \( \dim P - 1 \). □

**Definition 4.3** Let \( P \) be a treetope with base \( B \), and \((U, V)\) be a slice of \( P \) defined by canopy edge \( uv \). Then we define the stems of branch \( U \) to be subsets of \( U \), one for each edge \( uw \) where \( w \neq v \). If \( w \in B \), then its stem is the singleton set \( \{w\} \). Otherwise, its stem is the branch \( W \) containing \( w \) of the slice \((X, W)\) defined by edge \( uw \).
Fig. 4 A slice of the 3-treetope of Fig. 3 determined by an edge uv, partitioning it into two complementary branches U and V

**Definition 4.4** We say that a subset $S$ of the vertices of $B$ is *externally $k$-connected* if the graph formed from the graph of $B$ by contracting all vertices of $B \setminus S$ into a single supervertex is $k$-vertex-connected.

**Lemma 4.5** Let $P$ be a treetope with base $B$, and $(U, V)$ be a slice of $P$ defined by canopy edge $uv$. Then $U \cap B$ is externally $(\dim P - 1)$-vertex-connected.

**Proof** Let $d = \dim P$, and define a graph $G$ from the graph of $B$ by contracting all vertices of $B \setminus (U \cap B)$ into a single supervertex, as in Definition 4.4. Let $K$ be a set of at most $d - 2$ vertices in $G$. We prove by induction on the cardinality of $U$ that deleting $K$ from $G$ leaves a connected graph. In the base case, each stem of $U$ is a single vertex of $v$. Otherwise, by the induction hypothesis, each stem of $U$ is externally $(d - 1)$-connected, and each vertex in $K$ corresponds to at most one vertex in the contracted graph for each stem. It follows that the deletion of $K$ cannot disconnect the vertices within any stem. It remains to show that (both in the base case and the inductive case) each two stems remain connected to each other. Because all edges between two stems belong to $B$, this is equivalent to the statement that the graph formed from $G \setminus K$ by contracting the remaining vertices of each stem into a single supervertex is connected.

Let $L$ be the link of $u$, a polytope formed from $P$ by intersecting it with a hyperplane near $u$. This is a $(d - 1)$-dimensional polytope, whose faces are in one-to-one correspondence with the faces of $P$ that are incident to $u$. This correspondence changes the dimension of a face by one, so that an edge of the link corresponds to a 2-face of $P$, etc. Observe that, if we were not deleting the vertices in $K$, then the graph of $L$ is isomorphic to the graph formed by contracting each stem, and the complementary branch $V$, to a single vertex. For, each edge between two stems or between a stem and branch $V$ lifts to a 2-face of $P$ incident to $u$ (by Lemma 3.9) and therefore corresponds to an edge of $L$, and vice versa.
By Balinski’s theorem [3], the graph of $L$ is $(d - 1)$-vertex-connected. Deleting a vertex in $K$ may change this graph either by damaging the complementary branch $V$ (which we cannot assume to remain connected after the deletion because $V$ might not come before $U$ in the induction order) or by removing the endpoint of one of the edges linking two stems. However, as there are only $d - 2$ deletions, the graph remains connected after this damage, and therefore no two stems can be separated from each other.

\[\square\]

**Corollary 4.6** For any branch $U$, the subgraph of the graph of $B$ induced by $U \cap B$ is connected.

**Lemma 4.7** Let $(U, V)$ be a slice of $P$, and let $X$ and $Y$ be two distinct sets that are either stems of $U$ or the set $V$. Then there is at most one edge in $B$ connecting $X$ to $Y$.

**Proof** As in the proof of Lemma 4.5, consider the link $L$ of $u$. It has one vertex for each stem, and one vertex for $V$. Every edge in $B$ connecting $X$ to $Y$ lifts to a 2-face of $P$ that passes through $u$. This 2-face in turn corresponds to an edge between the two vertices in $L$ that correspond to $X$ and $Y$. Distinct edges in $B$ lift to distinct edges in $L$, but each pair of vertices in $L$ can be the endpoints of at most one edge. Therefore there can be at most one edge from $X$ to $Y$ in $B$. \[\square\]

For convenience, we again summarize the results of this section in a single theorem, describing the graph-theoretic properties of branches. We will use these properties to characterize the graphs of 4-treetopes in an algorithmically recognizable way.

**Theorem 4.8** If $P$ is a treetope with base $B$, and $(U, V)$ is a slice of $P$, then both $U$ and $V$ include at least $d - 1$ stems. Each pair of stems of $U$ (or one stem and the set $V$) are connected by at most one edge in $B$. Both $U \cap B$ and $V \cap B$ are externally $(\dim P - 1)$-vertex-connected.

### 5 Non-unique Bases

It is possible for a treetope to have more than one base. Given any $d$-treetope $P$ with base $B$, and any integer $k > 1$, we can construct a $(d + k - 1)$-treetope with $k$ bases as follows. Choose a Cartesian coordinate for $(d + k - 1)$-dimensional space, and embed $B$ into the space spanned by the first $d - 1$ of the coordinates. For each of the $k$ remaining coordinates, embed a copy of $P$ into the space spanned by $d$ and this coordinate. Form the convex hull of the resulting $k$ copies of $P$. Call the resulting polytope $P^{(k)}$. Combinatorially, the same construction may be obtained from the Cartesian product of $P$ with a $k$-vertex simplex, by collapsing sets of vertices in the product that come from the same vertex of $B$ into single vertices.

**Theorem 5.1** The polytope $P^{(k)}$ is a treetope with at least $k$ different choices of base facet.

**Proof** Within $P^{(k)}$ there are $k$ facets of the form $P^{(k-1)}$, the intersection of $P^{(k)}$ with an axis-parallel hyperplane through the origin, perpendicular to one of the final $k$
coordinates. We claim that each of these facets can be chosen as a base for $P^{(k)}$. That is, every face of $P^{(k)}$ that intersects $B$ in at most one point has dimension at most one.

To see this, let $f$ be an arbitrary face in $P^{(k)}$. By the construction of $P^{(k)}$, $f$ has a construction of the same form: it is the convex hull of up to $k'$ copies of some face $f'$ of $P$, in different axis-parallel subspaces. That is, $f = f^{k'}$ for some $k' \leq k$. We have the following cases:

- If $f'$ has dimension more than one, it intersects $B$ in more than one vertex, by the assumption that $P$ is a treetope. Therefore, it also intersects the chosen facet $P^{(k-1)}$ in more than one vertex.
- If $f'$ is one-dimensional (it is an edge) and $k' > 1$, then one of the copies of $f'$ lies in the chosen facet $P^{(k-1)}$, and again intersects it in more than one vertex.
- If $f'$ is a single point disjoint from $B$, and $k' > 2$, then $f$ is a $k'$-vertex simplex, all but one of whose vertices lie in the chosen facet $P^{(k-1)}$.
- In the remaining cases that $f'$ is one-dimensional but $k' = 1$, that $f'$ is a single vertex disjoint from $B$ with $k' = 2$, or that $f'$ is a vertex of $B$, then $f$ has dimension at most one.

