Separating Geodesic Structure and Product Structure

Laura Merker
Karlsruhe Institute of Technology, Germany

Lena Scherzer
University of Hamburg, Germany

Samuel Schneider
Karlsruhe Institute of Technology, Germany

Abstract

The geodesic treewidth of a graph G is the smallest k for which there is a partition $\mathcal{P}$ into geodesics such that $G/\mathcal{P}$ has treewidth k, where $G/\mathcal{P}$ is obtained from G by contracting each part of $\mathcal{P}$. Based on this notion, row treewidth was developed and is defined for a graph G as the smallest k such that $G \subseteq H \boxtimes P$ for some graph H of treewidth k and a path P. Equivalently, the row treewidth of a graph G is the smallest k for which there is a partition $\mathcal{P}$ into disjoint unions of geodesics that are aligned with respect to some layering such that $G/\mathcal{P}$ has treewidth k.

We separate the two notions by showing that bounded row treewidth does not imply bounded geodesic treewidth and by presenting a polynomial-time algorithm to decide whether a graph of treewidth 2 has geodesic treewidth 1, which is known to be NP-hard for row treewidth [Biedl, Eppstein, Ueckerdt, 2025]. More generally, we provide an algorithm to decide whether a given graph has geodesic treewidth at most d that is XP in the treewidth, whereas there is no such algorithm for row treewidth, unless P $=$ NP [Biedl, Eppstein, Ueckerdt, 2025]. On the other hand, we show that computing the geodesic treewidth is NP-hard and that every graph with geodesic treewidth 1 has bounded row treewidth. Moreover, we improve the best known lower bound on the geodesic treewidth of planar graphs to 5.

2012 ACM Subject Classification Mathematics of computing → Graph theory; Mathematics of computing → Graph algorithms

Keywords and phrases product structure, row treewidth, geodesic structure, geodesic treewidth

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.126

Related Version *Full Version*: `https://arxiv.org/abs/2607.02098v1` [32]

1 Introduction

Bounded *treewidth* allows using a wide range of algorithmic techniques to solve numerous important problems efficiently. However, many common graph classes, like planar graphs, do not have bounded treewidth. Hence, we are interested in generalizations of treewidth that apply to broader classes of graphs but still provide comparable structural benefits. In particular, we need to be able to handle large grids as they are planar graphs of high treewidth. A popular way to define such generalizations is to take a partition $\mathcal{P}$ of a graph G into (disjoint unions of) geodesics[1] and aim for small treewidth of the *quotient* $G/\mathcal{P}$, i. e., the graph that is obtained by contracting each (possibly disconnected) part of $\mathcal{P}$ into a single vertex. For a first example, note that a grid admits a partition into geodesics such that the quotient is a path and thus has small treewidth. The two generalizations of treewidth we

[1] A *geodesic* is a shortest path between any two vertices in the graph.

34th Annual European Symposium on Algorithms (ESA 2026).
Editors: Philip Bille, Seth Pettie, and Sabine Storandt; Article No. 126; pp. 126:1–126:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

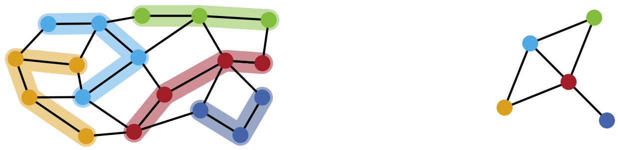

Figure 1 Left: A graph G with a partition $\mathcal{P}$ into geodesics. Right: The graph $G/\mathcal{P}$.

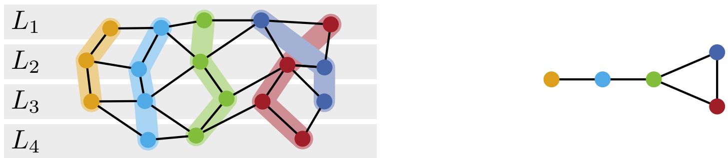

Figure 2 Left: A graph G with a layering $\mathcal{L} = (L_1, L_2, L_3, L_4)$ and a partition $\mathcal{P}$ of layered width 1. Right: The corresponding quotient $G/\mathcal{P}$ resulting from contracting each part of $\mathcal{P}$.

consider in this paper are a concept introduced by Pilipczuk and Siebertz [34], which we call geodesic treewidth, and row treewidth, introduced by Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [19] and Bose, Dujmović, Javarsineh, Morin, and Wood [11].

The *geodesic treewidth* of a graph G is the smallest k for which there is a partition $\mathcal{P}$ into geodesics such that $G/\mathcal{P}$ has treewidth k, see Figure 1. This concept is first introduced by Pilipczuk and Siebertz [34] to prove structural and algorithmic results on proper minor-closed graph classes, in particular planar graphs, concerning p-centered colorings, bounded expansion, and subgraph-isomorphism. If a graph class has bounded geodesic treewidth, we say it admits *geodesic structure*.

Based on geodesic structure, Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [19] develop the concept of product structure, which led to many structural [4, 5, 7, 9, 10, 12, 14–18, 20–30] and algorithmic [2, 6, 13, 33] results. In contrast to geodesic structure, product structure does not require a single geodesic in each part, but instead disjoint unions of geodesics, where the parts are aligned by a layering. For this, a partition $\mathcal{P}$ of a graph has *layered width* 1 with respect to some layering if each part of $\mathcal{P}$ contains at most one vertex in each layer. The *row treewidth* of a graph G is the smallest k for which there is a partition $\mathcal{P}$ of layered width 1 such that $G/\mathcal{P}$ has treewidth k, see Figure 2. We say a graph class admits *product structure* if it has bounded row treewidth. Another perspective on product structure is that a graph G has row treewidth at most k if and only if G is a subgraph of the strong product $H \boxtimes P$ of a graph H of treewidth k and a path P.

Although product structure has evolved from geodesic structure, only few results are known on the relationship between these two notions. Most prominently, it is known that planar graphs and bounded-genus graphs admit both geodesic structure and product structure [19, 34]. More general, for proper-minor closed graph classes, geodesic structure implies product structure since geodesic structure implies linear local treewidth[2] (even for general graphs), which is equivalent to product structure for proper minor-closed graph classes [19]. It is also remarkable that the best known upper bounds on the geodesic treewidth and row treewidth of planar graphs is 6 in both cases and is shown with a single construction that proves both upper bounds by providing a partition into geodesics of layered width 1 [37]. Moreover, such a partition can be computed in linear time [13]. Similarly, the lower bound

[2] This follows from the definitions. See [32, Appendix A] for the definition and a brief argument.

of 3 is certified by the same graph for both parameters [19]. While these results indicate that geodesic structure and product structure coincide on some interesting graph classes, in this paper we separate the two concepts both structurally and algorithmically. In particular, we show that geodesic treewidth is interesting from an algorithmic perspective since it is less hard to compute than row treewidth.

We assume familiarity with basic graph concepts and refer to [32, Appendix A] for details.

Contributions

We compare product structure to its predecessor, geodesic structure, and find that the two are not equivalent. For this, we show that a number of interesting graph classes admit product structure but not geodesic structure, including 1-planar graphs and graphs of row treewidth 1.

▶ **Theorem 1.** [Section 2] *Product structure does not imply geodesic structure.*

In addition, it is known that determining the row treewidth is NP-hard even for treewidth-2 graphs [6], so there is no XP-algorithm parametrized in the treewidth of the input graph, unless P = NP. In contrast, we give such an XP-algorithm for determining the geodesic treewidth. In particular, this yields a polynomial-time algorithm for the geodesic treewidth of treewidth-2 graphs.

▶ **Theorem 2.** [Section 3] *There is an $O(n^{2k+3}f(k))$-time algorithm that decides for a graph G of treewidth k and $d \in \mathbb{N}$ whether G has geodesic treewidth at most d, where f is a function.*

On the other hand, row treewidth and geodesic treewidth are closely related in other aspects. On the algorithmic side, we show that geodesic treewidth, too, is NP-hard.

▶ **Theorem 3.** [Section 4] *Determining if a graph has geodesic treewidth at most k is NP-hard for $k \geq 2$.*

Complementing Theorem 1, it is an interesting question whether geodesic structure implies product structure. For the special case of graph classes with geodesic treewidth 1 we show that they also admit product structure, which is not the case in reverse, i. e., there are graph classes with row treewidth 1 but no geodesic structure.

▶ **Theorem 4.** [Section 5] *Every graph with geodesic treewidth 1 has row treewidth at most 7.*

Lastly, we consider the row treewidth and geodesic treewidth of planar graphs. The best previously known lower bound for the row treewidth and geodesic treewidth of planar graphs is 3 [37]. For geodesic structure, we show a lower bound of 5, almost matching the upper bound of 6 [6].

▶ **Theorem 5.** [Section 6] *There is a planar graph whose geodesic treewidth is at least 5.*

2 Product Structure does not imply Geodesic Structure

We show that product structure does not imply geodesic structure by observing how geodesic structure and product structure behave differently when subdividing edges.

