Karlsruhe Institute of Technology

Comparing Variants of Product Structure

Master's Thesis of

Lena Scherzer

At the Department of Informatics
Institute of Theoretical Informatics (ITI)

Reviewer:	Dr. Torsten Ueckerdt
Second reviewer:	TT-Prof. Dr. Thomas Bläsius
Advisors:	Laura Merker
	Samuel Schneider

26.03.2025 – 26.09.2025

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

I declare that I have developed and written the enclosed thesis completely by myself. I have used ChatGPT with GPT 5 to familiarise myself with topics and conduct source research. I have not used any other than the aids that I have mentioned. I have marked all parts of the thesis that I have included from referenced literature, either in their original wording or paraphrasing their contents. I have followed the by-laws to implement scientific integrity at KIT.

Karlsruhe, 26.09.2025

. . . .

(Lena Scherzer)

Abstract

A graph class $\mathcal{G}$ has *geodesic structure* if there exists a constant k such that every $G \in \mathcal{G}$ has a partition $\mathcal{P}$ of $V(G)$ into geodesics such that $G/\mathcal{P}$ has treewidth at most k. A graph class $\mathcal{G}$ has *product structure* if there exists a constant k such that every $G \in \mathcal{G}$ has a layering $\mathcal{L}$ and a partition $\mathcal{P}$ of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k. Each set in a partition $\mathcal{P}$ of layered width 1 is allowed to contain at most one vertex in each layer in $\mathcal{L}$. The topic of this thesis is to investigate the relationships between these two variants and other related concepts. We show that product structure does not imply geodesic structure. For geodesic structure with $k = 1$, we show that it implies product structure. We also compare other variants of product structure and investigate the relationships between bounded layered treewidth, Baker treewidth and bounded local treewidth, which are all necessary conditions for product structure. Lastly, we show that some known results for product structure also hold for geodesic structure. For example, linear local treewidth is necessary for geodesic structure and calculating the geodesic treewidth is NP-hard. However, contrary to product structure, for graphs with treewidth 2, calculating the geodesic treewidth is possible in polynomial time.

Zusammenfassung

Eine Graphklasse $\mathcal{G}$ hat *Geodesic Structure*, wenn es ein k gibt, sodass jedes $G \in \mathcal{G}$ eine Partition $\mathcal{P}$ von $V(G)$ in Geodesics hat, sodass $G/\mathcal{P}$ Baumweite höchstens k hat. Eine Graphklasse $\mathcal{G}$ hat *Product Structure*, wenn es ein k gibt, sodass für jedes $G \in \mathcal{G}$ ein *Layering* $\mathcal{L}$ und eine Partition $\mathcal{P}$ mit layered width 1 existieren, sodass $G/\mathcal{P}$ Baumweite höchstens k hat. Jede Menge in der Partition $\mathcal{P}$ mit layered width 1 darf aus jeder Schicht in $\mathcal{L}$ höchstens einen Knoten enthalten. Das Thema dieser Arbeit ist die Untersuchung der Zusammenhänge zwischen diesen beiden Varianten von Product Structure und den Zusammenhängen mit anderen verwandten Konzepten. Wir zeigen, dass Product Structure nicht Geodesic Structure impliziert. Für Geodesic Structure mit $k = 1$ zeigen wir, dass es Product Structure impliziert. Wir vergleichen auch andere Varianten von Product Structure und untersuchen die Beziehungen zwischen Bounded Layered Treewidth, Baker Treewidth und Bounded Local Treewidth, welche alle notwendige Bedingungen für Product Structure sind. Auch zeigen wir, dass einige bekannte Ergebnisse für Product Structure auch für Geodesic Structure gelten. Beispielsweise ist Linear Local Treewidth notwendig für Geodesic Structure und die Berechnung der Geodesic Treewidth ist NP-schwer. Im Gegensatz zu Product Structure ist die Berechnung der Geodesic Treewidth, für Baumweite 2 Graphen, in polynomieller Zeit möglich.

Contents

1 Introduction

The treewidth of a graph vaguely describes how similar a graph is to a tree. Small treewidth is very useful for various applications, however, there are also many important graph classes with unbounded treewidth, for example, planar graphs. In this work, we consider and compare various generalizations of bounded treewidth that hold even for important graph classes with unbounded treewidth. We first introduce some of the more important generalizations and their relations to treewidth.

A graph class $\mathcal{G}$ has *product structure* if there exists a constant k such that every $G \in \mathcal{G}$ has a partition $\mathcal{P}$ into sets of *layered width* 1 such that $G/\mathcal{P}$ has treewidth at most k. This means a graph class has product structure if and only if, for each graph in the class, we can contract sets of layered width 1 such that the resulting graph has small treewidth. Given a layering $\mathcal{L}$ of G, sets of layered width 1 are sets that each contain at most one vertex from each layer. A simple example of a graph G with a layering and a partition $\mathcal{P}$ of layered width 1 is given in Figure 1.1. The corresponding graph $G/\mathcal{P}$ is also shown. It can be seen that, by definition, every graph class with bounded treewidth has product structure. Thus, product structure is a generalization of bounded treewidth. However, for example, planar graphs [1 , 2] and various beyond planar graph classes have product structure even though they have unbounded treewidth. Similar to bounded treewidth, product structure has also been used to prove and improve various bounds for parameters for the graph classes that admit product structure.

A different, but very similar, generalization of bounded treewidth is *geodesic structure*. It can be seen as the historic predecessor of product structure. The definition of geodesic structure has many similarities to the previous definition of product structure. As before, a graph class has geodesic structure if and only if, for each graph in the class, we can contract some sets such that the resulting graph has low treewidth. The only difference is in what type of sets we are allowed to contract. With product structure we contract sets of layered width 1. With geodesic structure we are allowed to contract only geodesics, where *geodesics* are shortest paths in a graph. Thus, a graph class $\mathcal{G}$ has geodesic structure if there exists a constant k such

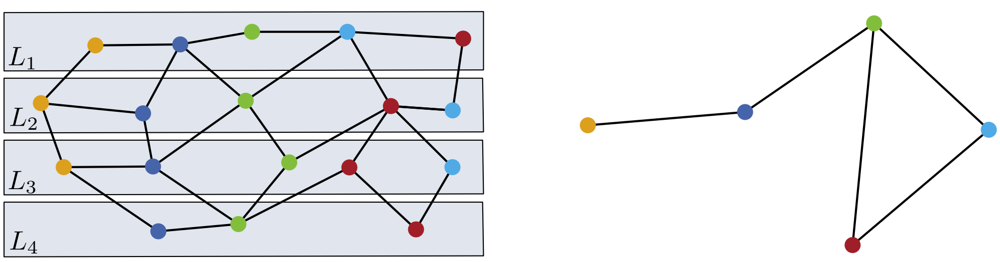


Figure 1.1: On the left a graph G with a layering $\mathcal{L} = \{L_1, L_2, L_3, L_4\}$ and a partition $\mathcal{P}$ into sets of layered width 1. On the right the corresponding graph $G/\mathcal{P}$ resulting from contracting each set in $\mathcal{P}$.

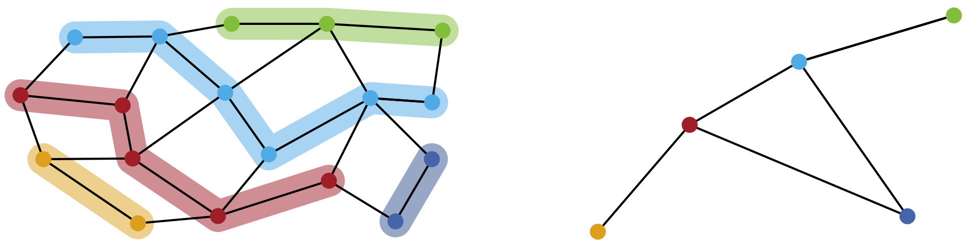


Figure 1.2: On the left a graph G with a partition $\mathcal{P}$ into geodesics. On the right the corresponding graph $G/\mathcal{P}$ resulting from contracting each geodesic in $\mathcal{P}$.

that every $G \in \mathcal{G}$ has a partition $\mathcal{P}$ into geodesics such that $G/\mathcal{P}$ has treewidth at most k. A simple example of a graph G with a partition $\mathcal{P}$ into geodesics is given in Figure 1.2 together with the corresponding graph $G/\mathcal{P}$.

Another similar variant of product structure that we define is BFS structure, which is product structure where the layerings of the graphs have to be *breadth first search* layerings[1] and the sets in the partition have to be connected. The sets that we contract in this case are vertical paths[2] in BFS layerings.

It can be seen that these generalizations have quite a few similarities in how they are defined. Additionally, historically they have been used in a similar context and are also known to exist for overlapping graph classes. Thus, we are interested in the exact relationships between these product structure variants. Other generalizations that we consider are various necessary conditions of product structure like bounded layered treewidth, linear local treewidth, bounded local treewidth and bounded Baker treewidth[3].

1.1 Outline

We begin by giving an overview of the related work on different variants of product structure and on the various necessary conditions for product structure in Section 1.2. We summarize our most important results in Section 1.3. In Chapter 2 we introduce basic notations and concepts and define the variants of product structure that we compare. In Chapter 3 we investigate the relationships between various variants of product structure with a focus on comparing geodesic structure and product structure. In Chapter 4, to better understand where product structure and geodesic structure behave the same, and where they differ, we investigate if some known results for product structure also apply for geodesic structure. In Chapter 5 we then investigate the relationships between various necessary conditions of product structure with a focus on comparing Baker treewidth and linear local treewidth. Lastly, in Chapter 6 we conclude this thesis by discussing our results. We also list open questions that could be the topic of future work.

[1] A *breadth-first search* layering of a graph results from a breath-first search, originating from a root vertex r, where each vertex is placed in layers based on their distance to r.

[2] A vertical path in a BFS layering is a path that contains at most one vertex from each layer.

[3] We give definitions for Baker treewidth and other necessary conditions for product structure in Section 2.3.

1.2 Related Work

In this section, we give an overview of the related work on product structure and various other variants of product structure.

Geodesic Structure. Of the product structure variants that we compare in this paper, geodesic structure was used at the earliest point in time and also directly inspired the other variants. Pilipczuk and Siebertz [3] first published the concept and show that planar graphs and bounded-genus graphs have geodesic structure. For planar graphs in particular, they show that every planar graph G has a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ has treewidth at most 8. Such a partition can be found in time $O(n^2)$. Note that they define and use geodesic structure as a concept, but do not give it a name. The name *geodesic structure* was chosen by us to more closely align with the name *product structure* that is widely used for the most popular graph structure variant. Pilipczuk and Siebertz [3] use the geodesic structure of planar graphs and bounded-genus graphs to show bounds for polynomial centered colourings for planar graphs, bounded-genus graphs and proper minor-closed graph classes. For planar graphs in particular Ueckerdt, Wood, and Yi [2] consider a variant of product structure that encompasses geodesic structure, and thus they improve the upper bound for the geodesic treewidth of planar graphs to 6. The lower bound of the geodesic treewidth of planar graphs has not been explicitly considered, however, the construction used for the lower bound of the row treewidth of planar graphs should, with some small changes to the arguments, also work here, and thus there is a planar graph with geodesic treewidth exactly 3 [2].

Product Structure. The concept of geodesic structure and its application directly inspired the idea of product structure. Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [1] first define the concept and show that planar graphs, bounded-genus graphs and proper minor-closed graph classes have product structure. For planar graphs in particular they show that every planar graph G has a partition $\mathcal{P}$ of layered width[4] 1 such that $G/\mathcal{P}$ has treewidth at most 8. Thus, planar graphs have row treewidth at most 8. This bound for the row treewidth of planar graphs was improved by Ueckerdt, Wood, and Yi [2], who show that every planar graph has row treewidth at most 6. They also consider the lower bound and provide a planar graph that has row treewidth exactly 3.

The idea of product structure became more widespread and many additional graph classes have been shown to admit product structure. The following graph classes have been shown to have product structure: k-nearest-neighbour graphs, (g, k)-planar graphs, d-map graphs, (g, d)-map graphs [4], h-framed graphs [5], (g, δ)-string graphs [4 , 6], k-th powers of planar graphs with bounded maximum degree [4 , 6 , 7], fan-planar graphs, k-fan-bundle graphs [7] and k-matching-planar graphs [8]. The results on product structure have been used to investigate different graph parameters and concepts in the following areas: adjacency labelling schemes [9 , 10 , 11], nonrepetitive colourings [12], p-centered colourings [13], clustered colourings [14 , 15], vertex rankings [16], queue layouts [1], reduced bandwidth [17], comparable box dimension [18], neighbourhood complexity [19], twin-width [5 , 20] and odd-colouring numbers [21 , 22].

It has also been shown that determining the row treewidth of graphs is NP-hard [23]. This is the case even for treewidth-2 graphs where the row treewidth is either 1 or 2.

[4] A detailed definition for partitions of layered width 1 and product structure in general is given in Section 2.2.

Stronger Variant of Product Structure. While product structure has been shown for many graph classes, in the first work on product structure Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [1] actually show that an even stronger property holds for planar graphs and bounded-genus graphs. For planar graphs in particular, they show that every planar graph G has a connected[5] partition $\mathcal{P}$ with layered width 1 such that $G/\mathcal{P}$ has treewidth at most 8. Moreover, there is such a partition for every BFS layering of G. It can be seen that the partition is a partition into sets of layer width 1 while at the same time being a partition into geodesics. Therefore, this stronger variant of product structure encompasses geodesics structure and product structure. The improvement by Ueckerdt, Wood, and Yi [2] that lowered the upper bound for the row treewidth of planar graphs to 6 preserves the BFS properties and thus also proves this stronger variant of product structure has an upper bound of 6.

Necessary Conditions for Product Structure. Product structure has also been investigated in relation to various broader generalizations of bounded treewidth. Bounded layered treewidth is one such generalization introduced by Dujmović, Morin, and Wood [24]. Among other results, they show that bounded genus graphs have bounded layered treewidth and that bounded layered treewidth implies the existence of layered separations of bounded width. Already in the first paper on product structure Dujmović, Joret, Micek, Morin, Ueckerdt, and Wood [1] show that bounded layered treewidth is a necessary condition for product structure. Dujmović, Morin, and Wood [24] show that bounded layered treewidth implies linear local treewidth and thus linear local treewidth is also a necessary condition for product structure. Bounded local treewidth is, of course, a further generalization of linear local treewidth. Linear local treewidth, as a necessary condition for product structure, has been used to show that graph classes do not admit product structure [1 , 7 , 25].

While it has been shown that these generalizations of bounded treewidth are necessary conditions for product structure, there is not much work on separating these concepts and proving that the broader generalizations are, in fact, weaker properties. The one defining work in this area is by Bose, Dujmović, Javarsineh, Morin, and Wood [26] who show that bounded layered treewidth does not imply product structure. However, for the case of proper minor closed graphs, Dujmović, Morin, and Wood [24] use the graph minor structure theorem of Robertson and Seymour [27] to prove that bounded local treewidth, linear local treewidth and bounded layered treewidth are all equivalent.

Baker treewidth relates to the previously named necessary conditions for product structure because it can be seen as a precursor to linear local treewidth that originates from Baker's technique [28]. It is based on the observation that deleting every k-th BFS layer from a planar graph leaves components of bounded treewidth. The technique was used to design approximation algorithms for maximum independent set, partition into triangles, minimum vertex cover, minimum dominating set, minimum edge dominating set and more [28]. The idea was also used to construct subgraph isomorphism, connectivity, and shortest path algorithms [29]. Baker Treewidth and Baker's technique were also applied to other graph classes like (g, k)-planar graphs [30], minor-closed classes [31] and bounded-degree H-induced-minor-free graphs [32]. While Hickingbotham [32] explicitly defines Baker treewidth, most other works do not formally define it as a graph parameter and instead use Baker's technique as an approach for designing approximation algorithms.

[5] A partition is connected if every set in the partition induces a connected subgraph.

1.3 Contributions

In this section, we introduce the most important results in this thesis. An overview of these results is given in Figure 1.3 where the arrows mean that one property of a graph class implies another property for the graph class. The figure also includes the known relationships between the properties with references in the caption.

The main topic of this thesis is comparing different variants of product structure. We begin by comparing product structure to its historic predecessor, geodesic structure, and get the following results:

Theorem 1.1: *[Section 3.1] Product structure does not imply geodesic structure.*

Theorem 1.2: *[Section 3.2] Every graph class with geodesic treewidth 1 has row treewidth at most 7.*

It is still an open question whether geodesic structure implies product structure in general or not. While comparing product structure and geodesic structure, we also get the following results that can be of independent interest:

Corollary 1.3: *[Section 3.1] Bounded layered treewidth does not imply geodesic structure.*

Observation 1.4: *[Section 3.1.1] The class of* 1*-planar graphs does not admit geodesic structure.*

Observation 1.4 in particular implies that many beyond planar graph classes that admit product structure, like k-planar graphs, fan-planar graphs and fan-bundle planar graphs, do not admit geodesic structure.

Next, we compare BFS Structure, as defined in Chapter 2, to the previous two variants of product structure. We show that BFS structure is stronger than both product structure and geodesic structure:

Theorem 1.5: *[Section 3.3] There are graph classes with geodesic structure but no BFS structure.*

Theorem 1.6: *[Section 3.3] There are graph classes with product structure but no BFS structure.*

Furthermore, to better understand what product structure and geodesic structure have in common and what differences there are, we investigate whether or not some known results for product structure also hold for geodesics structure. This gives us the following results on geodesic structure:

The best known lower bound for the row treewidth of planar graphs is 3 [2]. For geodesic structure, we show that the lower bound is at least 5:

Theorem 1.7: *[Section 4.2] There is a planar graph that has geodesic treewidth at least* 5.

This also implies that the lower bound for BFS structure is at least 5.

It is known that determining the row treewidth of graphs is NP-hard [23]. The same holds for the geodesic treewidth of graphs:

Theorem 1.8: *[Section 4.3.1] Determining if a given graph has geodesic treewidth at most k is NP-hard for $k \geq 2$.*

In contrast, while for row treewidth even determining the row treewidth of treewidth-2 graphs is NP-hard [23], it is in polynomial time for geodesic treewidth:

Theorem 1.9: *[Section 4.3.2] The geodesic treewidth of treewidth-2 graphs can be determined in time $O(n^5)$.*

Lastly, the question arises how we prove that a graph class does not admit product structure or geodesic structure. Bounded local treewidth and linear local treewidth, which are necessary conditions for product structure [1, 24], are commonly used tools, to show that a graph class does not admit product structure [1, 7, 25]. We show that linear local treewidth is also a necessary condition for geodesic structure.

Theorem 1.10: *[Section 4.1] If a graph class admits geodesic structure, then it has linear local treewidth.*

Bounded layered treewidth is an even stronger necessary condition for product structure. We also consider Baker treewidth, as defined in Section 2.3, which is an alternative necessary condition to linear local treewidth. Lastly, we investigate the relationships between the mentioned necessary conditions for product structure and get the following results:

Theorem 1.11: *[Chapter 5] If a graph class has bounded layered treewidth, then it has bounded Baker treewidth.*

Since bounded layered treewidth is a necessary condition for product structure this also implies that bounded Baker treewidth is a necessary condition for product structure. Next, we show that, just like linear local treewidth, bounded Baker treewidth implies bounded local treewidth:

Theorem 1.12: *[Chapter 5] For a graph class with bounded Baker treewidth with function $f(l)$, the local treewidth is bounded by $f(2k+1)$.*

We also compare bounded Baker treewidth to linear local treewidth and prove that:

Theorem 1.13: *[Chapter 5] There are graph classes with bounded Baker treewidth but without linear local treewidth.*

Lastly, we show that unbounded layered treewidth is still a stronger condition than bounded Baker treewidth.

Corollary 1.14: *[Chapter 5] There are graph classes with bounded Baker treewidth but unbounded layered treewidth.*

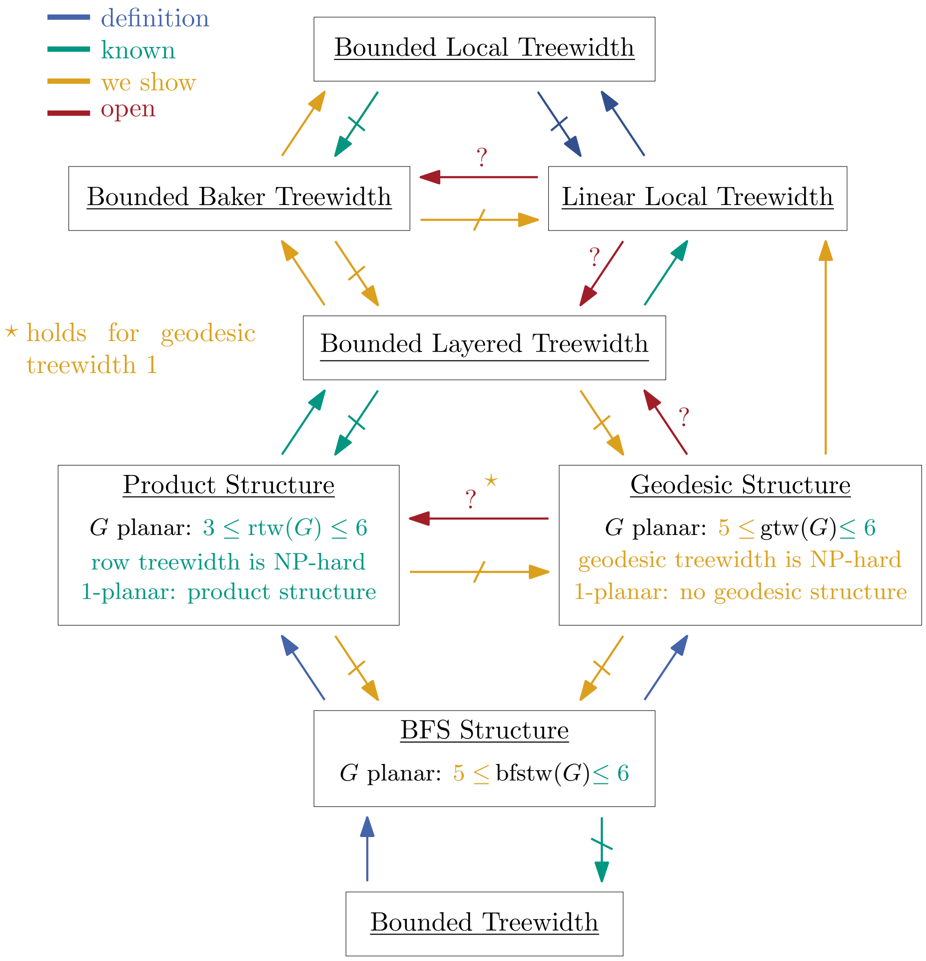


Figure 1.3: An overview of our results on the relationships between variants of products structure and their necessary conditions. Exact statements for all our results are given in Section 1.3. Here, we also give some short citations for the known results that have been shown by others:

- Bounded Local Treewidth $\not\Longrightarrow$ Bounded Baker Treewidth: $n \times n \times n$ grids with diagonals have unbounded Baker treewidth [33, 34]
- Bounded Layered Treewidth $\Longrightarrow$ Linear Local Treewidth: [24]
- Product Structure $\Longrightarrow$ Bounded Layered Treewidth: [1]
- Bounded Layered Treewidth $\not\Longrightarrow$ Product Structure: [26]
- BFS Structure $\not\Longrightarrow$ Bounded Treewidth: Planar graphs have BFS structure [1]

2 Preliminaries

In this chapter, we introduce and define all concepts that we use in this thesis. We begin by introducing the notation and giving definitions for basic concepts like quotients, treewidth and layerings. In the last two sections of this chapter, we define the variants of product structure that we compare and the various necessary conditions for product structure.

