

# Approximation and kernelization for chordal vertex deletion

**Citation for published version (APA):**

Jansen, B. M. P., & Pilipczuk, M. (2018). Approximation and kernelization for chordal vertex deletion. *SIAM Journal on Discrete Mathematics*, 32(3), 2258-2301. <https://doi.org/10.1137/17M112035X>

**DOI:**

[10.1137/17M112035X](https://doi.org/10.1137/17M112035X)

**Document status and date:**

Published: 10/07/2018

**Document Version:**

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

**Please check the document version of this publication:**

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

**Take down policy**

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.

## APPROXIMATION AND KERNELIZATION FOR CHORDAL VERTEX DELETION\*

BART M. P. JANSEN<sup>†</sup> AND MARCIN PILIPCZUK<sup>‡</sup>

**Abstract.** The CHORDAL VERTEX DELETION (CHVD) problem asks to delete a minimum number of vertices from an input graph to obtain a chordal graph. In this paper we develop a polynomial kernel for CHVD under the parameterization by the solution size. Using a new Erdős–Pósa-type packing/covering duality for holes in nearly chordal graphs, we present a polynomial-time algorithm that reduces any instance  $(G, k)$  of CHVD to an equivalent instance with  $\text{poly}(k)$  vertices. The existence of a polynomial kernel answers an open problem posed by Marx in 2006 [D. Marx, “Chordal Deletion Is Fixed-Parameter Tractable,” in *Graph-Theoretic Concepts in Computer Science*, Lecture Notes in Comput. Sci. 4271, Springer, 2006, pp. 37–48]. To obtain the kernelization, we develop the first  $\text{poly}(\text{OPT})$ -approximation algorithm for CHVD, which is of independent interest. In polynomial time, it either decides that  $G$  has no chordal deletion set of size  $k$ , or outputs a solution of size  $\mathcal{O}(k^4 \log^2 k)$ .

**Key words.** kernelization algorithm, approximation algorithm, chordal vertex deletion

**AMS subject classifications.** 05C85, 68Q25

**DOI.** 10.1137/17M112035X

**1. Introduction.** Many important combinatorial problems can be phrased in terms of vertex deletions: given an input graph  $G$  and an integer  $k$ , is it possible to delete at most  $k$  vertices from  $G$  to ensure the resulting graph belongs to a graph class  $\mathcal{G}$ ? For example, VERTEX COVER asks to delete at most  $k$  vertices to obtain an edgeless graph, while in (undirected) FEEDBACK VERTEX SET one deletes vertices to obtain an acyclic graph. By a classic result of Lewis and Yannakakis [49], the  $\mathcal{G}$  VERTEX DELETION problem is NP-hard for all nontrivial hereditary graph classes  $\mathcal{G}$ . This motivated intensive research into alternative algorithmic approaches to deal with these problems, such as polynomial-time approximation algorithms and fixed-parameter tractable (exact) algorithms, which test for a solution of size  $k$  in time  $f(k) \cdot n^{\mathcal{O}(1)}$  for some arbitrary function  $f$ . The notion of kernelization, which formalizes efficient and effective preprocessing, has also been used to cope with the NP-hardness of vertex deletion problems. A *kernelization* (or *kernel*) for a parameterized graph problem is a polynomial-time preprocessing algorithm that reduces any instance  $(G, k)$  to an equivalent instance  $(G', k')$  whose total size is bounded by  $g(k)$  for some function  $g$ , which is the *size* of the kernel. While all fixed-parameter

---

\*Received by the editors March 9, 2017; accepted for publication (in revised form) July 10, 2018; published electronically September 11, 2018. An extended abstract of this work appeared in *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms* (SODA 2017).

<http://www.siam.org/journals/sidma/32-3/M112035.html>

**Funding:** The majority of this work was done while the authors were participating in the “Fine-Grained Complexity and Algorithm Design” program at the Simons Institute for the Theory of Computing in Berkeley. The work of the first author was supported by the Netherlands Organisation for Scientific Research, Veni grant “Frontiers in Parameterized Preprocessing,” and Gravitation grant “Networks.” The work of the second author was supported by Polish National Science Centre grant DEC-2012/05/D/ST6/03214.

<sup>†</sup>Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven 5600 MB, The Netherlands (b.m.p.jansen@tue.nl, <https://www.win.tue.nl/~bjansen/>).