So in all cases, $P^{(k)}$ obeys the defining property of a treetope with the chosen copy of $P^{(k-1)}$ as its base.

Corollary 5.2 For any dimension $d > 1$, there exist infinitely many combinatorially distinct $d$-treetopes in which $d - 1$ different facets can be chosen as the base.

Proof Let $P_n$ be a two-dimensional convex polygon with $n$ vertices and construct the family of $d$-treetopes $P_n^{(d-1)}$.

Figure 5 (top) depicts a 4-treetope of this form; the bottom part of the figure depicts another 4-treetope, derived from the 3-treetope in Fig. 1 (left). Although 2-dimensional treetopes (that is, arbitrary convex polygons) can have an unbounded number of base faces, the same is not true for treetopes in higher dimensions.

Lemma 5.3 (Fomin and Thilikos [18]) Every 3-treetope (Halin graph) has at most four base faces.

Theorem 5.4 For $d \geq 3$, every $d$-treetope has at most $d + 1$ base facets.

Proof Let $P$ be a $d$-treetope with $k > 1$ base faces $B_1, B_2, \ldots, B_k$. For the representation of $P$ as a treetope with base $B_i$ (for $i < k$), Lemma 3.6 shows that face $B_k$ is itself a treetope, with $B_i \cap B_k$ as a base. $B_i \cap B_k$ is a ridge of $P$, and in any polytope, each ridge belongs to exactly two facets. So $B_k$ is a treetope with at least $k - 1$ different base faces, one for each $i < k$. The result follows by induction on dimension with Lemma 5.3 as the base case.

Every simplex provides an example of a $d$-treetope with exactly $d + 1$ base facets. This example, and the example of a triangular prism (a 3-treetope with three bases, the three square faces of the prism) shows that a treetope with $k > 2$ bases need not be formed from the construction $P^{(k)}$ described above. Nevertheless, as we now show, each two bases of the same treetope give an example of the construction with $k = 2$.  

 Springer
**Lemma 5.5** Let $P$ be a $d$-tope with $d > 2$ and with two bases $B_1$ and $B_2$. Then every vertex of $P$ belongs to $B_1 \cup B_2$, and the edges of $P$ between $B_1 \setminus B_2$ and $B_2 \setminus B_1$ form a perfect matching.

**Proof** Each vertex $v$ in $B_2 \setminus B_1$ has a unique neighbor $w$ that is not in $B_2$, the other endpoint of the lift of $v$ with respect to base $B_2$. By Theorem 3.10, the vertices of $B_2 \setminus B_1$ induce a connected subtree of the canopy of $P$ with respect to base $B_1$. Therefore, these non-$B_2$ neighbors for each vertex in $B_2 \setminus B_1$ must each lead to a distinct subtree of the canopy with respect to $B_1$. Each such subtree contains at least one leaf vertex in $B_1$: either $w$ itself is in $B_1$, or if not its branch contains at least $d - 1$ leaf vertices in $B_1$. By the assumption that $d > 2$, this number of leaves is greater than one.

So the number of vertices in $B_1 \setminus B_2$ is at least as large as the number of vertices in $B_2 \setminus B_1$, with equality only when these two sets of vertices are perfectly matched. By a symmetric argument, the number of vertices in $B_2 \setminus B_1$ is at least as large as the number of vertices in $B_1 \setminus B_2$, with equality only when these two sets of vertices are perfectly matched. Since two finite sets cannot each be larger than the other, the only possibility is that they are equal and that the vertices are matched.

**Theorem 5.6** Let $P$ be a $d$-tope with $d > 2$ and with two or more bases. Then any two of its bases are isomorphic to each other, and $P$ is isomorphic to $B^{(2)}$ where $B$ is any of the bases of $P$.

**Proof** Let $B_1$ and $B_2$ be any two bases of $P$. Then each face $f$ of $B_1$ corresponds to a face of $B_2$ obtained by lifting $f$ with respect to base $B_1$ and then taking the base of the resulting lifted face with respect to base $B_2$. By Lemma 5.5, the lift cannot be a
canopy edge disjoint from $B_2$, so this correspondence is well-defined on all faces. It is one-to-one and preserves the incidences between faces and sub-faces, from which it follows that $B_1$ and $B_2$ are isomorphic and that $P$ is isomorphic to $B^{(2)}$.  

This isomorphism between bases justifies the omission of the base from the notation $P^{(k)}$, as each choice of base will produce an isomorphic result.

6 Clustered Planarity

Our characterization of 4-treetopes is based on clustered planarity. An input pair $(C, G)$ where $G$ is a planar graph and $C$ is a family of subsets of the vertices of $G$ (a clustering of $G$) is said to be clustered planar when $G$ has a planar drawing such that each cluster can be drawn as a simple closed curve surrounding its vertices, without crossings between the cluster curves. Additionally, each edge can cross each cluster boundary at most once, and only when it has one endpoint inside the cluster and one endpoint outside. For such a drawing to exist, each two sets must either be disjoint or with one a subset of the other. Clustered planarity has been the subject of extensive research in graph drawing, and was recently proved to be testable in polynomial time [20]. We define a class of instances for the clustered planarity problem that are quite special, special enough to make the clustered planarity problem itself trivial for these instances. We construct non-planar graphs from these clustered planarity instances by adding a representative vertex for each cluster, and we use this construction to characterize the graphs of 4-treetopes.

6.1 Definitions

Definition 6.1 A polyhedral graph is a 3-vertex-connected planar graph. By Steinitz’s theorem a graph is polyhedral if and only if it is the graph of a 3-polytope.

Definition 6.2 If $G$ is a graph, and $C$ is a collection of subsets of the vertices of $G$, we say that $C$ is nested, and that $(C, G)$ is a clustering of $G$, if for every two sets $X$ and $Y$ in $C$ either $X \subset Y$, $Y \subset X$, or $X \cap Y = \emptyset$. A proper clustering is a clustering of an $n$-vertex graph in which the whole set of vertices is included as one of the clusters, each remaining cluster contains at least two and at most $n - 2$ vertices, no two clusters contain the same sets of vertices, and no two clusters contain complementary sets of vertices.

Rather than representing clusterings as planar embeddings with the clusters drawn as simple closed curves we instead represent the clusters themselves as vertices in a larger graph:

Definition 6.3 Let $(C, G)$ be a proper clustering. Then we define the cluster graph of $(C, G)$ to be a graph that contains $G$ as a subgraph, and in addition has one vertex $c_X$ for each cluster $X$ in $C$. Each vertex $v$ in $G$ is connected by an edge to the cluster vertex $c_X$ for the smallest cluster $X$ that contains $v$. There always exists at least one such cluster because of the inclusion of $V(G)$ as a cluster. Each cluster vertex $c_X$
Fig. 6 For a cycle graph (blue vertices) with connected clusters (yellow disks), the cluster graph (with an added red vertex in each region formed by the circles) is a Halin graph, and any Halin graph can be formed in this way (other than the one for $V(G)$) is connected by an edge to the cluster vertex $c_Y$ for the smallest cluster $Y$ that forms a strict superset of $X$.