2.1 Basic Concepts

All graphs that we consider are undirected graphs with no loops and no multi-edges. The set of vertices of a graph G is called $V(G)$ and the set of edges is called $E(G)$. For an edge $\{u, v\} \in E(G)$ we refer to it as $uv \in E(G)$.

A graph G' is a *subgraph* of G if and only if $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$.

For a vertex v in a graph G the k-th neighbourhood $N^k[v]$ of v is the set of vertices that have distance at most k from v. This also includes v itself.

In a graph G we can *contract* an edge uv by removing the edge uv and merging the corresponding vertices u and v. All neighbours that were previously neighbours of either u or v are now neighbours of the newly merged vertex.

We say a graph M is a *minor* of G if it can be obtained from G by contracting edges, deleting vertices and deleting edges.

For a graph G and a partition $\mathcal{P}$ of $V(G)$ into sets, the *quotient* $G/\mathcal{P}$ is defined as follows. The graph $G/\mathcal{P}$ has the sets in $\mathcal{P}$ as vertices and two sets $P_1, P_2 \in \mathcal{P}$ are connected by an edge if and only if there exist vertices $v_1 \in P_1$ and $v_2 \in P_2$ such that $v_1v_2 \in E(G)$. Thus, the quotient $G/\mathcal{P}$ corresponds to the graph obtained from G by merging each set in $\mathcal{P}$ into a single vertex.

A *layering* $\mathcal{L} = \{L_1, L_2, \dots\}$ of G is a partition of $V(G)$ into *layers* $L_1, L_2, \dots$ such that for every edge $uv \in E(G)$ with $u \in L_i, v \in L_j$ it holds that $|i - j| \leq 1$.

A *tree decomposition* $(T, \mathcal{X} = \{X_1, \dots, X_t\})$ of a graph G is a tree T where the vertices of T are the sets $\{X_1, \dots, X_t\}$ and the sets in $\mathcal{X}$ themselves are subsets of $V(G)$. The sets $\{X_1, \dots, X_t\}$ are called the *bags* of the tree decomposition. Additionally, $(T, \mathcal{X} = \{X_1, \dots, X_t\})$ has to fulfil the following conditions to be a tree decomposition of G:

1. Each vertex of G is in at least one bag.
2. For each edge $uv \in E(G)$ there is at least one bag that contains both u and v.
3. For a vertex v of G the subgraph of T, that is induced by all bags containing v, is connected.

The *width* of a tree decomposition $(T, \mathcal{X})$ is the size of the largest bag in $\mathcal{X}$ minus one. The *treewidth* of a graph G is the minimum width of a tree decomposition of G.

2.2 Variants of Product Structure

In this section, we define the variants of product structure that we compare in this thesis. For some variants, we also need to define some auxiliary concepts.

Product Structure. A graph class $\mathcal{G}$ has *product structure* if there exists a constant k such that every $G \in \mathcal{G}$ has a partition $\mathcal{P}$ of $V(G)$ into sets of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k. Here, a partition $\mathcal{P}$ of G into sets of *layered width* 1 is a partition of $V(G)$ such that there is a layering $\mathcal{L}$ of G such that each set in the partition contains at most one vertex from each layer in $\mathcal{L}$.

We define the *row treewidth* $\operatorname{rtw}(G)$ of a graph G as the minimum k for which G has a partition $\mathcal{P}$ of $V(G)$ into sets of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k. Thus, for a graph class, product structure is equivalent to bounded row treewidth.

We will mostly be using the previous definition of product structure in this thesis, as it most closely aligns with the definitions of the other variants of product structure. There is, however, an equivalent definition of product structure that is more widely used. This equivalent definition uses the *strong product* of graphs to define product structure. The strong product $G_1 \boxtimes G_2$ of two graphs G_1, G_2 is a graph G with $V(G) = V(G_1) \times V(G_2)$ and $(u, v), (x, y) \in V(G)$ are adjacent if and only if either $u = x$ and $vy \in E(G_2)$ or $v = y$ and $ux \in E(G_1)$ or $ux \in E(G_1)$ and $vy \in E(G_2)$. A graph class $\mathcal{G}$ has *product structure* if and only if there exists a constant k such that for every $G \in \mathcal{G}$ there exists a path P and a graph H such that $G \subseteq P \boxtimes H$ and H has treewidth at most k.

BFS Structure. We define *BFS structure* as a variant of product structure. A graph class $\mathcal{G}$ has *BFS structure* if there exists a constant k such that every $G \in \mathcal{G}$ has a connected partition $\mathcal{P}$ of $V(G)$ into sets of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k and the partition $\mathcal{P}$ is a partition into sets of layered width 1 for some *breadth-first search* layering of G. Every set in a connected partition induces a connected subgraph of G. A breadth-first search layering of a graph results from a breadth-first search, originating from a root vertex r, where vertices are placed in layers based on their distance to r. There is also a variant of BFS structure that is used in literature where the existence of such a partition $\mathcal{P}$ is required for every BFS layering of G. Here we just require that there exists such a partition $\mathcal{P}$ for *some* BFS layering of G. This makes separating this concept from the other related variants of product structure more interesting. In both variants, the sets in the partitions $\mathcal{P}$ are vertical paths in a BFS layering. A *vertical path* in a BFS layering is a path that contains at most one vertex from each layer.

We define the *BFS treewidth* $\operatorname{bfstw}(G)$ of a graph G as the minimum k for which G has a connected partition $\mathcal{P}$ of $V(G)$ into sets of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k and the partition $\mathcal{P}$ is a partition into sets of layered width 1 for some breadth-first search layering of G. Thus, for a graph class, BFS structure is equivalent to bounded BFS treewidth.

Geodesic Structure. A graph class $\mathcal{G}$ has geodesic structure if there exists a constant k such that every $G \in \mathcal{G}$ has a partition $\mathcal{P}$ of $V(G)$ into geodesics such that $G/\mathcal{P}$ has treewidth at most k. A *geodesic* is a shortest path in G.

We define the *geodesic treewidth* $\operatorname{gtw}(G)$ of a graph G as the minimum k for which G has a partition $\mathcal{P}$ of $V(G)$ into geodesics such that $G/\mathcal{P}$ has treewidth at most k. Thus, for a graph class, geodesic structure is equivalent to bounded geodesic treewidth.

2.3 Necessary Conditions for Product Structure

In this section, we define bounded layered treewidth, linear local treewidth and bounded local treewidth. All three concepts have been shown to be necessary conditions for product structure [1, 24]. We also define Baker treewidth for which we show in Chapter 5 that it is also a necessary condition for product structure.

Bounded Layered Treewidth. A graph class $\mathcal{G}$ has *bounded layered treewidth* if there exists a constant k such that for each $G \in \mathcal{G}$ there exists a tree decomposition D and a layering $\mathcal{L}$ of G such that the intersection between any bag B in D and any layer $L \in \mathcal{L}$ is at most k.

Local Treewidth. A graph class $\mathcal{G}$ has *bounded local treewidth* if there exists a function f such that for every graph $G \in \mathcal{G}$ and every vertex $v \in V(G)$ the treewidth of the graph induced by the k-th neighbourhood $N^k[v]$ of v is at most $f(k)$. If the function f is a linear function, then we say that $\mathcal{G}$ has *linear local treewidth.*

Baker Treewidth. A graph class $\mathcal{G}$ has *bounded Baker treewidth* if there exists a function f such that for every graph $G \in \mathcal{G}$ there exists a layering $\mathcal{L}$ of G such that each graph induced by the union of k consecutive layers in $\mathcal{L}$ has treewidth at most $f(k)$.

3 Product Structure vs. Geodesic Structure

In this chapter, we compare the variants of product structure. The main focus lies on investigating the relationship between product structure and geodesic structure.

3.1 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. For each $G \in \mathcal{G}$, let G' be the result of some number of subdivisions of each edge of G. The number of subdivisions of each edge does not have to be the same. We define $\mathcal{G}' = \{G' \mid G \in \mathcal{G}\}$ and thus $\mathcal{G}'$ contains, for each graph from $\mathcal{G}$, some subdivision of that graph.

In general, subdividing the edges of a graph can reduce the row and geodesic treewidth of the graph. We first illustrate this using the class of complete graphs:

The class of complete graphs $\mathcal{G}$ does not have product structure. For $K_n \in \mathcal{G}$ let P be a hamiltonian path of K_n. We construct K_n' by subdividing each edge $uv \notin P$ of K_n as often as the distance of u and v on P plus one. It can be seen that $\mathcal{G}'$ has product structure since every $K_n' \in \mathcal{G}'$ is a subgraph of $P \boxtimes S$ where S is a star. An example of this is given in Figure 3.1. Since every graph can be embedded in a complete graph, the following observation holds.

Observation 3.1: *For every graph class $\mathcal{G}$, there exists a graph class $\mathcal{G}'$ resulting from subdivision as defined above that has product structure.*

The class of complete graphs $\mathcal{G}$ also does not have geodesic structure. For $K_n \in \mathcal{G}$ let P be a hamiltonian path of K_n. We construct K_n' by subdividing each edge $uv \notin P$ of K_n as often as the distance of u and v on P minus one. Thus, P is a geodesic in K_n'. Let $\mathcal{P}$ be a partition into geodesics consisting of P and one additional geodesic for each edge of K_n that is not on P. It follows that $G_n/\mathcal{P}$ is a star and thus has treewidth 1.

Observation 3.2: *For the class of complete graphs $\mathcal{G}$, there exists a graph class $\mathcal{G}'$ resulting from subdivision as defined above that has geodesic treewidth 1.*

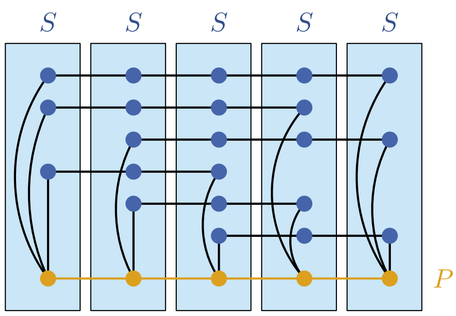
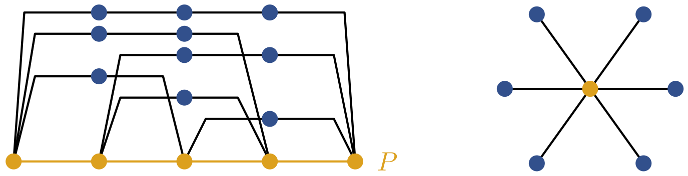


Figure 3.1: Subdivided K_5 that is a subgraph of $P \boxtimes S$, where S is a star.



(a) K_5 **(b)** $K_5/\mathcal{P}$

Figure 3.2: Subdivided K_5 that has geodesic treewidth 1.

However, row treewidth and geodesic treewidth still behave very differently under subdivision. In particular, if each edge is subdivided at least n^2 times, where n is the number of vertices in the graph, then the resulting graph is 1-planar. Since 1-planar graphs have row treewidth at most 28 [4, Theorem 3] the following observation follows:

Observation 3.3: *Let G be a graph and G' be the result after subdividing each edge of G at least $|V(G)|^2$ times. Then G' has row treewidth at most 28.*

For geodesic treewidth, it is, however, not the case that enough subdivisions of each edge lead to small geodesic treewidth. We prove this in the following lemma:

Lemma 3.4: *Let $\mathcal{G}$ be a graph class that does not admit geodesic structure. Let $\mathcal{G}'$ be the graph class resulting from subdividing each edge k times for each graph in $\mathcal{G}$. Then $\mathcal{G}'$ also does not admit geodesic structure.*

Proof. Assume $\mathcal{G}'$ does admit geodesic structure with geodesic treewidth c. For $G \in \mathcal{G}$, let G' be the graph resulting from subdividing each edge k times. Thus G' is in $\mathcal{G}'$ and therefore 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 k 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. ∎

There are even graph classes that do not have geodesic structure, no matter how often each edge is subdivided, unlike in the previous proof, where each edge is subdivided an equal number of times.

This difference in how row treewidth and geodesic treewidth behave under subdivision is used in the following to define two graph classes that admit product structure but no geodesic structure. Thus proving the following:

Theorem 1.1: *Product structure does not imply geodesic structure.*

The first graph class with product structure but no geodesic structure is defined as follows. Let K_n' be the graph resulting from subdividing each edge n^2 times in the complete graph K_n.

Lemma 3.5: *The graph class of all K_n' for $n \in \mathbb{N}$ admits product structure but not geodesic structure.*

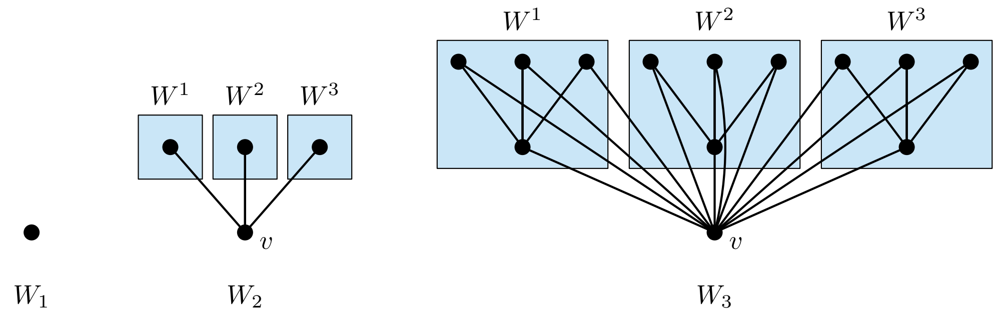


Figure 3.3: The graphs W_1, W_2, W_3.

Proof. By Observation 3.3 it holds that this graph class admits product structure. Since complete graphs do not admit geodesic structure by Lemma 3.4, it follows that the graph class of all K'_n also does not admit geodesic structure. ∎

Thus, product structure does not imply geodesic structure.

In the following, we give a different graph class that does not admit geodesic structure, regardless of the number of subdivisions of each edge. Using Observation 3.1 this thus also results in a graph class that admits product structure but not geodesic structure. The proof and graph class is inspired by the proof that subdivisions do not efficiently reduce row treewidth by Bose et al. [26].

For three graphs G_1, G_2, G_3, we construct graphs $G^{\star}_{1,2,3}$ as follows: connect a disjoint copy of G_1, G_2 and G_3 by a universal vertex v and subdivide the edges adjacent to v an arbitrary number of times. Each edge adjacent to v can be subdivided a different number of times. We prove:

Lemma 3.6: *If G_1, G_2 and G_3 each have geodesic treewidth at leasts k then $G^{\star}_{1,2,3}$ has geodesic treewidth at least $k+1$.*

Proof. We prove that $\text{gtw}(G^{\star}_{1,2,3}) \geq k+1$. Let $\mathcal{P}^{\star}$ be any partition of $G^{\star}_{1,2,3}$ into geodesics. Let v be the universal vertex in $G^{\star}_{1,2,3}$ and $P_v \in \mathcal{P}^{\star}$ be the geodesic containing v. Since P_v is a path either G_1, G_2 or G_3 does not intersect P_v. Assume without loss of generality that G_1 does not intersect P_v and let $\mathcal{P}^{\star}$ constrained to G_1 be called $\mathcal{P}_1$. It holds that $G_1/\mathcal{P}_1$ has at least treewidth k since $\text{gtw}(G_1) \geq k$. Let G'_1 be the graph $G_1/\mathcal{P}_1$ with an additional universal vertex. It is known that adding a universal vertex to a graph increases the treewidth by at least one, and thus G'_1 has treewidth at least $k+1$. The graph $G^{\star}_{1,2,3}/\mathcal{P}^{\star}$ has a G'_1-minor consisting of $G_1/\mathcal{P}_1$ and the vertex corresponding to P_v if the additional vertices from subdivisions are contracted into the vertex corresponding to P_v. Thus, $G^{\star}_{1,2,3}/\mathcal{P}^{\star}$ has treewidth at least $k+1$ and thus $\text{gtw}(G^{\star}_{1,2,3})) \geq k+1$. ∎

We iteratively construct the following graph class $\mathcal{W} = \bigcup W_i$. We define W_1 as a single vertex. For $i \geq 2$, the graph W_i is defined as the graph consisting of three disjoint copies W^1, W^2, W^3 of W_{i-1} and a vertex v that is adjacent to all other vertices. An example of this is shown in Figure 3.3.

Let $\mathcal{W}'$ be a graph class resulting from $\mathcal{W}$ by subdivision as defined at the beginning of this section. We prove that every $\mathcal{W}'$ does not admit geodesic structure.

Lemma 3.7: *Every $\mathcal{W}'$ resulting from $\mathcal{W}$ by subdivision does not admit geodesic structure.*

Proof. We use induction to prove that $W'_i \in \mathcal{W}'$ has geodesic treewidth at least $i-1$. For the base cases, W'_1 is a single vertex and thus trivially has geodesic treewidth 0. For W'_i and $i > 1$, it can be seen that W'_i consists of three graphs that have treewidth at least $i-1$ connected with a universal vertex v where the edges adjacent to v are possibly subdivided. Thus, using Lemma 3.6 it follows that W'_i has geodesic treewidth at least $i-1$. Therefore, the graph class $\mathcal{W}'$ does not admit geodesic structure. ∎

We have shown that product structure does not imply geodesic structure. Since product structure implies bounded layered treewidth [12] we get the following corollary.

Corollary 1.3: *Bounded layered treewidth does not imply geodesic structure.*

3.1.1 Beyond-planar graphs do not have geodesic structure

In Section 3.1 we define the graph class of all K'_n that consists of complete graphs where each edge is subdivided n^2 times. This graph class is 1-planar, however, it does not admit geodesic structure. Therefore, we make the following observation:

Observation 3.8: *The class of* 1*-planar graphs does not admit geodesic structure.*

However 1-planar graphs are part of many beyond-planar graph classes. Thus, for all these beyond-planar graph classes, it follows that they do not admit geodesic structure. Some examples of such beyond-planar graph classes that do not admit geodesic structure but have been shown to admit product structure are: (g, k)-planar graphs, fan-planar graphs, fan-bundle planar graphs and k-gap planar graphs [7].

3.2 Does geodesic structure imply product structure?

In this section, we show that graph classes with geodesic treewidth 1 have product structure. However, whether geodesic structure implies product structure in general remains an open question.

A graph class $\mathcal{G}$ admits *product structure* if there exists a constant k such that for every $G \in \mathcal{G}$ there exists a path P and a graph H with treewidth k such that G is a subgraph of $H \boxtimes P$. An equivalent definition for product structure is that there exists a constant k such that for every $G \in \mathcal{G}$ there is a partition $\mathcal{P}$ of layered width 1 such that $G/\mathcal{P}$ has treewidth at most k [1].

We observe that for all paths P and all constants c it holds that $P_c = P \cup P^2 \cup \dots \cup P^c \subseteq P \boxtimes K_c$. If we consider a graph class $\mathcal{G}$ for which there exist constants k and c such that for every $G \in \mathcal{G}$ there exists a path P and a graph H with treewidth k such that G is a subgraph of $H \boxtimes P_c$ it thus follows that $\mathcal{G}$ admits product structure with row treewidth $(k+1) \cdot c$ - 1. Using the equivalent definition $\mathcal{G}$ admits product structure if there exist constants k and c such that for every $G \in \mathcal{G}$ there is a partition $\mathcal{P}$ of layered width 1 with regard to some c*-layering* $\mathcal{L}$ such that $G/\mathcal{P}$ has treewidth at most k. A c-layering $\mathcal{L}$ is an ordered partition $(L_0, L_1, \dots)$ of $V(G)$ such that for every edge $vw \in E(G)$ with $v \in L_i$ and $w \in L_j$ we have $|i-j| \leq c$.

Let $\mathcal{G}$ be a graph class that admits geodesic structure with geodesic treewidth k, i.e. for each graph $G \in \mathcal{G}$ there exists a partition $\mathcal{P}$ into geodesics such that $G/\mathcal{P}$ has treewidth at most k. In this section we investigate whether there exists a constant c for $\mathcal{G}$ such that every $G \in \mathcal{G}$ has a c-layering $\mathcal{L}$ such that $\mathcal{P}$ is a partition of layered width 1 with regard to $\mathcal{L}$. This would imply that $\mathcal{G}$ also admits product structure.