<sup>‡</sup>Institute of Informatics, University of Warsaw, Warsaw 02-097, Poland (marcin.pilipczuk@mimuw.edu.pl, <https://www.mimuw.edu.pl/~malcin/>).

tractable problems have a kernel, a central challenge is to identify problems for which the size bound  $g(k)$  can be made polynomial, a so-called *polynomial kernel*; we refer the reader to surveys [46, 50, 51] for recent examples.

The paradigm of parameterized complexity fits graph modification problems particularly well. Assume that an input graph  $G$  represents results of an experiment or measurement. The experimental setup could result in the fact that  $G$  has certain properties, and therefore belongs to a specific graph class  $\mathcal{G}$ , if the measurements were completely error-free. Due to measurement errors, graph  $G$  might not be in  $\mathcal{G}$ , but close to a member of  $\mathcal{G}$ . Graph modification problems then capture the task of detecting and filtering out the errors. If the number of measurement errors is small, the solution size of the corresponding modification problem is small; an ideal target for parameterized algorithms.

The complexity of VERTEX COVER and FEEDBACK VERTEX SET is well understood with respect to all three aforementioned algorithmic paradigms: approximation algorithms, fixed-parameter algorithms, and polynomial kernelization. In this paper we analyze a generalization that has remained much more elusive: deleting vertices to make a graph *chordal*, i.e., to ensure the resulting graph contains no induced cycles of length at least four (*holes*). While VERTEX COVER and FEEDBACK VERTEX SET have long been known to be fixed-parameter tractable [10, 16], to have constant-factor approximation algorithms [4, 7, 54], and to have kernels of polynomial size [12, 15, 54, 56], our understanding of CHORDAL VERTEX DELETION (CHVD) is considerably lacking. The problem is known to be fixed-parameter tractable by a result of Marx [53], and the current-best runtime is  $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$  [19]. The kernelization and approximation complexity of CHVD has remained unresolved, however. The question of whether CHVD has a polynomial kernel or not has been open for 10 years [52] and has been posed by several groups of authors [19, 32, 40] and during the 2013 Workshop on Kernelization [22]. The approximability of CHVD is similarly open. To the best of our knowledge, nothing was known prior to this work.

**Results.** We resolve the kernelization complexity of CHVD and show that the problem has a kernel of polynomial size. Our main data reduction procedure is summarized by the following theorem.

**THEOREM 1.1.** *There is a polynomial-time algorithm that, given an instance  $(G, k)$  of CHORDAL VERTEX DELETION and a modulator  $M_0 \subseteq V(G)$  such that  $G - M_0$  is chordal, outputs an equivalent instance  $(G', k')$  with  $\mathcal{O}(k^{45} \cdot |M_0|^{29})$  vertices and  $k' \leq k$ .*

To be able to apply this reduction algorithm, a modulator  $M_0$  is needed to reveal the structure of the chordal graph  $G - M_0$ . If CHVD would have a polynomial-time constant-factor approximation algorithm, or even a poly(OPT)-approximation algorithm, one could bootstrap Theorem 1.1 with an approximate solution as modulator to obtain a polynomial kernel for CHVD. Prior to this work, no such approximation algorithm was known. We developed a poly(OPT)-approximation algorithm to complete the kernelization, which we believe to be of independent interest.

**THEOREM 1.2.** *There is a polynomial-time algorithm that, given an undirected graph  $G$  and an integer  $k$ , either correctly concludes that  $(G, k)$  is a no-instance of CHVD or computes a set  $X \subseteq V(G)$  of size  $\mathcal{O}(k^4 \log^2 k)$  such that  $G - X$  is chordal.*

Combining the two theorems, we obtain a kernel with  $\mathcal{O}(k^{161} \log^{58} k)$  vertices for CHVD. To appreciate the conceptual difficulties in obtaining a polynomial kernel for CHVD, let us point out why some popular kernelization strategies fail. In general, the difficulty of the  $\mathcal{G}$  VERTEX DELETION problem is tightly linked to the amount of structure of graphs in  $\mathcal{G}$ . If all graphs in  $\mathcal{G}$  have bounded treewidth, and  $\mathcal{G}$  is