Let $\mathcal{G}$ be a class of graphs. We define $\mathcal{G}' = \{G' \mid G \in \mathcal{G}\}$ where G' is the result of taking graph G and subdividing each edge $|V(G)|$ times. This reduces the row treewidth. This is somewhat in contrast to a result of Bose, Dujmović, Javarsineh, Morin, and Wood [11] who separate layered treewidth from row treewidth by showing that subdividing edges "efficiently" reduces the layered treewidth of a graph but not the row treewidth.

▶ **Lemma 6.** *For every graph class $\mathcal{G}$, the class $\mathcal{G}'$ has row treewidth* 1*.*

Proof. We consider a complete graph K_n and its subdivision K_n', and show that $K_n' \subseteq S \boxtimes P$ where P is a path and S is a subdivided star. In particular, the graph S is a tree and, thus, K_n' has row treewidth 1. Let P be the path on n vertices $p_1, \ldots, p_n$ and S the result of taking a star with center c and $n(n-1)/2$ leafs and subdividing each edge to become an n-vertex path. We embed the graph K_n' in $S \boxtimes P$ as follows. For the vertices $v_i \in V(K_n)$, we embed them as (c, p_i). We assign each edge v_iv_j with $i < j$ in K_n a leaf $l_{i,j}$ of S. Let $S_{i,j}$ be the path from $l_{i,j}$ to c in S. We embed the subdivided edge between v_i and v_j in K_n' in the subgraph $S_{i,j} \boxtimes P \subseteq S \boxtimes P$ without using vertices (c, p_k) for $k \notin \{i, j\}$. Note that $S_{i,j} \boxtimes P$ is an $n \times n$-grid with diagonals and indeed contains the required p_i-p_j-path with n internal vertices. Thus, it holds that $K_n' \subseteq S \boxtimes P$. Since this holds for complete graphs, it also holds for all other graphs and $\mathcal{G}'$ has row treewidth 1 for every graph class $\mathcal{G}$. ◀

For geodesic treewidth, it is not the case that subdivisions necessarily decrease it.

▶ **Lemma 7.** *For a graph class $\mathcal{G}$ that does not admit geodesic structure, the class $\mathcal{G}'$ also does not admit geodesic structure.*

Proof. Assume $\mathcal{G}'$ does admit geodesic structure with geodesic treewidth c. That is, for each $G' \in \mathcal{G}'$ there exists a partition $\mathcal{P}'$ of G' into geodesics such that $G'/\mathcal{P}'$ has treewidth at most c. We consider a geodesic $P' \in \mathcal{P}'$ and define $P = P' \cap V(G)$. We show that P is a geodesic in G. If P is not a geodesic in G, then there exists a shorter path S from the start of P to the end of P. Let S' be the path corresponding to the path S in G'. However, since G' is the result of subdividing each edge $|V(G)|$ times, it holds that S' is a shortcut for P'. Since this is impossible, it holds that P is a geodesic in G. Therefore, $\mathcal{P} = \{P' \cap V(G) \mid P' \in \mathcal{P}'\}$ is a partition of G into geodesics. It holds that $G/\mathcal{P}$ is a minor of $G'/\mathcal{P}'$ and thus has treewidth at most $\operatorname{tw}(G'/\mathcal{P}') = c$. Thus, the geodesic treewidth of G is at most c. Therefore, $\mathcal{G}'$ admitting geodesic structure implies that $\mathcal{G}$ admits geodesic structure. ◀

Note that with Lemmas 6 and 7 we obtain a rich collection of examples that admit product structure with row treewidth 1 but not geodesic structure by taking any graph class without geodesic structure and subdividing sufficiently often, for instance complete graphs.

▶ **Corollary 8.** *There are graph classes with row treewidth* 1 *and unbounded geodesic treewidth.*

As an additional corollary we observe that for every graph class $\mathcal{G}$, the class $\mathcal{G}'' = (\mathcal{G}')'$ is 1-planar since each edge of a graph $G \in \mathcal{G}$ is subdivided at least $|V(G)|^2$ times and thus can take $|V(G)|^2$ crossings. Hence, the 1-planar graphs include $\mathcal{G}''$ for $\mathcal{G}$ being the complete graphs, which do not admit geodesic structure. By Lemma 7, the latter transfers to $\mathcal{G}''$.

▶ **Corollary 9.** *The class of* 1*-planar graphs does not admit geodesic structure.*

Since 1-planar graphs admit product structure [22, Theorem 3], they provide another example for separating geodesic structure from product structure. We remark that Corollary 9 implies that many other beyond-planar graph classes that admit product structure, like k-planar graphs [22], fan-planar graphs, fan-bundle planar graphs [28], and k-matching planar graphs [26] also do not admit geodesic structure. Corollary 8 as well as the mentioned beyond-planar graph classes certify Theorem 1.

▶ **Theorem 1.** *Product structure does not imply geodesic structure.*

3 XP-Algorithm for Computing the Geodesic Treewidth

Biedl, Eppstein, and Ueckerdt [6] show that computing the row treewidth is NP-hard even for treewidth-2 graphs. In contrast, we show that the geodesic treewidth of treewidth-2 graphs can be computed in polynomial time. Moreover, we show that for all $k \in \mathbb{N}$ there is a polynomial-time algorithm that computes the geodesic treewidth of graphs with treewidth k.

▶ **Theorem 2.** *There is an $O(n^{2k+3}f(k))$-time algorithm that decides for a graph G of treewidth k and $d \in \mathbb{N}$ whether G has geodesic treewidth at most d, where f is a function.*

To show this theorem we use that treewidth-d graphs can be characterized via forbidden minors, i. e., there is a set $\mathcal{G}_d$ such that a graph has treewidth at most d if and only if it contains no graph in $\mathcal{G}_d$ as a minor, and $\mathcal{G}_d$ can be computed in time dependent only on d [1, 31]. While the best known upper bound on the size of $\mathcal{G}_d$ is doubly exponential in $O(d^5)$ [31], these lists are quite small for small d. For example, for $d = 1$ we have $\mathcal{G}_1 = \{K_3\}$ as a graph is a forest if and only if it does not contain K_3 as a minor, $\mathcal{G}_2 = \{K_4\}$ [8], and $\mathcal{G}_3$ contains K_5, the octahedron, the pentagonal prism graph, and the Wagner graph [3, 35].

To check whether G admits a partition $\mathcal{P}$ into geodesics such that the quotient $G/\mathcal{P}$ does not contain a minor from $\mathcal{G}_d$, we construct a dynamic program on a nice tree decomposition[3] of G. That is, we process the tree decomposition from bottom to top. Before we describe our algorithm, let us introduce some notation. We remark that when using the following definitions, we choose $J \subseteq G/\mathcal{P}$, where G is the input graph, $\mathcal{P}$ a partition of G into geodesics, and $\mathcal{Q}$ is a partition of the quotient J which we aim to use to identify a minor of J. During the algorithm it will be convenient to think of vertices of J as the subsets of $V(G)$ that are contracted to form the quotient. Thus, for this section we assume that for such a quotient $J \subseteq G/\mathcal{P}$ we have $V(J) \subset 2^{V(G)}$. Note that if each part of $\mathcal{Q}$ induces a connected subgraph of J, then $J/\mathcal{Q}$ is a minor of J. However, in the course of the algorithm, the parts in $\mathcal{Q}$ are not necessarily connected, which we handle with the following definitions. For a quotient $J/\mathcal{Q}$ with $\mathcal{Q}$ being a partition of J, we refer to the part $Q \in \mathcal{Q}$ that is contracted to form a vertex $v \in V(J/\mathcal{Q})$ as the *improper branch set* of v. Note that improper branch sets do not have to be connected, and thus $J/\mathcal{Q}$ is not necessarily a minor of J, hence the name *improper* branch set. Next, let $(H, \texttt{Components})$ be a tuple with H being a graph and `Components` being a function $V(H) \to 2^{2^{V(J)}}$, i. e., `Components` assigns each vertex of H a set of subsets of $V(J)$. We call $(H, \texttt{Components})$ an *improper minor* of some graph J if there is a partition $\mathcal{Q}$ of J such that $J/\mathcal{Q}$ is isomorphic to H and for each vertex $v \in V(J/\mathcal{Q})$ the improper branch set of v induces a subgraph of J whose connected components are precisely $\texttt{Components}(v)$. That is, H is a minor of J if and only if there is an improper minor $(H, \texttt{Components})$ of J such that $|\texttt{Components}(h)| = 1$ for all $h \in V(H)$. In our algorithm we use improper minors to keep track of possible minors of a quotient of our input graph. However, we remark that while it is important for us to maintain some improper minors with disconnected improper branch sets, at the end of the algorithm we check whether the improper minors are minors, so they should be thought of as potential minors.