Lemma 3.9: *Let G be a graph with a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree. Then there exists a 4-layering $\mathcal{L}$ of $V(G)$ such that $\mathcal{P}$ is a partition of layered width 1 with regard to $\mathcal{L}$.*

Proof. Let G be a graph with a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree. If $|\mathcal{P}| = 1$ then trivially there exists a 4-layering $\mathcal{L}$ of G such that $\mathcal{P}$ is a partition of layered width 1 and for each $P \in \mathcal{P}$ the vertices of P are embedded in consecutive layers of $\mathcal{L}$ according to their order in P. Assume that the same holds for $|\mathcal{P}| < n$. We show that for $|\mathcal{P}| = n$ there exists a 4-layering $\mathcal{L}$ of G such that $\mathcal{P}$ is a partition of layered width 1 and for each $P \in \mathcal{P}$ the vertices of P are embedded in consecutive layers of $\mathcal{L}$ according to their order in P.

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 $P_\Gamma = (v_0, v_1, \dots, v_m)$ and $P_\Delta = (w_1, w_2, \dots, w_n)$. Let $\mathcal{L} = (L_0, L_1, \dots)$ be a 4-layering of $G - P_\Gamma$ such that $\mathcal{P} \setminus P_\Gamma$ is a partition of layered width 1 and for each $P \in \mathcal{P} \setminus P_\Gamma$ the vertices of P are embedded in consecutive layers of $\mathcal{L}$ according to their order in P. Note that in the following construction some layers may be assigned negative indices.

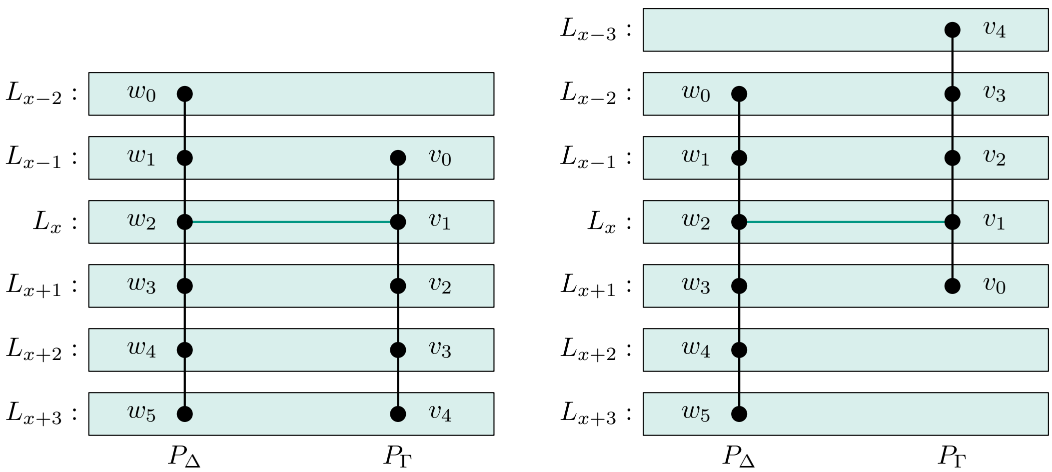


Figure 3.4: Layering $\mathcal{L}_1$ on the left and layering $\mathcal{L}_2$ on the right. The green edge is the alignment edge $e = v_i w_j$.

Since Γ is a leaf adjacent to Δ, there exists an $i \in [m]$ and $j \in [n]$ such that $e = v_i w_j \in E(G)$. We use this edge e to align the path P_Γ. Let L_x be the layer in $\mathcal{L}$ that contains w_j. We construct two potential layerings $\mathcal{L}_1, \mathcal{L}_2$ of G from $\mathcal{L}$. To construct $\mathcal{L}_1$ we take $\mathcal{L}$ and for every $l \in [m]$ we place the vertex v_l in the layer $L_{x+(l-i)}$. To construct $\mathcal{L}_2$ we take $\mathcal{L}$ and for every $l \in [m]$ we place the vertex v_l in the layer $L_{x-(l-i)}$. Thus, $\mathcal{L}_2$ only differs from $\mathcal{L}_1$ in that the path P_Γ is mirrored in the layering. An example of the two possible resulting layerings can be seen in Figure 3.4. For both layerings, it can be seen that $\mathcal{P}$ is a partition of layered width 1 and that for each $P \in \mathcal{P}$ the vertices of P are embedded in consecutive layers according to their order in P.

It remains to show that $\mathcal{L}_1$ or $\mathcal{L}_2$ is a 4-layering of G. For any layering $\mathcal{L}'$ let a k-*steep* edge be an edge with endpoints in layers L_a, L_b of $\mathcal{L}'$ and $|a - b| \geq k$. For the layerings $\mathcal{L}_1$ and $\mathcal{L}_2$, let a *crossing* edge be an edge with endpoints in layers L_a, L_b of $\mathcal{L}_1$ or $\mathcal{L}_2$ respectively and $a < x < b$, i.e. edges that cross the alignment edge e in the embedding shown in Figure 3.4. Note that as the vertices of P_Γ are embedded in consecutive layers according to their order in P_Γ, edges in P_Γ can not be 2-steep edges. Furthermore, non-crossing edges e' can not be

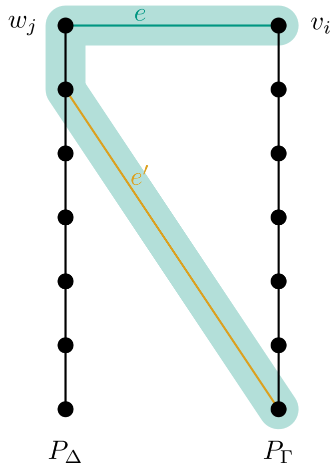


Figure 3.5: If a non-crossing edge e' is 3-steep, then together with the alignment edge e, this results in a shortcut for one of the two geodesics – in this example a shortcut on P_Γ that is highlighted green.

3-steep edges since else e and e' result in a shortcut for one of the two geodesics P_Γ or P_Δ as shown in Figure 3.5. Thus, if $\mathcal{L}_1$ contains no 5-steep crossing edge, then no 5-steep edge exists in the layering $\mathcal{L}_1$ and it is thus a valid 4-layering.

Assume now that there is a 5-steep crossing edge $e'' = v_q w_r$ in $\mathcal{L}_1$. We prove that in this case it follows that $\mathcal{L}_2$ has no 5-steep edges and is thus a valid 4-layering. Similar to $\mathcal{L}_1$, edges in P_Γ can not be 2-steep edges and non-crossing edges can not be 3-steep edges in $\mathcal{L}_2$. Thus, it only remains to show that $\mathcal{L}_2$ contains no 5-steep crossing edge.

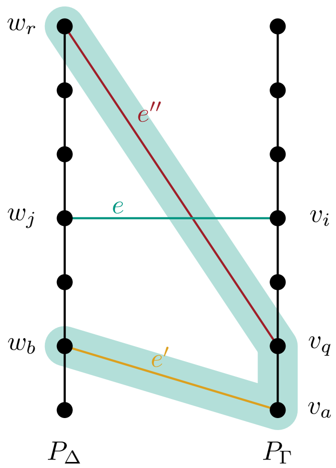


Figure 3.6: If $a \geq q$ then since e' is not steep the edges e'' and e' result in a shortcut for P_Γ that is highlighted in green.

Every crossing edge in $\mathcal{L}_1$ is non-crossing in $\mathcal{L}_2$. Thus, we consider non-crossing edges in $\mathcal{L}_1$ which correspond to crossing edges in $\mathcal{L}_2$. Let $e' = v_a w_b$ be such a non-crossing edge in $\mathcal{L}_1$. We already know that e' is not 3-steep in $\mathcal{L}_1$. Without loss of generality, let $a \geq i$ and $b \geq j$ and $q > i$ and $r < j$, meaning that e' is below the alignment edge e in the embedding of $\mathcal{L}_1$ shown in Figure 3.4. If $a \geq q$ then since e' is not 3-steep the edges e'' and e' result in a shortcut as shown in Figure 3.6. Thus, we assume now that $a < q$. Let $E' = \{e' = v_a w_b \mid |i-a|+|j-b| < 5\}$. Then, the edges in E' are not 5-steep in $\mathcal{L}_2$. The edges in E' correspond to the yellow edges in Figure 3.7.

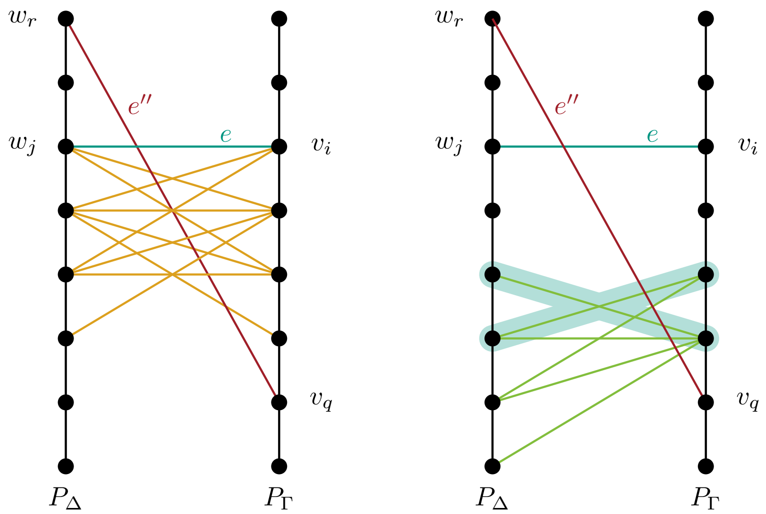


Figure 3.7: In yellow on the left are the edges $e' = w_b v_a \in E'$ which are not 5-steep edges in $\mathcal{L}_2$. In green on the right are the edges $e' = w_b v_a \in E'_2$ with $|a - i| \leq 3$ that together with e'' result in a shortcut for P_Δ. Highlighted are in green are the edges $v_{i+2}w_{j+3}$ and $v_{i+3}w_{j+2}$ which dominate the remaining edges. Combined the yellow and green edges are all non-3-steep non-crossing edges with $|a - i| \leq 3$.

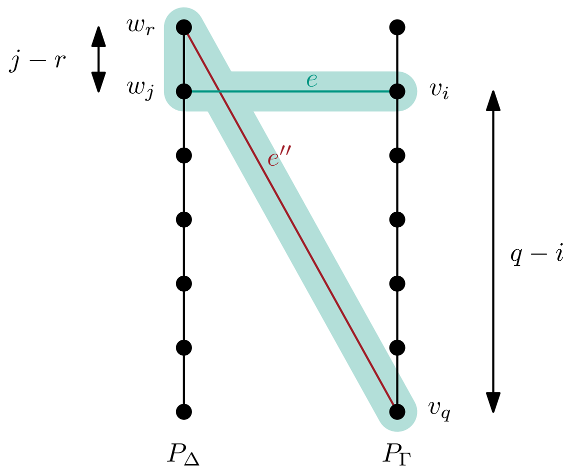


Figure 3.8: If $q - i > j - r + 2$ then e'' and e are a shortcut for P_Γ as shown in blue.

Lastly, we need to consider edges $e' = v_a w_b$ below the alignment edge e with $a < q$ and $e' \notin E'$ that could still be 5-steep crossing edges in $\mathcal{L}_2$. It holds that $q - i \leq j - r + 2$ else e'' and e are a shortcut for P_Γ as shown in Figure 3.8. Let S be the path from w_r to w_b using edges e'' and e' and the path P_Γ. An example for such a path S with all relevant distances is given in Figure 3.9. Note that S has length $2 + q - a$. Thus, for $a - i > 3$ it follows that $2 + q - a < q - i \leq j - r + 2 \leq b - r$. Thus, S is a shortcut for P_Δ from w_r to w_b.

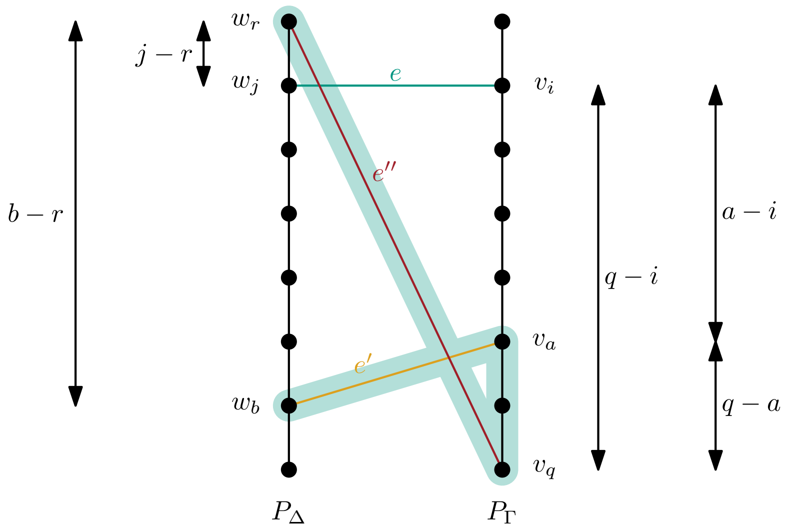


Figure 3.9: The shortcut S from w_r to w_b using edges e'' and e' is highlighted in green.

Finally, we consider the edges $E_2' = \{e' = v_a w_b \mid a - i \leq 3 \wedge e' \notin E'\}$. These edges can be seen in Figure 3.7 on the right. For $v_{i+2}w_{j+3}$ the path S has length $q-(i+2)+2 \leq j-r+2 < j-r+3$ and is thus a shortcut from w_r to w_{j+3}. For $v_{i+3}w_{j+2}$ the path S has length $q-(i+3)+2 \leq j-r+1 < j-r+2$ and is thus a shortcut from w_r to w_{j+2}. The remaining edges in E_2' are dominated by $v_{i+2}w_{j+3}$ and $v_{i+3}w_{j+2}$, meaning that if these two edges result in a shortcut when combined with e'' then the remaining edges in E_2' also result in a shortcut. Thus, the edges in E_2' would result in a shortcut for P_Δ and are thus not in the graph. Therefore, $\mathcal{L}_2$ has no 5-steep crossing edges and is a 4-layering. ∎

Theorem 1.2: *Every graph class with geodesic treewidth 1 has row treewidth at most 7.*

Proof. For each $G \in \mathcal{G}$ Lemma 3.9 gives us a 4-layering $\mathcal{L}$ of G such that $\mathcal{P}$ is a partition of layered width 1. Thus $\mathcal{G}$ admits product structure with row treewidth $(1+1)\cdot 4 - 1 = 7$. ∎

This construction works for graph classes where the quotient $G/\mathcal{P}$ is a tree. However, if the quotient $G/\mathcal{P}$ contains a cycle, a similar approach does not work in general. Consider the graph class consisting of all C_{3k} for $k \in \mathbb{N}$. Assume a constant c exists for this graph class such that for every C_{3k} and every partition $\mathcal{P}$ into geodesics there exists a c-layering where $\mathcal{P}$ is a partition of layered width 1 and for each $P \in \mathcal{P}$ the vertices of P are embedded in consecutive layers of $\mathcal{L}$ according to their order in P.

For C_{3k} we consider the partition $\mathcal{P} = \{P_1, P_2, P_3\}$ of the vertices into geodesics where each P_i is a path of length k as shown in Figure 3.10. Let $\mathcal{L}$ be a c-layering for C_{3k} where $\mathcal{P}$ is a partition of layered width 1 and for each P_i the vertices of P_i are embedded in consecutive layers of $\mathcal{L}$ according to their order in P_i. Thus, the orange edges e and f in Figure 3.10 are at most c-steep in $\mathcal{L}$. However, this implies that g is at least $k-2c$-steep and thus for large enough k the given $\mathcal{L}$ is not a c-layering. This contradicts our assumption and therefore for the graph class consisting of all C_{3k} no such constant c exists, for which we can find a c-layering as described in the proof of Lemma 3.9.

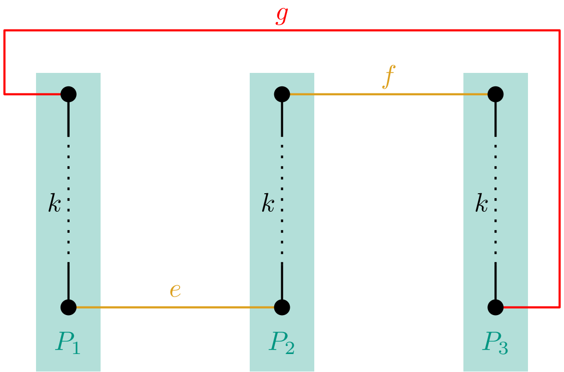


Figure 3.10: Graphs C_{3k} where each P_i is a path consisting of k vertices.

3.3 BFS Structure - Sufficient for Geodesic Structure and Product Structure

In this section, we compare BFS structure, as defined in Section 2.2, to the other variants of product structure. First, we observe that by definition BFS structure implies geodesic structure and product structure. Then we show that the reverse is not true in either case.

If a graph G has BFS treewidth k, then there exists a BFS layering of G and a connected partition $\mathcal{P}$ of $V(G)$ into sets of layered width 1 with regard to $\mathcal{L}$ such that $G/\mathcal{P}$ has treewidth at most k. Since this partition is a partition of layered width 1, it holds that G also has row treewidth at most k. Therefore, BFS structure for a graph class implies product structure. Since $\mathcal{P}$ is a connected partition, this means that every part in the partition is a path. More specifically, it is a vertical path in a BFS layering, which is also a shortest path. Therefore, $\mathcal{P}$ is a partition into geodesics and G has geodesic treewidth k. Thus, BFS structure for a graph class also implies geodesic structure.

In the following, we show that geodesic structure does not imply BFS structure by proving that the graph class we define has geodesic structure but no BFS structure.

We consider the graphs $\mathcal{G} = \{G_n \mid n \in \mathbb{N} \text{ even}\}$ where G_n is obtained from two $n \times n$-grids connected by a universal vertex v. In addition, all edges adjacent to v are subdivided $n/2$ times. To simplify notation, we only consider the case that n is an even number. An example of such a graph is shown in Figure 3.11.

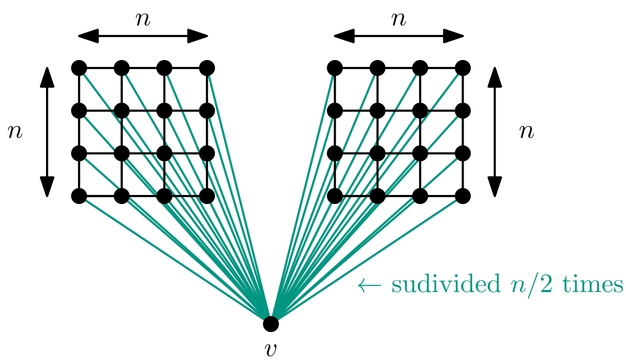


Figure 3.11: Graphs with geodesic structure and product structure but no BFS structure.

We first show that $\mathcal{G}$ has no BFS structure.

Lemma 3.10: *The graph class $\mathcal{G}$ does not admit BFS structure.*

Proof. Let $G_n \in \mathcal{G}$ and G', G'' be the two $n \times n$-grids in G_n. We consider any BFS layering of G_n. Without loss of generality, assume that the start of the breadth-first search is inside of G' or on any path from G' to v. At some point, the breadth-first search reaches v and from then on spreads evenly towards G''. Thus, G'' is contained in a single layer in the BFS layering. Let $\mathcal{P}$ be any partition of the vertices of G_n into vertical paths of the BFS layering. It holds that each path in $\mathcal{P}$ contains at most one vertex in G''. Thus $G/\mathcal{P}$ contains a $n \times n$-grid and has treewidth at least n. Therefore, $\mathcal{G}$ has no BFS structure. ∎

For a $n \times n$-grid consisting of n columns and n rows, let the partition of the vertices where the columns of the grid are chosen as parts be called the canonical partition. An example of a canonical partition of a grid is shown in Figure 3.12.

Next, for any $G_n \in \mathcal{G}$ let G', G'' be the two $n \times n$-grids in G_n. We define the following partition $\mathcal{P}_n$ of $V(G_n)$. Let $\mathcal{P}_n$ contain the canonical partitions of G' and G'' and every other vertex of G_n is in a part of its own. We show:

Lemma 3.11: *For $G_n \in \mathcal{G}$ the graph $G_n/\mathcal{P}_n$ has treewidth 2.*

Proof. The graphs G_n and $G_n/\mathcal{P}_n$ are shown in Figure 3.13. Note that deleting the vertex that corresponds to the part $\{v\} \in \mathcal{P}_n$ results in a forest and thus $G_n/\mathcal{P}_n$ has treewidth 2. ∎

We use the previous results to show that $\mathcal{G}$ has geodesic structure but no BFS structure.

Theorem 1.5: *There are graph classes with geodesic structure but no BFS structure.*

Proof. We show that the graph class $\mathcal{G}$ has geodesic structure but no BFS structure.

In Lemma 3.10 we show that $\mathcal{G}$ has no BFS structure.

For $G_n \in \mathcal{G}$ let G', G'' be the two $n \times n$-grids in G_n. We consider the partition $\mathcal{P}_n$ of $V(G_n)$ that contains the canonical partitions of G' and G'' and where every other vertex of G_n is in a part of its own. We show that this is a partition of $V(G_n)$ into geodesics. Firstly, the parts in the canonical partitions are columns in the grids and are thus paths. Secondly, since the paths between v and the grids have length $n/2$ and the paths in the canonical partitions of the grids G' and G'' have length n it follows that the paths in the canonical partitions of the grids G' and G'' are still geodesics in G_n. All remaining parts in $\mathcal{P}_n$ are single vertices and thus also geodesics. Therefore, $\mathcal{P}_n$ is a partition of $V(G_n)$ into geodesics. In Lemma 3.11 we show that $G_n/\mathcal{P}_n$ has treewidth 2. Therefore, every $G_n \in \mathcal{G}$ has geodesic treewidth 2. ∎

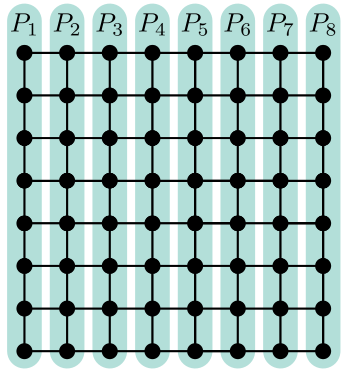


Figure 3.12: A 8×8-grid and its canonical partition $\{P_1, \ldots, P_8\}$.