closed under taking minors, then one can obtain a kernel by repeatedly replacing large *protrusions* [11, 31] in the graph by smaller gadgets. Since edgeless graphs and acyclic graphs are minor-closed and have constant treewidth, this explains the existence of kernels for VERTEX COVER and FEEDBACK VERTEX SET. This approach does not work for CHVD, as chordal graphs can have arbitrarily large treewidth (every complete graph is chordal). Another kernelization strategy for vertex deletion problems into hereditary classes  $\mathcal{G}$  is to model the problem as an instance of HITTING SET and apply a kernelization strategy based on the Sunflower Lemma (cf. [29, section 9.1]). This yields polynomial kernels when there is a characterization of  $\mathcal{G}$  in terms of a finite set of forbidden induced subgraphs, such as  $P_4$ -free graphs (equivalently, cographs). However, the number of forbidden induced subgraphs for chordal graphs is infinite as this set contains all induced cycles of length at least four, making this approach infeasible as well.

A key ingredient in our kernelization is an algorithmic version of Erdős–Pósa-type [26] covering/packing duality for holes in chordal graphs, expressed in terms of *flowers*. A  $v$ -flower of order  $\ell$  in graph  $G$  is a set  $\mathcal{C}$  of  $\ell$  holes  $\mathcal{C} = \{C_1, \dots, C_\ell\}$  in  $G$ , such that  $V(C_i) \cap V(C_j) = \{v\}$  for all  $i \neq j$ .

LEMMA 1.3. *There is a polynomial-time algorithm that, given a graph  $G$  and a vertex  $v$  such that  $G - v$  is chordal, outputs a  $v$ -flower  $C_1, \dots, C_\ell$  and a set  $S \subseteq V(G) \setminus \{v\}$  of size at most  $12\ell$  such that  $G - S$  is chordal.*

So there is a large packing of holes intersecting only in  $v$ , or all the holes can be intersected by a small set of vertices. We prove Lemma 1.3 by presenting a local search algorithm that builds a flower, followed by a greedy algorithm to construct a hitting set when the local search cannot make further progress. This procedure forms the first step in a series of arguments that *tidy the modulator* (in the sense of [58]) and narrow down the structure of the graph, which enables us to identify irrelevant parts of the graph which can be reduced. Section 3 describes the intuition behind the kernelization in more detail.

When it comes to the approximation algorithm, we combine linear-programming rounding techniques with structural insights into the behavior of nearly chordal graphs. Since chordal graphs have balanced clique separators (see, e.g., [38, Theorem 1] or [21, Lemma 7.19]), if  $(G, k)$  is a yes-instance of CHVD, then there is a balanced separator in  $G$  consisting of a clique plus  $k$  vertices. We use an approximation algorithm for minimum-weight vertex separators of Feige, Hajiaghayi, and Lee [28] together with an enumeration of maximal cliques to decompose the graph. This decomposition step reduces the problem of finding an approximate solution to the setting in which we have an input graph  $G$  and a clique  $X$  such that  $G - X$  is chordal, and the aim is to find a small chordal deletion set in  $G$  (smaller than  $|X|$ ). After a second decomposition step, an analysis of the structure of holes allows the problem to be phrased in terms of multicut on a directed graph, to which we apply a result of Gupta [39].

**Motivation.** The parameterized complexity of graph modification problems to chordal graphs and related graph classes gained significant interest in recent years. Several factors explain the popularity of such problems. First, chordal graphs and related graph classes, such as interval graphs, appear naturally in applications in solving linear equations [14], in scheduling [5], and in bioinformatics [55, 59]. Second, from a more theoretical perspective, these graph classes are closely related to widely used graph parameters, such as treewidth: a tree decomposition can be seen as a completion to a chordal graph, and similar relations hold between interval graphs

and pathwidth, proper interval graphs and bandwidth, or trivially perfect graphs and treedepth (cf. [25]). Third, to this day, we do not know of a vertex deletion problem that is fixed-parameter tractable (FPT) but has no polynomial kernel. (We do know of some that are not FPT, such as PERFECT DELETION [40].) It is tempting to conjecture that all FPT vertex deletion problems have polynomial kernels. CHVD is a candidate for refuting this conjecture, and resolving its complexity status gives us more insight into the landscape of kernelization complexity of vertex deletion problems. Fourth, as witnessed by numerous works in the past years [8, 9, 18, 20, 19, 25, 32, 42], most graph modification problems to chordal graphs and related graph classes turn out to be tractable but very challenging; the corresponding algorithmic results build upon deep graph-theoretical insights into the studied graph class, making this study particularly interesting from the point of view of graph theory.

