Polynomial Kernels for Strictly Chordal Edge Modification Problems

Maël Dumas
Univ. Orléans, INSA Centre Val de Loire, LIFO EA 4022, F-45067 Orléans, France

Anthony Perez
Univ. Orléans, INSA Centre Val de Loire, LIFO EA 4022, F-45067 Orléans, France

Ioan Todinca
Univ. Orléans, INSA Centre Val de Loire, LIFO EA 4022, F-45067 Orléans, France

Abstract

We consider the Strictly Chordal Editing problem, where one is given an undirected graph $G = (V, E)$ and a parameter $k \in \mathbb{N}$ and seeks to *edit* (add or delete) at most k edges from G to obtain a strictly chordal graph. Problems Strictly Chordal Completion and Strictly Chordal Deletion are defined similarly, by only allowing edge additions for the former, and only edge deletions for the latter. Strictly chordal graphs are a generalization of 3-leaf power graphs and a subclass of 4-leaf power graphs. They can be defined, e.g., as dart and gem-free chordal graphs. We prove the NP-completeness for all three variants of the problem and provide an $O(k^3)$ vertex-kernel for the completion version and $O(k^4)$ vertex-kernels for the two others.

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases Parameterized complexity, kernelization algorithms, graph modification, strictly chordal graphs

Digital Object Identifier 10.4230/LIPIcs.IPEC.2021.17

1 Introduction

Parameterized algorithms are among the most natural approaches to tackle NP-hard optimization problems [13]. In particular, they have been very successful in dealing with so-called edge modification problems on graphs: given as input an arbitrary graph $G = (V, E)$ and a parameter $k \in \mathbb{N}$, the goal is to transform G into a graph with some specific properties (i.e., belonging to a specific graph class $\mathcal{G}$) by adding and/or deleting at most k edges. Parameterized algorithms (also called FPT for *fixed parameter tractable*) aim at a time complexity of type $f(k) \cdot n^{O(1)}$, where f is some computable function, hence the combinatorial explosion is restricted to parameter k.

When the target class $\mathcal{G}$ is characterized by a finite family of forbidden induced subgraphs, modification problems are FPT by a result of Cai [8]. Indeed, as long as our graph contains one of the forbidden subgraphs, we can try each possibility to correct this obstruction and branch by recursive calls. On each branch, the budget k is strictly diminished, therefore the whole algorithm has a number of calls bounded by some function $f(k)$. The situation is more complicated when the target class $\mathcal{G}$ is characterized by an infinite family of forbidden induced subgraphs. Nonetheless, a large literature is devoted to edge modification problems towards chordal graphs (where we forbid all induced cycles with at least four vertices) as well as sub-classes of chordal graphs, typically obtained by requiring some fixed set of obstructions, besides the long cycles. Observe that, in this case, the situation remains relatively simple if we restrict ourselves to *edge completion* problems, where we are only allowed to *add* edges to the input graphs. Indeed, in this case, if a graph has a cycle of length longer than $k + 3$, it cannot be made chordal by adding at most k edges. Therefore we can use again the

16th International Symposium on Parameterized and Exact Computation (IPEC 2021).
Editors: Petr A. Golovach and Meirav Zehavi; Article No. 17; pp. 17:1–17:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

approach of Cai to deal with cycles of length at most $k+3$ and other obstructions, and either the algorithm finds a solution in $f(k)$ recursive calls, or we can conclude that we have a no-instance. The cases of *edge deletion* problems (where we are only allowed to remove edges) and *edge editing* problems (where we are allowed to both remove edges or add missing edges) are more complicated, since even long cycles can be eliminated by a single edge removal. Therefore more efforts and more sophisticated techniques were necessary in these situations, but several such problems turned out to be FPT [10, 14, 15]. The interested reader can refer to [11] for a broad and comprehensive survey on parameterized algorithms for edge modification problems.

We focus here on a sub-family of parameterized algorithms, namely on *kernelization*. The goal of kernelization is to provide a polynomial algorithm transforming any instance (I, k) of the problem into an equivalent instance (I', k') where k' is upper bounded by some function of k (in our case we will simply have $k' \leq k$), and the size of the new instance I' is upper bounded by some function $g(k)$. Hence the size of the reduced instance does not depend on the size of the original instance. While kernelization is possible for all FPT problems (the two notions are actually equivalent), the interesting question is whether a given FPT problem admits *polynomial kernels*, where the size of the reduced instance is bounded by some polynomial in k. Note that, under some complexity assumptions, not all FPT problems admit polynomial kernels [5, 6, 7, 9, 19, 24].

In this paper we provide polynomial kernels for Strictly Chordal Completion, Strictly Chordal Deletion and Strictly Chordal Editing. *Strictly chordal* graphs are a subclass of chordal graphs, also known as *block duplicate* graphs [22, 23, 18]. They can be obtained from block graphs, i.e., graphs in which every block (bi-connected component) induces a clique, by repeatedly choosing some vertex u and adding a *true twin* v of u, that is a vertex v adjacent to u and all other neighbors of u. They can also be characterized as dart, gem-free chordal graphs (see Figure 1 and next section). Strictly chordal graphs are known to be a subclass of 4-leaf power graphs [23], and a super-class of 3-leaf power graphs. Leaf power graphs have been introduced in the context of phylogeny reconstruction [28]. A graph is said to be p-leaf power, for some integer p, if its vertices can be bijectively mapped onto the leaves of some tree, such that two vertices are adjacent in the graph if the corresponding leaves are at distance at most p in the tree.

Related work

Kernelization for chordal completion goes back to the '90s and the seminal paper of Kaplan, Shamir and Tarjan [21]. Since then, several authors addressed completion, deletion and/or editing problems towards sub-classes of chordal graphs, as 3-leaf power graphs [3], split and threshold graphs [20], proper interval graphs [4], trivially perfect graphs [2, 16, 17, 20] or ptolemaic graphs [12]. All these classes have in common that they can be defined as chordal graphs, plus a constant number of obstructions. Several questions remain open, for example it is not known whether chordal deletion or chordal editing admit polynomial kernels [11]. We could also ask whether completion and editing problems towards 4-leaf power graphs admit a polynomial kernel, knowing that they are FPT [15].

Our contribution

Firstly, we prove that problems Strictly Chordal Completion, Strictly Chordal Deletion and Strictly Chordal Editing are NP-complete. Secondly, we give a kernelization algorithm for the Strictly Chordal Completion problem, producing a reduced instance with $O(k^3)$ vertices. Eventually, we discuss how to extend this approach in order to obtain an $O(k^4)$-vertex kernel for both Strictly Chordal Deletion and Strictly Chordal Editing. Above all, our purpose is to exhibit general techniques that might, we hope, be extended to kernelizations for edge modification problems towards other graph classes. Several such algorithms, e.g., [3, 17] share the following feature. Very informally, the target class $\mathcal{G}$ admits a tree-like decomposition, in the sense that the vertices of any graph $H \in \mathcal{G}$ can be partitioned into clique modules, and these modules can be mapped onto the nodes of a decomposition tree, the structure of the tree describing the adjacencies between modules. Therefore, if an arbitrary graph G can be transformed into graph H by at most k edge additions or deletions, at most $2k$ modules can be affected by the modifications. By removing the affected nodes from the decomposition tree, we are left with several components that correspond, in the initial graph G as well as in H, to induced subgraphs that may be large but that already belong to the target class. Moreover, these chunks are attached to the rest of graph G in a very regular way, through one or two nodes of the decomposition tree. The kernelization algorithms need to analyze these chunks and provide reduction rules, typically by ensuring a small number of nodes in the decomposition tree, plus the fact that each node corresponds to a module of small size.

The class of strictly chordal graphs does not have exactly a tree-like decomposition, but still can be decomposed into a structure of *block graph*, which can be seen as a generalization of a tree. Our algorithms exploit these informal observations and provide the necessary reduction rules together with the combinatorial analysis for the kernel size. Proofs of statements labeled with ($\star$) are omitted in this extended abstract.