During the algorithm, for each bag B of the tree decomposition of G and the subtree T_B below B (including B) we store the following: First, we store some information about all possible partitions of $G[T_B]$ into geodesics, where $G[T_B]$ is the graph induced by the bags of T_B. Clearly, we cannot store all of these partitions as the number of these partitions

[3] A *nice tree decompositions* has leaf bags, introduce bags, forget bags, and join bags. We refer to [32, Appendix A] for a formal definition.

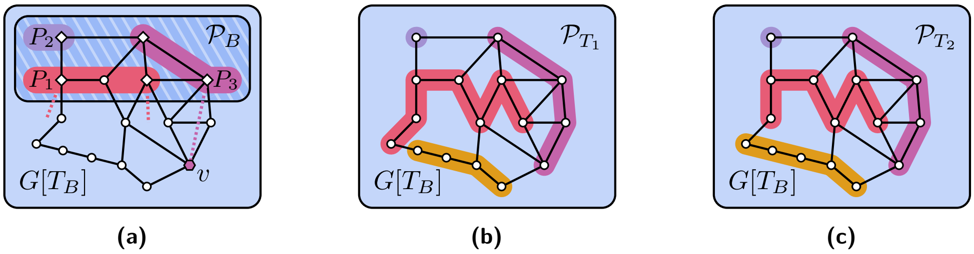

Figure 3 (a) An example for a bag B (hatched blue), a graph $G[T_B]$, and a partition $\mathcal{P}_B$ of B with the border-vertices of each part depicted as diamonds. The part P_1 has two end-vertices outside of B (that we do not store) indicated by the dashed lines. The part P_2 has no end-vertices outside of B. The part P_3 has one end-vertex v outside of B depicted as an 8-gon in the same color as P_3. It is associated with one of the border-vertices of P_3 indicated by the dashed line. **(b,c)** Two different partitions P_{T_1} and P_{T_2} of $G[T_B]$ into geodesics that results in $\mathcal{P}_B$ if restricted to B.

is exponential in n. Instead, for each partition $\mathcal{P}_T$ of $G[T_B]$ into geodesics, we store a partition $\mathcal{P}_B$ of B with some additional information. For each geodesic $P_T \in \mathcal{P}_T$ that is not disjoint from B we store a part P_B in $\mathcal{P}_B$ that contains the vertices $V(P_T) \cap B$ that are ordered according to their order in P_T. We refer to the first and last of these vertices as the *border-vertices* of P_B. Then, for each part P_B with corresponding geodesic P_T in $G[T_B]$ we store in a label attached to P_B whether both end-vertices of P_T are not in B, both end-vertices of P_B are in B or exactly one end-vertex v of P_T is not in B. In the last case we also store v and the border-vertex of P_B it is closest to. Figure 3 shows an example.

Note that this gives us at most $2n$ possible labels for each part P_B as there are $n-1$ options for the end-vertex, combined with two options for the border-vertex, and additionally the two options that none or both end-vertices are in B. Since each partition of B has at most $|B| \leq k+1$ parts, there are at most $(2n)^{k+1}$ different combinations of labels for a fixed partition of B. While we need to store more things, everything in the following only depends on k. Thus, since the number of partitions of B also depends only on k, in total we obtain $O(n^{k+1} \cdot f(k))$ distinct configurations for some function f.

In addition to the information about a partition $\mathcal{P}_T$ of $G[T_B]$ we already have, we store information about improper minors of $G[T_B]/\mathcal{P}_T$. To do this, let us define $\mathcal{S}_d$ as the set of all labeled subgraphs of graphs in $\mathcal{G}_d$. That is, we distinguish two isomorphic subgraphs if they are on different vertex-sets, e. g., a graph $H \in \mathcal{G}_d$ has $|V(H)|$ distinct subgraphs that consist of a single vertex. Recall that an improper minor of $J = G[T_B]/\mathcal{P}_T$ is a tuple $(H, \texttt{Components})$ with H being a graph and $\texttt{Components}$ being a function $V(H) \to 2^{2^{V(J)}}$. Our goal is to store all improper $\mathcal{S}_d$-minors of $G[T_B]/\mathcal{P}_T$, i. e., all improper minors $(H, \texttt{Components})$ of $G[T_B]/\mathcal{P}_T$ with $H \in \mathcal{S}_d$. However, since we want the number of these to depend only on k, we instead store something slightly different. That is, for each improper $\mathcal{S}_d$-minor $(H, \texttt{Components})$, we store the *B-improper minor* which is the tuple $(H, \texttt{Components}')$ with $\texttt{Components}'(v) = \{\{P \cap B \colon P \in C\} \colon C \in \texttt{Components}(v)\}$ for each vertex $v \in V(H)$, where $\texttt{Components}'(v)$ is a multi-set and $|\texttt{Components}'(v)| = |\texttt{Components}(v)|$. This means we only store the intersection of each geodesic in $\mathcal{P}_T$ with B instead of storing the entire geodesic. Moreover, if there is a vertex $v \in V(H)$ such that $\{\emptyset\} \in \texttt{Components}'(v)$ and $|\texttt{Components}'(v)| > 1$ we say that $(H, \texttt{Components}')$ is an *incompletable* B-improper minor and discard the tuple instead (cf. Figure 4). Otherwise, we refer to $(H, \texttt{Components}')$ as a *completable* B-improper minor of $G[T_B]/\mathcal{P}_T$. Note that for a completable B-improper minor $(H, \texttt{Components}')$, the set $\texttt{Components}'(v)$ contains each set only once for every $v \in V(H)$. Indeed, the multi-set is only needed to properly recognize the case that $\texttt{Components}'(v)$

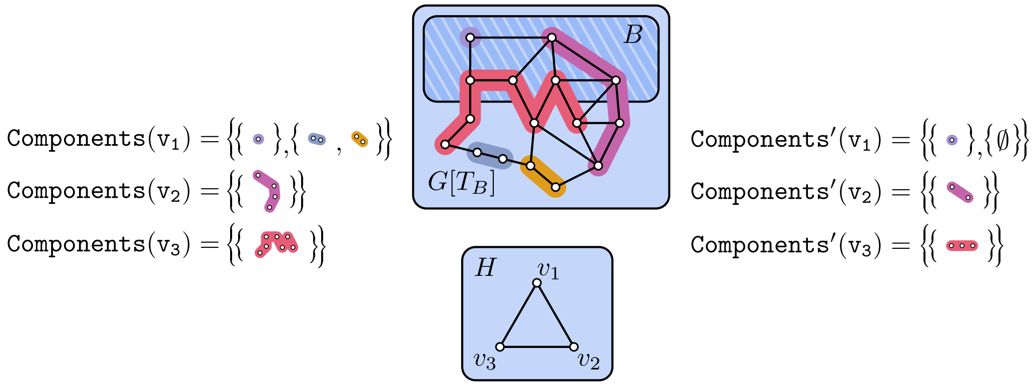

Figure 4 Middle: A graph $G[T_B]$ with a partition into geodesics and a graph H. Left: An improper minor $(H, \texttt{Components})$. Right: The corresponding (incompletable) B-improper minor $(H, \texttt{Components}')$. Each colored element in $\texttt{Components}$ and $\texttt{Components}'$ is a set of vertices.

contains $\{\emptyset\}$ more than once for some vertex $v \in V(H)$. This does not occur during the algorithm and, thus, we ignore it in the following and always assume that $\texttt{Components}'$ maps to a set. Note that the number of completable B-improper minors depends only on k.

In total for each partition $\mathcal{P}_T$ of $G[T_B]$ into geodesics with the set of improper $\mathcal{S}_d$-minors of $G[T_B]/\mathcal{P}_T$ we define a *configuration* of the bag B that consists of the following things:

- A partition $\mathcal{P}_B$ of B (corresponding to $\mathcal{P}_T$) with each part $P \in \mathcal{P}_B$ being ordered,
- a set L of labels indexed by the parts $P \in \mathcal{P}_B$ that tells us that two or none end-vertices associated with P are outside of B, or that associates a border-vertex of P with a vertex outside of B, and
- a set M of completable B-improper $\mathcal{S}_d$-minors (corresponding to the improper $\mathcal{S}_d$-minors of $G[T_B]/\mathcal{P}_T$).

Taking the union over all partitions $\mathcal{P}_T$ of $G[T_B]$ into geodesics, we obtain the set of all configurations of B. Before computing the configurations, we argue why it suffices to do so.