The same construction may be represented topologically rather than combinatorially. Let $(C, G)$ be a proper clustering. Represent the vertices of $G$ as points in the plane, and the clusters of $G$ as Jordan curves disjoint from each other and the points, with each cluster consisting of the points inside the corresponding curve (ignoring whether the edges can be routed to give a valid clustered planar drawing). Then the cluster graph has a cluster vertex for the region within each of these curves, adjacent to the vertices for adjacent regions and to the points within its region. For example, if $G$ is a cycle graph and $C$ is a nested collection of paths of two or more vertices in $G$, the resulting cluster graph is a Halin graph, and any Halin graph can be formed as a cluster graph in this way (Fig. 6).

The following criteria for a more special class of clusterings and cluster graphs (depicted in Fig. 7) are motivated by the properties described in Theorem 4.8.

**Definition 6.4** If $(C, G)$ is a proper clustering of a polyhedral graph, we say that $(C, G)$ is a well-connected clustering if it has the following properties:
Each cluster vertex \( c_X \) in the cluster graph has degree at least four.

For each two disjoint sets \( X \) and \( Y \) that are either clusters, complements of clusters, or singleton vertex sets, and whose union is not the entire vertex set, at most one edge of \( G \) has one endpoint in \( X \) and one endpoint in \( Y \).

For each cluster \( X \) in \( C \), other than the set of all vertices, and for the complementary set \( Y = V(G) \setminus X \), both \( X \) and \( Y \) are externally 3-vertex-connected in \( G \).

For instance, the clustering shown in Fig. 7 is well-connected. However, if the three central vertices were grouped into another cluster, the result would not be well-connected, because then there would exist disjoint but non-complementary pairs of clusters connected by more than one edge. These definitions have been set up in such a way as to make the following observation clear:

**Observation 6.5** If \( P \) is a 4-treetope with base \( B \), then the graph of \( P \) is the cluster graph of a well-connected clustering \((C, G)\) where \( G \) is the graph of \( B \).

**Proof** To form a well-connected clustering from \( P \), we choose an arbitrary vertex \( v \in B \) and define a cluster for each slice \((U, V)\), where the cluster is the intersection of \( B \) with the branch of the slice that does not contain \( v \). The resulting clusters are nested and their well-connectedness follows from Theorem 4.8. Each vertex \( u \) that is in one of the defined clusters is connected in the cluster graph to the vertex for the smallest cluster that contains it, which corresponds to the parent of \( u \) in \( P \). Each vertex that is not in any of these clusters (including \( v \) itself) is connected to the cluster vertex corresponding to the cluster of all vertices in \( G \), which again corresponds to its parent. Thus, the cluster graph and the graph of \( P \) are isomorphic. \( \square \)

**Definition 6.6** The graph \( G \) of every 3-polytope \( P \) has a well-connected clustering with only one cluster, the set of all vertices of \( G \). This clustering represents the 4-treetope formed as the pyramid over \( P \). We call it the *trivial clustering* of \( G \).

For some polyhedral graphs, the trivial clustering is the only well-connected clustering. For instance, each 2-face of the octahedral graph is a triangle, so each partition of the vertices is crossed by two edges that share an endpoint, preventing the octahedron from having any nontrivial well-connected clusterings. We do not know whether the existence of a nontrivial well-connected clustering for a given polyhedral graph can be tested efficiently.
6.2 Expansion and Contraction

**Definition 6.7** Let \((C, G)\) be a well-connected clustering with at least one nontrivial cluster, and let \(X\) be a cluster in \(C\) that is not a superset of any smaller clustering. Then the **contraction** of \(X\) is the clustering \((C', G')\) in which we remove \(X\) from the clustering and replace the vertices of \(X\) in \(G\) by a single supervertex, keeping all adjacencies to vertices outside \(C\). The other clusters containing vertices of \(X\) should also be modified in the obvious way, by replacing these vertices by the new supervertex.

**Lemma 6.8** With \(C, G,\) and \(X\) as above, the contraction of \(X\) is another well-connected clustering.

**Proof** \(G'\) remains polyhedral: it is 3-vertex-connected and has at least four vertices by the external connectivity of \(V \setminus X\). The contraction does not change the required properties of any of the other clusters in \(C'\).

We define an **expansion** to be the opposite operation to a contraction. More precisely:

**Definition 6.9** Let \((C, G)\) be a well-connected clustering, let \(v\) be a designated vertex in \(G\), and let \(H\) be a polyhedral graph containing a vertex \(v'\) of the same degree as \(v\). Additionally, suppose that we have identified a one-to-one order-preserving correspondence between the edges incident to \(v\) (in the cyclic order given by the embedding of \(G\)) and the edges incident to \(v'\) (in the cyclic order given by the embedding of \(H\)). Then we define the **expansion** of \(v\) by \(H\) to be a graph formed from \(G\) by deleting \(v\), adding \(H - v'\) in its place, and reconnecting each of the edges that was incident to \(v\) in \(G\) to the corresponding neighbor of \(v'\) in \(H\). We then add to \(C\) another cluster for the vertices in \(H - v'\) that were added to the graph, and modify the existing clusters in \(C\) in the obvious way, by replacing \(v\) in each cluster that contains it by the vertices of \(H - v'\).

For instance, the graph in Fig. 7 can be formed by starting with the (4-regular 6-vertex) graph of an octahedron and its trivial clustering, and then performing three expansions, each of which uses the graph of the octahedron as \(H\) and creates one of the three nontrivial clusters in the figure.

**Observation 6.10** Expansions and contractions are inverse to each other: if we expand a vertex and then contract the resulting new cluster, or if we contract a cluster and then expand the resulting vertex by the graph defining the property of external connectivity of the contracted cluster, the result in either case is the original clustering.

**Lemma 6.11** With \(C, G, v,\) and \(H\) as above, the expansion of \(v\) by \(H\) is another well-connected clustering.

**Proof** The new cluster has the required degree in the cluster graph, because of the definitional requirement that the polyhedral graph \(H\) has at least four vertices. It is externally 3-connected by 3-connectivity of both \(G\) and \(H\). Its addition does not change the cluster graph degree, or external 3-connectivity of the other clusters, nor can it cause two edges to share endpoints when they did not do so previously. And the overall graph remains 3-connected, because the change cannot introduce any new 2-vertex cuts.
We summarize the results of this section in a theorem:

**Theorem 6.12** The well-connected clusterings are exactly the clusterings that can be obtained from the trivial clustering of a polyhedral graph by a sequence of expansion operations. Every expansion operation can be undone by a contraction operation, and vice versa. Both expansion and contraction preserve the property of being a well-connected clustering.

### 7 Realization

In this section we prove that the cluster graphs of well-connected clusterings can always be realized as 4-treetopes. We will realize our 4-treetopes by an inductive process in which we add one canopy vertex in each step. A proof along the same lines was used by Aichholzer et al. [1] to prove that every Halin graph has a polyhedral realization in which the base face is horizontal and all other faces have equal slopes, or equivalently that every tree can be realized as the straight skeleton of a convex polygon.