2 Preliminaries

We consider simple, undirected graphs $G = (V, E)$ where V denotes the *vertex set* and $E \subseteq (V \times V)$ the *edge set* of G. We will sometimes use $V(G)$ and $E(G)$ to clarify the context. Given a vertex $u \in V$, the *open neighborhood* of u is the set $N_G(u) = \{v \in V : uv \in E\}$. The *closed neighborhood* of u is defined as $N_G[u] = N_G(u) \cup \{u\}$. Two vertices u and v are *true twins* if $N_G[u] = N_G[v]$. Given a subset of vertices $S \subseteq V$, $N_G[S]$ is the set $\cup_{v \in S} N_G[v]$ and $N_G(S)$ is the set $N_G[S] \setminus S$. We will omit the mention to G whenever the context is clear. The subgraph *induced* by S is defined as $G[S] = (S, E_S)$ where $E_S = \{uv \in E : u \in S, v \in S\}$. For the sake of readability, given a subset $S \subseteq V$ we define $G \setminus S$ as $G[V \setminus S]$. A subgraph C is a *connected component* of G if it is a maximal connected subgraph of G. A subset of vertices $M \subseteq V$ is a *module* if for every vertices $x, y \in M$, $N(x) \setminus M = N(y) \setminus M$. A set $S \subseteq V$ is a *separator* of G if $G \setminus S$ is not connected. Given two vertices u and v of G, the separator S is a uv-separator if u and v lie in distinct connected components of $G \setminus S$. Moreover, S is a *minimal* uv-separator if no proper subset of S is a uv-separator. Finally, a separator S is *minimal* if there exists a pair $\{u, v\}$ such that S is a minimal uv-separator.

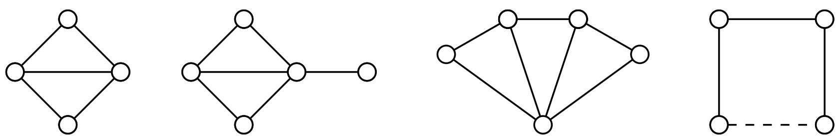

Figure 1 The diamond, dart, gem and cycle of length at least 4.

▶ **Definition 1.** *Given a graph $G = (V, E)$, a* critical clique *of G is a set $K \subseteq V$ such that $G[K]$ is a clique, K is a module and is inclusion-wise maximal under this property.*

Notice that K is a maximal set of true twins and that the set $\mathcal{K}(G)$ of critical cliques of any graph G partitions its vertex set $V(G)$. This leads to the following definition.

▶ **Definition 2** (Critical clique graph). *Let $G = (V, E)$ be a graph. The* critical clique graph *of G is the graph $\mathcal{C}(G) = (\mathcal{K}(G), E_{\mathcal{C}})$ with $E_{\mathcal{C}} = \{KK' \mid \forall u \in K, \forall v \in K', uv \in E\}$.*

Strictly chordal graphs

Block graphs are graphs in which every block (bi-connected component) is a clique. They can also be characterized as chordal graphs that do not contain diamonds as induced subgraphs [1]. A natural generalization of block graphs are strictly chordal graphs, also known as block duplicate graphs [22, 23, 18], that are obtained from block graphs by adding true twins [18].

▶ **Theorem 3** ([26]). *Let $G = (V, E)$ be a graph. The following conditions are equivalent:*
1. *G is a strictly chordal graph,*
2. *The critical clique graph $\mathcal{C}(G)$ is a block graph,*
3. *G does not contain any dart, gem or hole as an induced subgraph (see Figure 1),*
4. *The minimal separators of G are pairwise disjoint.*

We consider the following problem:

Strictly Chordal Editing
Input: A graph $G = (V, E)$, a parameter $k \in \mathbb{N}$
Question: Does there exist a set of pairs $F \subseteq (V \times V)$ of size at most k such that the graph $H = (V, E \triangle F)$ is strictly chordal, with $E \triangle F = (E \setminus F) \cup (F \setminus E)$?

The Strictly Chordal Completion and Strictly Chordal Deletion problems are defined similarly by requiring F to be disjoint from (resp. included in) edge set E. Given a graph $G = (V, E)$, a set $F \subseteq (V \times V)$ such that the graph $H = (V, E \triangle F)$ is strictly chordal is called an *edition* of G. When F is disjoint from E (resp. included in E) it is called a *completion* (resp. a *deletion*) of G. For the sake of simplicity we use $G \triangle F$, $G + F$ and $G - F$ to denote the resulting strictly chordal graphs in all version of the problem. In all cases, F is *optimal* whenever it is minimum-sized. Given an instance $(G = (V, E), k)$ of any of the aforementioned problems, we say that F is a k*-edition* (resp. k-completion, k-deletion) whenever F is an edition (resp. completion, deletion) of size at most k. Finally, a vertex is *affected* by F whenever it is contained in some pair of F. We say that an instance $(G = (V, E), k)$ of any of the aforementioned problems is a yes-instance whenever it admits a k-edition (resp. k-completion, k-deletion).

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two vertex-disjoint graphs and let $S_1 \subseteq V_1, S_2 \subseteq V_2$. The *join composition* of G_1 and G_2 on S_1 and S_2, denoted $(G_1, S_1) \otimes (G_2, S_2)$, is the graph $(V_1 \cup V_2, E_1 \cup E_2 \cup (S_1 \times S_2))$.

▶ **Lemma 4** ([22]). *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two strictly chordal graphs and let $S_1 \subseteq V_1, S_2 \subseteq V_2$. The graph $G = (G_1, S_1) \otimes (G_2, S_2)$ is strictly chordal if for each $i \in \{1, 2\}$, S_i is a critical clique or is a clique included in exactly one maximal clique of G_i.*

We will use the following result that guarantees that any clique module of a given graph G will remain a clique module in any optimal edge modification towards some hereditary class of graphs, in particular towards strictly chordal graphs.

▶ **Lemma 5** ([3]). *Let $\mathcal{G}$ be an hereditary class of graphs closed under true twin addition. For every graph $G = (V, E)$, there exists an optimal edition (resp. completion, deletion) F into a graph of $\mathcal{G}$ such that for any two critical cliques K and K' either $(K \times K') \subseteq F$ or $(K \times K') \cap F = \emptyset$.*

In the remainder of this paper we always assume that the considered optimal editions (resp. completions, deletions) satisfy Lemma 5.

2.1 Hardness results

The NP-completeness of Strictly Chordal Completion follows directly from the proof of NP-completeness of 3-Leaf Power Completion from [14]. We can use the same reduction from Biclique Deletion, taking the complement of a bipartite graph and adding an universal vertex.

▶ **Theorem 6.** *Strictly Chordal Completion is NP-complete.*

We show the NP-completeness of Strictly Chordal Editing and Strictly Chordal Deletion by giving a reduction from Cluster Editing and Cluster Deletion, known to be NP-complete [25, 30, 27].

▶ **Theorem 7.** *Strictly Chordal Editing and Strictly Chordal Deletion are NP-complete.*

Proof. A graph is a cluster graph if it does not contain any induced path on three vertices (so-called P_3). Given an instance $(G = (V, E), k)$ of Cluster Editing, we construct an instance of Strictly Chordal Editing by adding a clique $U = \{u_1, \ldots, u_{k+1}\}$ of size $k + 1$ adjacent to all vertices of V, and for each vertex x in V, $k + 1$ vertices $\{v_1^x, \ldots, v_{k+1}^x\}$ adjacent only to x. Let $(G' = (V', E'), k)$ be the produced instance. We show that the graph G admits a k-edition into a cluster graph if and only if G' admits a k-edition into a strictly chordal graph. Suppose first that there is a k-edition F of G into a cluster graph. The graph $G \triangle F$ is a graph without any P_3 as induced subgraph. Now consider the graph $H' = G' \triangle F$. By construction $H'[V]$ contains no P_3, so H' is chordal and contains neither gems nor dart. By Theorem 3, it follows that H' is strictly chordal. Now suppose that there exists a k-edition F' of G' into a strictly chordal graph. By contradiction, suppose that G' contains a P_3 $\{x, y, z\}$ where x, z are the ends of the path. Then, there exist $i, j \in \{1, \ldots, k + 1\}$ such that $\{x, y, z, u_i, v_j^y\}$ forms a dart, contradicting that $G' \triangle F'$ is strictly chordal, thus $G \triangle F$ is a cluster graph.