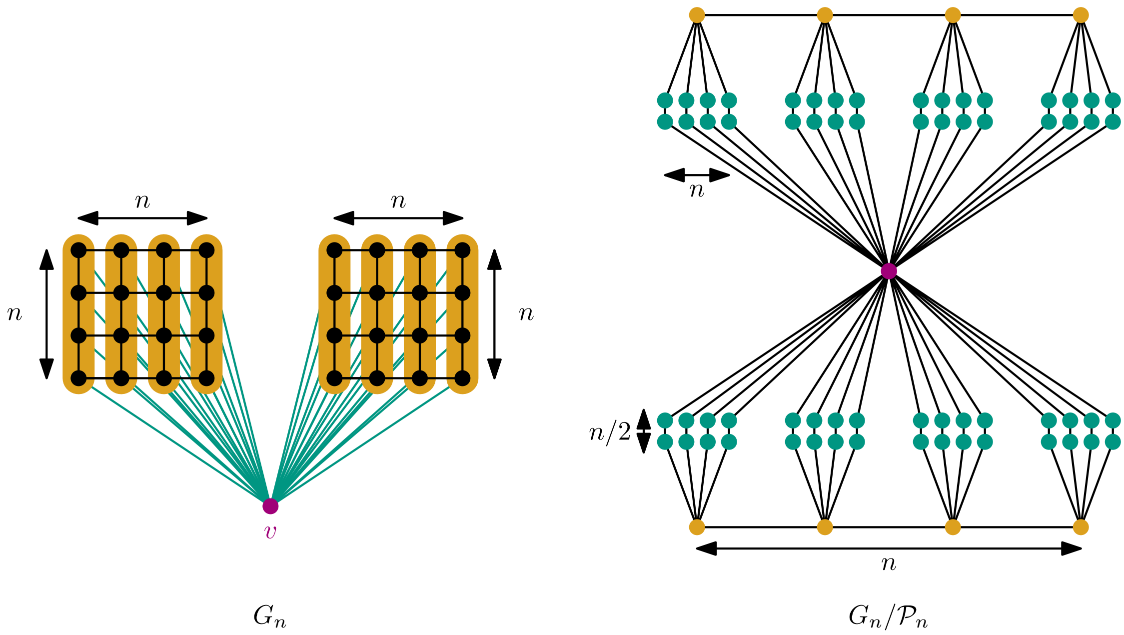


Figure 3.13: On the left: the graph G_n. On the right: the corresponding graph $G_n/\mathcal{P}_n$ where the vertex resulting from the contraction of $\{v\}$ is in the centre, connected to two paths of length n by subdivided multi-edges.

Next, we observe that the partition $\mathcal{P}_n$ is also a partition of layered width 1 for some layering and thus $\mathcal{G}$ has product structure. Thus, the following theorem holds:

Theorem 1.6: *There are graph classes with product structure but no BFS structure.*

Proof. We show that the graph class $\mathcal{G}$ has product structure but no BFS structure.

In Lemma 3.10 we show that $\mathcal{G}$ has no BFS structure.

For $G_n \in \mathcal{G}$ let G', G'' be the two $n \times n$-grids in G_n. We consider the partition $\mathcal{P}_n$ of $V(G_n)$ that contains the canonical partitions of G' and G'' and where every other vertex of G_n is in a part of its own. We show that this is a partition of $V(G_n)$ into sets of layered width 1 for some layering. We define a layering $\mathcal{L}$ of G_n. In $\mathcal{L}$ we place the i-th rows of the two grids in G_n in layer L_i and we place v in layer $L_{n/2}$. Since the paths from the grids to v have length at least $n/2$, we can place the remaining path vertices in layers in $\mathcal{L}$ such that $\mathcal{L}$ is a layering of

G_n. Note that the columns of the grids are sets of layered width 1 in $\mathcal{L}$. Thus $\mathcal{P}_n$ is a partition into sets of layered width 1. In Lemma 3.11 we show that $G_n/\mathcal{P}_n$ has treewidth 2. Therefore, every $G_n \in \mathcal{G}$ has row treewidth 2. ∎

4 Investigating Geodesic Structure

In this chapter, we investigate whether some known results for product structure also apply for geodesic structure.

4.1 Necessary condition: Linear Local Treewidth

It is known that linear local treewidth is a necessary condition for product structure [1]. Here, we prove that linear local treewidth is also a necessary condition for geodesic structure.

Theorem 1.10: *If a graph class admits geodesic structure, then it has linear local treewidth.*

Proof. Let $\mathcal{G}$ be a graph class that admits geodesic structure with geodesic treewidth c and let $G \in \mathcal{G}$. Thus, there exists a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ has treewidth at most c. Let $(\mathcal{X}, T)$ be a tree decomposition of $G/\mathcal{P}$ with width at most c. Let $v \in V(G)$ be any vertex of G and $N^k[v]$ be the k-th neighbourhood of v. We now construct a tree decomposition $(\mathcal{X}', T')$ of $G[N^k[v]]$. For each vertex $x \in V(G/\mathcal{P})$ let the corresponding geodesic be called P_x. We construct the bags $\mathcal{X}'$ by replacing in each bag from $\mathcal{X}$ each vertex $x \in V(G/\mathcal{P})$ with $V(P_x) \cap N^k[v]$. Assume $P \in \mathcal{P}$ is a geodesic with $V(P) \cap N^k[v] > 2k + 1$. Let x, y be the two outermost vertices on P that are contained in $N^k[v]$. It can be seen that P between x and y has at least length $2k + 1$. However, the shortest path from x to v and the shortest path from v to y have at most length k. Thus P between x and y does not constitute a shortest path between x, y and therefore P is not a geodesic. It follows that the number of vertices of a geodesic inside $N^k[v]$ is at most $2k + 1$. Thus the maximum bag size in $\mathcal{X}'$ is at most $(2k + 1)(c + 1)$. Note that some bags may be empty after the replacement, however, these bags can be removed, still resulting in a connected T'. Since $\mathcal{P}$ is a partition of G, every vertex of $N^k[v]$ is contained in at least one bag of $\mathcal{X}'$. For each edge $xy \in E(G[N^k[v]])$ it is either part of a geodesic in $\mathcal{P}$ and thus x, y are in a shared bag or it is between two geodesics $P_{x'}, P_{y'} \in \mathcal{P}$ where x', y' are the corresponding vertices in $G/\mathcal{P}$. Since x' and y' are in a shared bag in $\mathcal{X}$, it follows that x and y are in a shared bag in $\mathcal{X}'$. Lastly, since the subgraph induced by bags in $\mathcal{X}$ containing a vertex $x \in G/\mathcal{P}$ is connected, the subgraph induced by bags in $\mathcal{X}'$ containing a vertex $x' \in N^k[v]$ is also connected. Thus, $(\mathcal{X}', T')$ is a tree decomposition of $G[N^k[v]]$ and $G[N^k[v]]$ has treewidth at most $(2k + 1)(c + 1) - 1$ for every $v \in V(G)$ and and every $k \in \mathbb{N}$. ∎

4.2 Lower Bound for Geodesic Treewidth of Planar Graphs

Ueckerdt, Wood, and Yi [2] show that every planar graph has BFS treewidth at most 6. Thus 6 is also an upper bound for the row treewidth and geodesic treewidth of planar graphs.

Dujmović et al. show that 3 is a lower bound for the row treewidth of planar graphs by constructing a planar graph with row treewidth 3 [1]. We remark that the graph they construct also has geodesic treewidth 3 and thus 3 is also a lower bound for the geodesic treewidth of planar graphs.

In this section, we improve this bound for the geodesic treewidth of planar graphs by constructing a planar graph G that has geodesic treewidth at least 5.

Theorem 1.7: *There is a planar graph that has geodesic treewidth at least* 5.

Proof. In the following, we construct a planar graph G with $\mathrm{gtw}(G) \geq 5$. During the construction, we note Observations (i)-(iv) about the geodesics in G. For this purpose, we fix a partition $\mathcal{P}$ of G into geodesics.

Whenever we add new vertices to G, we take care to respect the following property: every shortest path between two previously added vertices stays a shortest path and no path of the same length between them is newly created. This makes sure that the observed properties (i)-(iv) also hold at the end of our construction.

We start the construction of G with a triangle with vertices v_0, v_1 and v_2. Since v_0, v_1 and v_2 form a triangle, it follows that:

(i) At least two vertices of v_0, v_1 and v_2 are not part of the same geodesic in $\mathcal{P}$.

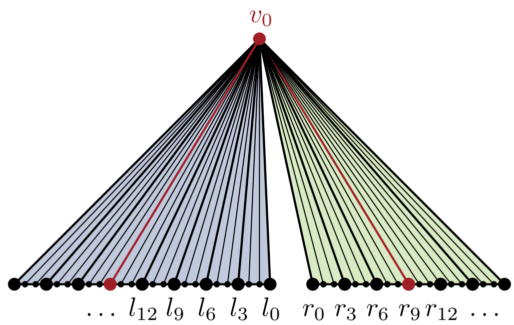


Figure 4.1: The right fan is highlighted in green and the left fan in blue. A possible geodesic containing v_0 is drawn in red. Note that every geodesic has at most three vertices in a fan. Thus, the thick/labelled vertices that are not red belong to pairwise distinct geodesics.

We add two fans at each v_i, where v_i is the centre. In the following, we only describe the construction of the fans containing v_0, however, disjoint copies of the same fans are added at v_1 and v_2 too. The vertices of the right fan with centre v_0 are called $r_0, r_1, \dots$ and the vertices of the left fan $l_0, l_1, \dots$. We will discuss the exact fan sizes at the end of the proof. These vertices together with v_0 form fans, meaning that $r_j r_{j+1}$ and $l_j l_{j+1}$ are edges in G. An example of v_0 and the adjacent fans is shown in Figure 4.1. Observe that every geodesic has at most three vertices in a fan and thus the geodesic containing v_i contains at most two other vertices in a fan with centre v_i. Thus it holds that:

(ii) The geodesics containing every third vertex in a fan with centre v_i, excluding the geodesic containing v_i, are pairwise distinct.

At this point in the construction, we thus know that for each fan the graph $G/\mathcal{P}$ contains a cycle of arbitrary length if the fan is large enough.

Next, for each $i \in [0, 2]$ we add vertices adjacent to the right fan of v_i. Again, we only describe the construction for v_0, however, equivalent disjoint copies are added for v_1 and v_2. For v_0 we add vertices $z_0, z_1, \dots$ where each vertex z_k is adjacent to r_{6k+3}. Each vertex z_k is also connected by disjoint paths of length 6 to the vertices r_{6k} and $r_{6(k+1)}$. An example of

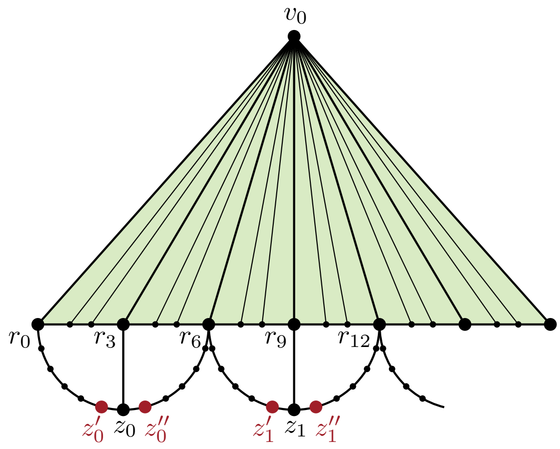


Figure 4.2: The right fan with its $z_0, z_1, \ldots$ vertices. Note that for all k, either z'_k or z''_k are part of a geodesic that contains no vertices in the fan.

v_0 and its right fan with its $z_0, z_1, \ldots$ vertices is shown in Figure 4.2. The first vertex on the length-6 path from z_k to r_{6k} is called z'_k and the first vertex on the length-6 path from z_k to $r_{6(k+1)}$ is called z''_k. Observe that for all vertices z'_k and z''_k the shortest path to any vertex in the right fan of v_0 includes the vertex r_{6k+3}. Hence, there is no geodesic containing all three vertices r_{6k+3}, z'_k and z''_k. Thus, it follows that:

(iii) Either z'_k or z''_k are part of a geodesic that contains no vertices of the right fan that they are adjacent to.

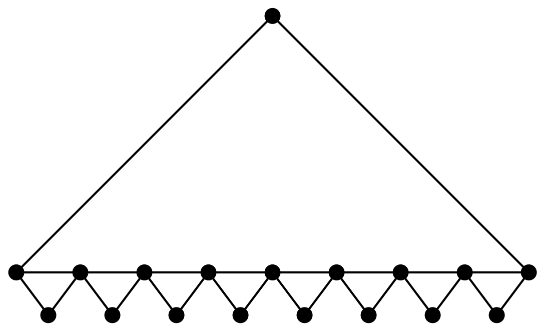


Figure 4.3: A cycle with tips.

At this point in the construction, we thus know that for each right fan, if the fan is large enough, the graph $G/\mathcal{P}$ contains a minor that consists of a cycle of arbitrary length with tips as shown in Figure 4.3. The tips result from vertices z'_k or z''_k as shown in Observation (iii).

Lastly for $i \in [0, 2]$ and any k we connect z'_k of the copy containing v_i to l_{12k} and l_{12k+3} of copy containing v_{i+1} using two disjoint paths of length 6. We also connect z''_k of the copy containing v_i to l_{12k+6} and l_{12k+9} of the copy containing v_{i+1} using another two disjoint paths of length 6. Here, the lower index of v_{i+1} is always taken modulo 3. A sketch of the resulting graph is given in Figure 4.4 and there it can also be seen that G is planar. By construction, the paths are long enough such that the shortest path between z'_k and l_{12k} or l_{12k+3} is of length 5 and contains v_i and v_{i+1}. An analogous observation holds for z''_k. It thus follows that:

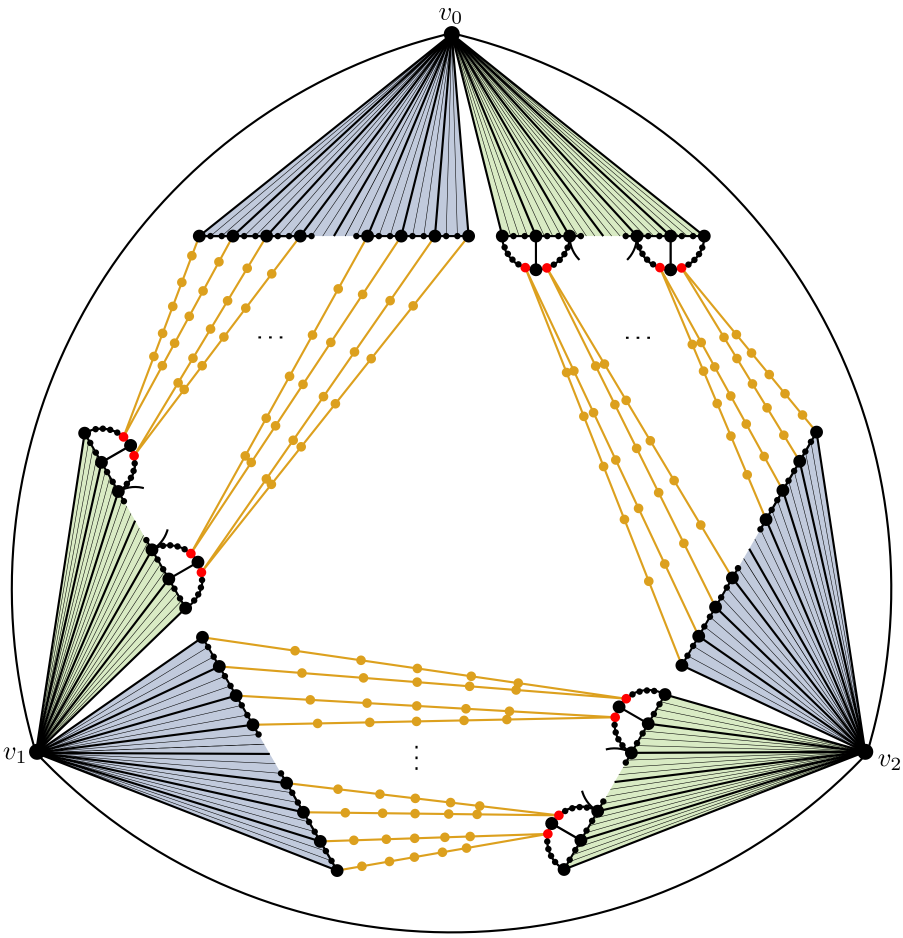


Figure 4.4: The planar graph with geodesic treewidth 5. The right fans are highlighted in green and the left fans are highlighted in blue.

(iv) If v_i and v_j are not part of the same geodesic then there is no geodesic that both contains vertices of the right fan with centre v_i (including the vertices z_k and paths connecting to the right fan) and vertices of the left fan with centre v_j.

For any sub-fan F of the right fan with centre v_i we define the adjacent sub-fan of v_{i+1} as the sub-fan of the left fan with centre v_{i+1} that contains v_{i+1} and all vertices that are connected using the length 6 paths to a vertex z'_k or z''_k with z_k adjacent to F.

The construction of G is now completed and we use the Observations (i)-(iv) to analyse $G/\mathcal{P}$ and aim to show that $G/\mathcal{P}$ contains the graph H in Figure 4.8 as a minor. The graph H is known to have treewidth 5.

Observation (i) is that at least two of the vertices v_0, v_1 and v_2 are not part of the same geodesic. Let without loss of generality v_0 and v_1 not be part of the same geodesic and P_i be the geodesic containing v_i for $i \in \{0, 1\}$. Since P_0 contains at most three vertices in the right fan of v_0, the geodesic P_0 splits this fan into at most three sub-fans. If we choose the right fans large enough, then by pigeonhole-principle one of the three sub-fans is large enough such that it contains r_{6k}, r_{6k+3} and $r_{6(k+1)}$ for at least 14 consecutive k. Let this sub-fan be called F'. The geodesic P_1 contains at most two vertices, besides v_1, in the left fan of v_1. Therefore since $3 \cdot 4 + 2 = 14$ it holds that there is a sub-fan F'' of F' that is large enough that it contains

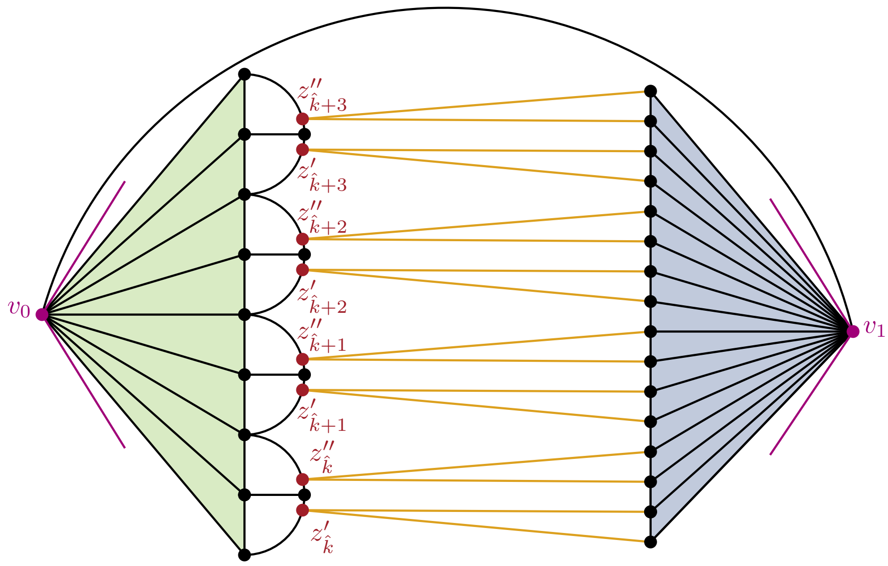


Figure 4.5: In green the right sub-fan of v_0 and in blue the left sub-fan of v_1. Subdivision vertices are not drawn for better readability. In purple the geodesics containing v_0 and v_1 that do not intersect the sub-fans besides in v_0 and v_1.

r_{6k}, r_{6k+3} and $r_{6(k+1)}$ for at least four consecutive k and the adjacent sub-fan of v_1 does not intersect P_1 outside of v_1. Let the four consecutive k be $\hat{k}, \hat{k}+1, \hat{k}+2$ and $\hat{k}+3$. A sketch of the sub-fan F'' and the adjacent sub-fan of v_1 is given in Figure 4.5. We calculate that a fan size for the right fans of $(14 \cdot 3 + 4) \cdot 6 + 1 = 277$ suffices, which corresponds to a fan size of $(14 \cdot 3 + 4) \cdot 4 \cdot 3 - 2 = 550$ for the left fans. Observation (iii) states that for any k, either z'_k or z''_k are part of a geodesic that contains no vertices in the right fan of v_0 . Without loss of generality let the geodesics containing $z'_{\hat{k}}, z'_{\hat{k}+1}, z'_{\hat{k}+2}$ and $z'_{\hat{k}+3}$ not contain any vertex that is part of the right fan of v_0. Lastly using Observations (ii) and (iv) we can thus conclude that the geodesics of every third vertex in the right sub-fan of v_0, of every third vertex of the left fan of v_1 and of $z'_{\hat{k}}, z'_{\hat{k}+1}, z'_{\hat{k}+2}$ and $z'_{\hat{k}+3}$ are all pairwise distinct. The mentioned vertices are shown in Figure 4.6.

With the geodesics of all these vertices being distinct, we compare Figure 4.6 and the alternative drawing of H in Figure 4.7. It can easily be seen that $G/\mathcal{P}$ contains a H-minor where the geodesics containing $z'_{\hat{k}}, z'_{\hat{k}+1}, z'_{\hat{k}+2}, z'_{\hat{k}+3}$ correspond to b_1, b_2, b_3, b_4. The vertices a_1, a_2, a_3, a_4 form by contracting some geodesics in the left fan of v_1 and the vertices a_1, a_2, a_3, a_4 by contracting some geodesics in the right fan of v_0. Since H has treewidth 5 it thus follows that G is a planar graph that has geodesic treewidth at least 5.