One motivation for studying the particular case of vertex deletion to a chordal graph comes from the following fact: given a graph  $G$  and a modulator  $M$  such that  $G - M$  is chordal, one can find a maximum independent set, maximum clique, and minimum vertex cover of  $G$  in time  $2^{|M|} \cdot n^{\mathcal{O}(1)}$  (cf. [17, section 6]). Hence an efficient algorithm for chordal vertex deletion allows these fundamental optimization problems to be solved efficiently on nearly chordal graphs.

**Related work.** Kernels are known for a wide variety of vertex deletion problems [31, 32, 37, 42, 43, 48, 54, 56]. The recent kernels for ODD CYCLE TRANSVERSAL [48] and PROPER INTERVAL VERTEX DELETION [32, 42] are noteworthy since, like chordal graphs, bipartite graphs and proper interval graphs do not have bounded treewidth and do not have a characterization by a finite set of forbidden induced subgraphs. There are linear kernels for CHVD on planar graphs [11] and  $H$ -topological-minor-free graphs [45]. Concerning nonstandard parameterizations, such as structural ones, the CHVD problem is known to have polynomial kernels when parameterized by the size of a vertex cover [30], when parameterized by the size of a set of candidate vertices from which the solution is required to be selected [40], and on bounded-expansion graphs when parameterized by a constant-treedepth modulator [33]. Approximation algorithms have been developed for (weighted) FEEDBACK VERTEX SET [4, 7], ODD CYCLE TRANSVERSAL [35], and its edge deletion variant [1].

Several recent kernelization results rely on matroid theory [47, 48] (resulting in randomized kernels), or on *finite integer index*, or the Graph-Minor Theorem [11, 31, 33, 45] (resulting in proofs that kernelization algorithms exist without showing how they should be constructed). In contrast, our kernelization algorithm is deterministic and fully explicit; it is based on new graph-theoretical insights into the structure of chordal graphs.

**Organization.** We present preliminaries on graphs in section 2. Section 3 contains an informal overview of the data reduction procedure. Section 4 is devoted to the proof of Lemma 1.3. We introduce an annotated version of CHVD in section 5, which captures some of the structure we can obtain in the instance by appropriate use of Lemma 1.3. Sections 6–9 present reduction rules for instances of the annotated problem. The approximation algorithm is given in section 10, which allows us to give the final kernel in section 11.

**2. Preliminaries.** The set  $\{1, \dots, n\}$  is abbreviated as  $[n]$ . A set  $S$  *avoids* a set  $U$  if  $S \cap U = \emptyset$ . For a set  $U$ , by  $\binom{U}{2}$  we denote the family of all unordered pairs of distinct members of  $U$ . We often use  $xy$  instead of  $\{x, y\}$  for an element of  $\binom{U}{2}$ . A graph  $G$  consists of a vertex set  $V(G)$  and edge set  $E(G) \subseteq \binom{V(G)}{2}$ . For

a vertex  $v \in V(G)$  its open neighborhood is  $N_G(v) := \{u \in V(G) \mid uv \in E(G)\}$  and its closed neighborhood is  $N_G[v] := N_G(v) \cup \{v\}$ . The open neighborhood of a set  $S \subseteq V(G)$  is  $N_G(S) := \bigcup_{v \in S} N_G(v) \setminus S$ , and the closed neighborhood is  $N_G[S] := \bigcup_{v \in S} N_G[v]$ . The graph obtained from  $G$  by deleting all vertices in the set  $S$  and their incident edges is denoted  $G - S$ ; for a single vertex  $v \in V(G)$  we use  $G - v$  as shorthand for  $G - \{v\}$ .

An *st-walk* in a graph  $G$  is a sequence of vertices  $s = v_1, v_2, \dots, v_\ell = t$  such that  $v_i = v_{i+1}$  or  $v_i v_{i+1} \in E(G)$  for  $1 \leq i < \ell$ . An *st-path* is an *st-walk* on which all vertices are distinct. For a walk  $W$  or path  $P$  we use  $V(W)$  and  $V(P)$  to denote the vertices of  $W$  and  $P$ , respectively. For an *st-path*  $P$ , the vertices  $s$  and  $t$  are its *endpoints* and  $V(P) \setminus \{s, t\}$  are its *interior vertices*.