The same reduction can be done from Cluster Deletion to Strictly Chordal Deletion. This concludes the proof. ◀

3 Kernelization algorithm for Strictly Chordal Completion

We begin this section by providing a high-level description of our kernelization algorithm. We use the critical clique graph of strictly chordal graphs to bound the number of vertices of a reduced instance. Let us consider a positive instance $(G = (V, E), k)$ of Strictly Chordal Completion, F a suitable solution and $H = G + F$. Denote by $\mathcal{C}(H)$ the critical clique graph of H as described Definition 2 and recall that $\mathcal{C}(H)$ is a block graph (Theorem 3). Since $|F| \leq k$, we know that at most $2k$ critical cliques of $\mathcal{C}(H)$ may contain affected vertices. Let A be the set of such critical cliques, T the minimum subgraph of $\mathcal{C}(H)$ that spans all critical cliques of A and A' the set of critical cliques of degree at least 3 in T. We shall see later that $|A' \backslash A| \leq 3 \cdot |A|$ (Lemma 19). We will define the notion of block-branch, corresponding to subgraphs of G that are strictly chordal. We will focus our interest on two types of block-branches: the ones that are connected to the rest of the graph by only one critical clique, called *1-block-branches*, and the ones that are connected to the rest of the graph by exactly two critical cliques, called *2-block-branches*. The connected components of the graph $T \backslash (A \cup A')$ correspond to parts of 2-block-branches and the length of these 2-block-branches will be bounded by 3. We will see that there are at most $4k$ such connected components, thus there is $O(k)$ critical cliques in T. Finally, the connected components of the graph $\mathcal{C}(H) \backslash V(T)$ correspond to 1-block-branches or sets of connected 1-block-branches. Each 1-block-branch will be reduced to 2 critical cliques, and to each critical clique or maximal clique of T there is a linear number of 1-block-branches of $\mathcal{C}(H) \backslash V(T)$ adjacent to it. Altogether, the graph $\mathcal{C}(H)$ contains $O(k^2)$ critical cliques, and each critical clique is of size at most $k+1$, hence the graph G contains $O(k^3)$ vertices.

3.1 Classical reduction rules

We first give classical reduction rules when dealing with modification problems. The first rule is safe for any target graph class hereditary and closed under disjoint union. The second one comes from the fact that strictly chordal graphs are hereditary and closed under true twin addition, combined with Lemma 5.

▶ **Rule 1.** *Let C be a strictly chordal connected component of G. Remove $V(C)$ from G.*

▶ **Rule 2.** *Let $K \subseteq V$ be a set of true twins of G such that $|K| > k+1$. Remove $|K| - (k+1)$ arbitrary vertices in K from G.*

▶ **Lemma 8** (Folklore, [3]). *Rules 1 and 2 are safe and can be applied in polynomial time.*

3.2 Block-branch reduction rules

We now consider the main structure of our kernelization algorithm, namely *block-branches.*

▶ **Definition 9** (block-branch). *Let $G = (V, E)$ be a graph. We say that a subgraph B of G is a* block-branch *if $V(B)$ is an union of critical cliques $K_1, \ldots, K_r \in \mathcal{K}(G)$ such that the subgraph of $\mathcal{C}(G)$ induced by $K_1, \ldots, K_r$ is a connected block graph.*

We emphasize that our definition of a block-branch B is stronger than simply requiring B to be an induced strictly chordal graph. For example, if G is the dart graph, the subgraph obtained by removing the pendant vertex is strictly chordal, but it is not a block-branch because the corresponding critical cliques do not form a block graph in $\mathcal{C}(G)$. Let B be a block-branch of graph G and let $K_1, \ldots, K_r$ be the critical cliques of G contained in $V(B)$. We say that K_i is an *attachment point* of B if $N_G(K_i) \setminus V(B) \neq \emptyset$. A block-branch B is a p-block-branch if it has exactly p attachment points. We denote B^R the subgraph of B in which all attachment points have been removed.

We first give structural Lemmata on block-branches that will be helpful to prove the safeness of our rules.

▶ **Lemma 10.** *Let $G = (V, E)$ be a graph and B a block-branch of G. For any attachment point P of B, let $B' = B \backslash P$, consider the connected components $G_1, G_2, \ldots, G_r$ of B' and let $Q_i = N_B(P) \cap V(G_i)$. For every i, $1 \leq i \leq r$, Q_i is a critical clique of G_i or Q_i is included in exactly one maximal clique of G_i.*

Proof. First, we show that all sets Q_i are cliques. Suppose that Q_i is not a clique for some $1 \leq i \leq r$. Let x and y be non adjacent vertices of Q_i and $z \in P$. Since G_i is connected, take a shortest path π between x and y in G_i. The subgraph induced by the vertices $\{x, y, z\}$ and those of π contains either a cycle of length at least 4 if z is not adjacent to any inner vertex of π, which is a contradiction since B is a block graph, or a diamond with z being one of its vertices of degree 3. In the latter case, since z is not in the same critical clique of $\mathcal{C}(G)$ as its true twin in the diamond, the critical cliques of $\mathcal{C}(G)$ that contain some vertices of this diamond also form a diamond in $\mathcal{C}(G)$. This diamond is formed by critical cliques of G contained in B, contradicting the definition of a block-branch. In all cases we have a contradiction, it remains that Q_i is a clique in G.

Now, suppose that Q_i, $1 \leq i \leq r$ is included in two or more maximal cliques in G_i and let $u, v \in V(G_i) \backslash Q_i$ be two vertices adjacent to Q_i such that $uv \notin E(G_i)$. If Q_i is not a critical clique of G_i, there are two cases, either Q_i is not a module of G_i or Q_i is included in a larger module of G_i. In the first case, there are two vertices $x, y \in Q_i$ that are not in the same module of G_i. Let $z \in V(G_i) \backslash Q_i$ be a vertex adjacent to only x or y, say w.l.o.g. x. The subgraph of G induced by $\{u, v, x, y, z\}$ is either a dart if z is not adjacent to u nor v or a gem if it is adjacent to exactly one of them. If z is adjacent to both u and v, then $\{z, u, y, v\}$ forms a C_4 in this order, leading to a contradiction. In the second case, that is Q_i is included in a larger module of G_i, let $x \in Q_i$, y a vertex in the same module as x in G_i and $z \in P$. Observe that $zy \notin E(G)$. The set $\{x, y, z, u, v\}$ induces a dart in G. In all cases we have a contradiction, since the forbidden structure is also contained in B. It remains that Q_i is a critical clique of G_i or Q_i is included in exactly one maximal clique of G_i. ◀

▶ **Lemma 11.** *Let $G = (V, E)$ be a graph and B a block-branch of G. Let F be an optimal completion of G that respects Lemma 5, and $H = G + F$. For any attachment point P of B, let C be the critical clique of H which contains P. Then $C' = C \backslash V(B^R)$ is a critical clique of $H' = H \backslash V(B^R)$ or is included in exactly one maximal clique of H'.*