∎

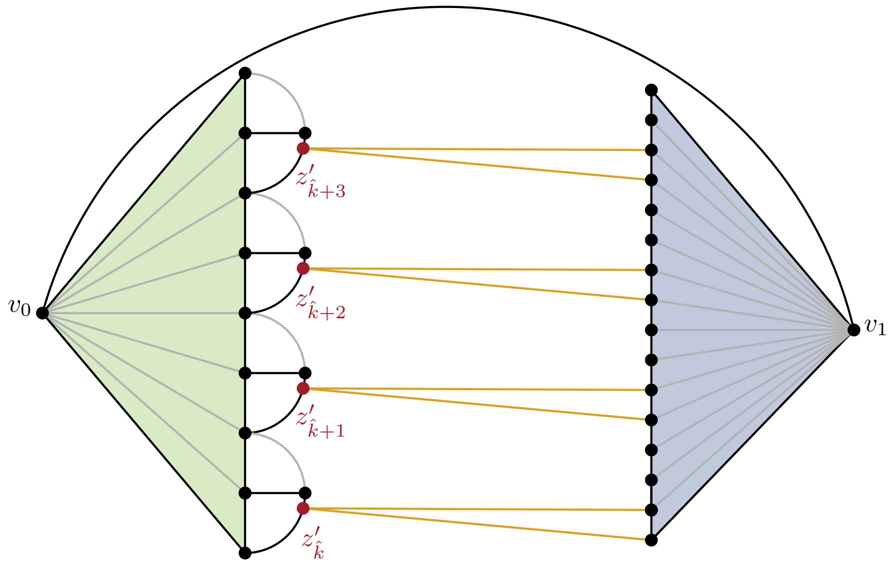


Figure 4.6: In green the right sub-fan of v_0 and in blue the left sub-fan of v_1. Subdivision vertices are not drawn and only every third vertex of the fans is drawn for better readability. All the drawn vertices of the fans and z'_k, z'_{k+1}, z'_{k+2} and z'_{k+3} are all part of pairwise distinct geodesics. The edges and paths in light grey are not needed for the H minor.

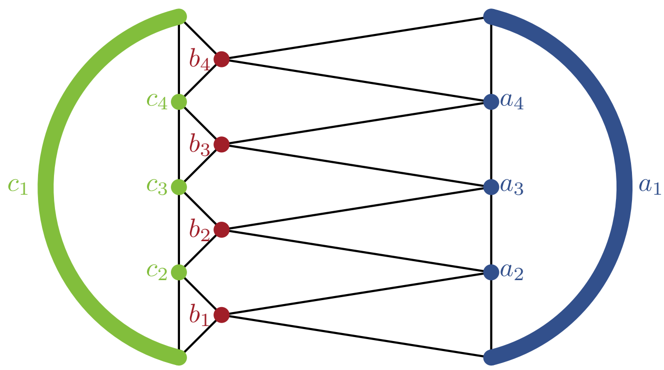


Figure 4.7: Alternative drawing of graph H that is known to have treewidth 5. Note that c_1 and a_1 are vertices that are drawn as half-circles to match the embedding of g in Figure 4.6. The graph H is a minor of Figure 4.6 where the geodesics containing $z'_{\hat{k}}, z'_{\hat{k}+1}, z'_{\hat{k}+2}, z'_{\hat{k}+3}$ correspond to b_1, b_2, b_3, b_4. The vertices a_1, a_2, a_3, a_4 form by contracting some geodesics in the left fan of v_1 and the vertices a_1, a_2, a_3, a_4 by contracting some geodesics in the right fan of v_0.

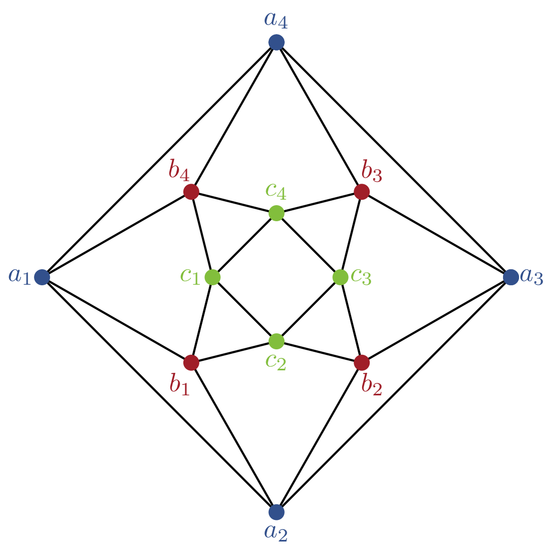


Figure 4.8: Graph H that is known to have treewidth 5.

4.3 Computing Geodesic Treewidth

4.3.1 Computing the Geodesic Treewidth is NP-Hard

For row treewidth, it is known that computing the row treewidth of treewidth 2 graphs is NP-hard [23]. In the following, we prove that computing the geodesic treewidth of graphs is also NP-hard.

Theorem 4.1: *Determining whether a given graph has geodesic treewidth at most 2 is NP-hard.*

Proof. To show that this problem is NP-hard, we give a reduction from a SAT variant that is known to be NP-complete. Tovey [35] has shown that SAT is NP-complete if every clause contains 2 or 3 variables and every variable occurs at most trice. In particular, in their reduction, they show 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 and construct the following graph. We begin with a path P of length $2n+1$ where n is the number of variables V in I. 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 neighbourhood as the vertices we are duplicating. Let the resulting graph be called G_1 and let $\{(x_i, \overline{x_i}) \mid x_i \in P \wedge \overline{x_i} \textit{ is the duplicate vertex of } x_i\}$ be the pairs of vertices that were duplicated. An example of such a graph G_1 is shown in Figure 4.9 on the left. For a s-t-path S, we say $x_i \in V$ is true if and only if $x_i \in S$, thus every path S corresponds uniquely to a truth assignment of the variables V. For each clause C of I 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 4.9 in the centre. In addition for each clause $c = \{l_1, l_2\} \in C$ 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 4.9 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.

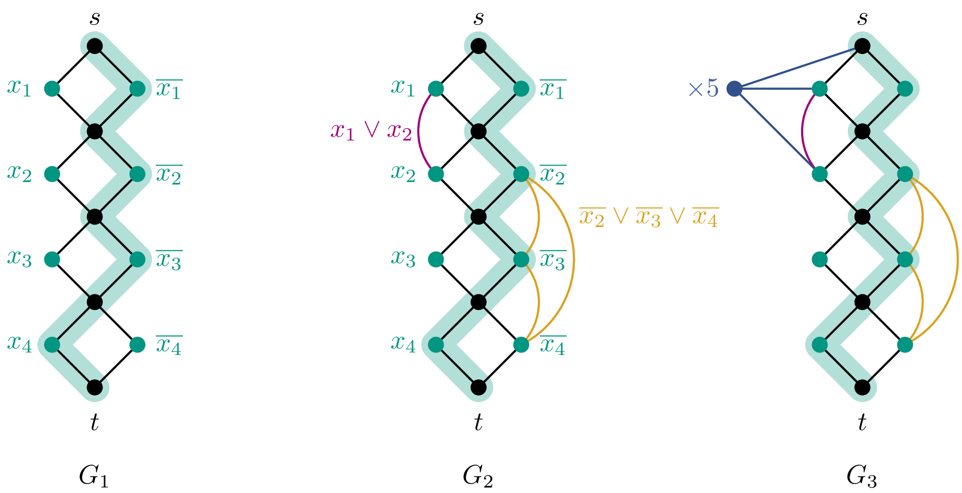


Figure 4.9: On the left an example of G_1 with a shortest s-t-path highlighted in green that uniquely corresponds to a truth assignment φ of the variables where $\varphi(x_4) = t$ and $\varphi(x_i) = f$ for $i \in [3]$. In the centre a corresponding G_2 with a SAT instance consisting of two clauses. On the right a corresponding G_3, the dark blue vertex has four additional copies with the same neighbourhood.

For a graph G with $s, t \in V(G)$, let a s-t-partition of G into geodesics be a partition into geodesics that contains a shortest s-t-path. We show that I is satisfiable if and only if there exists a s-t-partition $\mathcal{P}$ of G_4 into geodesics such that $G_4/\mathcal{P}$ has treewidth at most 2.

Firstly, let us assume that I is a non-satisfiable instance. Let $\mathcal{P}$ be a s-t-partition of G_4 into geodesics with $S \in \mathcal{P}$ being a shortest s-t-path. We prove that $\operatorname{tw}(G_4/\mathcal{P}) \geq 3$. Since I is non-satisfiable, there exists a clause $c \in C$ such that no literal in c is on S. Assume c is a clause of size 3 and let $c = \{l_1, l_2, l_3\}$. We show that the graph $G_4/\mathcal{P}$ contains a K_4 minor obtained from S and the geodesics in $\mathcal{P}$ containing l_i. This is the case since for any two $l_i, l_j \in c$ with $l_i \neq l_j$ it holds that l_i and l_j are not part of the same geodesic in $\mathcal{P}$ since every shortest path between l_i and l_j includes vertices in S. Therefore, the vertices in $G_4/\mathcal{P}$ corresponding to the geodesic S and the geodesics containing l_1, l_2 and l_3 respectively form a K_4 minor. Else if $c = \{l_1, l_2\}$ is a clause of size 2 then the graph $G_4/\mathcal{P}$ also contains a K_4 minor. Indeed, there are five vertices p_i that are connected to l_1, l_2 and s in G_4 and it can be seen that at most four of them can be part of the same geodesic as l_1 or l_2. Thus, without loss of generality, let p_1 not be part of the geodesics corresponding to l_1 and l_2. Recall that since $S \subseteq G_1$ it holds that $p_1 \notin S$. Additionally, as before, it holds that l_1 and l_2 are not part of the same geodesic in $\mathcal{P}$ since any shortest path between l_1 and l_2 includes vertices in S. Therefore, the vertices in $G_4/\mathcal{P}$ corresponding to the geodesic S and the geodesics containing l_1, l_2 and p_1 respectively are part of a K_4 minor. Thus, assuming that a shortest path between s and t is a geodesic that gets contracted, the geodesic treewidth of G_4 is at least 3 if the corresponding SAT instance is non-satisfiable.

Secondly, let us assume that I is a satisfiable instance. We show that in this case the geodesic treewidth of G_4 is at most 2. Let φ be a satisfying truth assignment for I and let S be the corresponding s-t-path. It holds that for every $c \in C$ at least one literal of c is on S. We first consider the graph G_2', which results from G_2 by contracting S. Let σ be the vertex corresponding to S in G_2'. We consider the graph $G_2' - \sigma$. Let l_1 be a vertex in $G_2' - \sigma$ and thus $\varphi(l_1) = f$. Next, we observe that l_1 has degree at most 1 in $G_2' - \sigma$. This is the case since

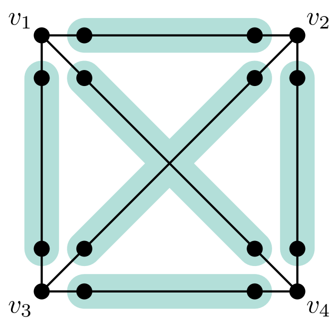


Figure 4.10: K_4 with vertices v_1, v_2, v_3, v_4 where each highlighted edge gets replaced by a copy of G_4.

$\varphi(l_1) = f$, so for every clause $c = \{l_1, l_2\}$ the literal l_2 is true and thus $l_2 \in S$. Thus, edges introduced for clauses of size 2 do not contribute to the degree of l_1 in $G_2' - \sigma$. Similarly, for every clause $c = \{l_1, l_2, l_3\}$ either l_2 or l_3 is true and thus $l_2 \in S$ or $l_3 \in S$. Since l_1 occurs in at most one clause of length 3 it holds that l_1 has degree at most 1 in $G_2' - \sigma$. Since every vertex in $G_2' - \sigma$ has at most degree 1, $G_2' - \sigma$ is the disjoint union of edges and singletons. Therefore, the treewidth of $G_2' - \sigma$ is 1 and thus G_2' has at most treewidth 2.

Next, we lift the argument to G_3. For this, let G_3' be the graph resulting from G_3 by contracting S. The graph G_3' compared to G_2' only has the additional vertices p_i for $c \in C$. However since for any $c = \{l_1, l_2\} \in C$ either l_1 or l_2 is on S it holds that the neighbourhood $N_{G_3'}(p_i)$ of p_i is either an edge or a vertex for all $c \in C$ and $i \in [5]$. Thus, p_i can be added to the tree decomposition by adding a bag containing p_i and $N_{G_3'}(p_i)$ adjacent to a bag containing $N_{G_3'}(p_i)$. Therefore, the treewidth of G_3' is at most 2.

Finally, we lift the argument to G_4. For this, let G_4' be the graph resulting from G_4 by contracting S. It can be seen that G_4' differs from G_3' only by adding subdivisions and multi-edges, however, since these do not increase the treewidth beyond 2, the graph G_4' also has treewidth at most 2. Lastly we define the following partition into geodesics $\mathcal{P} = \{S\} \cup \{\{v\} \mid v \in G_4 - S\}$. It holds that $G_4/\mathcal{P} = G_4'$ and thus the geodesic treewidth of G_4 is at most 2. Note that we can guarantee that $G_4/\mathcal{P}$ has treewidth 2 while $\mathcal{P}$ consists only of a path from s to t and other geodesics of length 1.

Finally, we will use G_4 as a gadget to construct G, a graph for which it holds that I is satisfiable if and only if G has geodesic treewidth at most 2. For this purpose, we take a K_4 with vertices v_1, v_2, v_3, v_4, subdivide each edge twice and replace the resulting six edges whose endpoints are subdivision vertices each by a copy of G_4 where s and t correspond to the two subdivision vertices. A sketch of this construction is given in Figure 4.10.

We claim that G has geodesic treewidth at most 2 if I is a satisfiable instance and geodesic treewidth at least 3 if I is a non-satisfiable instance.

First, let I be a satisfiable instance. As shown before G_4 thus has geodesic treewidth at most 2 and we can guarantee that $G_4/\mathcal{P}$ has treewidth 2 for $\mathcal{P} = \{S\} \cup \{\{v\} \mid v \in G_4 - S\}$ where S is a shortest path from s to t. Note that the geodesics corresponding to the geodesics in $\mathcal{P}$ are also geodesics in G since every path between two vertices in the same copy of G_4 that uses vertices outside this copy has length at least $4n$. We define a partition $\mathcal{P}'$ of G into geodesics as follows. Let $\mathcal{P}'$ contain the geodesics of $\mathcal{P}$ for all copies of G_4. Let $S_{i,j}$ be the geodesic corresponding to the s-t-path in the copy of G_4 between v_i and v_j. Finally we extend $S_{1,2}$ by v_1 and v_2 and $S_{3,4}$ by v_3 and v_4 to get $\mathcal{P}'$. Let $\sigma_{i,j}$ be the vertex in $G/\mathcal{P}'$ corresponding to the,

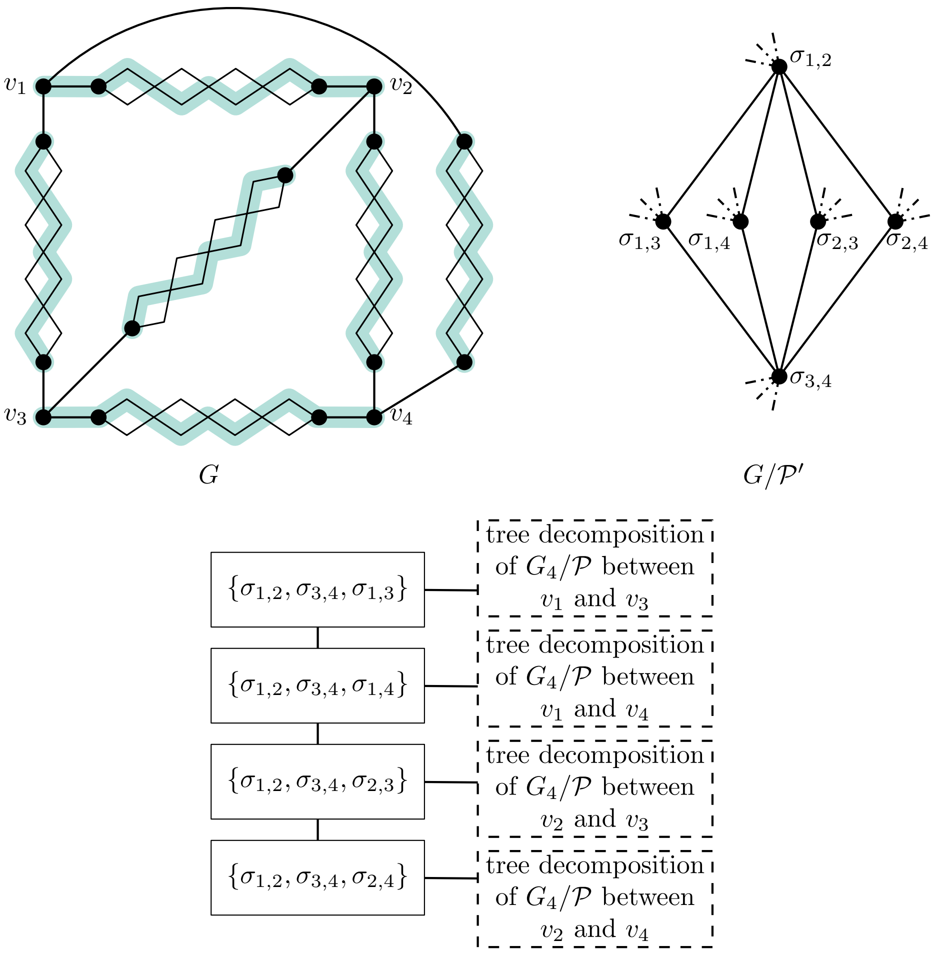


Figure 4.11: On the top left is an approximation of a graph G with the geodesics $S_{i,j}$ highlighted in green. On the top right is the corresponding graph $G/\mathcal{P}'$. On the bottom it is shown how to append the tree decompositions of the copies of $G_4/\mathcal{P}$ such that it results in a tree decomposition of $G/\mathcal{P}'$ with bag size 3.

possibly extended, geodesic $S_{i,j}$. It holds that $G/\mathcal{P}'$ has treewidth 2 since we can append the tree decompositions of the copies of $G_4/\mathcal{P}$ to the tree decomposition given in Figure 4.11 to get a tree decomposition of $G/\mathcal{P}'$. Thus G has geodesic treewidth at most 2 if I is a satisfiable instance.

Finally, let I be a non-satisfiable instance. Let $\mathcal{P}'$ be any partition into geodesics of G. If no two vertices v_i and v_j for $i \neq j$ are contracted in $G/\mathcal{P}'$, then $G/\mathcal{P}'$ contains a K_4 minor and thus has treewidth at least 3. If v_i and v_j are contracted in $G/\mathcal{P}'$ then let $\mathcal{P}$ be the geodesics of $\mathcal{P}'$ restricted to the copy of G_4 between v_i and v_j. It thus holds that $\mathcal{P}$ is a s-t-partition into geodesics of G_4. In addition $G_4/\mathcal{P}$ is a subgraph of $G/\mathcal{P}'$, so if $G/\mathcal{P}'$ has treewidth at most 2 then $G_4/\mathcal{P}$ also has treewidth at most 2. However, as shown before, assuming that a shortest s-t-path is a geodesic in $\mathcal{P}$, the geodesic treewidth of G_4 is at least 3 given that I is non-satisfiable. Thus, the geodesic treewidth of G is at least 3 if I is a non-satisfiable instance. ∎

Recall that for three graphs G_1, G_2, G_3 the graph $G^\star_{1,2,3}$ is defined as follows: connect a disjoint copy of G_1, G_2 and G_3 by a universal vertex v and subdivide the edges adjacent to v an arbitrary number of times. Each edge adjacent to v can be subdivided a different number of times. For a graph G with geodesic treewidth k let $G = G_1 = G_2 = G_3$ and let each edge adjacent to v be subdivided exactly $|V(G)|$ times. In this special case let the resulting graph $G^\star_{1,2,3}$ be called $G^\star$.

Lemma 3.6 gives us that $G^\star$ has geodesic treewidth at least $k+1$. In the following, we show that $G^\star$ has geodesic treewidth exactly $k+1$.

Lemma 4.2: *If G has geodesic treewidth k then $G^\star$ has geodesic treewidth at most $k+1$.*

Proof. We prove that $\text{gtw}(G^\star) \leq k+1$. Let $\mathcal{P}$ be a partition of G into geodesics such that $G/\mathcal{P}$ has treewidth k. The geodesics in $\mathcal{P}$ are also geodesics in $G^\star$ since any possible shortcut using v has length at least $2|V(G)|$. We thus define a partition $\mathcal{P}^\star$ of $G^\star$ into geodesics consisting of $\mathcal{P}$ for each copy of G and $\{\{x\} \mid x \in V(G^\star) \setminus (V(G_1) \cap V(G_2) \cap V(G_3))\}$. Thus $\mathcal{P}^\star$ has all geodesics corresponding to $\mathcal{P}$ for the copies of G and all other vertices are geodesics of length 0. We consider the graph G' that consists of three disjoint copies of $G/\mathcal{P}$ joined by a universal vertex v. It is known that adding a universal vertex to a graph increases the treewidth by at most one and thus G' has treewidth at most $k+1$. Observe that $G^\star/\mathcal{P}^\star$ is isomorphic to G' with some additional subdivisions and thus $\text{tw}(G^\star/\mathcal{P}^\star) \leq k+1$. Therefore, $G^\star$ has geodesic treewidth at most $k+1$. ∎

The following theorem follows from Theorem 4.1, Lemma 3.6 and Lemma 4.2.

Theorem 1.8: *Determining if a given graph has geodesic treewidth at most k is NP-hard for $k \geq 2$.*

4.3.2 Computing the Geodesic Treewidth of Treewidth-2 Graphs in Polynomial Time

Computing the row treewidth of treewidth-2 graphs is NP-hard [23]. In Section 4.3.1 we show that computing the geodesic treewidth in general is also NP-hard. Here we show that, different from row treewidth, computing the geodesic treewidth of graphs of treewidth 2 is not NP-hard.

We first give an algorithm that determines the geodesic treewidth of series-parallel graphs, which we will then use as a part of our algorithm for treewidth-2 graphs in general.