▶ **Lemma 10.** *Let G be a graph with a nice tree decomposition with root bag B. Then, the graph G has geodesic treewidth at most d if and only if B has a configuration with a set M of completable B-improper $\mathcal{S}_d$-minors such that for each $(H, \texttt{Components}') \in M$ we have $H \notin \mathcal{G}_d$ or $|\texttt{Components}'(v)| > 1$ for a vertex $v \in V(H)$.*

Proof. Let G have geodesic treewidth at most d. Then, by definition, there is a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ has treewidth at most d. As $\mathcal{G}_d$ contains precisely the forbidden minors of treewidth d, the graph $G/\mathcal{P}$ contains no graph $H \in \mathcal{G}_d$ as a minor. Thus, by the definition of improper minors, we have for every improper minor $(H', \texttt{Components})$ of $G/\mathcal{P}$ that $H' \notin \mathcal{G}_d$ or $|\texttt{Components}(v)| > 1$ for a vertex $v \in V(H')$. In the first case we are done, and in the second we have $|\texttt{Components}'(v)| = |\texttt{Components}(v)| > 1$.

Now, let G have geodesic treewidth at least $d+1$. Then, by the definition of geodesic treewidth, for every partition $\mathcal{P}$ of G into geodesics the graph $G/\mathcal{P}$ has treewidth at least $d+1$. Thus, the graph $G/\mathcal{P}$ contains some graph from $\mathcal{G}_d$ as a minor. Now, consider a configuration of B with partition $\mathcal{P}_B$ of B and a set M of completable B-improper $\mathcal{S}_d$-minors. Our goal is to show that there is a completable B-improper minor $(H, \texttt{Components}') \in M$ with $H \in \mathcal{G}_d$ and $|\texttt{Components}'(v)| = 1$ for each $v \in V(H)$. To do this recall that by definition, a configuration results from a partition $\mathcal{P}$ of $G[T_B] = G$. By assumption, we have $\mathrm{tw}(G/\mathcal{P}) \geq d+1$ and thus $G/\mathcal{P}$ has a minor $H \in \mathcal{G}_d$. Now let $(H, \texttt{Components})$

be the improper minor with $|\texttt{Components}(v)| = 1$ for each $v \in V(H)$ obtained from the branch sets of H, which exists by definition. Consider the corresponding B-improper minor $(H, \texttt{Components}')$. Since $|\texttt{Components}'(v)| = |\texttt{Components}(v)| = 1$ for each $v \in V(H)$, we know that $(H, \texttt{Components}')$ is completable and, thus, that $(H, \texttt{Components}') \in M$. ◀

Let us now come to a summary of how our algorithm computes the set of configuration for all bags of a nice tree decomposition of width k. A detailed description and a proof of correctness is in [32, Appendix B]. We start with a pre-computation of the distances between any two vertices, verify $d \leq k$ (we are done otherwise), and compute the sets $\mathcal{G}_d$ and $\mathcal{S}_d$.

Leaf bags are brute-forced. For an introduce bag B with child $B_1 = B - \{v\}$ and a configuration with a partition $\mathcal{P}_1$ of B_1, we consider every way to add v. We can add $\{v\}$ as a new part, add it to an existing part (if it remains a geodesic) or merge two parts with v in between (if this results in a geodesic). For a resulting partition of B we construct a configuration of B by iterating over all completable B_1-improper minors in the configuration of B_1 and consider every way to add v to them. Similarly, we handle forget bags by checking whether removing the vertex yields a completable B-improper minor. For a join bag B with children B_1 and B_2 we iterate over all pairs of configurations of B_1 and B_2 with partitions $\mathcal{P}_1$, $\mathcal{P}_2$. If $\mathcal{P}_1 = \mathcal{P}_2$ and the labels agree, we check for all pairs of completable B_1- and B_2-improper minors whether we can combine them to a completable B-improper minor.

We remark that the join bags are crucial for the runtime, taking $O(n^{2k+2} \cdot f(k))$ time for all pairs of $O(n^{k+1} \cdot f(k))$ configurations. With $O(n)$ bags, we obtain a runtime of $O(n^{2k+3} \cdot f(k))$. We discuss in [32, Remark 24] that the geodesics are a bottleneck as we obtain an $O(n \cdot f(k))$ algorithm if we want to partition G into any induced paths instead of geodesics.

4 Computing the Geodesic Treewidth is NP-Hard

We complement our algorithm by showing that, similar to row treewidth, it is NP-hard to decide whether a graph has geodesic treewidth 2. For this, we give a reduction from a SAT variant that is known to be NP-complete. Tovey [36] shows that SAT is NP-complete if every clause contains 2 or 3 variables and every variable occurs at most once in a clause of size 3 and at most twice in clauses of size 2. We consider such an instance I of SAT with clauses C and variables U and construct the following graph. We begin with a path P of length $2n + 1$ where $n = |U|$. Let the first vertex on this path be called s and the last t. Beginning with the vertex adjacent to s we now duplicate every other vertex on the path, meaning we add new vertices with the same neighborhood as the vertices we are duplicating. Let the resulting graph be called G_1 and let $\{(x_i, \overline{x_i}) \mid x_i \in P \wedge \overline{x_i}$ is the duplicate vertex of $x_i\}$ be the pairs of vertices that are duplicated. An example of such a graph G_1 is shown in Figure 5 on the left. For an s-t-path S, we say $x_i \in U$ is true if and only if $x_i \in S$. Thus, every path S corresponds uniquely to a truth assignment of the variables U.

For each clause $c \in C$ and literals $l_1, l_2 \in c$, we add the edge $l_1 l_2$ to G_1 to construct G_2 as shown in Figure 5 in the center. That is, each clause in C corresponds to a clique in G_2. In addition, for each clause $\{l_1, l_2\} \in C$ of size 2 we add five new vertices $p_1, \ldots, p_5$ that are adjacent to l_1, l_2, and s. The resulting graph G_3 can be seen in Figure 5 on the right. Lastly, to construct G_4, every edge $e \in E(G_3) \setminus E(G_1)$ is subdivided $2n$ times. As a result, every shortest path from s to t in G_4 only uses edges of G_1 and, thus, we retain the unique correspondence between shortest paths from s to t and truth assignments of the variables.

For a graph G with $s, t \in V(G)$, let an *s-t-partition* of G into geodesics be a partition into geodesics that contains a shortest s-t-path. We refer to [32, Appendix C] for a detailed version the following proof.

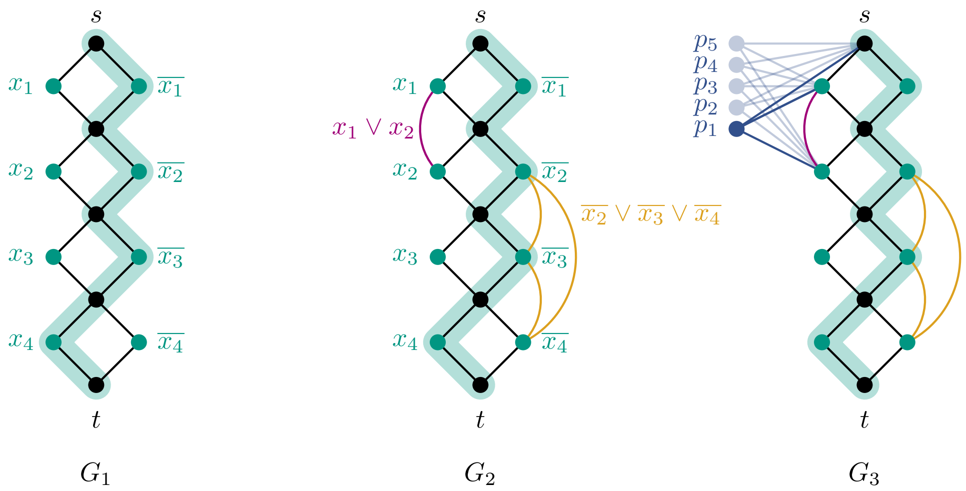

Figure 5 Left: An example of G_1 with a shortest s-t-path highlighted in green that uniquely corresponds to the truth assignment φ with $\varphi(x_4) = t$ and $\varphi(x_i) = f$ for $i \in [3]$. Middle: A corresponding G_2 with a SAT instance consisting of two clauses. Right: A corresponding G_3.

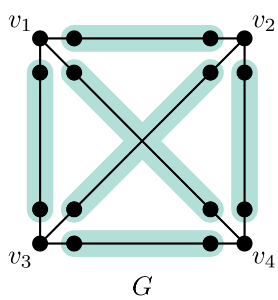

Figure 6 K_4 with vertices $v_1, \dots v_4$ where each highlighted edge gets replaced by a copy of G_4.

▶ **Lemma 11.** *The instance I is satisfiable if and only if there exists an s-t-partition $\mathcal{P}$ of G_4 into geodesics such that $G_4/\mathcal{P}$ has treewidth at most 2.*

Proof sketch. If I is not satisfiable then for every s-t-partition with shortest s-t-path S there is a clause $c \in C$ such that no literal of c is in S. It can be seen that the geodesics containing the literals of c, the geodesics containing the corresponding vertices $p_1, \dots, p_5$ (if c is a clause of size 2), and the s-t-geodesic form a K_4 minor.