Proof. Assume that C' is included in two or more maximal cliques in H'. Let $u, v \in V(H') \backslash C'$ be two vertices adjacent to C' such that $uv \notin E(H')$. If C' is not a critical clique of H', then either C' is not a module of H' or C' is included in a larger module of H'. In the first case, there are two vertices $x, y \in C'$ that are not in the same module of H'. Let $z \in V(H') \backslash C'$ be a vertex adjacent to only x or y, say w.l.o.g. x. The subgraph of H induced by $\{u, v, x, y, z\}$ is either a dart if z is not adjacent to u nor v or a gem if it is adjacent to exactly one of them. If z is adjacent to both u and v, then $\{z, u, y, v\}$ forms a C_4 in this order, leading to a contradiction. In the second case, C' is included in a larger module of H'. Let $x \in C'$, y be a vertex in the same module as x in H' and $z \in N_B(P)$. The set $\{x, y, z, u, v\}$ induces a dart in H if zy is not in $E(H)$, else a gem in H. In all cases we have a contradiction since H is strictly chordal. It remains that C' is a critical clique of H' or C' is included in exactly one maximal clique of H'. ◀

▶ **Lemma 12.** *Let $G = (V, E)$ be a graph and B a 1-block-branch of G with attachment point P. There exists an optimal completion F of G such that:*

- *The set of vertices of B affected by F is included in $P \cup N_B(P)$.*
- *In $H = G + F$ the vertices of $N_B(P)$ are all adjacent to the same vertices of $V(G) \backslash V(B^R)$.*

Proof. Let F be an optimal completion of G. Let C be the critical clique of H which contains P and let $C' = C \backslash V(B^R)$. Let $Q_1, \ldots, Q_r$ be the cliques that partition $N_B(P)$ (Lemma 10) and G_i the connected component of B^R which contains Q_i. The graphs $H' = H \backslash V(B^R)$ and G_i, for $1 \leq i \leq r$, are strictly chordal by heredity. By Lemma 10, for $1 \leq i \leq r$, Q_i is a critical clique of G_i or is in exactly one maximal clique of G_i. By Lemma 11, C' is a critical clique of H' or is in exactly one maximal clique of H'. Thus, by Lemma 4 the graph H^* corresponding to the graph $(\bigcup_{1 \leq i \leq r} G_i, \bigcup_{1 \leq i \leq r} Q_i) \otimes (H', C')$ is a strictly chordal graph. Let F^* be such that $H^* = G + F^*$. By construction $F^* \subseteq F$, and the desired properties are verified. ◀

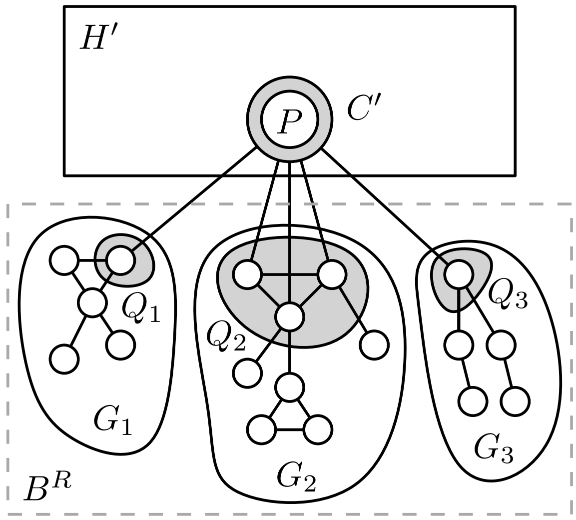

Figure 2 Illustration of the proof of Lemma 12 for $r = 3$. The graph $H^* = (G_1 \cup G_2 \cup G_3, Q_1 \cup Q_2 \cup Q_3) \otimes (H', C')$ is strictly chordal by Lemma 4.

▶ **Rule 3.** *Let $(G = (V, E), k)$ be an instance of* Strictly Chordal Completion. *If G contains a 1-block-branch B with attachment point P, then remove from G the vertices of $V(B^R)$ and add a clique K of size $min\{|N_B(P)|, k + 1\}$ adjacent to P.*

▶ **Lemma 13** (⋆)**.** *Rule 3 is safe.*

▶ **Lemma 14.** *Let $(G = (V, E), k)$ be a yes-instance of* Strictly Chordal Completion*, reduced by Rule 2. Let $B_1, \ldots, B_l$ be disjoint 1-block-branches of G with attachment points $P_1, \ldots, P_l$ which have the same neighborhood N in $G \setminus \bigcup_{i=1}^{l} V(B_i)$ and form a disjoint union of cliques $Q_1, \ldots Q_r$ in $G[P_1 \cup \cdots \cup P_l]$. If $\Sigma_{i=1}^{l}|P_i| > 2k+1$, then for every k-completion F of G, N has to be a clique of $H = G+F$. Moreover, if $r > 1$ and $(\Sigma_{i=1}^{l}|P_i|) - max_{1 \le j \le r}\{|Q_j|\} > k$ then N is a critical clique of H.*

Proof. If $r = 1$, assume for a contradiction that N is not a clique in H and let $x, y \in N$ such that $xy \notin E(H)$. By our hypothesis, $|Q_1| > 2k + 1$. Recall that since G is reduced by Rule 2, its critical cliques have at most $k + 1$ vertices, in particular each P_i is of size at most $k + 1$. At least one block-branch B_i contains some edge uz with $u \in P_i$ and $z \in V(B_i) \setminus P_i$ (otherwise each block-branch B_i is formed only of P_i, so Q_1 would be a clique module in G, hence contained in some critical clique of G, contradicting the fact that all critical cliques of G are of size at most $k + 1$). In graph G, z is not adjacent to any other block-branch B_j, for any $j \neq i$. Since $\sum_{j \neq i} |P_j| > k$ and F is of size at most k, there must exist some $j \neq i$ and some vertex $v \in P_j$ such that $vz \notin E(H)$. Let us examine the subgraph of H induced by vertices $\{z, u, v, x, y\}$. If z is not adjacent to any of $\{x, y\}$, we obtain a dart. If it is adjacent to exactly one of them, the five vertices induce a gem. Finally, if z is adjacent to both x and y in H, then $\{z, x, v, y\}$ forms a C_4 in this order. In all cases we have a contradiction. It remains that N is a clique in H.

If $r > 1$, suppose for a contradiction that N is not a clique in H, then there exist two vertices $x, y \in N$ such that $xy \notin E(H)$. For any pair of vertices $u_i \in Q_i, u_j \in Q_j, i \neq j$, the set $\{x, u_i, y, u_j\}$ induces a C_4 in G. Thus (u_j, u_i) has to be in F, implying $|F| > k$, a contradiction. Hence N has to be a clique in H. Suppose now that N is not a module in H, then there exists $x, y \in N$ and $z \in V(G) \setminus (N \cup Q_1 \cup \cdots \cup Q_r)$ adjacent to only one of the vertices x or y, say w.l.o.g. x. For any pair of vertices $u_i \in Q_i, u_j \in Q_j, i \neq j$, if $(u_j, u_i) \notin F$

the set $\{x, y, u_i, u_j, z\}$ induces a dart if neither u_i nor u_j is adjacent to z, a gem if one of them is adjacent to z, or if both of them are adjacent to z, $\{y, u_i, u_j, z\}$ induces a C_4 in H. Thus (u_j, u_i) has to be in F, implying $|F| > k$, a contradiction. Hence N must be a module in H. If N is not a critical clique of H, then it is strictly contained in some critical clique N'. A vertex $x \in N' \backslash N$ must have been made adjacent to $N_H[N]$. If x is not in some P_i or $N_{B_i}(P_i)$, then it must have been made adjacent to all vertices of $P_1 \cup \cdots \cup P_l$, implying $|F| > k$, a contradiction. If x is in some P_i or or $N_{B_i}(P_i)$, since $(\Sigma_{i=1}^{l}|P_i|) - max_{1\leq j\leq r}\{|Q_j|\} > k$, then it must have been made adjacent to the vertices of every $P_j, j \neq i$, implying $|F| > k$, a contradiction. It remains that N is a critical clique in H. ◀