We define series-parallel graphs as follows. A single edge is a series-parallel graph where one vertex is the source s and the other the sink t. Given two disjoint series-parallel graphs G_1 and G_2 we can combine them using parallel composition by identifying the sources of G_1 and G_2 to create the source of the new graph and identifying the sinks of G_1 and G_2 to create the sink of the new graph. We can also combine G_1 and G_2 using series composition by identifying the sink of G_1 with the source of G_2. The source of G_1 is the source of the new graph and the sink of G_2 is the sink of the new graph. A series–parallel decomposition D of a series-parallel graph G is a recursive decomposition tree that records how G can be constructed from single edges via series compositions and parallel compositions. Such a series-parallel decomposition of G can be calculated in linear time [36].

The following theorem by Bodlaender [37] gives an intuition for why our algorithm for series-parallel graphs might be useful for treewidth-2 graphs.

Theorem 4.3 (Bodlaender [37])**:** *A graph G has treewidth at most 2, if and only if every biconnected component of G is a series-parallel graph.*

Lemma 4.4: *The geodesic treewidth of series-parallel graphs can be determined in time $O(n^5)$.*

Proof. We give an algorithm that determines the geodesic treewidth of a series-parallel graph G in time $O(n^5)$.

The first step of the algorithm is to pre-compute and store the all-pair-shortest-path distances in the given graph G. This can be done in at most time $O(n^2)$, using for example n breadth-first searches.

The second step is to compute a series–parallel decomposition D of G in linear time [36].

Our algorithm for determining the geodesic treewidth of series-parallel graphs is a DP algorithm that processes the subgraphs along a bottom-up order in the series-parallel decomposition D.

We first define the following concepts that our algorithm relies on:

Let G' be a subgraph of G in D with source s and sink t. For a partition $\mathcal{P}'$ of G' into geodesics and a vertex $x \in G'$ let the geodesic containing x in $\mathcal{P}'$ be called P'_x. We define $\nu(P'_x)$ as the vertex in $G'/\mathcal{P}'$ that corresponds to the geodesic P'_x. Next, we define a *valid partition* $\mathcal{P}'$ of G' as a partition of the vertices of G' into geodesics such that the geodesics in $\mathcal{P}'$ are also geodesics in G and additionally $G'/\mathcal{P}'$ fulfils one of the following conditions. Either $G'/\mathcal{P}'$ is a tree or identifying $\nu(P'_s)$ with $\nu(P'_t)$ in $G'/\mathcal{P}'$ results in a tree. If $G'/\mathcal{P}'$ fulfils the first condition, then $\mathcal{P}'$ is a case 1 valid partition and if $G'/\mathcal{P}'$ only fulfils the second condition, then $\mathcal{P}'$ is a case 2 valid partition.

For a valid partition $\mathcal{P}'$ of G' we define the corresponding *valid configuration* as the following set of information:

- $\deg_{\mathcal{P}'}(s)$ the number of neighbours of s that are in the same geodesic as s in $\mathcal{P}'$, this value is in $\{0, 1, 2\}$
- $\deg_{\mathcal{P}'}(t)$ the number of neighbours of t that are in the same geodesic as t in $\mathcal{P}'$, this value is in $\{0, 1, 2\}$
- if $\deg_{\mathcal{P}'}(s) = 1$, then the other endpoint of the geodesic starting at s, there are at most n possible endpoints
- if $\deg_{\mathcal{P}'}(t) = 1$, then the other endpoint of the geodesic starting at t, there are at most n possible endpoints

- whether $\mathcal{P}'$ is a case 1 or case 2 valid partition
- length l of the longest $v(P'_s)$-$v(P'_t)$-path in $G'/\mathcal{P}'$, we only store the exact distance for $l \in \{0, 1, 2\}$, else we just store value greater than 2, we define $l = 0$ if $P'_s = P'_t$

Note that different valid partitions of the graph can result in the same valid configuration. It can be seen from the definition of valid configurations that the following claim holds.

Claim 1. The number of valid configurations that exist for a subgraph G' is at most $3^2 \cdot n^2 \cdot 2 \cdot 4$.

We prove that the following claim also holds:

Claim 2. The geodesic treewidth of G is 1 if and only if there exists a valid case 1 configuration of G. Else, the geodesic treewidth of G is 2.

Proof. Assume there is a valid case 1 configuration of G. This implies that there exists a partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree. Thus G has geodesic treewidth 1.

Assume G has geodesic treewidth 2 and therefore there is no partition $\mathcal{P}$ of G into geodesics such that $G/\mathcal{P}$ is a tree. Since such a partition $\mathcal{P}$ is a case 1 valid partition of G, it thus holds that no valid case 1 configuration of G exists.

Since series-parallel graphs have treewidth 2, it holds that G has geodesic treewidth at most 2. ∎

Claim 2 means that if we are able to determine all possible valid configurations of G, then we know the geodesic treewidth of G.

Our DP algorithm that processes the subgraphs along a bottom-up order in the series-parallel decomposition D calculates and stores all valid configurations of the current subgraph G' in each step.

There are three possible cases for G':

Case 1: A single edge. The current subgraph G' is the simplest series-parallel graph, an edge st. Then there are only two valid partitions of G'. One where s and t are part of one geodesic and one where s and t are not part of the same geodesic. Thus, the following claim holds:

Claim 3. If G' is an edge st, we can compute all valid configurations in time $O(1)$.

For the cases that G' is a series composition or a parallel composition of two children, we show the following:

We first describe how to decide in time $O(1)$, given some valid configurations of the children of the current subgraph, if the combination of these two valid configurations results in a valid configuration for G'. We then show that the approach of testing all combinations of valid configurations for the children indeed finds all valid configurations for the current subgraph.

Case 2: Series composition. The current subgraph G' is the result of the series composition of two solved subgraphs G_1 and G_2, where we identify the sink t_1 of G_1 with the source s_2 of G_2. We define how we combine a valid partition $\mathcal{P}_1$ of G_1 and a valid partition $\mathcal{P}_2$ of G_2 into a partition of G'.

Let P_{s_1} and P_{t_1} be the geodesics in $\mathcal{P}_1$ that contain s_1 and t_1 respectively. Let P_{s_2} and P_{t_2} be the geodesics in $\mathcal{P}_2$ that contain s_2 and t_2 respectively. Lastly, let $v(P_{s_i})$ and $v(P_{t_i})$ be the vertices in $G_i/\mathcal{P}_i$ that correspond to P_{s_i} and P_{t_i} respectively for $i \in \{1, 2\}$.

We combine a valid partition $\mathcal{P}_1$ of G_1 and a valid partition $\mathcal{P}_2$ of G_2 into a partition $\mathcal{P}'$ of G' by taking the geodesics in $\mathcal{P}_1$ and $\mathcal{P}_2$ and combining P_{t_1} and P_{s_2} into a single part. We describe how to check if $\mathcal{P}'$ is a valid partition of G'. During the description, note that if $\mathcal{P}'$ is valid or not depends only on properties of $\mathcal{P}_1$ and $\mathcal{P}_2$ that are stored in the corresponding valid configurations. Thus, our algorithm will check for pairs of valid configurations of the children if they result in a valid configuration of G' instead of having to test pairs of valid partitions.

Claim 4. In the case of series composition, we can check if $\mathcal{P}'$ is a partition into geodesics that are also geodesics in G, in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Proof. $\mathcal{P}'$ is a partition into paths if and only if $\deg_{\mathcal{P}_1}(t_1) + \deg_{\mathcal{P}_2}(s_2) < 3$. If this is the case then we check if $\mathcal{P}'$ is a partition of G' into geodesics that are also geodesics in G. All geodesics that do not contain the vertex i resulting from the identification of the sink t_1 of G_1 with the source s_2 of G_2 are in $\mathcal{P}_1$ or $\mathcal{P}_2$ and thus geodesics in G. Let x_1 and x_2 be the endpoints of the geodesics starting at t_1 and s_2 in $\mathcal{P}_1$ and $\mathcal{P}_2$ respectively. For the path in $\mathcal{P}'$ that contains the identification vertex i we check if it is a geodesic in G by checking if $d(x_1, i) + d(x_2, i) \leq d(x_1, x_2)$ where $d(u, v)$ refers to the precomputed lengths of shortest paths between the vertices. If this is the case, then $\mathcal{P}'$ is a partition of G' into geodesics that are also geodesics in G. ∎

Claim 5. In the case of series composition, let $\mathcal{P}'$ be a partition into geodesics that are also geodesics in G. We can check if $\mathcal{P}'$ is a *valid* partition and compute the corresponding valid configuration in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Proof. Let $\mathcal{P}'$ be a partition into geodesics in G. If $\mathcal{P}_1$ and $\mathcal{P}_2$ are case 1 valid partitions then $G_1/\mathcal{P}_1$ is a tree and $G_2/\mathcal{P}_2$ is a tree and thus $G'/\mathcal{P}'$ is also a tree. Therefore, $\mathcal{P}'$ is a case 1 valid partition of G'.

Consider the case that $\mathcal{P}_1$ is a case 2 valid partition. In this case $G_1/\mathcal{P}_1$ is not a tree however if the vertices $v(P_{s_1})$ and $v(P_{t_1})$ in $G_1/\mathcal{P}_1$ are identified then this results in a tree. Observe that $\mathcal{P}'$ is a valid partition if the source and sink of G_2 are in the same geodesic in $\mathcal{P}_2$. In this case, $\mathcal{P}'$ is a case 2 valid partition for G'. Else if the source and sink of G_2 are not in the same geodesic in $\mathcal{P}_2$ then $\mathcal{P}'$ is not a valid partition since $G'/\mathcal{P}'$ is not a tree and identifying $v(P'_s)$ with $v(P'_t)$ in $G'/\mathcal{P}'$ will also not result in a tree. Whether the source and sink of G_2 are in the same geodesic or not can be checked using only information stored in the valid configurations in time $O(1)$ by checking if the length l_2 of the longest $v(P_{s_2})$-$v(P_{t_2})$-path in $G_2/\mathcal{P}_2$ is 0. An analogous statement is true if $\mathcal{P}_2$ is a case 2 valid partition. If both $\mathcal{P}_1$ and $\mathcal{P}_2$ are case 2 valid partitions, then $\mathcal{P}'$ is not a valid partition. ∎

Combining Claim 4 and Claim 5 gives us the following result:

Claim 6. In the case of series composition, we can check if $\mathcal{P}'$ is a *valid* partition and compute the corresponding valid configuration in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Case 3: Parallel composition. The current subgraph G' is the result of the parallel composition of two solved subgraphs G_1 and G_2, where we identify the two sources s_1, s_2 and the two sinks t_1, t_2. We define how we combine a valid partition $\mathcal{P}_1$ of G_1 and a valid partition $\mathcal{P}_2$ of G_2 into a partition of G'.

Let P_{s_1} and P_{t_1} be the geodesics in $\mathcal{P}_1$ that contain s_1 and t_1 respectively. Let P_{s_2} and P_{t_2} be the geodesics in $\mathcal{P}_2$ that contain s_2 and t_2 respectively. Lastly, let $v(P_{s_i})$ and $v(P_{t_i})$ be the vertices in $G_i/\mathcal{P}_i$ that correspond to P_{s_i} and P_{t_i} respectively for $i \in \{1, 2\}$.

We combine a valid partition $\mathcal{P}_1$ of G_1 and a valid partition $\mathcal{P}_2$ of G_2 into $\mathcal{P}'$ by taking the geodesics in $\mathcal{P}_1$ and $\mathcal{P}_2$. Then we combine P_{s_1} and P_{s_2} into a single part and also combine P_{t_1} and P_{t_2} into a single part. We describe how to check if $\mathcal{P}'$ is a valid partition of G'. During the description, note that if $\mathcal{P}'$ is valid or not depends only on properties of $\mathcal{P}_1$ and $\mathcal{P}_2$ that are stored in the corresponding valid configurations. Thus, our algorithm will check for pairs of valid configurations of the children if they result in a valid configuration of G' instead of having to test pairs of valid partitions.

Claim 7. In the case of parallel composition, we can check if $\mathcal{P}'$ is a partition into geodesics that are also geodesics in G, in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Proof. $\mathcal{P}'$ is a partition into paths if and only if $\deg_{\mathcal{P}_1}(s_1) + \deg_{\mathcal{P}_2}(s_2) < 3$ and $\deg_{\mathcal{P}_1}(t_1) + \deg_{\mathcal{P}_2}(t_2) < 3$. The only exception to this is if the merged geodesics are a cycle. However, we detect this cycle when checking if the merged geodesics are a geodesic in G. Without loss of generality, let us consider two geodesics merged at the source s of G' with endpoints x_1, x_2. The merged path is a geodesic in G if and only if $d(x_1, s) + d(x_2, s) \le d(x_1, x_2)$ where for two vertices u, v the distance $d(u, v)$ refers to the precomputed lengths of shortest paths between the vertices u, v. This can be checked using only information stored in the valid configurations in time $O(1)$. An analogous statement is true for paths merged at the sink of G'. All other geodesics in $\mathcal{P}'$ are also geodesics in $\mathcal{P}_1$ or $\mathcal{P}_2$ and are thus geodesics in G. ■

Claim 8. In the case of parallel composition, let $\mathcal{P}'$ be a partition into geodesics that are also geodesics in G. We can check if $\mathcal{P}'$ is a *valid* partition and compute the corresponding valid configuration in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Proof. Let l_1 be the length of the longest $v(P_{s_1})$-$v(P_{t_1})$-path in $G_1/\mathcal{P}_1$, where $v(P_{s_1})$ and $v(P_{t_1})$ refer to the vertices corresponding to the geodesics containing s_1 and t_1 respectively. Let l_2 be the length of the longest $v(P_{s_2})$-$v(P_{t_2})$-path in $G_2/\mathcal{P}_2$, where $v(P_{s_2})$ and $v(P_{t_2})$ refer to the vertices corresponding to the geodesics containing s_2 and t_2 respectively. In the case of parallel composition, let $\mathcal{P}'$ be a partition into geodesics that are also geodesics in G. First assume that $G_1/\mathcal{P}_1$ is a tree, and $G_2/\mathcal{P}_2$ is a tree. The case that $G_1/\mathcal{P}_1$ or $G_2/\mathcal{P}_2$ is not a tree is handled afterwards.

We show that if $G_1/\mathcal{P}_1$ is a tree, and $G_2/\mathcal{P}_2$ is a tree then $G'/\mathcal{P}'$ is also a tree if and only if $l_1 + l_2 = 2$. Else if $G'/\mathcal{P}'$ is not a tree, it results in a tree if we identify the vertices corresponding to P_s and P_t if and only if $l_1 \le 2$ and $l_2 \le 2$.

We prove this by checking all cases.

The first case is that $l_1 = l_2 = 1$. An example for the geodesics in this case is Case 1 in Figure 4.12. Since $l_1 = 1$ it holds that $v(P_{s_1})$ and $v(P_{t_1})$ are adjacent in $G_1/\mathcal{P}_1$. Since $l_2 = 1$ it holds that $v(P_{s_2})$ and $v(P_{t_2})$ are adjacent in $G_2/\mathcal{P}_2$. Thus, $G'/\mathcal{P}'$ is the result of identifying two edges from two trees with each other, which results in a tree.

The second case is that, without loss of generality, $l_1 = 2$ and $l_2 = 0$. In this case, s and t are connected by a geodesic in $\mathcal{P}_2$ and thus also in $\mathcal{P}'$. An example for the geodesics in this case is Case 2 in Figure 4.12. In $G_1/\mathcal{P}_1$ the vertices $v(P_{s_1})$ and $v(P_{t_1})$ have distance 2 and

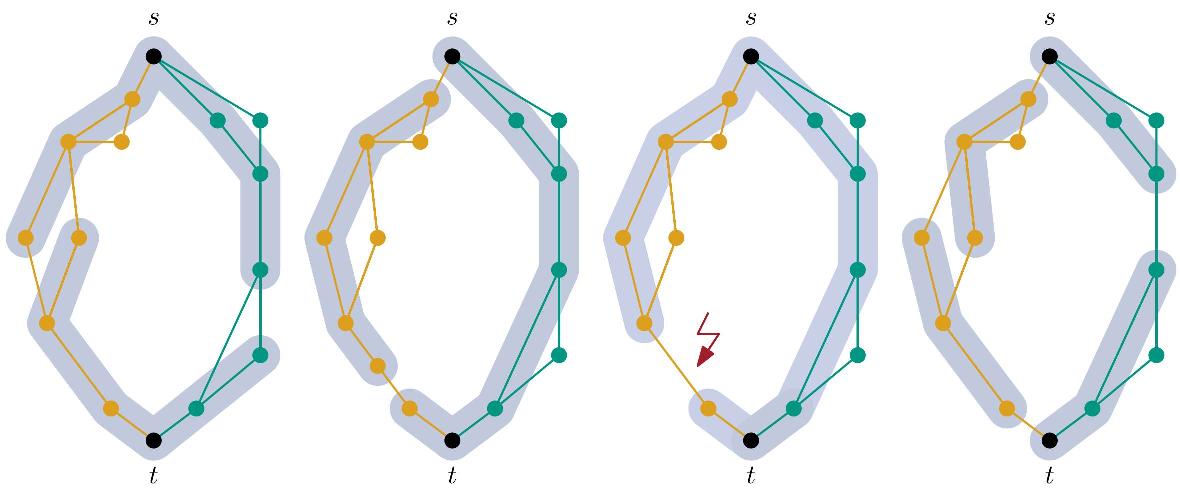


Figure 4.12: Cases to prove that if $\mathcal{P}'$ is a partition into geodesics, $G_1/\mathcal{P}_1$ is a tree and $G_2/\mathcal{P}_2$ is a tree then $G'/\mathcal{P}'$ is also a tree if and only if $l_1 + l_2 = 2$. From left to right: Case 1: $l_1 = l_2 = 1$, Case 2: w.l.o.g. $l_1 = 2$ and $l_2 = 0$, Case 3: $l_1 + l_2 \leq 1$ and Case 4: $l_1 + l_2 \geq 3$

thus identifying them results in a tree. The graph $G'/\mathcal{P}'$ can then be constructed by first identifying the vertices $v(P_{s_1})$ and $v(P_{t_1})$ in $G_1/\mathcal{P}_1$ and then identifying the resulting vertex with the vertex $v(P_{s_2}) = v(P_{t_2})$ in the tree $G_2/\mathcal{P}_2$. Thus $G'/\mathcal{P}'$ is a tree.

The third case is that $l_1 + l_2 \leq 1$. An example for the geodesics in this case is Case 3 in Figure 4.12. In this case, the geodesics adjacent to the sinks and sources are all merged into a single geodesic in $\mathcal{P}'$. This implies that there is a cycle in G' where all vertices are part of a single geodesic, which is a contradiction. Thus, the third case never occurs, since $\mathcal{P}'$ is not a partition into geodesics in G which we detect in a previous step.

The fourth case is that $l_1 + l_2 \geq 2$. An example for the geodesics in this case is Case 4 in Figure 4.12. In this case, in $G'/\mathcal{P}'$ there are two internally disjoint paths between the vertices corresponding to P_s and P_t and thus $G'/\mathcal{P}'$ is not a tree.

In the case that $G'/\mathcal{P}'$ is not a tree, we check if it is a tree if we identify the vertices corresponding to P_s and P_t. This is the case if and only if $l_1 \leq 2$ and $l_2 \leq 2$. An example for the geodesics in this case is shown in Figure 4.13.

For the case that $G_1/\mathcal{P}_1$ is a tree and $G_2/\mathcal{P}_2$ is a tree, we described how to check if $\mathcal{P}'$ is a valid partition and if it is a case 1 or case 2 valid partition. It remains to consider the case that $G_1/\mathcal{P}_1$ or $G_2/\mathcal{P}_2$ is not a tree and thus $\mathcal{P}_1$, $\mathcal{P}_2$ or both are case 2 valid partitions. We show that if s and t are connected by a geodesic in $\mathcal{P}'$, then $\mathcal{P}'$ is a case 1 valid partition of G'. If s and t are not connected by a geodesic in $\mathcal{P}'$, then $\mathcal{P}'$ is a case 2 valid partition of G' if and only if $l_1 \leq 2$ and $l_2 \leq 2$.

First, if s and t are connected by a geodesic in $\mathcal{P}'$, then in at least one of the two subgraphs G_1 or G_2 the sink and source are connected by a geodesic. Without loss of generality let s_2 and t_2 in G_2 be part of the same geodesic in $\mathcal{P}_2$ and therefore $P_{s_2} = P_{t_2}$. Thus identifying $v(P_{s_2})$ with $v(P_{t_2})$ makes no difference and therefore $\mathcal{P}_2$ is a case 1 valid partition and $G_2/\mathcal{P}_2$ is a tree. Since $\mathcal{P}_1$ is a case 2 valid partition identifying $v(P_{s_1})$ and $v(P_{t_1})$ in $G_1/\mathcal{P}_1$ results in a tree. Therefore $G'/\mathcal{P}'$ is a tree since $G'/\mathcal{P}'$ is the result of identifying $v(P_{s_1})$ and $v(P_{t_1})$ in $G_1/\mathcal{P}_1$ and then identifying the resulting vertex with the vertex $v(P_{s_2}) = v(P_{t_2})$ of the tree $G_2/\mathcal{P}_2$. Thus $\mathcal{P}'$ is a case 1 valid partition of G'.

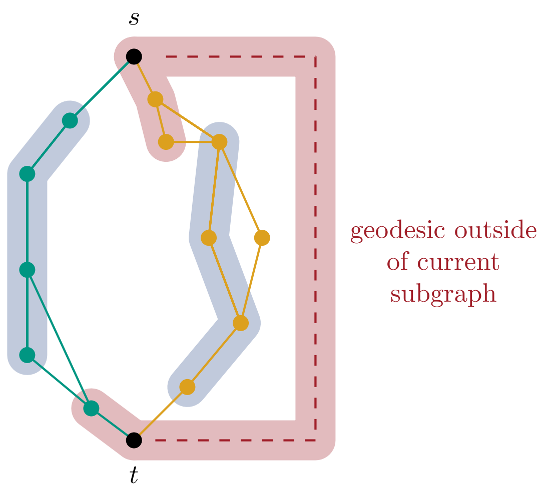


Figure 4.13: Example where $G'/\mathcal{P}'$ is not a tree, however it is a tree if we contract P_s and P_t. This corresponds to s and t being part of the same geodesic outside the current subgraph.