If I is satisfiable then deleting the vertices of an s-t-path S corresponding to a satisfying truth assignment results in the disjoint union of some trees and paths. Thus, for any partition $\mathcal{P}$ of G_4 into geodesics that contains S, the graph $G_4/\mathcal{P}$ has treewidth at most 2. ◀

Finally, we construct G by subdividing a K_4 with vertices $v_1, \dots, v_4$ twice and replacing the six edges whose endpoints are subdivision vertices by copies of G_4 as shown in Figure 6.

▶ **Lemma 12.** *The graph G has geodesic treewidth at most 2 if and only if I is satisfiable.*

Proof sketch. If I is satisfiable we take an s-t partition $\mathcal{P}$ of G_4 such that $G_4/\mathcal{P}$ has treewidth at most 2 (Lemma 11) for each copy of G_4 in G. Then, we include $v_1, \dots, v_4$ in some of the s-t-paths as depicted in Figure 7a. This yields a partition $\mathcal{P}'$ into geodesics such that $G/\mathcal{P}'$ (Figure 7b) is a $K_{2,4}$ with some attached copies of $G_4/\mathcal{P}$ and, thus, has treewidth at most 2.

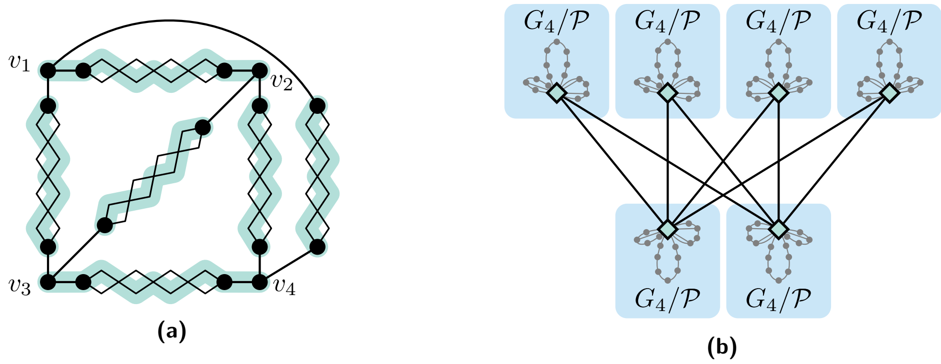

Figure 7 **(a)** A sketch of G with possibly extended s-t-paths highlighted in green, **(b)** the corresponding graph $G/\mathcal{P}'$. The diamond vertices correspond to the green geodesics in **(a)**.

If I is not satisfiable, by Lemma 11 no s-t partition $\mathcal{P}$ of G_4 yields $\text{tw}(G_4/\mathcal{P}) \leq 2$. Then, v_1, v_2, v_3, and v_4 are in distinct geodesics, giving a K_4-minor which has treewidth 3. ◀

▶ **Corollary 13.** *Determining whether a graph has geodesic treewidth at most* 2 *is* NP*-hard.*

Next we generalize the NP-hardness result from 2 to k and show that determining whether a given graph has geodesic treewidth at most k is NP-hard. For a graph G, we define the graph $G^\star$ as follows: connect three disjoint copies G_1, G_2, G_3 of G by a universal vertex v and for each edge adjacent to v subdivide it $|V(G)|$ times.

▶ **Lemma 14.** *G has geodesic treewidth k if and only if $G^\star$ has geodesic treewidth $k+1$.*

Proof sketch. Due to the subdivisions, geodesics in G are also geodesics in $G^\star$, so we may reuse the partition of G into geodesics and put v and the subdivision vertices into its own geodesic each. For the other direction, we consider a partition $\mathcal{P}^\star$ of $G^\star$ into geodesics and find a suitable partition for G by restricting $\mathcal{P}^\star$ to one of the copies. Refer to [32, Appendix C] for a proof that the described partitions indeed yield quotients of treewidth $k+1$, respectively k. ◀

Now, the following theorem follows immediately from Corollary 13 and Lemma 14.

▶ **Theorem 3.** *Determining if a graph has geodesic treewidth at most k is* NP*-hard for $k \geq 2$.*

5 Geodesic Treewidth 1 implies Product Structure

In this section, we show that graph classes with geodesic treewidth 1 have bounded row treewidth. However, whether geodesic structure implies product structure in general remains an open question. Note that this is in contrast to Corollary 8, which shows that there are graphs with row treewidth 1 but unbounded geodesic treewidth.

For this, we define a *c-layering* $\mathcal{L}$ of a graph G as an ordered partition $(L_0, L_1, \dots)$ of $V(G)$ into *layers* such that for every edge $vw \in E(G)$ with $v \in L_i$ and $w \in L_j$ we have $|i-j| \leq c$. We say a partition $\mathcal{P}$ of some graph G has *c-layered width* d if there is a c-layering $\mathcal{L}$ such that each part of $\mathcal{P}$ has at most d vertices in each layer of $\mathcal{L}$. Note that a 1-layering is a layering and analogously 1-layered width coincides with layered width. Similarly to layered width [19], the notion of c-layered width turns out useful to upper-bound the row treewidth and thereby to prove Theorem 4.

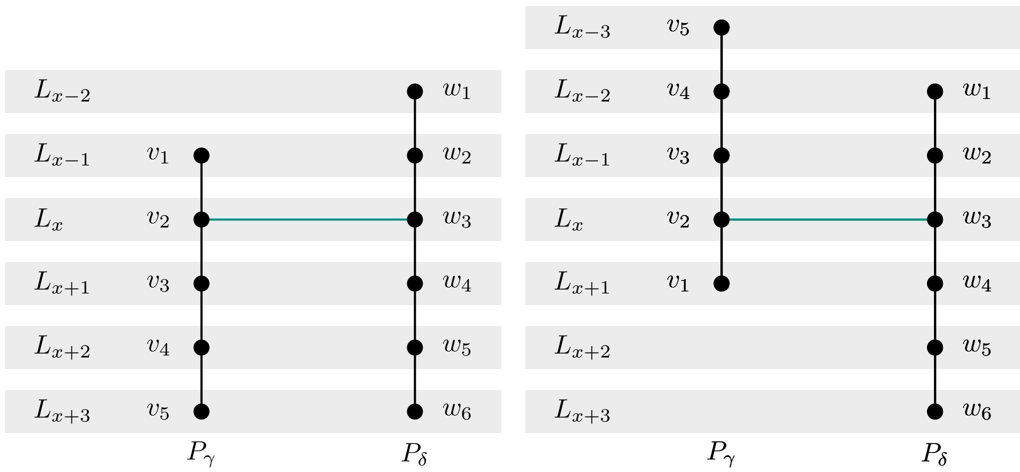

Figure 8 Layering $\mathcal{L}_1$ on the left and layering $\mathcal{L}_2$ on the right. The green edge is the alignment edge $e = v_i w_j$, where $i = 2, j = 3$ in this example.

▶ **Lemma 15.** *Let G be a graph with a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree. Then, $\mathcal{P}$ has 4-layered width 1.*

Proof sketch. For a partition $\mathcal{P}$ of some graph into geodesics, we call a c-layering $\mathcal{L}$ *aligned* if for each geodesic $P \in \mathcal{P}$ the vertices of P are embedded in consecutive layers of $\mathcal{L}$ according to their order in P. Note that only edges in geodesics are required to have their endpoints in two consecutive layers, whereas edges between distinct geodesics are allowed skip up to $c-1$ layers. We prove the following stronger statement by induction on the number of geodesics in $\mathcal{P}$. Note that since by definition, every aligned c-layering has only one vertex of each geodesic in each layer, Claim 16 implies that the 4-layered width of $\mathcal{P}$ is 1.

▷ Claim 16. For every graph G and every partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree, there is an aligned 4-layering.

If $|\mathcal{P}| = 1$, then there is an aligned 4-layering of G by putting each vertex into its own layer, ordered according to the single geodesic in $\mathcal{P}$. Next, consider a graph G with a partition $\mathcal{P}$ into at least two geodesics such that $G/\mathcal{P}$ is a tree. Let $\gamma, \delta \in V(G/\mathcal{P})$ with γ being a leaf adjacent to δ. Let P_γ and P_δ be the geodesics corresponding to γ and δ, respectively, with $P_\gamma = (v_1, \dots, v_m)$ and $P_\delta = (w_1, \dots, w_n)$. Let $\mathcal{L} = (L_0, L_1, \dots)$ be an aligned 4-layering of $G - P_\gamma$ which we obtain by induction. We construct two aligned c-layerings $\mathcal{L}_1, \mathcal{L}_2$ of G from $\mathcal{L}$ and show that one of them indeed is an aligned 4-layering. For this, note that since γ is adjacent to δ in $G/\mathcal{P}$, there exist $i \in [m]$ and $j \in [n]$ such that $e = v_i w_j \in E(G)$. We denote the layer in $\mathcal{L}$ that contains w_j by L_x and use the edge e to align the path P_γ. That is, we put v_i in layer L_x and obtain two options how to continue P_γ from there. The two options are shown in Figure 8 and yield the layerings $\mathcal{L}_1$ and $\mathcal{L}_2$. For a layering $\mathcal{L}'$ let a *k-steep* edge be an edge with endpoints in layers L_a, L_b of $\mathcal{L}'$ and $|a - b| = k$.