▶ **Rule 4.** *Let $(G = (V, E), k)$ be an instance of* Strictly Chordal Completion *and $B_1, \ldots, B_l$ disjoint 1-block-branches of G with attachment points $P_1, \ldots, P_l$ having the same neighborhood N in $G \backslash \bigcup_{i=1}^{l} V(B_i)$. Assume that $\Sigma_{i=1}^{l}|P_i| > 2k + 1$ and let $Q_1, \ldots Q_r$ be the disjoint union of cliques in $G[P_1 \cup \cdots \cup P_l]$.*

- *If $r = 1$, remove the vertices $\bigcup_{i=1}^{l} V(B_i)$, add two adjacent cliques K and K' of size $k+1$ with neighborhood N and a vertex u_K adjacent to every vertex of K.*
- *If $r > 1$ and $(\Sigma_{i=1}^{l}|P_i|) - max_{1\leq j\leq r}\{|Q_j|\} > k$, remove the vertices $\bigcup_{i=1}^{l} V(B_i)$ and add two non-adjacent cliques of size $k + 1$ with neighborhood N.*

▶ **Lemma 15** ($\star$)**.** *Rule 4 is safe.*

3.3 Reducing the 2-block-branchs

Let B be a 2-block-branch of a graph G reduced by Rule 3, with attachment points P_1 and P_2. We say that B is *clean* if B^R is connected, and that the *length* of a clean 2-block-branch is the length of a shortest path between its two attachment points in $\mathcal{C}(B)$.

▶ **Lemma 16.** *Let $G = (V, E)$ be graph and B a clean 2-block-branch of length at least 3 with attachment points P_1, P_2 that are in different connected components of $G \backslash V(B^R)$. There exists an optimal completion F of G such that:*

- *The set of vertices of B affected by F is included in $P_1 \cup N_B(P_1) \cup P_2 \cup N_B(P_2)$,*
- *In $H = G + F$, the vertices of $N_B(P_1)$ (resp. $N_B(P_2)$) are all adjacent to the same vertices of $V(G) \backslash V(B^R)$.*

Proof. Let F be an optimal completion of G, and $H = G + F$. Recall that by hypothesis P_1 and P_2 are in different connected components of $G \setminus V(B^R)$ and let G_1 and G_2 be these components. Consider the graphs $H_1 = H[V(G_1) \cup V(B^R)]$, $H_2 = H[V(G_2) \cup V(B^R)]$ and $H_3 = H[V(G) \backslash (V(G_1) \cup V(G_2) \cup V(B^R))]$, which are strictly chordal graphs by heredity. Let C_i be the critical clique of H_i which contains P_i, and $C'_i = C_i \backslash V(B^R)$, $i \in \{1, 2\}$. By Lemma 11, C'_i is a critical clique of $H'_i = H_i \backslash V(B^R)$ or is in exactly one maximal clique of H'_i. Since P_1 are P_2 are at distance at least 3 to each other, $N_B(P_1)$ and $N_B(P_2)$ are disjoint. Since B^R is connected, Lemma 10 gives that $N_B(P_1)$ and $N_B(P_2)$ are cliques in G and are critical cliques of B^R or are each in exactly one maximal clique of B^R. By Lemma 4, the graph H^* corresponding to the disjoint union of $((B^R, N_B(P_1)) \otimes (H'_1, C'_1)), N_B(P_2)) \otimes (H'_2, C'_2)$ and H_3 is strictly chordal and the completion $F^* \subseteq F$, such that $H^* = G + F^*$, respects the desired properties. ◀

▶ **Rule 5.** *Let $(G = (V, E), k)$ be an instance of* Strictly Chordal Completion *and B a clean 2-block-branch of G of length at least 3 with attachment points P_1, P_2 that are in different connected components of $G \backslash V(B^R)$. Remove the vertices of B^R, and add two new cliques K_1 and K_2 of size respectively $min\{|N_B(P_1)|, k + 1\}$ and $min\{|N_B(P_2)|, k + 1\}$ with the edges $(P_1 \times K_1), (K_1 \times K_2), (K_2 \times P_2)$.*

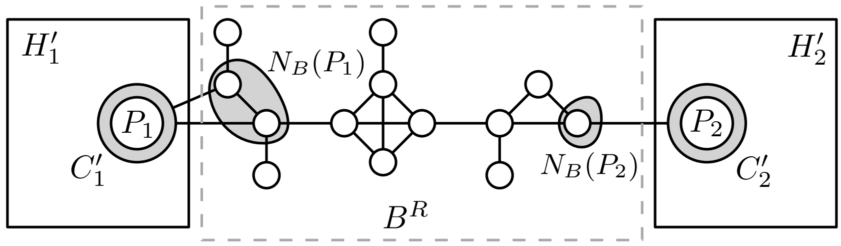

Figure 3 Illustration of the proof of Lemma 16. The graph $H^* = ((B^R, N_B(P_1)) \otimes (H'_1, C'_1)), N_B(P_2)) \otimes (H'_2, C'_2)$ is strictly chordal by Lemma 4.

▶ **Lemma 17** ($\star$). *Rule 5 is safe.*

4 Size of the kernel

We first state that reduction rules involving block-branches can be applied in polynomial time.

▶ **Lemma 18.** *Given an instance $(G = (V, E), k)$ of* Strictly Chordal Completion*, Rules 3 to 5 can be applied in polynomial time.*

Proof. We rely on a linear time computation of the critical clique graph $\mathcal{C}(G)$ of G [29] and a linear time recognition algorithm for block graphs [1]. We show that we can enumerate all 1-block-branches and 2-block-branches in polynomial time. Since an attachment point is by definition a critical clique of G, one can detect 1-block-branches by removing a critical clique of $\mathcal{C}(G)$ and look among the remaining connected components those that induce a connected block graph together with P (in $\mathcal{C}(G)$). Considering a maximal set of such components together with P gives a 1-block-branch B. We proceed similarly to detect clean 2-block-branches by removing a pair of critical cliques P_1, P_2 of $\mathcal{C}(G)$, and look among the remaining connected components those that induce a connected block graph (in $\mathcal{C}(G)$) together with P_1 and P_2. Such components together with P_1 and P_2 gives a clean 2-block-branch B. Recall that Rule 5 only applies if P_1 and P_2 are not in the same component of $G \setminus V(B^R)$, which can easily be verified. Since there is $O(|V(G)|)$ critical cliques, the result follows. ◀

▶ **Lemma 19.** *Let $G = (V, E)$ be a connected block graph, a set $A \subseteq V(G)$ and T_A be a minimal connected induced subgraph of G that spans all vertices of A. Denote by $f(T_A)$ the set of vertices of degree at least 3 in T_A. We have:*

(i) *The subgraph T_A is unique,*
(ii) $|f(T_A) \setminus A| \leq 3 \cdot |A|$,
(iii) *The graph $T_A \setminus (A \cup f(T_A))$ contains at most $2 \cdot |A|$ connected components.*

Proof. A convenient way to represent the tree-like structure of block graph G is its block-cut tree $T_{BC}(G)$. Recall that the block-cut tree of a graph G has two types of nodes: the *block nodes* correspond to blocks of G (i.e, bi-connected components which, in our case, are precisely the maximal cliques of G) and the *cut nodes* correspond to cut-vertices of G. We put an edge between a cut node and a block node in $T_{BC}(G)$ if the corresponding cut-vertex belongs to the corresponding block of G.

Each vertex of G that is not a cut-vertex belongs to a unique block of G. Therefore we can map each vertex v of $V(G)$ on a unique node $n(v)$ of $T_{BC}(G)$ as follows: if v is a cut-vertex, we map it on its corresponding cut node in $T_{BC}(G)$, otherwise we map it on the block node corresponding to the unique block of G containing v. Observe that, for any two vertices $u, v \in A$, any (elementary) path P from u to v in G corresponds to the path P_{BC} from $n(u)$ to $n(v)$ in $T_{BC}(G)$. In particular, P contains all vertices corresponding to cut nodes of P_{BC}, therefore T_A must contain all these vertices. Altogether, the vertices of T_A are precisely the vertices of A, plus all cut-vertices of G corresponding to the cut nodes of such paths, implying the unicity of T_A.