#### 7.1 Face and Cone Shape Realizability

To achieve the desired placement of new vertices in each step of the inductive proof, we will use projective duality together with a known method for realizing polyhedra with specified face shapes.

**Lemma 7.1** (Barnette and Grünbaum [4]) Let $G$ be a 3-vertex-connected planar graph, $f$ be a 2-face of its combinatorial embedding, and $B$ be a realization of $f$ as a convex polygon in the $xy$-plane of three-dimensional Euclidean space. Then there exists a three-dimensional polyhedron $P$ whose graph is isomorphic to $G$, with $B$ as the face corresponding to $f$.

By applying a projective transformation that fixes the plane of $B$ we can additionally ensure that, for a given viewpoint $\alpha$ on the opposite side of that plane from $P$, $f$ is the only face of $P$ visible from $\alpha$. For our purposes we need a projectively dual version of this result, for which we need some more definitions.

**Definition 7.2** Suppose that finitely many halfspaces all have boundaries that pass through a common point $p$, and that $p$ is the only point in the intersection of their boundaries. In such a case we call the intersection of the halfspaces a **convex polyhedral cone**, and we call $p$ the apex of the cone. If $Q$ is the intersection of a convex polyhedral cone $C$ with finitely many additional halfspaces (none of which contain the apex $p$), and every infinite face of $Q$ is a subset of an infinite face of the cone, we call $Q$ a **cone polyhedron**, and we call $C$ the cone of $Q$. The faces of a cone polyhedron may be defined in the same way as for convex polyhedra. Equivalently, a cone polyhedron is an intersection of finitely many halfspaces with the property that the hyperplanes containing unbounded faces of the intersection intersect in a single point, the apex.

The **graph** of a cone polyhedron $Q$ is an undirected graph with a vertex for each 0-face of $Q$ together with one additional vertex, the cone vertex of $Q$. It has an edge
for each 1-face of \( Q \), of two types: a 1-face that is a finite line segment connects two 0-faces, and a 1-face that is an infinite ray connects a 0-face with the cone vertex.

**Lemma 7.3** Let \( C \) be a three-dimensional convex polyhedral cone with \( k \) sides (2-faces), and let \( G \) be a polyhedral graph with a designated vertex \( v \) of degree \( k \), and with a fixed order-preserving correspondence between the edges incident to \( v \) and the rays of the cone. Then there exists a cone polyhedron \( Q \) whose graph is isomorphic to \( G \), such that the isomorphism maps \( v \) to the cone vertex and respects the correspondence between edges and rays, and such that \( C \) is the cone of \( Q \).

**Proof** Let \( \omega \) be a point interior to \( C \) and let \( \tau \) be a projective duality transformation that maps \( \omega \) to the plane at infinity. Then \( \tau \) maps the apex of \( C \) to a non-infinite plane \( \pi \), and it maps the planes through the sides of \( C \) to points in convex position in \( \pi \), forming the vertices of a convex polygon \( B \). Additionally, \( \tau \) maps the plane at infinity into a non-infinite point \( \alpha \) that does not belong to plane \( \pi \).

Apply Lemma 7.1 to realize the dual graph of \( G \) as a polyhedron \( P \) in which the face dual to \( v \) is realized as polygon \( B \), and additionally (by performing a projective transformation of \( P \)) arrange the realization in such a way that \( P \) is on the other side of \( \pi \) from \( \alpha \) and \( B \) is the only face of \( P \) visible from \( \alpha \). Then \( \tau^{-1}(P) \) (a cell in the projective arrangement of hyperplanes dual to the vertices of \( P \)) has one vertex \( v \) separated from all the others by the plane at infinity. That is, when viewed as a subset of Euclidean space, this cell in the arrangement has two connected components, one of which is a cone polyhedron and the other of which is a polyhedral cone (either \( C \) or its reflection through the apex). The cone polyhedron component, reflected if necessary to lie within \( C \), gives the desired realization \( Q \). \( \square \)

### 7.2 Characterizing 4-Treetopes

**Theorem 7.4** A graph \( G \) is the graph of a 4-treetope \( P \) with base \( B \) if and only if \( G \) is the cluster graph of a well-connected clustering \((C, F)\) of a polyhedral graph \( F \), with \( F \) forming the graph of \( B \).

**Proof** One direction, the claim that every 4-treetope graph is a cluster graph, is Observation 6.5. In the other direction, let \( G \) be the cluster graph of a well-connected clustering \((C, F)\); we will prove by induction on the number of clusters that \( G \) can be realized as a 4-treetope. As a base case, if there is only one cluster (the set of all vertices of the base graph) then we may realize the base graph \( F \) as a 3-polytope \( B \) by Steinitz’s theorem, and then realize \( G \) itself as the pyramid over \( B \).

Otherwise, by Theorem 6.12, let \((C, F)\) be obtained from a smaller well-clustered graph \((C', F')\) by an expansion operation. This operation replaces a vertex \( v \) of \( F' \) by a new cluster, whose cluster vertex may be called \( c \). Let \( H \) be the polyhedral graph used to form the expansion, and let \( v' \) be the vertex that is removed from \( H \) as part of the expansion (with \( v \) and \( v' \) having equal degrees). By induction, the cluster graph \( G' \) of \((C', F')\) can be represented as a 4-treetope \( P' \) with base \( B' \), with \( F' \) isomorphic to the graph of \( B' \).

Let \( C \) be a polyhedral cone in the three-dimensional affine hull of \( B' \), formed by the intersection of the 2-faces of \( B' \) that are incident to \( v \). By Lemma 7.3 we may find a
realization of $H$ as a cone polyhedron $K$, with $v'$ as the cone vertex and with $C$ as the cone of $K$, respecting the correspondence between neighbors of $v$ and neighbors of $v'$. Scale this cone polyhedron to be small enough so that all of its infinite rays have starting points that lie within the edges of $B'$ incident to $v$. Create a new base polyhedron $B$ by adding the vertices of the scaled cone polyhedron to $B'$ and removing $v$ (Fig. 8). Add another vertex representing $c$ anywhere on the edge from $v$ to its parent in $P'$, and compute $P$ as the convex hull of the set of vertices obtained in this way.

Then, in the new polytope $P$, for each vertex $u$ that was a neighbor of $v$ there exists a new vertex within line segment $uv$, so the change from $P'$ to $P$ does not change the link of any vertex that belongs to both $P$ and $P'$. However, in $P$, each vertex of $K$ must have a neighbor outside $B$, for otherwise it would have a two-dimensional link. Since all vertices with changed links belong to $B \cup \{c\}$, the only choice for a vertex outside $B$ to connect to is $c$. Therefore each vertex of $K$ has an edge to $c$, but to no other vertices outside $B$. Thus, we have formed a 4-treetope whose canopy now includes $c$, and where the base vertices reached from $c$ have the correct topology, realizing $G$ as required.

\[\qed\]

8 Recognition

Our recognition algorithm for the graphs of 4-treetoposes is based on the idea of repeatedly finding and contracting a cluster in the clustering corresponding to the treetope. To this end, we seek the vertices that represent contractible clusters.