We show that $\mathcal{L}_1$ or $\mathcal{L}_2$ is an aligned 4-layering, i.e., all edges are at most 4-steep. The key arguments are first that edges which do not cross e in the embedding of Figure 8 are at most 2-steep (Figure 9a). And second, if $\mathcal{L}_1$ contains a 5-steep edge e_1, then every edge e_2 crossing e in $\mathcal{L}_2$ has its endpoints closer to e than e_1 (Figure 9b). Combining these arguments, one can show that e_2 is at most 4-steep in $\mathcal{L}_2$ by otherwise finding a shortcut for P_δ using e_1 and e_2 (see [32, Appendix D]). ◀

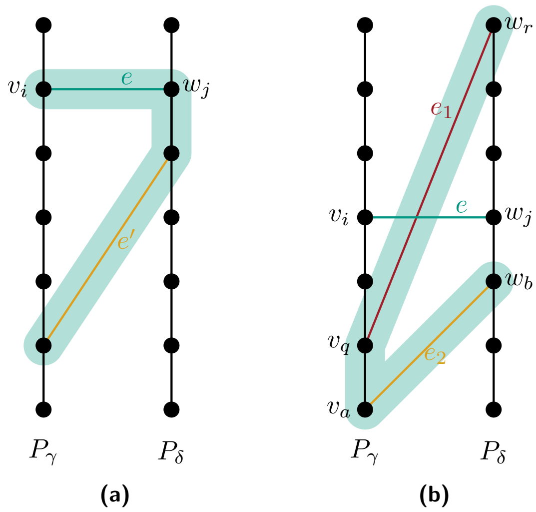

Figure 9 Key arguments for Lemma 15. Recall that geodesics, in particular P_γ and P_δ, do not have shortcuts. **(a)** If a non-crossing edge e' is 3-steep, then together with edge e, this results in a shortcut, highlighted green. **(b)** Embeddings according to $\mathcal{L}_1$ and assuming $a \geq q$. Recall that e_2 crosses e in $\mathcal{L}_2$ and thus does not cross e in $\mathcal{L}_1$. Since e_2 is at most 2-steep and e_1 is at least 5-steep, the edges e_1 and e_2 result in a shortcut for P_δ that is highlighted in green.

It is now left to show that Lemma 15 implies Theorem 4.

▶ **Theorem 4.** *Every graph with geodesic treewidth* 1 *has row treewidth at most* 7*.*

Proof. Let G be a graph with a partition $\mathcal{P}$ of G into geodesics such that $T = G/\mathcal{P}$ is a tree. By Lemma 15, there is a 4-layering $\mathcal{L}$ of G such that $\mathcal{P}$ has layered width 1 with respect to $\mathcal{L}$. We partition the layers of $\mathcal{L}$ into consecutive blocks of size 4, where block k contains layers $4k$ to $4(k+1)-1$, for $k \in \mathbb{Z}$, and merge the layers in a block to get a layering $\mathcal{L}'$. Note that $\mathcal{P}$ has layered width 4 with respect to $\mathcal{L}'$. Thus $G \subseteq T \boxtimes P \boxtimes K_4$ [19, Obs. 35] and therefore G has row treewidth at most $\operatorname{tw}(T \boxtimes K_4) \leq (1+1) \cdot 4 - 1 = 7$. ◀

6 Lower Bound for Geodesic Treewidth of Planar Graphs

Ueckerdt, Wood, and Yi [37] show that every planar graph has row treewidth at most 6 by providing a partition into so-called vertical paths, i.e., paths along a BFS-tree. In addition, Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [19] give a lower bound by constructing a planar graph with row treewidth 3. It is asked by different authors [27,37] to close the gap between these two bounds for planar graphs or subclasses thereof. Interestingly, the paths in [37] not only adhere to a layering, which suffices for product structure, but they are also geodesics. That is, [37] also provides an upper bound of 6 on the geodesic treewidth and, even stronger, a partition of any planar graph into paths that certifies both upper bounds simultaneously. Similarly, we remark that the lower-bound construction in [19] also works for geodesic treewidth. In this section, we attack this gap between 3 and 6 by constructing a planar graph with geodesic treewidth 5. We consider it an interesting open question whether our construction can be adapted to also improve the lower bound on the row treewidth or, most intriguing, to a graph that lower-bounds both parameters at the same time.

▶ **Theorem 5.** *There is a planar graph whose geodesic treewidth is at least* 5*.*

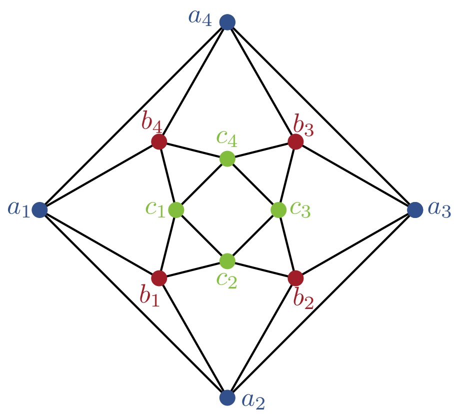

Figure 10 The cuboctahedron is known to have treewidth 5.

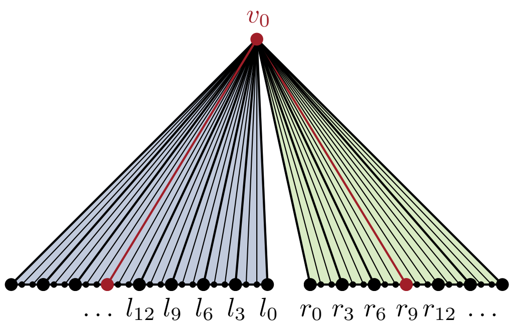

Figure 11 Two fans at v_0 in G_1, where the left fan is highlighted in blue and the right fan in green. A possible geodesic containing v_0 is drawn in red. Note that every geodesic has at most three vertices in any fan. Thus, the thick vertices that are not red belong to pairwise distinct geodesics.

We remark that the treewidth in Theorem 5 is indeed exactly 5, but we only prove the lower bound. To do so, we construct a sequence of graphs $G_1 \subseteq G_2 \subseteq G_3$ and show that for every partition $\mathcal{P}$ of G_3 into geodesics, the quotient $G_3/\mathcal{P}$ contains the so-called cuboctahedron (Figure 10) as a minor, a planar graph of treewidth 5. While constructing G_3, we outline how to find the cuboctahedron step by step. For a full proof refer to [32, Appendix E], where we also describe and analyze the construction in more detail. For an overview, G_1 yields the blue and the green cycle in Figure 10, consisting of the a-vertices, respectively, c-vertices. Then G_2 adds the red b-vertices to the green cycle, and finally G_3 provides the edges between b- and c-vertices.

During the construction, we make sure that every newly created path between already present vertices is strictly longer than the shortest path between them. In particular, distances are preserved and we do not obtain new geodesics between previously introduced vertices. Hence, observations concerning geodesics of some G_i also hold in subsequent supergraphs.

We start with a triangle whose vertices are called v_0, v_1, and v_2. Note that for every partition of a triangle into geodesics, at least two of the three vertices are in distinct geodesics. We aim to lift this property to larger parts of our construction. That is, we describe a graph of which we add three copies, one for each vertex of the initial triangle, and show that for every partition into geodesics, there are two copies that do not share geodesics. In these two copies, we then find the desired cuboctahedron-minor.

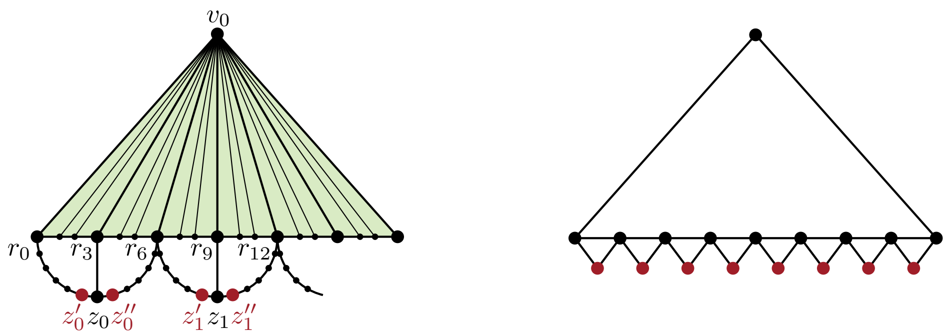

Figure 12 Left: A right fan in G_2 with z_k-attachments. Note that for all k, either z_k' or z_k'' are part of a geodesic that contains no vertices in the fan. As an example, the z_0-attachment consists of the vertices $z_0, r_0, \dots, r_6$ and the length 6-paths in between that contain z_0' and z_0''. Right: The cycle with stacked vertices that is contained in the quotient $G_2/\mathcal{P}$ for every partition $\mathcal{P}$.