**OBSERVATION 2.1.** *For any st-walk  $W$  in a graph  $G$ , any shortest st-path in  $G[V(W)]$  is an induced st-path  $P$  in  $G$  with  $V(P) \subseteq V(W)$ .*

A *hole* is an induced (i.e., chordless) cycle of length at least four. A graph is *chordal* if it does not contain any holes; we say that such graphs are *hole-free*. The *independence number* of a graph  $G$ , denoted  $\alpha(G)$ , is the maximum cardinality of a set of vertices in  $G$  in which no pair is connected by an edge. Similarly, the *clique number*  $\omega(G)$  is the maximum cardinality of a set of vertices that are pairwise all adjacent.

**PROPOSITION 2.1.** *Any chordal graph  $G$  satisfies  $\alpha(G) \geq |V(G)|/\omega(G)$ .*

*Proof.* As any chordal graph is perfect [13], the chromatic number of  $G$  equals  $\omega(G)$ . A proper coloring with  $\omega(G)$  colors is a partition into  $\omega(G)$  independent sets. At least one color class has size  $|V(G)|/\omega(G)$ , guaranteeing an independent set of that size.  $\square$

**Tree decompositions.** A *tree decomposition* of a graph  $G$  is a pair  $(T, \beta)$ , where  $T$  is a tree and  $\beta: V(T) \rightarrow 2^{V(G)}$  assigns to every node  $p \in V(T)$  a subset of the vertices of  $G$ , called a *bag*, such that the following hold:

1.  $\bigcup_{p \in V(T)} \beta(p) = V(G)$ .
2. For each edge  $uv \in E(G)$ , there is a node  $p \in V(T)$  such that  $u, v \in \beta(p)$ .
3. For each  $v \in V(G)$ , the nodes of  $T$  whose bag contains  $v$  induce a connected subtree of  $T$ .

For a vertex  $v \in V(G)$  we use  $\beta^{-1}(v)$  to denote the set of nodes of  $T$  whose bag contains  $v$ . The third property ensures that  $\beta^{-1}(v)$  is a subtree of  $T$  for all  $v \in V(G)$ . The *adhesion* of an edge  $e = pq$  of the decomposition tree is  $\text{ADH}(e) := \beta(p) \cap \beta(q)$ .

Very often in our work, the tree  $T$  will be rooted. If this is the case, we use  $T_p$  to denote the subtree of  $T$  rooted at  $p \in V(T)$ . For a rooted tree decomposition  $(T, \beta)$  of  $G$  and a vertex  $v \in V(G)$ , define  $\text{TOP}(v)$  to be the node  $p \in V(T)$  whose bag  $\beta(p)$  contains  $v$  and whose distance in  $T$  to the root node is minimum. The third property of tree decompositions ensures that this is well-defined. For two nodes  $p, q \in V(T)$ , we use  $\text{LCA}(p, q)$  to denote the lowest common ancestor of  $p$  and  $q$  in  $T$ .

For a graph  $G$  and two vertex sets  $X$  and  $Y$ , we use  $d_G(X, Y)$  to denote the number of edges on a shortest path from a vertex in  $X$  to a vertex in  $Y$  (or  $+\infty$  if such a path does not exist). When  $(T, \beta)$  is a tree decomposition of  $G$  and  $s, t \in V(G)$ , this means that  $d_T(\beta^{-1}(s), \beta^{-1}(t))$  is the distance between the subtree where  $s$  is represented and the subtree where  $t$  is represented. A path  $P$  in  $G$  *connects* vertex sets  $X$  and  $Y$  if one endpoint of  $P$  belongs to  $X$  and the other endpoint belongs to  $Y$ . It is a *minimal* path connecting  $X$  and  $Y$  when no interior vertex belongs to  $X$  or  $Y$ . Observe that for disjoint connected vertex subsets  $X$  and  $Y$  of a tree  $T$ , the minimal path connecting  $X$  and  $Y$  is unique. This implies that the minimal path

connecting  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  is unique when  $s$  and  $t$  do not appear in the same bag of a tree decomposition  $(T, \beta)$  of  $G$ . We use  $\Pi_T(s, t)$  to refer to a minimal path connecting  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  in  $T$ . We also use  $\Pi_T(s, \text{ROOT})$  to refer to the path from  $\text{TOP}(s)$  to the root of  $T$ , which is the minimal path connecting  $\beta^{-1}(s)$  to the root node.