8.1 Extremal Vertices and Extremal Clusters

Definition 8.1 Let $G$ be the graph of a 4-treetope $P$ with base $B$. Then a vertex $v$ of $G$ is extremal if $v$ is disjoint from $B$ and has exactly one neighbor in $G$ that is also disjoint from $B$. An extremal cluster of $G$ is the set of vertices consisting of $v$ and its neighbors in $B$.

Because the vertices and edges of a treetope that are disjoint from the base form a tree, whose leaves are the extremal vertices, we have:

Observation 8.2 Every 4-treetope that is not a pyramid contains at least two extremal vertices.
**Observation 8.3** Let $G$ be the graph of a 4-treetope $P$ with base $B$, and let $v$ be an extremal vertex of $G$. Then $G$ is the cluster graph of a well-connected clustering $(C, H)$ where $H$ is the graph of $B$, in which the neighbors in $B$ of $v$ form a minimal cluster in $H$.

**Proof** Choose a vertex $w$ of $B$ that is not a neighbor of $v$, and for each slice of $P$ form a cluster in $B$ consisting of the base vertices in the branch of the slice that does not contain $w$. ⊓⊔

**Observation 8.4** Let $G$ be the graph of a 4-treetope $P$ with base $B$, let $v$ be an extremal vertex of $G$, and let $G$ be the cluster graph of a well-connected clustering $(C, H)$ in which the extremal cluster of $v$ is one of the minimal clusters. Then the operation in $G$ of contracting the cluster of $v$ into a single supervertex produces the cluster graph of the clustering formed by contracting the extremal cluster of $v$.

### 8.2 Candidate Vertices

Intuitively, the overall outline of our algorithm will be to repeatedly identify and contract extremal clusters until reaching the graph of a pyramid. We would like to do this by using the properties of cluster graphs to identify their extremal vertices. However, these vertices cannot be uniquely identified, as the example of a tetrahedral prism demonstrates. This 4-polytope has two tetrahedral facets and four triangular-prism facets; it can form a treetope in four different ways, with any one of the triangular-prism facets as its base and with the two remaining vertices that are outside this facet as its extremal vertices. Thus, in this polytope, every vertex is extremal, but not all choices of extremal clusters are compatible with each other. In other, larger 4-treetopes, there can also exist vertices that are necessarily part of the base of the treetope, but whose local neighborhoods look like the neighborhoods of extremal vertices. Therefore, we define a broader class of vertices, the candidate vertices, that include the extremal vertices and possibly some other non-extremal vertices.

**Definition 8.5** Let $G$ be an arbitrary graph. We define a candidate vertex to be a vertex $v$ of $G$ with the following properties:

- $v$ has at least four neighbors.
- The graph induced in $G$ by the neighbors of $v$ is planar, and has exactly two connected components, one of which is an isolated vertex.
- If $v$ is deleted from $G$, the nontrivial component of the neighbors of $v$ induces an externally 3-vertex-connected subgraph of the remaining graph.
- The set of edges connecting the vertices in the nontrivial component of the neighbors of $v$ to vertices (other than $v$) outside this component forms a matching in $G$, with no two of these edges sharing an endpoint.

**Observation 8.6** The conditions for being a candidate vertex are checkable in polynomial time and are satisfied by every extremal vertex.

Despite candidate vertices not necessarily being extremal vertices, they can be used to identify extremal clusters:
Definition 8.7  Let $G$ be a graph containing a candidate vertex $v$. The candidate cluster of $v$ is the set of $v$ and the vertices in the nontrivial connected component of neighbors of $v$.

Lemma 8.8  Let $G$ be the graph of a 4-treetope $P$ with base $B$, let $v$ be a candidate vertex, and let $Q$ be the candidate cluster of $v$. Then $Q$ is an extremal cluster for $G$ and $B$.

Proof  We first observe that $v$ cannot be a non-base vertex that is not extremal, for every such vertex has a neighborhood that induces a graph with at least two isolated vertices (the canopy neighbors of $v$). And if $v$ is extremal, the result is true by definition. So the remaining case is that $v$ is a candidate vertex but that it belongs to $B$. In this case, let $u$ be the parent of $v$, and let $w$ be the isolated vertex in the neighborhood of $v$.

Then we have the following:

- The neighborhood of $u$ contains exactly one cluster vertex, or equivalently, $u$ is extremal. There must be at least one cluster vertex neighbor of $u$, for otherwise we would have a trivial clustering, and the neighbors of $v$ would form a single component connected through $u$, violating the assumption that there are two components. And $u$ cannot have two cluster vertex neighbors, for that would violate the condition that the edges connecting neighbors of $v$ to the rest of the graph form a matching.

- The extremal cluster of $u$ contains $v$. This follows from $u$ being be the parent of $v$.

- The extremal cluster of $u$ contains at least one neighbor $x$ of $v$ in $Q$. For otherwise the only possible neighbor of $v$ in the cluster of $u$ would be its isolated neighbor $w$, and the cluster could not be externally 3-connected.

- The extremal cluster of $u$ contains every neighbor of $v$ in $Q$. For, if not, by the connectivity of the neighborhood, there would exist some two adjacent vertices $y$ and $z$ in $Q$ such that the cluster of $u$ contained $y$ but not $z$. But then there would be two edges from $z$ to the cluster of $u$ (one to $y$ and one to $v$), violating the requirement that no two edges into the cluster can share an endpoint.

- The extremal cluster of $u$ does not contain any base vertex $z$ outside $Q$. For, if it did, $u$ would belong to $Q$ but would have two neighbors outside $Q$ (the vertex $z$ and one cluster vertex neighbor), violating the requirement on the candidate vertex $v$ that the edges from the cluster of $v$ to the rest of the graph form a matching.

We conclude from this chain of reasoning that $Q$ is the extremal cluster of $u$ and that it is an extremal cluster. □

Lemma 8.9  Let $G$ be an arbitrary graph, let $v$ be a candidate vertex, and let $Q$ be the candidate cluster of $v$. Let $G'$ be the graph formed by contracting $Q$ to a single supervertex $v'$, and suppose that $G'$ is the graph of a 4-treetope in which $v'$ belongs to the base. Then $G$ is also the graph of a 4-treetope in which $Q$ is an extremal cluster.

Proof  The reversal of the contraction operation can be interpreted as an expansion operation in a well-connected clustering whose cluster graph is $G'$, taking it to a well-connected clustering whose cluster graph to $G$. The result follows by Theorem 7.4. □
8.3 The Algorithm

Based on the analysis of the previous sections, we can test whether a given graph $G$ is the graph of a 4-treetope as follows:

- Initialize a set $K$ of known base vertices of $G$ to be the empty set.
- While $G$ contains a candidate vertex $v$ that does not belong to $K$:
  - Contract the candidate cluster of $v$ into a single supervertex $v'$.
  - Add $v'$ to $K$.
- If the remaining graph contains a universal vertex $u$ that does not belong to $K$, and the vertices other than $u$ induce a polyhedral graph, return yes. Otherwise, return no.

Theorem 8.10 The algorithm described above correctly tests whether its input is the graph of a 4-treetope, in polynomial time.