Let $T_{BC}(A)$ denote the subtree of $T_{BC}(G)$ spanning all nodes of $n(A) = \{n(a) \mid a \in A\}$. We count the vertices of $f(T_A) \setminus A$, so let a' be such a vertex. By the previous observation, it is a cut-vertex of G, so $n(a')$ is a cut node of $T_{BC}(A)$. Let v be a neighbor of a' in T_A. By construction of T_A and $T_{BC}(A)$, there is a block node b adjacent to $n(a')$ in $T_{BC}(A)$, such that v and a' are in the maximal clique of G corresponding to node b. Moreover, v is in A, or v is a cut-vertex of G such that the cut node $n(v)$ is adjacent to b in $T_{BC}(A)$. Hence we have:

1. $n(a')$ is of degree at least 3 in $T_{BC}(A)$, or
2. $n(a')$ is the neighbor of a block node b of degree at least three in $T_{BC}(A)$, or, if none of these hold, then
3. $n(a')$ has exactly two neighbors b and b' in $T_{BC}(A)$, the corresponding maximal cliques of G contain at least p vertices of A, where p plus the degrees of b and b' in T_{BC} is at least three.

Let k be the number of leaves of $T_{BC}(A)$. Observe that for any leaf l of $T_{BC}(A)$, there is some $a \in A$ such that $n(a) = l$. Choose for each leaf l a unique vertex $a_l \in A$ such that $n(a_l) = l$, we call a_l a leaf vertex. Note that the number of vertices $a' \in f(A)$ corresponding to the first two items of the above enumeration is upper bounded by $3k$. Indeed, $n(a')$ is incident to an edge of $T_{BC}(A)$, having an end node of degree at least 3. One can easily check that, in any tree of k leaves, the number of such edges is bounded by $3k$ (this can be shown by induction on the number of leaves of the tree, adding a new leaf node at a time). We count now the vertices $a' \in f(A)$ of the third type. By the third item, a' has at least one neighbor $a \in A$ in graph T_A, such that a is *not* a leaf vertex. Observe that a can be in the neighborhood of at most two vertices $a' \in f(A)$ of this third type. Altogether it follows that $|f(A) \setminus A \leq 3k + 2(|A| - k) \leq 3 \cdot |A|$.

To prove the third item, the number of components of $T_A \setminus (A \cup f(A))$, we visualize again the situation in $T_{BC}(A)$. Recall that for any node of degree at least three in $T_{BC}(A)$ it is either a cut node, in which case it is in $f(A)$ (the first case of the enumeration above), or it is a block node but then all its neighbors in $T_{BC}(A)$ correspond to vertices of $f(A)$ (the second case of the enumeration above). Therefore the components of $T_A \setminus f(A)$ correspond to the components of $T_{BC}(A)$ after removal of all nodes of degree at least 3. Hence the number of such components is upper bounded by $2k$. Removing the leaf vertices of A does not increase the number of components, and the removal of each other vertex of A increases the number of components by at most one. Thus $T_A \setminus (A \cup f(A))$ has at most $2\ cdot|A|$ components, concluding the proof of our lemma. ◀

▶ **Lemma 20** ([29]). *Let $G = (V, E)$ be a graph and $H = G \triangle \{uv\}$ for $u, v \in V$, then $\mathcal{K}(H) \leq \mathcal{K}(G) + 2$.*

▶ **Theorem 21.** *Strictly Chordal Completion admits a kernel with $O(k^3)$ vertices.*

Proof. Let $(G = (V, E), k)$ be a reduced yes-instance of Strictly Chordal Completion, F a k-completion of G and $H = G + F$. We assume that G is connected, the following arguments can be easily adapted if this is not the case by summing over all connected components of G. The graph H is strictly chordal, thus the graph $\mathcal{C}(H)$ is a block graph.

We first show that $\mathcal{C}(H)$ has $O(k^2)$ vertices. We say that a critical clique of $\mathcal{C}(H)$ is affected if it contains a vertex affected by F. Let A be the set of affected critical cliques of $\mathcal{C}(H)$. Since $|F| \leq k$, we have $|A| \leq 2k$. Let T be the connected minimal subgraph of H that spans all critical cliques of A, and A' the set of vertices of degree at least 3 in T.

We first show that $|V(T)|$ is $O(k)$. By Lemma 19, $|A' \setminus A| \leq 3 \cdot |A| \leq 6k$. The connected components of the graph $T \backslash (A \cup A')$ are paths since every vertex is of degree at most 2 and by Lemma 19 there are at most $4k$ such paths. Let R be one of these paths, it is composed of unaffected critical cliques, thus there exists a 2-block-branch B of G that contains R. Moreover, the extremities of R are the attachment points of B, which have been reduced by Rule 5. Thus R is of length at most 3. It remains that $|V(T)| = 2k + 6k + 4 \cdot 4k = O(k)$.

We will now show that $\mathcal{C}(H) \backslash V(T)$ contains $O(k^2)$ critical cliques. First observe that each connected component of $\mathcal{C}(H) \backslash V(T)$ is adjacent to some vertices of T since the graph is reduced by Rule 1. Since $\mathcal{C}(H)$ is a block graph, connected components of $\mathcal{C}(H) \backslash V(T)$ are adjacent to either a critical clique of T or a maximal clique of T (else, there would be a diamond in $\mathcal{C}(H)$). We claim that there are $O(k^2)$ critical cliques in the connected components of $\mathcal{C}(H) \backslash V(T)$ adjacent to maximal cliques of T. Since T is a block graph and $|V(T)| = O(k)$, there are $O(k)$ maximal cliques in T. Moreover, there is only one connected component of $\mathcal{C}(H) \backslash V(T)$ adjacent to each maximal clique, otherwise there would be a diamond in $\mathcal{C}(H)$. Take K a maximal clique of T and let X be its adjacent connected component of $\mathcal{C}(H) \backslash V(T)$. Observe that X has to be an union of 1-block-branches and their attachment points form a clique. By Rule 4, there are at most $2k + 1$ 1-block-branches in X and each one has been reduced by Rule 3, thus X contains at most $4k + 2$ critical cliques.

Finally, we claim that there are $O(k^2)$ critical cliques in the connected components of $\mathcal{C}(H) \backslash V(T)$ adjacent to critical cliques of T. First take a critical clique of $T \backslash A$ and its adjacent connected components of $\mathcal{C}(H) \backslash V(T)$. Observe that they form a 1-block-branch, reduced by Rule 3, thus the adjacent connected component consists in only one critical clique. Next, take a critical clique a of A, and let $C_1, \ldots, C_r$ be the connected components of $\mathcal{C}(H) \backslash V(T)$ adjacent to a and Y the union of these connected components. Observe that each C_i has to be an union of 1-block-branches and their attachment points form a clique Q_i. Let $B_1, \ldots, B_l$ with attachment points $P_1, \ldots, P_l$ be the 1-block-branches of Y. By Rule 4, there are at most $2k + 1$ 1-block-branches by connected component and if $\Sigma_{i=1}^{l} |P_i| > 2k + 1$, then $(\Sigma_{i=1}^{l} |P_i|) - max_{1 \leq j \leq r}\{|Q_j|\} \leq k$, implying that there are at most $3k + 1$ 1-block-branches in C. Each of these 1-block-branches is reduced by Rule 3, hence Y contains at most $6k + 2$ critical cliques in total.