**OBSERVATION 2.2.** *Let  $(T, \beta)$  be a tree decomposition of graph  $G$ . If  $s, t$  are distinct vertices of  $G$  and  $\Pi$  is a minimal path in  $T$  connecting  $\beta^{-1}(s)$  to  $\beta^{-1}(t)$ , then for every edge  $e$  on  $\Pi$  all  $st$ -paths in  $G$  contain a vertex of  $\text{ADH}(e)$ .*

**OBSERVATION 2.3.** *Let  $(T, \beta)$  be a tree decomposition of graph  $G$  and let  $p \in V(T)$ . If  $S \subseteq V(G)$  such that  $G[S \setminus \beta(p)]$  is connected, then there is a single tree  $T'$  in the forest  $T - p$  such that all bags containing a vertex of  $S \setminus \beta(p)$  are contained in  $T'$ .*

**Clique trees.** A *clique tree* is a tree decomposition  $(T, \beta)$  such that each bag  $\beta(p)$  is a maximal clique in  $G$ . It is known that a graph is chordal if and only if it has a clique tree [13, Theorem 1.2.3]. A clique tree of a chordal graph can be found in linear time [13, Theorem 1.2.4]. The definition implies that if  $(T, \beta)$  is a clique tree of  $G$  and  $S \subseteq V(G)$  is a clique in  $G$ , then  $S \subseteq \beta(p)$  for some node  $p \in V(T)$ .

Note that in a chordal graph  $G$  with a clique tree  $(T, \beta)$ , two vertices  $s, t$  appear in the same bag if and only if they are adjacent. Thus, in a chordal graph  $G$  with a clique tree  $(T, \beta)$ , there is a unique minimal path connecting  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  for every nonadjacent pair of vertices  $s, t$ .

In our proofs we often work with a clique tree  $(T, \beta)$  of a chordal graph  $G$ , and we will need to refer to both paths in  $G$  and paths in  $T$ . To maximize readability, we use the convention that identifiers such as  $P$  and  $P'$  are used for paths in  $G$ , while identifiers such as  $\Pi$  and  $\Pi'$  are used for paths in the decomposition tree  $T$ .

**PROPOSITION 2.2.** *Let  $(T, \beta)$  be a clique tree of a graph  $G$ , let  $U \subseteq V(G)$ , and let  $s$  and  $t$  be distinct vertices in  $V(G) \setminus U$ . If there is a path  $\Pi$  in  $T$  connecting  $\beta^{-1}(s)$  to  $\beta^{-1}(t)$  such that  $\text{ADH}(e) \setminus U \neq \emptyset$  for each edge  $e$  on  $\Pi$ , then there is an induced  $st$ -path in  $G - U$ .*

*Proof.* If  $st \in E(G)$ , the claim is trivial. Otherwise, the trees  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  are vertex-disjoint since each bag of the clique tree forms a clique in  $G$ . Consider a path  $\Pi$  as described, consisting of edges  $e_1, \dots, e_\ell$  with  $\ell \geq 1$ . For each edge  $e_i$  with  $i \in [\ell]$  pick a vertex  $v_i \in \text{ADH}(e_i) \setminus U$  and note that, by definition of adhesion,  $v_i$  occurs in the bags of both endpoints of  $e_i$ . As  $v_1$  is in a common bag with  $s$  and each bag is a clique in  $G$ , we know  $sv_1 \in E(G)$ . Similarly we have  $v_\ell t \in E(G)$ . Finally, edges  $e_i$  and  $e_{i+1}$  share an endpoint whose bag contains both  $v_i$  and  $v_{i+1}$  for  $i < \ell$ , implying that  $s, v_1, \dots, v_\ell, t$  is an  $st$ -walk in  $G$  that avoids  $U$ . By Observation 2.1, this yields an induced  $st$ -path in  $G$  that avoids  $U$ .  $\square$