Secondly, assume s and t are not connected by a geodesic in $\mathcal{P}'$ and $l_1 \leq 2$ and $l_2 \leq 2$. Then $G'/\mathcal{P}'$ is still a tree if we identify $v(P'_s)$ and $v(P'_t)$ since both $G_1/\mathcal{P}_1$ and $G_2/\mathcal{P}_2$ result in a tree if we identify $v(P_{s_1})$ with $v(P_{t_1})$ and $v(P_{s_2})$ with $v(P_{t_2})$. If we assume that s and t are not connected by a geodesic in $\mathcal{P}'$ and $l_1 > 2$ or $l_2 > 2$ then $\mathcal{P}'$ is not a valid partition since if we do not identify $v(P'_s)$ with $v(P'_t)$ then $G'/\mathcal{P}'$ contains a cycle in the subgraph with a case 2 valid configuration and if we do identify $v(P'_s)$ with $v(P'_t)$ then $G'/\mathcal{P}'$ contains a cycle in the subgraph G_i where $l_i > 2$. ∎

Combining Claim 7 and Claim 8 gives us the following result:

Claim 9. In the case of parallel composition, we can check if $\mathcal{P}'$ is a *valid* partition and compute the corresponding valid configuration in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Combining the results for both the series composition with Claim 6 and the parallel composition with Claim 9 gives us the following claim:

Claim 10. We can check if $\mathcal{P}'$ is a *valid* partition and compute the corresponding valid configuration in time $O(1)$ using only information stored in the valid configurations corresponding to $\mathcal{P}_1$ and $\mathcal{P}_2$.

Our algorithm checks for all pairs of valid configurations of the two children whether a valid configuration for G' results. Since there are at most $O(n^2)$ valid configurations for each subgraph (Claim 1) and we can check each pair in time $O(1)$ (Claim 10), this requires a time of $O(n^4)$.

The last step is to prove that checking all pairs of valid configurations of the children results in us finding all valid configurations of the current subgraph.

Claim 11. If $\mathcal{P}'$ is a valid partition of G' and G' has two children G_1 and G_2 in D, then there are valid partitions $\mathcal{P}_1$ and $\mathcal{P}_2$ of G_1 and G_2, respectively, such that the combination of $\mathcal{P}_1$ and $\mathcal{P}_2$ results in $\mathcal{P}'$.

Proof. Let $\mathcal{P}'$ be a valid partition of G'. Let P'_s and P'_t be the geodesics in $\mathcal{P}'$ containing the source s and the sink t of G', respectively. Let $v(P'_s)$ and $v(P'_t)$ be the vertices in $G'/\mathcal{P}'$ that correspond to P'_s and P'_t respectively. We define $\mathcal{P}_1$ and $\mathcal{P}_2$ as the partitions of G_1 and G_2 into geodesics that result from restricting $\mathcal{P}'$ to these subgraphs.

Without loss of generality, assume that $\mathcal{P}_1$ is not a valid partition of G_1. Let s_1 and t_1 be the source and sink of G_1 and P_{s_1}, P_{t_1} be the geodesics containing s_1 and t_1 respectively in $\mathcal{P}_1$. Let $v(P_{s_1})$ and $v(P_{t_1})$ be the vertices in $G'/\mathcal{P}'$ that correspond to P_{s_1} and P_{t_1} respectively. We consider the two cases that s_1 and t_1 are connected by a geodesic in $\mathcal{P}'$ or not.

First, let s_1 and t_1 be connected by a geodesic in $\mathcal{P}'$. The graph $G_1/\mathcal{P}_1$, after identifying $v(P_{s_1})$ with $v(P_{t_1})$, contains a cycle C as otherwise P_1 is a case 2 valid partition. Observe that C is also a cycle in $G'/\mathcal{P}'$ after s_1 and t_1 are connected by a geodesic in $\mathcal{P}'$. Thus, in this case $\mathcal{P}'$ is not a case 1 valid partition of G'. It remains to show that identifying $v(P'_s)$ with $v(P'_t)$ in $G'/\mathcal{P}'$ does not result in a tree and thus $\mathcal{P}'$ is also not a case 2 valid partition of G'. Note that the geodesics corresponding to the vertices in C only interact with the rest of the graph G' in s_1 and t_1. Since s_1 and t_1 are connected by a geodesic in $\mathcal{P}'$, the corresponding vertex in C is the only vertex in C that is affected by identifying $v(P'_s)$ with $v(P'_t)$ in $G'/\mathcal{P}'$. Thus the cycle C remains a cycle in $G'/\mathcal{P}'$ after identifying $v(P'_s)$ with $v(P'_t)$ which implies that $\mathcal{P}'$ is also not a case 2 valid partition of G'. This, however, contradicts the assumption that $\mathcal{P}'$ is a valid partition of G'.

Secondly, assume that s_1 and t_1 are not connected by a geodesic in $\mathcal{P}'$. The graph $G_1/\mathcal{P}_1$ contains a cycle C, as otherwise P_1 is a case 1 valid partition. Observe that C is also a cycle in $G'/\mathcal{P}'$ if s_1 and t_1 are not connected by a geodesic in $\mathcal{P}'$. Thus, in this case $\mathcal{P}'$ is not a case 1 valid partition of G'. To reach a contradiction, it remains to show that $\mathcal{P}'$ is also not a case 2 valid partition of G'. Assume that $s_1 \notin P'_s$ or $t_1 \notin P'_t$. In this case identifying $v(P'_s)$ with $v(P'_t)$ does not remove C from $G'/\mathcal{P}'$. Therefore, in this case, $\mathcal{P}'$ is not a case 2 valid partition of G'. Thus we assume that $s_1 \in P'_s$ and $t_1 \in P'_t$. In this case identifying $v(P'_s)$ with $v(P'_t)$ in $G'/\mathcal{P}'$ corresponds to identifying $v(P_{s_1})$ with $v(P_{t_1})$ in $G_1/\mathcal{P}_1$. This, however, also results in a cycle since $\mathcal{P}_1$ is not a case 2 valid partition. Therefore, $\mathcal{P}'$ is also not a case 2 valid partition of G'. Thus, in all cases $\mathcal{P}'$ is not a valid partition of G', which contradicts our assumption. ∎

Claim 11 guarantees that if a valid configuration exists for the current subgraph, then we find it by checking all pairs of valid configurations of the children.

Since the algorithm tests at most $O(n^4)$ combinations for at most $O(n)$ composition steps, the algorithm can find all valid configurations of the graph G in time $O(n^5)$. ∎

We now extend the algorithm to treewidth-2 graphs in general. In the extended algorithm, we consider the *block-cut tree* of the given graph. The block-cut tree is a tree where the vertices are the biconnected components and cut vertices of G. An edge between a biconnected component and a cut vertex exists in the block-cut tree if and only if the cut vertex is part of the biconnected component. For calculating the runtime of the extended algorithm, we also need the following property of block-cut trees:

Lemma 4.5: *The block-cut tree of a graph G has at most $|V(G)| + n_c$ vertices, where n_c is the number of cut vertices in G.*

Proof. We first show that a graph G contains at most n biconnected components. Consider the block-cut tree of G and root it in a biconnected component. Assign each cut vertex to its parent block. Thus, all blocks, besides the blocks that are leaf blocks in the block-cut tree, are assigned at least one cut vertex. The leaf blocks contain at least one non-cut vertex.

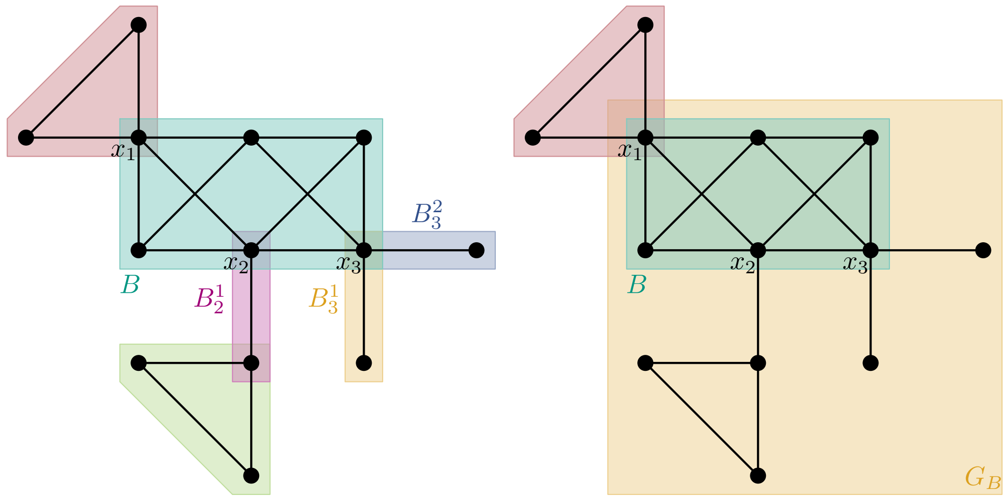


Figure 4.14: On the left: a graph G where the biconnected components are highlighted. Let the central green biconnected component B be the biconnected component that is processed in the current step of the algorithm. The vertices x_1, x_2 and x_3 are the cut vertices of G in B. The vertex x_1 is the parent cut vertex of B. On the right the subgraph G_B that is rooted at B is shown.

Thus, there are at most n biconnected components. Since vertices in the block-cut tree are either cut vertices or biconnected components of G it holds that the block-cut tree has at most $|V(G)| + n_c$ vertices. ∎

Theorem 1.9: *The geodesic treewidth of treewidth-2 graphs can be determined in time $O(n^5)$.*

Proof. We first describe how the algorithm works and then prove that it is correct.

Given a treewidth-2 graph G, we begin by calculating the block-cut tree of G in linear time [38]. Theorem 4.3 says that a graph G has treewidth at most 2, if and only if every biconnected component of G is a series-parallel graph. Thus, the biconnected components in the block-cut tree of G are series-parallel graphs.

Our algorithm processes the block-cut tree in bottom-up order. Let B be the biconnected component that is processed in the current step. An example of a graph and a current biconnected component is shown in Figure 4.14. We define the subgraph G_B of G that is rooted at B as follows: G_B consists of all biconnected components of G for which B is on the path to the root of the block-cut tree. Let $x_1, x_2, \dots$ be the cut vertices of G that are in B. Let x_1 be the parent cut vertex of B and all other cut vertices are children of B. For $i \geq 2$ and a child cut vertex x_i, let $B_i^1, B_i^2, \dots$ be the adjacent biconnected components in the block-cut tree that are not B. Let the subgraphs $G_{B_i^1}, G_{B_i^2}, \dots$ be the subgraphs of G that are rooted at $B_i^1, B_i^2, \dots$ respectively for $i \geq 2$. The subgraphs $G_{B_i^1}, G_{B_i^2}, \dots$ were processed before the current step since we are working in bottom-up order.

For each subgraph G_B rooted at a biconnected component B, that we have solved, we store the following information:

- whether or not there exists a partition $\mathcal{P}_B$ of this subgraph G_B into geodesics such that $G_B/\mathcal{P}_B$ is a tree
- across all such possible partitions $\mathcal{P}_B$ we store the minimum number of neighbours n_B of x_1 that are in the same geodesic as x_1 in $\mathcal{P}_B$ where x_1 is the parent cut vertex of B

We call a partition $\mathcal{P}_B$ of a subgraph G_B, where the number of neighbours of x_1 that are in the same geodesic as x_1 in $\mathcal{P}_B$ is exactly n_B, a minimum degree partition of G_B.

Since the biconnected components in the block-cut tree of G are series-parallel graphs [37], the graph B is a series-parallel graph. We apply the algorithm from Lemma 4.4 to B with some slight alterations. Let $\mathcal{P}$ be a valid partition that the algorithm considers in some step. We additionally require that for all $i \geq 2$ the number of neighbours of a cut vertex x_i that are in the same geodesic as x_i in $\mathcal{P}$ plus $n_{B_i^1} + n_{B_i^2} + \dots$ is at most 2. Only if these degree restrictions are fulfilled, $\mathcal{P}$ combined with minimum degree partitions of $G_{B_i^1}, G_{B_i^2}, \dots$ for $i \geq 2$ results in a partition into geodesics. Lastly, we store the number of neighbours of x_1 that are in the same geodesic as x_1 in $\mathcal{P}$ as part of the valid configuration as well. This allows us, if the algorithm succeeds, to determine the minimum number of neighbours of x_1 that are in the same geodesic as x_1 such that a case 1 valid partition of G_B exists. This minimum number is then stored as the value n_B for the current solved subgraph.

If no case 1 valid partition of B exists that fulfils the additional degree restrictions, then the algorithm fails and determines that G has geodesics treewidth 2.

Else if we have solved the last biconnected component in the block-cut tree using this altered algorithm and found a valid case 1 configuration at the root, then we say G has geodesic treewidth 1.

We now prove that the algorithm for treewidth-2 graphs is correct.

We consider a step in the algorithm at a biconnected component B that solves the subgraph G_B. Let the cut vertices and relevant other subgraphs be defined as described in the algorithm.

We first show that a case 1 valid partition $\mathcal{P}$ of B, combined with minimum degree partitions $\mathcal{P}_{B_i^1}, \mathcal{P}_{B_i^2}, \dots$ of $G_{B_i^1}, G_{B_i^2}, \dots$ for $i \geq 2$, results in a case 1 valid partition of G_B if for all $i \geq 2$ the number of neighbours of a cut vertex x_i, that are in the same geodesic as x_i in $\mathcal{P}$, plus $n_{B_i^1} + n_{B_i^2} + \dots$ is at most 2. An example of a case 1 valid partition $\mathcal{P}$ of B that gets combined with case 1 valid partitions of the subgraphs, that correspond to the biconnected components connected to B at child cut vertices, is shown in Figure 4.15.

Let $\mathcal{P}_B$ be the partition resulting from combining the partition $\mathcal{P}$ of B with minimum degree partitions of $G_{B_i^1}, G_{B_i^2}, \dots$ for $i \geq 2$. If for all $i \geq 2$ the number of neighbours of a cut vertex x_i that are in the same geodesic as x_i in $\mathcal{P}$ plus $n_{B_i^1} + n_{B_i^2} + \dots$ is at most 2 then $\mathcal{P}_B$ is a partition of G_B into paths. The paths in $\mathcal{P}_B$ are geodesics when restricted to biconnected components. However, any shortcut inside such a path would have to be inside a biconnected component, which is not possible. Thus, $\mathcal{P}_B$ is a partition of G_B into geodesics. Additionally, the graph $G_B/\mathcal{P}_B$ is a tree since it is the result of identifying the tree $B/\mathcal{P}$ with the trees $G_{B_i^1}/\mathcal{P}_{B_i^1}, G_{B_i^2}/\mathcal{P}_{B_i^2}, \dots$ at a single vertex each. Thus, $\mathcal{P}_B$ is a case 1 valid partition of G_B.

We next show that if there exists a case 1 valid partition of G_B then there exists a case 1 valid partition of B that fulfils the additional degree restrictions.

Now assume that there exists a case 1 valid partition $\mathcal{P}_B$ of G_B. We show that in this case the algorithm finds a case 1 valid partition of B where the number of neighbours of x_i that are in the same geodesic as x_i in $\mathcal{P}$ plus $n_{B_i^1} + n_{B_i^2} + \dots$ is greater than 2 for every $i \geq 2$. For $i \geq 2$ and $G_{B_i^1}, G_{B_i^2}, \dots$ let $\mathcal{P}_{B_i^1}, \mathcal{P}_{B_i^2}, \dots$ be $\mathcal{P}_B$ restricted to the corresponding subgraphs. Note that $\mathcal{P}_{B_i^1}, \mathcal{P}_{B_i^2}, \dots$ are case 1 valid partitions for the subgraphs $G_{B_i^1}, G_{B_i^2}, \dots$. Thus, for all j and $i \geq 2$ it holds that $n_{B_i^j}$ is at most the number of neighbours of x_i that are in the same geodesic as x_i in $\mathcal{P}_{B_i^j}$. Let $\mathcal{P}$ be the result of restricting $\mathcal{P}_B$ to B. Since $\mathcal{P}_B$ is a partition into geodesics it follows that for $i \geq 2$ the number of neighbours of x_i that are in the same geodesic as x_i in $\mathcal{P}$ combined with $n_{B_i^1} + n_{B_i^2} + \dots$ is at most 2. This implies that $\mathcal{P}$ is a case 1 valid partition for B which fulfils all additional degree restrictions.

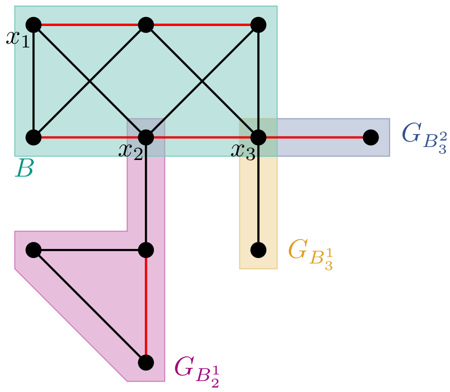


Figure 4.15: A subgraph G_B of G that is rooted at a biconnected component B. The vertices x_1, x_2 and x_3 are the cut vertices of G in B. The vertex x_1 is the parent cut vertex of B. For B and for the subgraphs $G_{B_i^1}, G_{B_i^2}, \ldots$ corresponding to the biconnected components connected to B at child cut vertices, some case 1 valid partitions are given. The geodesics in these partitions are highlighted in red. These partitions combine into a case 1 valid partition of G_B even though the partition of $G_{B_3^2}$ is not a degree minimum partition. Note that this partition of $G_{B_3^2}$ can be replaced by a degree minimum partition of $G_{B_3^2}$ and the combined partition of G_B remains a case 1 valid partition into geodesics. The number of neighbours of x_1 that are in the same geodesic as x_1 is 1. If there are no case 1 valid partitions of G_B where this value is smaller then it holds that $n_B = 1$.

This proves that the algorithm is correct. In the altered algorithm for series-parallel graphs that we apply to a biconnected component B, we need to check for the cut vertices $x_2, x_3, \ldots$ that the number of neighbours of x_i that are in the same geodesic as x_i combined with $n_{B_i^1} + n_{B_i^2} + \ldots$ is at most 2. However, we only need to perform this check if a child cut vertex x_i is the current sink or source, for which we already store the needed values in the corresponding valid configuration. For internal vertices, this check was passed at an earlier point and the number of neighbours that are in the same geodesic does not increase afterwards. Thus, this check, combined with storing the number of neighbours of the parent cut vertex x_1 that are in the same geodesic as x_1, does not increase the runtime significantly. A single biconnected component is still solved in time $O(n^5)$.

We claim that applying the algorithm bottom-up to all biconnected components of G is still in $O(n^5)$. For this purpose we first estimate the sum $S_V = |B_1| + |B_2| + \ldots$ of the sizes of all biconnected components $B_1, B_2, \ldots$ of G. Each vertex that is not a cut vertex contributes at most 1 to this sum. The cut vertices contribute as much to this sum as the number of edges in the block-cut tree. Lemma 4.5 gives us that the block-cut tree has at most $n + n_c$ vertices and thus also at most $n + n_c$ edges, where $n = |V(G)|$ and n_c is the number of cut vertices in G (Lemma 4.5). Thus S_V is at most $2|V(G)|$. Since it holds that $\sum_i |V(B_i)| \leq S_V$ it follows that $\sum_i O(|V(B_i)|)^5 \leq O(S_V)^5 \leq O(2V(G))^5 = O(n^5)$. ■

5 Baker Treewidth vs. Linear Local Treewidth

In this chapter, we investigate the relationships between bounded layered treewidth, linear local treewidth, bounded local treewidth and bounded Baker treewidth. It is known that bounded layered treewidth is a necessary condition for product structure [1]. Dujmović, Morin, and Wood show that bounded layered treewidth implies linear local treewidth [24], and by definition, linear local treewidth implies bounded local treewidth. Thus, linear local treewidth and bounded local treewidth are necessary conditions for product structure as well. In Section 4.1 we show that linear local treewidth and thus bounded local treewidth are necessary conditions for geodesic structure, too.

We begin by proving that bounded layered treewidth implies bounded Baker treewidth. This also implies that bounded Baker treewidth is a necessary condition for product structure. The proof is very similar to the proof that bounded layered treewidth implies linear local treewidth by Dujmović, Morin, and Wood [24].

Theorem 1.11: *If a graph class has bounded layered treewidth, then it has bounded Baker treewidth.*

Proof. Let $\mathcal{G}$ be a graph class where every graph $G \in \mathcal{G}$ has layered treewidth at most k. Thus, there exists a layering $\mathcal{L}$ of G and a tree decomposition $\mathcal{T}$ of G such that the intersection between any layer $L \in \mathcal{L}$ and any bag of $\mathcal{T}$ has size at most k. Let G' be a graph induced by l consecutive layers in the layering $\mathcal{L}$ of G. By removing the vertices not in G' from $\mathcal{T}$, we get a tree decomposition of G' where each bag has size at most $l \cdot k$. Therefore, $\mathcal{G}$ has bounded Baker treewidth with function $f(l) = l \cdot k$. ∎

Next, we show that bounded Baker treewidth implies bounded local treewidth.

Theorem 1.12: *For a graph class with bounded Baker treewidth with function $f(l)$, the local treewidth is bounded by $f(2k+1)$.*

Proof. Let $\mathcal{G}$ be a graph class with bounded Baker treewidth with function $f(l)$. Thus, for $G \in \mathcal{G}$ there exists a layering $\mathcal{L}$ such that the subgraph induced by l consecutive layers of $\mathcal{L}$ has treewidth at most $f(l)$. Let v be a vertex in G and G' be the subgraph induced by the k-th neighbourhood of v. It holds that G' is contained in at most $2k+1$ consecutive layers of $\mathcal{L}$ thus, the treewidth of G' is at most $f(2k+1)$. ∎

It is known that bounded local treewidth does not imply linear local treewidth. An example for this is the 3D grid, which has quadratic local treewidth but no linear local treewidth. We observe that bounded local treewidth also does not imply bounded Baker treewidth.