We conclude that $|V(\mathcal{C}(H))| = O(k^2)$ and by Lemma 20, we have $|V(\mathcal{C}(G))| \leq |V(\mathcal{C}(H))| + 2k$, therefore by Rule 2, $|V(G)| = O(k^3)$. To conclude, we claim that a reduced instance can be computed in polynomial time. Indeed, Lemma 8 states that it is possible to reduce exhaustively a graph under Rules 1 and 2. Once this is done, it remains to apply exhaustively Rules 3 to 5 which is ensured by Lemma 18. ◀

5 Kernels for edition and deletion

In this section we present the kernel for Strictly Chordal Editing, all the following arguments also hold for Strictly Chordal Deletion. It is clear that Rules 1 and 2 are also safe for Strictly Chordal Editing, as well as Lemma 11. Lemma 12 also holds although the proof needs to be adapted to take in consideration the possibility of disconnecting the attachment point of a 1-block-branch B and B^R. It follows that Rule 3 is safe for the edition version. Next, Lemma 14 holds for edition with similar arguments for the

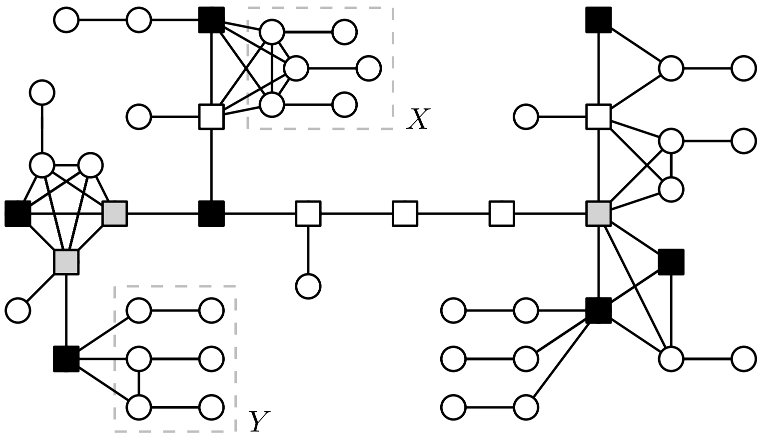

Figure 4 Illustration of the critical clique graph of an reduced instance in the proof of Theorem 21. Square nodes correspond to vertices of T, the ones filled in black are vertices of A, the ones filled in grey are vertices of $A' \backslash A$.

proof, implying the safeness of Rule 4 for edition. The main difference lies in Rule 5 handling 2-block-branches. Indeed, in this case, an optimal edition may affect vertices that are not contained in an attachment point nor their neighborhood in the branch. However, this case can be dealt with with more intricate arguments (Rule 6).

Finally, we can observe that for a clean 2-block-branch B of some graph G, an edition can remove edges and disconnect B. We thus have to consider 2-block-branches whose attachment points lie in the same connected component of $G \backslash V(B^R)$. We hence have to take this into consideration for our new reduction rule. To that aim, we use a minimum-sized (P_1, P_2)-cut of B where P_1 and P_2 are its attachment points. More precisely, we define *min-cut*(B) as a set $M \subseteq E(B)$ of minimum size such that P_1 and P_2 are not in the same connected component of $B - M$. We can observe that *min-cut*(B) contains the edges between a pair of consecutive critical cliques of a shortest path between P_1 and P_2 and the edges between one of these critical cliques and the critical cliques they have in their common neighborhood.

▶ **Observation 22.** *Let F be an optimal edition of G and $F_1 \subseteq F$. If F_2 is an optimal edition of the graph $G \triangle F_1$, then $F_1 \cup F_2$ is an optimal edition of G.*

▶ **Lemma 23.** *Let $G = (V, E)$ be graph and B a clean 2-block-branch of length at least $k + 4$ with attachment points P_1, P_2. There exists an optimal edition F of G such that:*

- *If P_1 and P_2 are not in the same connected component of $B \triangle F$, then F may contain edges of $min\text{-}cut(B)$.*
- *In each case, the other vertices of B affected by F are included in $P_1 \cup N_B(P_1) \cup P_2 \cup N_B(P_2)$,*
- *In $G \triangle F$ the vertices of $N_B(P_1)$ (resp. $N_B(P_2)$) are all adjacent to the same vertices of $V(G) \backslash V(B^R)$.*

Proof. Let F be an optimal edition of G, $H = G \triangle F$. Let C_1 (resp. C_2) be the critical clique of H which contains P_1 (resp. P_2), $C_1' = C_1 \backslash (B^R)$ and $C_2' = C_2 \backslash (B^R)$. We first consider the case where B is not connected in $B \triangle F$. If $F_1 = (P_1 \times N_B(P_1)) \subseteq F$, consider the graph $G_1 = G \triangle F_1$. Observe that $B_1 = B \backslash P_1$ is a 1-block-branch of G_1 with attachment point P_2. By Lemma 12 there exists an optimal edition F_2 of G_1 where the vertices of B_1 affected by F_2 are in $P_2 \cup N_{B_1}(P_2)$. By Observation 22, $F_1 \cup F_2$ is an optimal edition and it satisfies the desired properties. The same arguments can be used if $(P_2 \times N_B(P_2)) \subseteq F$.

If F contains neither $(P_1 \times N_B(P_1))$ nor $(P_2 \times N_B(P_2))$, since F is optimal, it must contain $F_1 = \mathit{min\text{-}cut}(B)$. We consider the graph $G_1 = G \triangle F_1$, there are two components in $G_1[V(B)]$, say B_1 the one containing P_1 and B_2 the other one containing P_2. These connected components are 1-block-branches with attachment points P_1 and P_2. As before Lemma 12 applies on the 1-block-branches B_1 and B_2, thus there is an optimal edition F_2 of G_1 where the only vertices of B_1 and B_2 affected are $(P_1 \times N_B(P_1))$ and $(P_2 \times N_B(P_2))$. Thus, the edition $F_1 \cup F_2$ is an optimal edition of G that respects the desired properties.

In the case where B is connected in $B \triangle F$, observe that C'_1 and C'_2 have to be in different connected components of $H \backslash V(B^R)$. Otherwise there would be a shortest path p_H from $c_1 \in C'_1$ to $c_2 \in C'_2$ in $H \backslash V(B^R)$ (of length potentially 0 if $c_1 = c_2$) and a shortest path p_B from c_1 to c_2 in B of length at least $k+4$. There is no edge between the two paths in G, so $H[V(p_B) \cup V(p_H)]$ admits at least one cycle of length at least 4, contradiction. Since B^R is connected, by Lemma 10, $N_B(P_1)$ and $N_B(P_2)$ are cliques in G, $N_B(P_1)$ and $N_B(P_2)$ are critical cliques of B^R or each are in exactly one maximal clique of B^R. By Lemma 11, C'_1 and C'_2 are either critical cliques of $H' = H \backslash V(B^R)$ or are in exactly one maximal clique of the connected component H'_1 (resp. H'_2) of H' which contains C'_1 (resp. C'_2). Let H_3 be the union of connected components of H that do not contain B. By Lemma 4, the graph H^* corresponding to the disjoint union of $((B^R, N_B(P_1)) \otimes (H'_1, C'_1)), N_B(P_2)) \otimes (H'_2, C'_2)$ and H_3 is strictly chordal and the completion F^* such that $H^* = G + F^*$ respects the desired properties. ◀

▶ **Rule 6.** *Let $(G = (V, E), k)$ be an instance of* Strictly Chordal Editing *and B a clean 2-block-branch of G of length at least $k+4$ with attachment points P_1, P_2. Then remove the vertices of B^R, and add the following path of $k+5$ cliques:*

$$K_{min\{|N_B(P_1)|,k+1\}} - K_{k+1} - K_1 - K_{|min\text{-}cut(B)|} - K^1_{k+1} - \cdots - K^k_{k+1} - K_{min\{|N_B(P_2)|,k+1\}}$$

where K_n is the clique of size n and $K_{min\{|N_B(P_1)|,k+1\}}$ (resp. $K_{min\{|N_B(P_2)|,k+1\}}$) is adjacent to P_1 (resp. P_2).