Proof Each step involves testing graph properties such as planarity that are already known to be polynomial, and each iteration of the loop reduces the size of the graph by at least one vertex, so the polynomial time bound for the algorithm is clear.

If $G$ is the graph of a 4-treetope $P$ with base $B$, then by Lemma 8.8 each iteration will correctly perform a contraction of an extremal cluster in $G$, and will correctly mark the resulting supervertex as part of the base of the contracted graph. Therefore, in this case, the algorithm will eventually reach a 4-treetope that has no extremal vertex, which by Observation 8.2 must be a 4-pyramid. In such a graph, there does exist a universal vertex whose neighborhood is polyhedral, and the algorithm will correctly answer yes.

Conversely, suppose that the algorithm does answer yes. Then it will have found a sequence of contractions that reduce the given graph to the graph of a 4-pyramid, whose apex does not belong to the set $K$. Then by Lemma 8.9 each contraction made by the algorithm can be reversed to produce a 4-treetope whose canopy is disjoint from $K$. Therefore, the algorithm’s “yes” answer is correct.

Recognizing the graphs of pyramids over arbitrary 4-polytopes, and therefore also recognizing the graphs of 5-treetopes, is as difficult as recognizing the graphs of arbitrary 4-polytopes, which we expect to be complete for the existential theory of the reals.

9 Graph-Theoretic Properties of 4-Treetope Graphs

As unions of planar graphs and trees, the graphs of 4-treetopes are necessarily sparse graphs. But although the graphs of 3-treetopes (the Halin graphs) have bounded treewidth, the same is not true for 4-treetopes, because they include the graphs formed by adding an apex to arbitrary planar graphs. More strongly, as we show below, the 4-treetopes are not contained in any nontrivial minor-closed graph family. Nevertheless, they obey stronger forms of sparsity than merely having a low ratio of edges to
vertices. In particular, they have bounded expansion in the sense described by Nešetřil and de Mendez [31].

Although Halin graphs are Hamiltonian and more strongly almost pancyclic (only one cycle length can be missing) [37], the same is not true for 4-treetope graphs. Grünbaum and Motzkin constructed 3-dimensional polyhedra in which all simple paths have sublinear length [21], and the pyramids over these polyhedra are 4-treetopes in which all simple cycles are again sublinear.

9.1 Knotted Embeddings and Arbitrary Minors

To prove that 4-treetope graphs do not have any forbidden minors, we study the knots, links, and graphs that can be embedded on the boundary of a 4-polytope (topologically a 3-sphere) as cycles of vertices and edges of the polytope. Here, to avoid pathological cases, we define a piecewise-linear knot as a simple closed curve in $\mathbb{R}^3$ formed as the union of finitely many line segments, and a knot to be a curve topologically equivalent to a piecewise-linear knot, meaning that it can be obtained by a homeomorphism of $\mathbb{R}^3$. Similarly, a piecewise-linear link is the disjoint union of finitely many piecewise-linear knots, and a link is anything topologically equivalent to a piecewise-linear link; this should be distinguished from the unrelated meaning of the link of a polytope. An embedding of a knot or link is any other knot or link that is topologically equivalent. A knot is trivial (also called the unknot) if it is topologically equivalent to a circle, and a link is trivial if it is topologically equivalent to a collection of circles with disjoint bounding balls. For 4-polytopes that are pyramids, the graph of the polytope cannot contain any nontrivial knots or links. For, in this case, every cycle either remains entirely on the base of the pyramid or it forms a path on the base together with two edges connecting the path endpoints to the apex. Thus, with respect to the 2-sphere boundary of the base, it forms at most a 1-bridge knot, which must therefore be the unknot. However, this does not extend to treetopes:

**Observation 9.1** Let $K$ be an arbitrary knot or link. Then there exists a 4-treetope $P$ whose graph contains a knot or link that is embedded into the boundary of $P$ in a way that is topologically equivalent to $K$. 
Proof Project a piecewise-linear representation of $K$ onto a plane with a projection direction in general position, so that the result is a diagram of $K$ as a piecewise-linear self-crossing curve in the plane with only two segments of the curve meeting at each crossing point. Draw a circle surrounding each crossing point, small enough that it does not cross or contain any other such circle and intersecting the diagram of $K$ in exactly four crossing points. Add a subdivision vertex at each crossing of the diagram with itself or with these circles. Add additional subdivision vertices along the strands of $K$ outside these circles, and edges between these vertices, as necessary so that the result becomes 3-vertex-connected. Form a well-clustered graph by adding a cluster consisting of each original crossing point of $K$ and the four points where the circle surrounding it crosses $K$ (Fig. 9), and realize this clustering as a 4-treetope. Then, at each crossing of $K$, one of the two strands of $K$ may be replaced by a two-edge path through the cluster vertex of the crossing, separating it from the other strand. □

More generally, every piecewise-linear embedding of a graph into three-dimensional space has a topologically equivalent embedding (with its edges subdivided into paths) as a subgraph of a 4-treetope within the boundary 3-sphere of the treetope. To find a treetope that contains a subdivision of a given embedded graph, perform the following steps:

- Find a projection of the 3-dimensional embedding of the graph onto the plane so that the only nonplanarities are simple crossings between pairs of edges.
- Choose a casing (above-below relationship of the two edges at each crossing) consistent with the original 3d embedding.
- Replace each crossing by a clustered five-vertex subgraph as in Fig. 9.
- Finally, expand each edge of the original graph into a path through the cluster graph of the resulting clustered planar drawing, following a path through the five-vertex subgraph for the lower edge in each crossing and instead following a path through the cluster graph for the upper edge in each crossing.

Applying this construction to large complete graphs shows that there are no forbidden minors for the graphs of 4-treetopes. In this respect the 4-treetopes differ from the 4-pyramids, for which the seven graphs of the Petersen family, and many others, are known forbidden minors [32].

### 9.2 Separators and Bounded Expansion

A separator of an $n$-vertex graph is a subset of vertices the removal of which partitions the remaining subgraph into connected components whose number of vertices is at most a constant fraction of $n$. As is well known, planar graphs have separators of size $O(\sqrt{n})$; this property forms the basis for many efficient algorithms for these graphs. We will use a stronger form of this planar separator theorem:

**Lemma 9.2** (Miller [2,30]) Every maximal planar graph has a separator of size $O(\sqrt{n})$ that forms a simple cycle, such that at most $2n/3$ vertices are inside the cycle and at most $2n/3$ vertices are outside the cycle.
As we will show, the graphs of 4-treetopes also obey a similar separator theorem. To prove this, we define a superclass of the clustered planar drawings used to define 4-treetopes.

**Definition 9.3** We define a *sparse clustering* to be a clustered planar drawing with the property that for each two disjoint vertex sets \( X \) and \( Y \) that are clusters, complements of clusters, or singleton vertex sets, and whose union is not the entire vertex set, at most one edge of \( G \) has one endpoint in \( X \) and one endpoint in \( Y \). We define a *sparse cluster graph* to be the cluster graph of a sparse clustering.