**PROPOSITION 2.3.** *Let  $(T, \beta)$  be a clique tree of a graph  $G$ , let  $P$  be an induced  $st$ -path in  $G$  with  $|V(P)| \geq 5$  for  $s, t \in V(G)$ , and let  $T^*$  be the tree in the forest  $T - (\beta^{-1}(s) \cup \beta^{-1}(t))$  that contains the interior nodes of  $\Pi_T(s, t)$ . For any vertex  $u$  of  $P$  that is not among the first two or last two vertices on  $P$ , the subtree  $\beta^{-1}(u)$  is contained entirely in  $T^*$ .*

*Proof.* Assume the preconditions hold. Since  $u$  is not among the first two or last two vertices on the induced  $st$ -path  $P$ , vertex  $u$  is not adjacent to  $s$  or  $t$  and therefore  $\beta^{-1}(u)$  is vertex-disjoint with  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$ . It follows that  $\beta^{-1}(u)$  is contained entirely within one tree  $T'$  of the forest  $T - (\beta^{-1}(s) \cup \beta^{-1}(t))$ . Assume for a contradiction that  $T' \neq T^*$ . Then either  $\Pi_T(u, s)$  goes through  $\beta^{-1}(t)$ , or  $\Pi_T(u, t)$

goes through  $\beta^{-1}(s)$ . Assume without loss of generality (by symmetry) that  $\Pi_T(u, s)$  goes through  $\beta^{-1}(t)$ .

The subpath  $P'$  of  $P$  from  $s$  to  $u$  is induced. None of the vertices on  $P'$  are adjacent to  $t$ , since  $P$  is induced and  $u$  itself is not adjacent to  $s$  or  $t$ . Let  $e$  be an edge on  $\Pi_T(u, s)$  that has an endpoint  $p$  in  $\beta^{-1}(t)$ . By Observation 2.2, the  $su$ -path  $P'$  contains a vertex  $w$  of  $\text{ADH}(e) \subseteq \beta(p)$ , with  $p \in \beta^{-1}(t)$ . This shows that  $w$  is in a common bag with  $t$ . Since each bag of  $T$  forms a clique, vertex  $w$  on  $P'$  is adjacent to  $t$ , a contradiction.  $\square$

**PROPOSITION 2.4.** *Let  $(T, \beta)$  be a clique tree of graph  $G$ , and let  $P$  be an induced  $st$ -path in  $G$ . For every interior vertex  $u$  of  $P$ , the tree  $\beta^{-1}(u)$  contains a node of  $\Pi_T(s, t)$ .*

*Proof.* Assume for a contradiction that there is an interior vertex  $u$  of  $P$  for which  $\beta^{-1}(u)$  contains no node of  $\Pi_T(s, t)$ . Orient  $P$  from  $s$  to  $t$  and consider the first vertex  $u_1$  for which this holds. Let  $u_0$  be the predecessor of  $u_1$  on  $P$ . Let  $u_2$  be the first vertex after  $u_1$  on  $P$  for which  $\beta^{-1}(u_2)$  contains a node of  $\Pi_T(s, t)$ , which is well-defined since  $t$  is such a vertex.

Let  $T'$  be the tree in  $T - V(\Pi_T(s, t))$  containing  $\beta^{-1}(u_1)$ , which is well-defined since  $u_1$  does not occur in a bag of  $\Pi_T(s, t)$  and the occurrences of  $u_1$  form a connected subtree. Let  $q$  be the unique node of  $\Pi_T(s, t)$  that has a neighbor in  $T'$ . Since  $u_0$  occurs in a common bag with  $u_1$  and also occurs in a bag on  $\Pi_T(s, t)$ , by the connectivity property of tree decompositions we have  $u_0 \in \beta(q)$ . Since  $u_2$  is the first vertex after  $u_1$  that occurs in a bag of  $\Pi_T(s, t)$ , all interior vertices  $u$  of the subpath of  $P$  from  $u_0$  to  $u_2$  occur only in bags of  $T'$  by Observation 2.3. Since  $u_2$  occurs in a common bag with the predecessor of  $u_2$  on  $P$ , it follows that  $u_2$  occurs both in a bag of  $T'$  and in a bag of  $\Pi_T(s, t)$ , and consequently  $u_2 \in \beta(q)$ . Hence  $u_0$  and  $u_2$  occur in a common clique, implying that  $u_0$  and  $u_2$  are connected in  $G$ . But this edge is a chord on  $P$  since  $u_0$  and  $u_2$  are not consecutive, a contradiction to the assumption that  $P$  is induced.  $\square$

**3. Informal description of the kernelization.** In this section we give the high-level ideas behind the kernelization algorithm. Given an instance  $(G, k)$  of CHVD, we run the approximation algorithm of Theorem 1.2. If it concludes that  $(G, k)$  is a no-instance, then we output a constant-size no-instance as the result of the kernelization. Otherwise, we obtain an approximate solution  $M_0 \subseteq V(G)$  of size  $\text{poly}(k)$  such that  $G - M_0$  is chordal. The reduction process proceeds in four phases.