To construct G_1 we add two fans at each v_i, a *left fan* and a *right fan*, whose vertices are labeled as shown in Figure 11. Observe that two vertices whose distance in the path of the fan is at least 3 can only be in the same geodesic if their geodesic contains the center. Since the fans are separated from the remainder of the graph by their center, this transfers to G_1.

▶ **Observation 17.** *For every partition of G_1 into geodesics and each of the three pairs of fans, the vertices r_{3k}, ℓ_{3k} for $k \geq 0$ are in pairwise distinct geodesics, except for the vertices that are in the same geodesic as the center.*

We use Observation 17 to find cycles which serve as a base to build the cuboctahedron. Indeed, for every partition $\mathcal{P}$ of G_1 in geodesics, the quotient $G_1/\mathcal{P}$ contains six arbitrarily large cycles, at least one for each fan, provided the fans are sufficiently large.

We extend G_1 to G_2 by adding gadgets, called *z_k-attachments*, with three dedicated vertices z_k, z_k', z_k'' to the right fans as shown in Figure 12. Observe that for each k, the vertices z_k' and z_k'' have distance 3 to v_i, and thus distance at most 4 to every vertex of their fan. Hence, the shortest path to any vertex in the right fan of v_i includes the vertex r_{6k+3}. Since a geodesic containing r_{6k+3} can contain only one of z_k' and z_k'', we obtain the following.

▶ **Observation 18.** *For every partition of G_2 into geodesics, each of the three right fans F, and every k, the geodesic with z_k' or the geodesic with z_k'' contains no vertices of F.*

This allows us to extend the cycle from the quotient of G_1 by stacking vertices on the edges of the cycle as shown in Figure 12 (right). The additional vertices are obtained from the geodesic containing z_k', respectively z_k'', provided by Observation 18, whereas the other of the two vertices is not used.

Finally, to construct G_3 we connect the three pairs of fans with each other as shown in Figure 13. To verify that distances are preserved, recall that in G_2 the distance between z_k', respectively z_k'', to the center of its fan is 3, and hence at most 5 to any vertex in a fan. Since the newly added paths in G_3 are of length 6, all paths between vertices of G_2 using a new vertex are strictly longer than the shortest path in G_2. Moreover, for each two vertices in different copies, the shortest path contains the respective centers.

▶ **Observation 19.** *For every partition of G_3 into geodesics, it holds that every geodesic that contains vertices of two distinct copies C_i and C_j, $i, j \in \{0, 1, 2\}$, also contains v_i and v_j.*

In particular, since v_0, v_1, v_2 induce a triangle and thus cannot be in a single geodesic, at least two copies do not share geodesics. We use these two copies to find the desiredcuboctahedron.

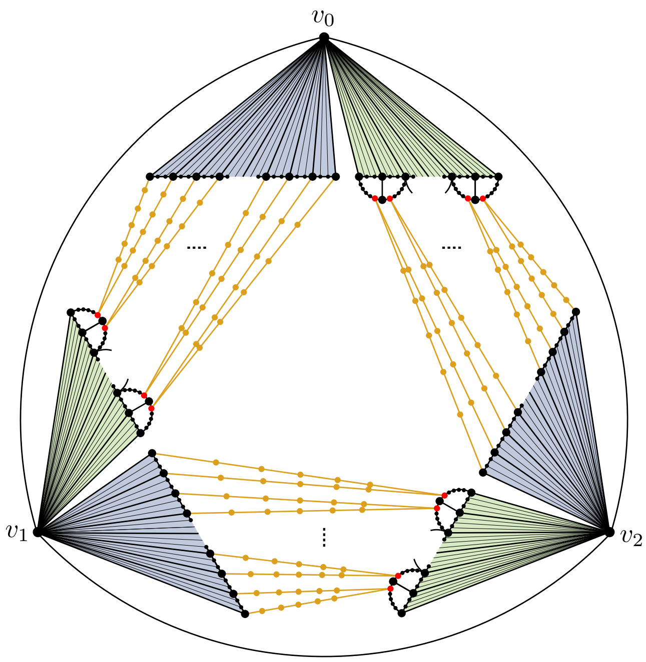

Figure 13 The planar graph G_3, which has geodesic treewidth 5. The right fans are highlighted in green and the left fans are highlighted in blue. The yellow paths are newly added in G_3.

▶ **Lemma 20.** *For every partition $\mathcal{P}$ of G_3, the quotient $G_3/\mathcal{P}$ contains the cuboctahedron as a minor. In particular, the geodesic treewidth of G_3 is at least* 5*.*

Proof sketch. Observations 17–19 give two copies such that no geodesic contains vertices of both copies as outlined during the construction. Figures 14 and 15 show how to find the cuboctahedron within these two copies. For a detailed proof, refer to [32, Appendix E]. ◀

7 Open Questions

While we show that product structure does not imply geodesic structure, it is an intriguing open question whether geodesic structure implies product structure. To this end, we show a positive result for geodesic treewidth 1. That is, a possible counterexample needs to be a

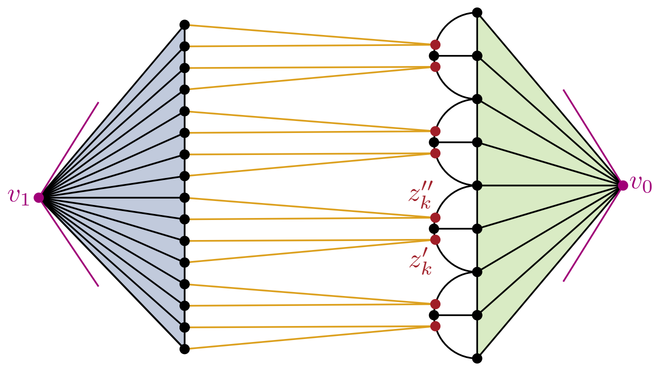

Figure 14 A left sub-fan (blue) and a right sub-fan (green). Subdivision vertices are not drawn. The geodesics containing v_0 and v_1 (purple) do not intersect the sub-fans besides in v_0 and v_1.

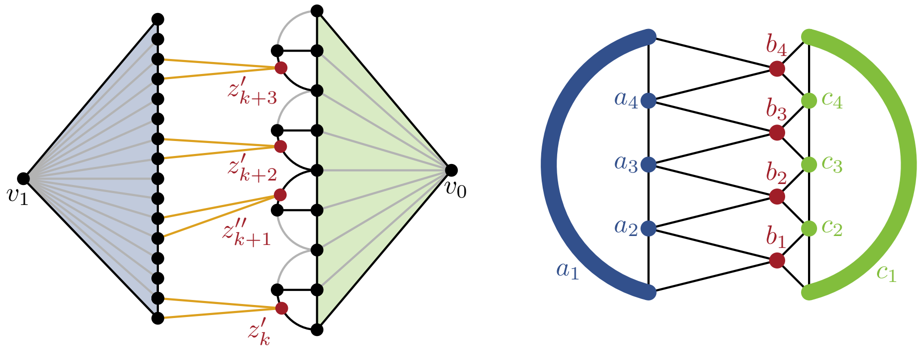

Figure 15 Left: Parts of the left fan of v_1 (blue) and the right fan of v_0 (green). Some vertices are omitted for readability. All drawn vertices of the fans and $z'_k, z''_{k+1}, z'_{k+2}$ and z'_{k+3} belong to pairwise distinct geodesics. Gray edges are not used. Right: Alternative drawing of the cuboctahedron. Note that c_1 and a_1 are vertices that are drawn as half-circles to match the embedding of G_3 on the left.

graph class that is not minor-closed and has geodesic treewidth at least 2. We remark that both a positive and a negative answer may have interesting consequences. If the two notions are incomparable, i. e., neither implies the other, there is a chance to handle graph classes that do not admit product structure. On the other hand, if geodesic structure is strictly stronger than product structure, there is hope to strengthen results obtained by the latter.

In light of the different behavior of geodesic treewidth and row treewidth in their computational complexity, we ask whether determining the geodesic treewidth is FPT in the treewidth of the input graph. Note that by the NP-hardness result of Biedl, Eppstein, and Ueckerdt [6], row treewidth cannot be expected to admit such an FPT-algorithm. In addition, we remark that the concept of geodesic structure was developed to compute p-centered colorings for graphs of proper minor-closed graph classes and based on this to solve the subgraph-isomorphism problem on this graph class [34]. We ask for further problems that can be solved more efficiently on graphs with bounded geodesic treewidth than in general.

Finally, we ask for the maximum geodesic treewidth and row treewidth of planar graphs, which is between 5 and 6, respectively 3 and 6, by Theorem 5 and [19, 37].