That is, we keep one of the main requirements of a well-connected clustering, but we forgo the requirements of minimum degree four per cluster vertex and external 3-vertex-connectivity of each cluster.

**Theorem 9.4** Every \( n \)-vertex subgraph of a sparse cluster graph has a separator of size \( O(\sqrt{n}) \). In particular this is true for the graphs of 4-treetopes.

**Proof** Let \( G \) be a subgraph of a cluster graph of a sparse clustering. We can assume that the whole cluster graph of the same clustering does not include any additional vertices, for if we had a clustering with additional vertices in the underlying graph or additional curves defining more cluster vertices than the ones in \( G \), we could delete those vertices or curves from the clustering and obtain another clustering of which \( G \) is also a subgraph. And since additional edges only make it more difficult to obtain a small separator, we may assume without loss of generality that \( G \) is the whole cluster graph rather than a proper subgraph.

We define a planar graph \( H \) from the clustering by the following steps:

- Augment the underlying planar graph by a new vertex at each crossing point of an edge and a cluster boundary (subdividing the edge at that point)
- Add a cycle of edges that follow the curve surrounding each cluster, connecting the crossing points on that cluster. If there are only one or two crossing points, add a constant number of additional vertices on the curve so that it can be completed to a cycle of edges without forming a multigraph.
- Complete the resulting planar embedded graph to a maximal planar graph, preserving its embedding.

Figure 10 shows part of this construction, for the clustering of Fig. 7, prior to the maximal planar completion. The number of crossing points added on one of the curves of the clustering is at most proportional to the number of children of the corresponding cluster in the cluster hierarchy, from which it follows that \( H \) has \( O(n) \) vertices. By repeatedly applying Lemma 9.2 we can find a collection of \( O(1) \) cycles that together partition \( H \) into connected subgraphs of at most \( n/3 \) vertices, and that each have length \( O(\sqrt{n}) \).

We construct a separator that includes each vertex of the underlying planar graph that belongs to one of these separating cycles. We also include in the separator a cluster vertex for each region of the clustered drawing that is crossed by one of the separating cycles. Thus, the number of cluster vertices included in the separator is proportional to the number of crossing vertices of \( H \) included in the separating cycles in \( H \). This
separator partitions \( G \) into subgraphs that correspond to the connected components of \( H \).

Each vertex in one of the connected components of \( H \) that remain after removing the cycles either directly corresponds to a vertex of \( G \) (if it is a vertex of the underlying planar graph) or to two cluster vertices in \( G \) (if it is a crossing vertex in \( H \)). Thus, each of the remaining connected subgraphs of \( G \) after the separator vertices are removed has at most \( 2n/3 \) vertices.

If this result is used to recursively subdivide the graph of a treetope, the sizes of the separators at each level of subdivision decrease by a constant factor at each level of the recursion. One can interpret this recursive subdivision as a tree-decomposition of the graph, with width at most the sum of the contributions from each level. Bounding this sum by a geometric series, it follows that the treewidth of an \( n \)-vertex treetope is \( O(\sqrt{n}) \), from which many algorithmic consequences follow.

**Definition 9.5** A \( t \)-shallow minor of a given graph \( G \) is another graph obtained from \( G \) by contracting a collection of vertex-disjoint connected subgraphs of radius at most \( t \), and then performing an arbitrary sequence of edge and vertex deletions on the result. A family of graphs has bounded expansion if there exists a function \( f \) such that every \( t \)-shallow minor of a graph in the family has a ratio of edges to vertices that is at most \( f(t) \) [31]. More strongly, a family of graphs has polynomial expansion if it has bounded expansion with a function \( f \) that is bounded by a polynomial in \( t \) [12].

These properties are important in some algorithmic applications. In particular, subgraph isomorphism is fixed-parameter tractable (parametrized by subgraph size) for graph families of bounded expansion [31], and several graph optimization problems including maximum independent set and minimum dominating set have polynomial-time approximation schemes on graphs of polynomial expansion [24].
Theorem 9.6 Subgraphs of sparse cluster graphs, and in particular the graphs of treetopes, have polynomial expansion.

Proof The subgraphs of sparse cluster graphs form a hereditary graph class: a subgraph of a subgraph of a sparse cluster graph is itself a subgraph of a sparse cluster graph. As Dvořák and Norin [12] show, every hereditary graph class obeying a separator theorem with separator size $O(n^{1-\epsilon})$ for constant $\epsilon > 0$ has polynomial expansion. The result follows from Theorem 9.4. □

The assumption that the clustering is sparse is a necessary condition for this result. Without this constraint, every graph that can be decomposed into a Hamiltonian cycle and a perfect matching can be represented as a subdivision of a cluster graph of a clustered planar drawing. The construction is to represent the Hamiltonian cycle as the underlying planar graph of the clustering, and to form a collection of nested
clusters, each of which adds the two vertices of a matched pair to the next inner cluster in the nesting. Figure 11 illustrates an example of this construction for the Heawood graph, a 14-vertex 3-regular graph with girth 6. The graphs that can be represented in this way include Hamiltonian 3-regular expander graphs, which have no sublinear separators. Therefore, the cluster graphs of arbitrary clustered planar drawings do not have polynomial expansion. By modifying the construction to add a central vertex connected to all the other vertices of the cycle (making the underlying planar graph be a wheel instead of a cycle) one can additionally perform this construction with a clustered planar drawing in which all clusters are connected.

10 Conclusions

We have defined an interesting class of polytopes, generalizing the Halin graphs to higher dimensions. We have characterized the graphs of four-dimensional polytopes in this class in terms of the cluster graphs of certain clustered planar graph drawings, and used this characterization to develop polynomial-time recognition algorithms for these graphs. We have also begun a preliminary graph-theoretic investigation of the properties of these graphs, showing that (unlike the graphs of Halin graphs and three-dimensional polyhedra) they do not have bounded treewidth or forbidden minors, but they do have bounded expansion.

Our algorithms can be used to recognize the graphs of 4-treetopes quickly, but do not immediately lead to a polynomial-time algorithm for constructing a realization of these polytopes, because of our dependence on the Barnette–Grünbaum realization of a three-dimensional polytope with a pre-specified face shape. Even if this specified face has small integer coordinates, the use of induction by Barnette and Grünbaum may cause their realization to have doubly exponential coordinates, requiring an exponential number of bits to represent precisely. This issue naturally raises several questions: Can 4-treetopes always be realized with integer coordinates (as 3-polytopes can and general 4-polytopes cannot)? If so, can we represent those coordinates using a polynomial number of bits per coordinate? And if that is also true, can we construct a realization in polynomial time?