▶ **Lemma 24** ($\star$)**.** *Rule 6 is safe.*

Notice that Lemma 18 allows to detect any clean 2-block-branch. For the size of the kernel, the proof is similar, however, in this case the paths are of length $O(k)$, thus there are $O(k^2)$ vertices and maximal cliques in the inclusion-minimal subgraph spanning the affected critical cliques. Thus a reduced graph contains $O(k^4)$ vertices.

▶ **Theorem 25.** Strictly Chordal Editing *and* Strictly Chordal Deletion *admits a kernel with $O(k^4)$ vertices.*

6 Conclusion

We presented polynomial size kernels for the three variants of strictly chordal edge modification problems. Our conviction is that the approach based on decompositions of the target class, combined with the ability of reducing the size of the bags of the decomposition and of limiting the number of affected bags to $O(k)$ is a promising starting point for edge modification problems, especially into subclasses of chordal graphs. The technique has been employed especially for classes that admit a tree-like decompositions with disjoint bags (e.g., 3-leaf power and trivially perfect graphs [3, 17]), also for other types of tree-like decompositions with non-disjoint bags (e.g., ptolemaic graphs [12]). We generalize it here to scrictly chordal graphs, that have a decomposition into disjoint bags as nodes of a block graph.

The difficulty is that, at this stage, each class requires ad-hoc arguments and reduction rules, based on its specific decomposition. An ambitious goal would be to obtain a generic algorithm for edge modification problems into any class of chordal graphs, plus a finite set of obstructions, as conjectured by Bessy and Perez [4]. We also ask whether 4-leaf power completion, deletion and editing problems admit a polynomial kernel.

References

1. Hans-Jürgen Bandelt and Henry Martyn Mulder. Distance-hereditary graphs. *J. Comb. Theory, Ser. B*, 41(2):182–208, 1986. `doi:10.1016/0095-8956(86)90043-2`.
2. Gabriel Bathie, Nicolas Bousquet, and Théo Pierron. (sub)linear kernels for edge modification problems towards structured graph classes. *CoRR*, abs/2105.09566, 2021. `arXiv:2105.09566`.
3. Stéphane Bessy, Christophe Paul, and Anthony Perez. Polynomial kernels for 3-leaf power graph modification problems. *Discret. Appl. Math.*, 158(16):1732–1744, 2010. `doi:10.1016/j.dam.2010.07.002`.
4. Stéphane Bessy and Anthony Perez. Polynomial kernels for proper interval completion and related problems. *Inf. Comput.*, 231:89–108, 2013. `doi:10.1016/j.ic.2013.08.006`.
5. Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *J. Comput. Syst. Sci.*, 75(8):423–434, 2009. `doi:10.1016/j.jcss.2009.04.001`.
6. Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM J. Discret. Math.*, 28(1):277–305, 2014. `doi:10.1137/120880240`.
7. Hans L. Bodlaender, Stéphan Thomassé, and Anders Yeo. Kernel bounds for disjoint cycles and disjoint paths. *Theor. Comput. Sci.*, 412(35):4570–4578, 2011. `doi:10.1016/j.tcs.2011.04.039`.
8. Leizhen Cai. Fixed-parameter tractability of graph modification problems for hereditary properties. *Inf. Process. Lett.*, 58(4):171–176, 1996. `doi:10.1016/0020-0190(96)00050-6`.
9. Leizhen Cai and Yufei Cai. Incompressibility of h-free edge modification problems. *Algorithmica*, 71(3):731–757, 2015. `doi:10.1007/s00453-014-9937-x`.
10. Yixin Cao and Dániel Marx. Chordal editing is fixed-parameter tractable. *Algorithmica*, 75(1):118–137, 2016. `doi:10.1007/s00453-015-0014-x`.
11. Christophe Crespelle, Pål Grønås Drange, Fedor V. Fomin, and Petr A. Golovach. A survey of parameterized algorithms and the complexity of edge modification. *CoRR*, abs/2001.06867, 2020. `arXiv:2001.06867`.
12. Christophe Crespelle, Benjamin Gras, and Anthony Perez. Completion to chordal distance-hereditary graphs: a quartic vertex-kernel. Accepted for publication at International Workshop on Graph-Theoretic Concepts in Computer Science, 2021.
13. Marek Cygan, Fedor V Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*, volume 5. Springer, 2015.
14. Michael Dom, Jiong Guo, Falk Hüffner, and Rolf Niedermeier. Error compensation in leaf power problems. *Algorithmica*, 44(4):363–381, 2006. `doi:10.1007/s00453-005-1180-z`.
15. Michael Dom, Jiong Guo, Falk Hüffner, and Rolf Niedermeier. Closest 4-leaf power is fixed-parameter tractable. *Discret. Appl. Math.*, 156(18):3345–3361, 2008. `doi:10.1016/j.dam.2008.01.007`.
16. Pål Grønås Drange and Michał Pilipczuk. A polynomial kernel for trivially perfect editing. *Algorithmica*, 80(12):3481–3524, 2018. `doi:10.1007/s00453-017-0401-6`.
17. Maël Dumas, Anthony Perez, and Ioan Todinca. A cubic vertex-kernel for trivially perfect editing. *CoRR*, abs/2105.08549, 2021. `arXiv:2105.08549`.
18. Martin Charles Golumbic and Uri N. Peled. Block duplicate graphs and a hierarchy of chordal graphs. *Discret. Appl. Math.*, 124(1-3):67–71, 2002. `doi:10.1016/S0166-218X(01)00330-4`.

19 Sylvain Guillemot, Frédéric Havet, Christophe Paul, and Anthony Perez. On the (non-)existence of polynomial kernels for P_l-free edge modification problems. *Algorithmica*, 65(4):900–926, 2013. doi:10.1007/s00453-012-9619-5.

20 Jiong Guo. Problem kernels for np-complete edge deletion problems: Split and related graphs. In Takeshi Tokuyama, editor, *Algorithms and Computation, 18th International Symposium, ISAAC 2007, Sendai, Japan, December 17-19, 2007, Proceedings*, volume 4835 of *Lecture Notes in Computer Science*, pages 915–926. Springer, 2007. doi:10.1007/978-3-540-77120-3_79.

21 Haim Kaplan, Ron Shamir, and Robert Endre Tarjan. Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs. *SIAM J. Comput.*, 28(5):1906–1922, 1999. doi:10.1137/S0097539796303044.

22 William Kennedy. Strictly chordal graphs and phylogenetic roots. Master's thesis, University of Alberta, 2005.

23 William Kennedy, Guohui Lin, and Guiying Yan. Strictly chordal graphs are leaf powers. *Journal of Discrete Algorithms*, 4(4):511–525, 2006. doi:10.1016/j.jda.2005.06.005.

24 Stefan Kratsch and Magnus Wahlström. Two edge modification problems without polynomial kernels. *Discret. Optim.*, 10(3):193–199, 2013. doi:10.1016/j.disopt.2013.02.001.

25 Mirko Křivánek and Jaroslav Morávek. Np-hard problems in hierarchical-tree clustering. *Acta Informatica*, 23(3):311–323, 1986. doi:10.1007/BF00289116.

26 Lilian Markenzon and Christina Fraga Esteves Maciel Waga. New results on ptolemaic graphs. *Discret. Appl. Math.*, 196:135–140, 2015. doi:10.1016/j.dam.2014.03.024.

27 Assaf Natanzon. *Complexity and approximation of some graph modification problems.* University of Tel-Aviv, 1999.

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

29 Fábio Protti, Maise Dantas da Silva, and Jayme Luiz Szwarcfiter. Applying modular decomposition to parameterized cluster editing problems. *Theory Comput. Syst.*, 44(1):91–104, 2009. doi:10.1007/s00224-007-9032-7.

30 Ron Shamir, Roded Sharan, and Dekel Tsur. Cluster graph modification problems. *Discret. Appl. Math.*, 144(1-2):173–182, 2004. doi:10.1016/j.dam.2004.01.007.