References

1 Isolde Adler, Martin Grohe, and Stephan Kreutzer. Computing excluded minors. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 641–650, USA, 2008. Society for Industrial and Applied Mathematics. URL: `http://dl.acm.org/citation.cfm?id=1347082.1347153`.

2 Shinwoo An, Seonghyuk Im, Seokbeom Kim, and Myounghwan Lee. An efficient algorithm for $\mathcal{F}$-subgraph-free edge deletion on graphs having a product structure, 2025. `doi:10.48550/arXiv.2510.14674`.

3 Stefan Arnborg, Andrzej Proskurowski, and Derek G. Corneil. Forbidden minors characterization of partial 3-trees. *Discrete Mathematics*, 80(1):1–19, 1990. `doi:10.1016/0012-365X(90)90292-P`.

4 Michael A. Bekos, Giordano Da Lozzo, Petr Hliněný, and Michael Kaufmann. Graph product structure for h-framed graphs. *The Electronic Journal of Combinatorics*, pages P4–55, 2024. `doi:10.37236/12123`.

5 Michael A. Bekos, Martin Gronemann, and Chrysanthi N. Raftopoulou. An Improved Upper Bound on the Queue Number of Planar Graphs. *Algorithmica*, 85:544–562, February 2023. `doi:10.1007/s00453-022-01037-4`.

6 Therese Biedl, David Eppstein, and Torsten Ueckerdt. On the complexity of embedding in graph products. *Computing in Geometry and Topology*, 4(2):5:1–5:18, May 2025. `doi:10.57717/cgt.v4i2.56`.

7 Thomas Bläsius, Emil Dohse, Deborah Haun, and Laura Merker. Product Structure and Treewidth of Hyperbolic Uniform Disk Graphs. In Hee-Kap Ahn, Michael Hoffmann, and Amir Nayyeri, editors, *42nd International Symposium on Computational Geometry (SoCG 2026)*, volume 367 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 18:1–18:17, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.SoCG.2026.18`.

8 Hans L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209(1):1–45, 1998. `doi:10.1016/S0304-3975(97)00228-4`.

9 Marthe Bonamy, Cyril Gavoille, and Michał Pilipczuk. Shorter Labeling Schemes for Planar Graphs. *SIAM Journal on Discrete Mathematics*, 36(3):2082–2099, September 2022. Publisher: Society for Industrial and Applied Mathematics. `doi:10.1137/20M1330464`.

10 Édouard Bonnet, O-joung Kwon, and David R. Wood. Reduced bandwidth: A qualitative strengthening of twin-width in minor-closed classes (and beyond). *Journal of Combinatorial Theory, Series B*, 178:27–66, 2026. `doi:10.1016/j.jctb.2025.11.010`.

11 Prosenjit Bose, Vida Dujmović, Mehrnoosh Javarsineh, Pat Morin, and David R. Wood. Separating layered treewidth and row treewidth. *Discrete Mathematics & Theoretical Computer Science*, vol. 24, no. 1(Graph Theory), May 2022. `doi:10.46298/dmtcs.7458`.

12 Prosenjit Bose, Vida Dujmović, Mehrnoosh Javarsineh, and Pat Morin. Asymptotically optimal vertex ranking of planar graphs, 2020. `doi:10.48550/arXiv.2007.06455`.

13 Prosenjit Bose, Pat Morin, and Saeed Odak. An Optimal Algorithm for Product Structure in Planar Graphs. In Artur Czumaj and Qin Xin, editors, *18th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2022)*, volume 227 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19:1–19:14, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.SWAT.2022.19`.

14 Michał Dębski, Piotr Micek, Felix Schröder, and Stefan Felsner. Improved bounds for centered colorings. *Advances in Combinatorics*, August 2021. `doi:10.19086/aic.27351`.

15 Marc Distel, Robert Hickingbotham, Michał T. Seweryn, and David R. Wood. Powers of planar graphs, product structure, and blocking partitions. *Innovations in Graph Theory*, 1:39–86, 2024. `doi:10.5802/igt.4`.

16 Vida Dujmović, Louis Esperet, Cyril Gavoille, Gwenaël Joret, Piotr Micek, and Pat Morin. Adjacency labelling for planar graphs (and beyond). *J. ACM*, 68(6), October 2021. `doi:10.1145/3477542`.

17 Vida Dujmović, Louis Esperet, Gwenaël Joret, Bartosz Walczak, and David R. Wood. Planar graphs have bounded nonrepetitive chromatic number. *Advances in Combinatorics*, March 2020. `doi:10.19086/aic.12100`.

18 Vida Dujmović, Louis Esperet, Pat Morin, and David R. Wood. Proof of the clustered hadwiger conjecture. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1921–1930, 2023. `doi:10.1109/FOCS57990.2023.00116`.

19 Vida Dujmović, Gwenaël Joret, Piotr Micek, Pat Morin, Torsten Ueckerdt, and David R. Wood. Planar graphs have bounded queue-number. *J. ACM*, 67(4), August 2020. `doi:10.1145/3385731`.

20 Vida Dujmović, Pat Morin, and Saeed Odak. Odd colourings of graph products, 2022. `doi:10.48550/arXiv.2202.12882`.

21 Vida Dujmović, Louis Esperet, Pat Morin, Bartosz Walczak, and David R. Wood. Clustered 3-colouring graphs of bounded degree. *Combinatorics, Probability and Computing*, 31(1):123–135, 2022. `doi:10.1017/S0963548321000213`.

22 Vida Dujmović, Pat Morin, and David R. Wood. Graph product structure for non-minor-closed classes. *Journal of Combinatorial Theory, Series B*, 162:34–67, 2023. `doi:10.1016/j.jctb.2023.03.004`.

23 Zdeněk Dvořák, Daniel Gonçalves, Abhiruk Lahiri, Jane Tan, and Torsten Ueckerdt. On comparable box dimension. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry (SoCG 2022)*, volume 224 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 38:1–38:14, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.SoCG.2022.38`.

24 Louis Esperet, Gwenaël Joret, and Pat Morin. Sparse universal graphs for planarity. *Journal of the London Mathematical Society*, 108(4):1333–1357, 2023. `doi:10.1112/jlms.12781`.

25 Miriam Goetze, Fabian Klute, Kolja Knauer, Irene Parada, Juan Pablo Peña, and Torsten Ueckerdt. Strong odd coloring in minor-closed classes, 2025. `doi:10.48550/arXiv.2505.02736`.

26 Kevin Hendrey, Nikolai Karol, and David R. Wood. Structure of k-matching-planar graphs, 2025. `doi:10.48550/arXiv.2507.22395`.

27 Robert Hickingbotham, Paul Jungeblut, Laura Merker, and David R. Wood. The product structure of squaregraphs. *Journal of Graph Theory*, 105(2):179–191, 2024. `doi:10.1002/jgt.23008`.

28 Robert Hickingbotham and David R. Wood. Shallow minors, graph products, and beyond-planar graphs. *SIAM Journal on Discrete Mathematics*, 38(1):1057–1089, 2024. `doi:10.1137/22M1540296`.

29 Gwenaël Joret and Clément Rambaud. Neighborhood complexity of planar graphs. *Combinatorica*, 44(5):1115–1148, 2024. `doi:10.1007/s00493-024-00110-6`.

30 Daniel Kráľ, Kristýna Pekárková, and Kenny Štorgel. Twin-Width of Graphs on Surfaces. In Rastislav Královič and Antonín Kučera, editors, *49th International Symposium on Mathematical Foundations of Computer Science (MFCS 2024)*, volume 306 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 66:1–66:15, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.MFCS.2024.66`.

31 Jens Lagergren. Upper bounds on the size of obstructions and intertwines. *Journal of Combinatorial Theory, Series B*, 73(1):7–40, 1998. `doi:10.1006/jctb.1997.1788`.

32 Laura Merker, Lena Scherzer, and Samuel Schneider. Separating geodesic structure and product structure, 2026. `arXiv:2607.02098v1`.

33 Pat Morin. A fast algorithm for the product structure of planar graphs. *Algorithmica*, 83(5):1544–1558, 2021. `doi:10.1007/S00453-020-00793-5`.

34 Michał Pilipczuk and Sebastian Siebertz. Polynomial bounds for centered colorings on proper minor-closed graph classes. *Journal of Combinatorial Theory, Series B*, 151:111–147, 2021. `doi:10.1016/j.jctb.2021.06.002`.

35 A. Satyanarayana and L. Tung. A characterization of partial 3-trees. *Networks*, 20(3):299–322, 1990. `doi:10.1002/net.3230200304`.

36 Craig A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984. `doi:10.1016/0166-218X(84)90081-7`.

37 Torsten Ueckerdt, David R. Wood, and Wendy Yi. An improved planar graph product structure theorem. *The Electronic Journal of Combinatorics*, pages P2–51, 2022. `doi:10.37236/10614`.