Consider the graph U_n obtained from a $n \times n \times n$ grid by adding all diagonals to its unit subcubes. Berger, Dvořák, and Norin [33] show that for every k and any partition A_1, A_2 of the vertex set of U_n, either A_1 or A_2 induces a subgraph of treewidth at least k if n is large enough. Dvořák [34] considers the partition of U_n into odd and even numbered layers and

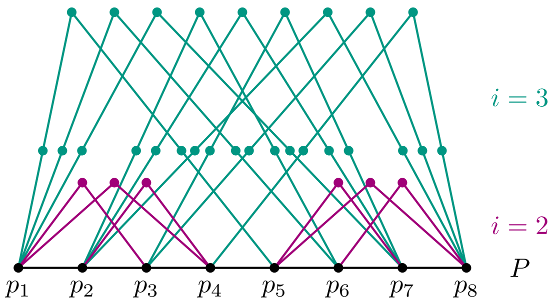


Figure 5.1: The graph G_3. In purple all disjoint paths of length 2 between non-adjacent vertices p_u, p_v where $\lfloor u/(2^2+1)\rfloor = \lfloor v/(2^2+1)\rfloor$. In green all disjoint paths of length 4 between non-adjacent vertices p_u, p_v where $\lfloor u/(2^3+1)\rfloor = \lfloor v/(2^3+1)\rfloor$.

observes that the result of Berger, Dvořák, and Norin thus prevents the existence of bounded treewidth layerings of U_n. Since the layering considered by Baker treewidth is a bounded treewidth layering, where the graph induced by a layer L has treewidth at most $f(1)$, it thus follows that U_n does not have bounded Baker treewidth.

Lemma 5.1 (Berger, Dvořák, and Norin [33] and Dvořák [34])**:** *The graphs U_n do not have bounded Baker treewidth.*

We observe that the graphs U_n have quadratic local treewidth and thus bounded local treewidth. Together with Lemma 5.1 this proves that bounded local treewidth does not imply bounded Baker treewidth.

The relations of bounded Baker treewidth to bounded local treewidth and bounded layered treewidth are quite similar to the known relations of linear local treewidth to bounded local treewidth and bounded layered treewidth. Thus, the last question we investigate in this section is how linear local treewidth relates to bounded Baker treewidth.

Theorem 1.13: *There are graph classes with bounded Baker treewidth but without linear local treewidth.*

Proof. We construct a graph class $\mathcal{G}$ that has bounded Baker treewidth but no linear local treewidth. For every $h \in \mathbb{N}$ we begin the construction of $G_h \in \mathcal{G}$ with a path $P = p_1, \ldots, p_{2^h}$ consisting of 2^h vertices. For any non-adjacent vertices p_u, p_v on this path let i be the smallest integer such that $\lfloor u/(2^i+1)\rfloor = \lfloor v/(2^i+1)\rfloor$. Connect p_u to p_v by a disjoint path of length $2(i-1)$. An example of a graph G_3 is shown in Figure 5.1. This results in all pairs of vertices of P being connected by independent paths. However, the size of the resulting subdivided cliques is small if we disconnect the longer independent paths.

The graph G_h has diameter at most $2h$. However, it contains a subdivided clique of size 2^h and thus has treewidth at least 2^h. Therefore, $\mathcal{G}$ does not have linear local treewidth.

We show that $\mathcal{G}$ has bounded Baker treewidth with function $f(l) = 2^l$. For each $G_h \in \mathcal{G}$ we construct a layering $\mathcal{L}$ as follows. A vertex $v \in G$ is assigned to layer L_{d_v} in the layering where d_v is the shortest distance from v to the path P. An example of this layering is shown in Figure 5.2.

We consider a graph induced by l consecutive layers in $\mathcal{L}$. If the consecutive layers do not include L_0, then the induced graph is a disjoint union of paths and thus has treewidth 1. If the consecutive layers include L_0, then we can split the vertices of the path P into 2^{h-l} sets consisting of 2^l consecutive vertices. Between two different sets, there are no connecting

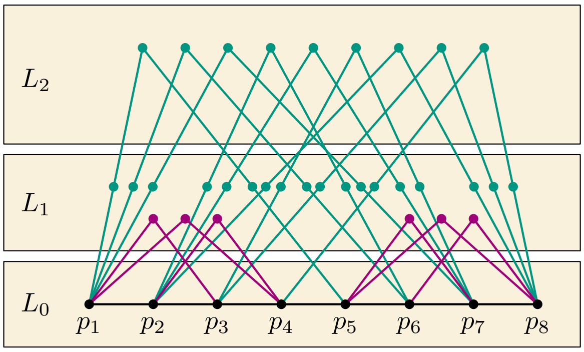


Figure 5.2: The graph G_3 with layering $\mathcal{L}$ where the layer of a vertex is determined by its distance from P.

disjoint paths besides P since the longer disjoint paths have vertices outside the l consecutive layers. A single set of path vertices, together with all vertices reachable using no path vertices from outside the set, is a subdivided clique of size 2^l with some more disjoint paths connected only at one end to the clique. Thus, the graph induced by l consecutive layers has treewidth at most 2^l and thus $\mathcal{G}$ has bounded Baker treewidth. ■

Since bounded layered treewidth implies linear local treewidth [24] and bounded Baker treewidth does not imply linear local treewidth, we get the following corollary.

Corollary 1.14: *There are graph classes with bounded Baker treewidth but unbounded layered treewidth.*

We have shown that bounded Baker treewidth does not imply linear local treewidth, however, it is still possible that linear local treewidth implies bounded Baker treewidth.

6 Conclusion

In this thesis, we compared product structure, geodesic structure and BFS structure. We also investigated the relationships between bounded layered treewidth, Baker treewidth, linear local treewidth and bounded local treewidth. We also translated many results known only for product structure to geodesic structure or showed that they do not hold. However, there are some relationships between these variants of product structure and related concepts that are still unknown. An overview can be seen in Figure 6.1. We give some detailed questions that could be the basis of future work:

Question 6.1: *Does geodesic structure imply product structure?*

We only answered this question for graph classes with geodesic treewidth 1. For these, we showed that they also admit product structure. In general, it is unknown if geodesic structure implies product structure. This might also be due to the lack of graph classes for which we know that they admit geodesic structure. As a first step, we could observe graph classes for which the geodesic treewidth is significantly smaller than the row treewidth. However, we only found some ideas for graph classes where the row treewidth is twice as high as the geodesic treewidth, which is not sufficient for separating geodesic structure from product structure.

Question 6.2: *Does geodesic structure imply bounded layered treewidth?*

If Question 6.1 could be answered positively, then Question 6.2 is also true. Reversely, if geodesic structure does not imply bounded layered treewidth, then geodesic structure also does not imply product structure. Thus, answering Question 6.2 would either solve Question 6.1 or at least be a step in the right direction.

Question 6.3: *Does geodesic structure imply bounded Baker treewidth?*

This is another open question that we have not considered in this work. If Question 6.2 holds, then this question also holds and geodesic structure implies bounded Baker treewidth. However, it may be easier to answer this question first.

Question 6.4: *Does linear local treewidth imply bounded layered treewidth?*

It is known that bounded layered treewidth implies linear local treewidth [24]. However, it is currently not known if the reverse also holds. If linear local treewidth also implies bounded layered treewidth, then the two concepts would be one and the same, which would be a very surprising result. This open question also relates to the next open question:

Question 6.5: *Does linear local treewidth imply bounded Baker treewidth?*

We showed that the reverse does not hold. Note that if Question 6.4 is true, then Question 6.5 is also true since we have shown that bounded layered treewidth implies bounded baker treewidth.

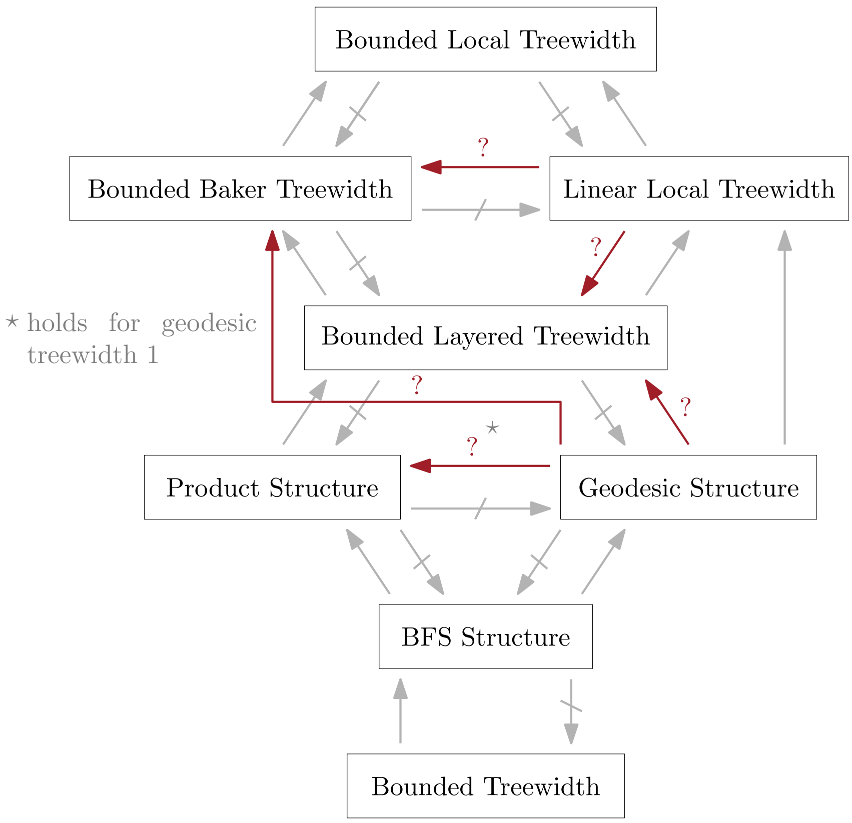


Figure 6.1: The relationships between variants of product structure and related concepts that are still open are drawn in red.

An interesting observation is that all these open questions have a common structure. Geodesic structure and linear local treewidth are defined without connections to any layering of the graphs, while product structure, bounded layered treewidth and Baker treewidth all use layerings. Thus, to answer the open questions, we are always required to either find a convenient layering or to show that there exists no such convenient layering, which has turned out to be quite difficult.

Another interesting area of research is to investigate how the relationships of the variants of product structure change if we only consider proper minor-closed graphs.

Question 6.6: *What are the relationships between the variants of product structure for minor closed graph classes?*

This question is interesting because for proper minor-closed graph classes it is known that product structure, bounded layered treewidth, linear local treewidth and bounded local treewidth are all equivalent [1 , 39]. Since geodesic structure implies linear local treewidth, it follows that geodesic structure implies product structure for proper minor-closed graph classes. This answers Question 6.1 positively for proper minor-closed graph classes. However, the reverse question is still open for proper minor-closed graph classes.

Lastly, we could continue investigating geodesic structure and check if more known results for product structure also hold for geodesic structure. For example, we could investigate whether or not bounded degree planar graphs have geodesic structure, where the quotients also have bounded degree. For product structure, it is known that this is not the case [40].

Question 6.7: *Do bounded-degree planar graphs have geodesic structure where the quotients also have bounded degree?*

Additionally, the algorithm from Section 4.3 for computing the geodesic treewidth of treewidth-2 graphs could be improved in runtime. Our algorithm only aims to show that the problem is solvable in polynomial time. The question if the problem is solvable in linear time could be of separate interest.

However, further research into geodesic structure is hard to justify with the current limited known applications for it. In some cases, we think that the bounds for parameters shown for graph classes with product structure could be translated for graph classes with geodesic structure. However, to justify using geodesics structure we would either have to find graph classes that have geodesic structure and no product structure (Question 6.1) or show that geodesic structure results in better bounds for some parameters:

Question 6.8: *Are there applications where geodesic structure performs better than product structure?*

Bibliography

[1] Vida Dujmović, Gwenaël Joret, Piotr Micek, Pat Morin, Torsten Ueckerdt, and David R. Wood. “Planar Graphs Have Bounded Queue-Number”. In: *J. ACM* Volume 67 (Aug. 2020). DOI: *10.1145/3385731*.

[2] Torsten Ueckerdt, David R. Wood, and Wendy Yi. “An Improved Planar Graph Product Structure Theorem”. In: *The Electronic Journal of Combinatorics* (2022), P2–51. DOI: *10.37236/10614*.

[3] Michał Pilipczuk and Sebastian Siebertz. “Polynomial bounds for centered colorings on proper minor-closed graph classes”. In: *Journal of Combinatorial Theory, Series B* Volume 151 (2021), pp. 111–147. ISSN: 0095-8956. DOI: *10.1016/j.jctb.2021.06.002*.

[4] Vida Dujmović, Pat Morin, and David R. Wood. “Graph product structure for non-minor-closed classes”. In: *Journal of Combinatorial Theory, Series B* Volume 162 (2023), pp. 34–67. ISSN: 0095-8956. DOI: *10.1016/j.jctb.2023.03.004*.

[5] Michael A. Bekos, Giordano Da Lozzo, Petr Hliněný, and Michael Kaufmann. “Graph Product Structure for h-Framed Graphs”. In: *The Electronic Journal of Combinatorics* (2024), P4–55. DOI: *10.37236/12123*.

[6] Marc Distel, Robert Hickingbotham, Michał T. Seweryn, and David R. Wood. “Powers of planar graphs, product structure, and blocking partitions”. In: *Innovations in Graph Theory* Volume 1 (2024), pp. 39–86. DOI: *10.5802/igt.4*.

[7] Robert Hickingbotham and David R. Wood. “Shallow Minors, Graph Products, and Beyond-Planar Graphs”. In: *SIAM Journal on Discrete Mathematics* Volume 38 (2024), pp. 1057–1089. DOI: *10.1137/22M1540296*.

[8] Kevin Hendrey, Nikolai Karol, and David R. Wood. *Structure of k-Matching-Planar Graphs*. 2025. DOI: *10.48550/arXiv.2507.22395*.

[9] Marthe Bonamy, Cyril Gavoille, and Michał Pilipczuk. “Shorter Labeling Schemes for Planar Graphs”. In: *SIAM Journal on Discrete Mathematics* Volume 36 (Sept. 2022), pp. 2082–2099. DOI: *10.1137/20M1330464*.

[10] Vida Dujmović, Louis Esperet, Cyril Gavoille, Gwenaël Joret, Piotr Micek, and Pat Morin. “Adjacency Labelling for Planar Graphs (and Beyond)”. In: *J. ACM* Volume 68 (Oct. 2021). DOI: *10.1145/3477542*.

[11] Louis Esperet, Gwenaël Joret, and Pat Morin. “Sparse universal graphs for planarity”. In: *Journal of the London Mathematical Society* Volume 108 (2023), pp. 1333–1357. DOI: *10.1112/jlms.12781*.

[12] Vida Dujmović, Louis Esperet, Gwenaël Joret, Bartosz Walczak, and David R. Wood. “Planar graphs have bounded nonrepetitive chromatic number”. In: *Advances in Combinatorics* (Mar. 2020). DOI: *10.19086/aic.12100*.

[13] Michał Dębski, Piotr Micek, Felix Schröder, and Stefan Felsner. “Improved Bounds for Centered Colorings”. In: *Advances in Combinatorics* (Aug. 2021). DOI: *10.19086/aic.27351*.

[14] 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)*. 2023, pp. 1921–1930. DOI: *10.1109/FOCS57990.2023.00116*.

[15] Vida Dujmović, Louis Esperet, Pat Morin, Bartosz Walczak, and David R. Wood. “Clustered 3-colouring graphs of bounded degree”. In: *Combinatorics, Probability and Computing* Volume 31 (2022), pp. 123–135. DOI: *10.1017/S0963548321000213*.

[16] Prosenjit Bose, Vida Dujmović, Mehrnoosh Javarsineh, and Pat Morin. *Asymptotically Optimal Vertex Ranking of Planar Graphs*. 2020. DOI: *10.48550/ARXIV.2007.06455*.

[17] Édouard Bonnet, O-joung Kwon, and David R. Wood. *Reduced bandwidth: a qualitative strengthening of twin-width in minor-closed classes (and beyond)*. 2022. DOI: *10.48550/ARXIV.2202.11858*.

[18] Zdeněk Dvořák, Daniel Gonçalves, Abhiruk Lahiri, Jane Tan, and Torsten Ueckerdt. “On Comparable Box Dimension”. In: *38th International Symposium on Computational Geometry (SoCG 2022)*. Edited by Xavier Goaoc and Michael Kerber. Vol. 224. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, 38:1–38:14. DOI: *10.4230/LIPIcs.SoCG.2022.38*.

[19] Gwenaël Joret and Clément Rambaud. “Neighborhood complexity of planar graphs”. In: *Combinatorica* Volume 44 (2024), pp. 1115–1148. DOI: *10.1007/s00493-024-00110-6*.

[20] Daniel Kráľ, Kristýna Pekárková, and Kenny Štorgel. “Twin-Width of Graphs on Surfaces”. In: *49th International Symposium on Mathematical Foundations of Computer Science (MFCS 2024)*. Edited by Rastislav Královič and Antonín Kučera. Vol. 306. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 66:1–66:15. ISBN: 978-3-95977-335-5. DOI: *10.4230/LIPIcs.MFCS.2024.66*.

[21] Vida Dujmović, Pat Morin, and Saeed Odak. *Odd Colourings of Graph Products*. 2022. DOI: *10.48550/ARXIV.2202.12882*.

[22] 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*.

[23] Therese Biedl, David Eppstein, and Torsten Ueckerdt. “On the Complexity of Embedding in Graph Products”. In: *Computing in Geometry and Topology* Volume 4 (May 2025), 5:1–5:18. DOI: *10.57717/cgt.v4i2.56*.

[24] Vida Dujmović, Pat Morin, and David R. Wood. “Layered separators in minor-closed graph classes with applications”. In: *Journal of Combinatorial Theory, Series B* Volume 127 (2017), pp. 111–147. DOI: *10.1016/j.jctb.2017.05.006*.

[25] Laura Merker, Lena Scherzer, Samuel Schneider, and Torsten Ueckerdt. “Intersection Graphs with and Without Product Structure”. In: *32nd International Symposium on Graph Drawing and Network Visualization (GD 2024)*. Edited by Stefan Felsner and Karsten Klein. Vol. 320. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 23:1–23:19. ISBN: 978-3-95977-343-0. DOI: *10.4230/LIPIcs.GD.2024.23*.

[26] Prosenjit Bose, Vida Dujmović, Mehrnoosh Javarsineh, Pat Morin, and David R. Wood. “Separating layered treewidth and row treewidth”. In: *Discrete Mathematics & Theoretical Computer Science* Volume vol. 24, no. 1 (May 2022). DOI: *10.46298/dmtcs.7458*.

[27] Neil Robertson and Paul D. Seymour. "Graph Minors. XVI. Excluding a non-planar graph". In: *Journal of Combinatorial Theory, Series B* Volume 89 (2003), pp. 43–76. ISSN: 0095-8956. DOI: *10.1016/S0095-8956(03)00042-X*.

[28] Brenda S. Baker. "Approximation algorithms for NP-complete problems on planar graphs". In: *J. ACM* Volume 41 (Jan. 1994), pp. 153–180. ISSN: 0004-5411. DOI: *10.1145/174644.174650*.

[29] David Eppstein. "Subgraph Isomorphism in Planar Graphs and Related Problems". In: *Journal of Graph Algorithms and Applications* Volume 3 (1999), pp. 1–27. DOI: *10.7155/jgaa.00014*.

[30] Alexander Grigoriev and Hans L. Bodlaender. "Algorithms for graphs embeddable with few crossings per edge". In: *Algorithmica* Volume 49 (2007), pp. 1–11. DOI: *10.1007/s00453-007-0010-x*.

[31] David Eppstein. "Diameter and treewidth in minor-closed graph families". In: *Algorithmica* Volume 27 (2000), pp. 275–291. DOI: *10.1007/s004530010020*.

[32] Robert Hickingbotham. *Induced Minors, Asymptotic Dimension, and Baker's Technique.* 2025. DOI: *10.48550/arXiv.2508.06190*.

[33] Eli Berger, Zdeněk Dvořák, and Sergey Norin. "Treewidth of Grid Subsets". In: *Combinatorica* Volume 38 (2018), pp. 1337–1352. DOI: *10.1007/s00493-017-3538-5*.

[34] Zdeněk Dvořák. "Thin graph classes and polynomial-time approximation schemes". In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms.* New Orleans, Louisiana: Society for Industrial and Applied Mathematics, 2018, pp. 1685–1701. ISBN: 9781611975031. DOI: *10.1137/1.9781611975031.110*.

[35] Craig A. Tovey. "A simplified NP-complete satisfiability problem". In: *Discrete Applied Mathematics* Volume 8 (1984), pp. 85–89. ISSN: 0166-218X. DOI: *10.1016/0166-218X(84)90081-7*.

[36] Jacobo Valdes, Robert E. Tarjan, and Eugene L. Lawler. "The Recognition of Series Parallel Digraphs". In: *SIAM Journal on Computing* Volume 11 (1982), pp. 298–313. DOI: *10.1137/0211023*.

[37] Hans L. Bodlaender. "A partial k-arboretum of graphs with bounded treewidth". In: *Theoretical Computer Science* Volume 209 (1998), pp. 1–45. ISSN: 0304-3975. DOI: *10.1016/S0304-3975(97)00228-4*.

[38] John Hopcroft and Robert Tarjan. "Algorithm 447: efficient algorithms for graph manipulation". In: *Commun. ACM* Volume 16 (June 1973), pp. 372–378. ISSN: 0001-0782. DOI: *10.1145/362248.362272*.

[39] Erik D. Demaine and MohammadTaghi Hajiaghayi. "Equivalence of Local Treewidth and Linear Local Treewidth and Its Algorithmic Applications". In: *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms.* New Orleans, Louisiana: Society for Industrial and Applied Mathematics, 2004, pp. 840–849. URL: *https://dl.acm.org/doi/abs/10.5555/982792.982919*.

[40] Vida Dujmović, Gwenaël Joret, Piotr Micek, Pat Morin, and David R. Wood. "Bounded-degree planar graphs do not have bounded-degree product structure". In: *The Electronic Journal of Combinatorics* (2024), P2.51. DOI: *10.37236/11712*.