References

1. Aichholzer, O., Cheng, H., Devadoss, S.L., Hackl, T., Huber, S., Li, B., Risteski, A.: What makes a tree a straight skeleton? In: Proceedings of the 24th Canadian Conference on Computational Geometry (CCCG’12) (2012). https://2012.cccg.ca/papers/paper30.pdf
2. Alon, N., Seymour, P., Thomas, R.: Planar separators. SIAM J. Discrete Math. 7(2), 184–193 (1994)
3. Balinski, M.: On the graph structure of convex polyhedra in \( n \)-space. Pac. J. Math. 11(2), 431–434 (1961)
4. Barnette, D., Grünbaum, B.: Preassigning the shape of a face. Pac. J. Math. 32, 299–306 (1970)
5. Booth, K.S., Lueker, G.S.: Testing for the consecutive ones property, interval graphs, and graph planarity using \( P \overline{Q} \)-tree algorithms. J. Comput. Syst. Sci. 13(3), 335–379 (1976)
6. Boyer, J.M., Myrvold, W.J.: On the cutting edge: simplified \( O(n) \) planarity by edge addition. J. Graph Algorithms Appl. 8(3), 241–273 (2004)

Springer
7. Chiba, N., Nishizeki, T., Abe, A., Ozawa, T.: A linear algorithm for embedding planar graphs using $PQ$-trees. J. Comput. Syst. Sci. 30(1), 54–76 (1985)
8. Cornuéjols, G., Naddef, D., Pulleyblank, W.R.: Halin graphs and the travelling salesman problem. Math. Program. 26(3), 287–294 (1983)
9. Cortese, P.F., Di Battista, G.: Clustered planarity. In: Proceedings of the 21st Symposium on Computational Geometry (SCG’05), pp. 32–34. ACM, New York (2005)
10. de Fraysseix, H., de Mendez, P.O., Rosenstiehl, P.: Trémaux trees and planarity. Int. J. Found. Comput. Sci. 17(5), 1017–1030 (2006)
11. Demaine, E.D., Demaine, M.L., Uehara, R.: Zipper unfolding of domes and prisms. In: Proceedings of the 25th Canadian Conference on Computational Geometry (CCCG 2013), pp. 43–48 (2013). https://cglab.ca/~cccg/proceedings/2013/papers/paper_10.pdf
12. Dvořák, Z., Norin, S.: Strongly sublinear separators and polynomial expansion. SIAM J. Discrete Math. 30(2), 1095–1101 (2016)
13. Eppstein, D.: Simple recognition of Halin graphs and their generalizations. J. Graph Algorithms Appl. 20(2), 323–346 (2016)
14. Eppstein, D., Kuperberg, G., Ziegler, G.M.: Fat 4-polytopes and fatter 3-spheres. In: Bezdek, A. (ed.) Discrete Geometry: In Honor of W. Kuperberg’s 60th Birthday. Pure and Applied Mathematics, vol. 253, pp. 239–265. Marcel Dekker, New York (2003)
15. Eppstein, D., Löffler, M.: Bounds on the complexity of halfspace intersections when the bounded faces have small dimension. Discrete Comput. Geom. 50(1), 1–21 (2013)
16. Feigenbaum, J., Hershberger, J., Schäffer, A.A.: A polynomial time algorithm for finding the prime factors of Cartesian-product graphs. Discrete Appl. Math. 12(2), 123–138 (1985)
17. Feng, Q.W., Cohen, R.F., Eades, P.: Planarity for clustered graphs. In: Proceedings of the 3rd European Symposium on Algorithms (ESA 1995). Lecture Notes in Computer Science, vol. 979, pp. 213–226. Springer, Berlin (1995)
18. Fomin, F.V., Thilikos, D.M.: A 3-approximation for the pathwidth of Halin graphs. J. Discrete Algorithms 4(4), 499–510 (2006)
19. Friedman, E.J.: Finding a simple polytope from its graph in polynomial time. Discrete Comput. Geom. 41(2), 249–256 (2009)
20. Fulek, R., Tóth, C.D.: Atomic embeddability, clustered planarity, and thickenability. In: Proceedings of the 31st ACM-SIAM Symposium on Discrete Algorithms (2020) (to appear)
21. Grünbaum, B., Motzkin, T.S.: Longest simple paths in polyhedral graphs. J. Lond. Math. Soc. 37, 152–162 (1962)
22. Grünbaum, B., Shephard, G.C.: Convex polytopes. Bull. Lond. Math. Soc. 1(3), 257–300 (1969)
23. Halin, R.: Studies on minimally $n$-connected graphs. In: Welsh, D.J.A. (ed.) Combinatorial Mathematics and its Applications, pp. 129–136. Academic Press, London (1971)
24. Har-Peled, S., Quannrud, K.: Approximation algorithms for polynomial-expansion and low-density graphs. SIAM J. Comput. 46(6), 1712–1744 (2017)
25. Hopcroft, J., Tarjan, R.E.: Efficient planarity testing. J. ACM 21(4), 549–568 (1974)
26. Imrich, W., Peterin, I.: Recognizing Cartesian products in linear time. Discrete Math. 307(3–5), 472–483 (2007)
27. Kawarabayashi, K.: Planarity allowing few error vertices in linear time. In: Proceedings of the 50th IEEE Symposium on Foundations of Computer Science (FOCS 2009), pp. 639–648. IEEE Computer Society, Los Alamitos (2009)
28. Kirkman, T.P.: On the enumeration of $x$-edra having triedral summits and an $(x - 1)$-gonal base. Philos. Trans. R. Soc. Lond. 146, 399–411 (1856)
29. Koch, E., Perles, M.A.: Covering efficiency of trees and $k$-trees. In: Proceedings of the 7th Southeastern Conference on Combinatorics, Graph Theory, and Computing. Congressus Numerantium, vol. 17, pp. 391–420. Utilitas Mathematica, Winnipeg (1976)
30. Miller, G.L.: Finding small simple cycle separators for 2-connected planar graphs. J. Comput. Syst. Sci. 32(3), 265–279 (1986)
31. Nešetřil, J., de Mendez, P.O.: Sparsity: Graphs, Structures, and Algorithms. Algorithms and Combinatorics, vol. 28. Springer, Heidelberg (2012)
32. Pierce, M.: Searching for and classifying the finite set of minor-minimal non-apex graphs. Honours thesis, California State University, Chico (2014). https://www.csuchico.edu/~tmattman/mpthesis.pdf
33. Rademacher, H.: On the number of certain types of polyhedra. Ill. J. Math. 9, 361–380 (1965)
34. Richter-Gebert, J., Ziegler, G.M.: Realization spaces of 4-polytopes are universal. Bull. Am. Math. Soc. New Ser. 32(4), 403–412 (1995)
35. Schaefer, M.: Complexity of some geometric and topological problems. In: Proceedings of the 19th International Symposium on Graph Drawing (GD 2009). Lecture Notes in Computer Science, vol. 5849, pp. 334–344. Springer, Berlin (2010)
36. Shih, W.K., Hsu, W.L.: A new planarity test. Theor. Comput. Sci. 223(1–2), 179–191 (1999)
37. Skowrońska, M.: The pancyclicity of Halin graphs and their exterior contractions. In: Alspach, B.R., Godsil, C.D. (eds.) Cycles in Graphs. Annals of Discrete Mathematics, vol. 27, pp. 179–194. North-Holland, Amsterdam (1985)
38. Steinitz, E.: Polyeder und Raumeinteilungen. Encyclopädie der Mathematischen Wissenschaften, Band 3 (Geometries), pp. 1–139 (1922)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.