1. We prove the promised Erdős-Pósa property (Lemma 1.3). This allows us to slightly enlarge the modulator  $M_0$  to a modulator  $M$  with the following property: for every  $v \in M$ , the graph  $G - (M \setminus \{v\})$  is chordal.
2. We find a vertex set  $Q \supseteq M$  of size  $\text{poly}(k, |M|)$  such that all the connected components  $A$  of  $G - Q$  (which are chordal graphs since  $Q$  contains  $M$ ) are simple in the following way: all vertices in  $A$  have the same neighborhood in  $M$ , and the neighborhood of  $A$  in the rest of the graph (i.e.,  $Q \setminus M$ ) consists of two cliques of size  $\text{poly}(k, |M|)$ . One could think of  $Q$  as a union of a number of separators that break  $G - M$  into, in some sense, homogeneous pieces.
3. We introduce a reduction rule that bounds the number of connected components of  $G - Q$  in terms of  $|Q|$ .
4. We introduce a reduction rule that shrinks the size of each component  $A$  of  $G - Q$  to polynomial in the sizes of the two boundary cliques.

This process reduces the size of the graph to polynomial in  $k$ , when starting with a modulator  $M_0$  of size  $\text{poly}(k)$ .



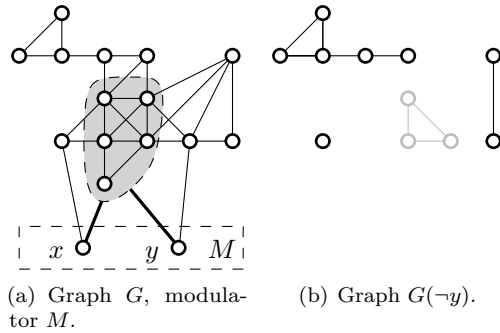


FIG. 1. (a) A graph  $G$  with modulator  $M$  such that  $G - (M \setminus \{v\})$  is chordal for each  $v \in M$ . To prevent clutter, the common neighborhood of the vertices  $\{x, y\} \in M$  is represented by a single cloud of five vertices. The thick lines from  $\{x, y\}$  to this cloud encode that all possible edges are present. (b) The graph  $G(\neg y)$  is drawn in black. The three gray vertices form a clique, which is the neighborhood of the rightmost connected component of  $G(\neg y)$ .

**Reduction phase 1.** We prove Lemma 1.3: if  $G$  is a graph containing a distinguished vertex  $v$  such that  $G - v$  is chordal, then either there are  $k + 1$  holes in  $G$  that are vertex-disjoint except for intersecting at  $v$ , or there is a set  $S_v \subseteq V(G) \setminus \{v\}$  of  $12k$  vertices such that  $G - S_v$  is chordal. The proof is algorithmic and yields a polynomial-time algorithm that computes the set of  $k + 1$  holes or the set  $S_v$ , depending on the outcome. Given the input graph  $G$  and the modulator  $M_0$ , we apply this algorithm to the graph  $G_v := G - (M_0 \setminus \{v\})$  for each  $v \in M_0$ ; note that  $G_v - v$  is chordal. When  $G_v$  contains  $k + 1$  holes that intersect only in  $v$ , we know that any size- $k$  solution to the original instance must contain  $v$ . Hence we can delete such vertices and decrease the budget  $k$  by one. After exhaustively applying this rule, we can collect the hitting sets  $S_v$  for the remaining vertices and add them to  $M_0$ , obtaining a larger modulator  $M$ . It yields the following useful structure: for every  $v \in M$ , the graph  $G - (M \setminus \{v\})$  is chordal. This property is exactly the *tidyness of the modulator* as defined by van Bevern, Moser, and Niedermeier [58].

**Reduction phase 2.** For the next phases, we switch to an *annotated version* of the CHVD problem where, apart from the graph  $G$ , the target solution size  $k$ , and the modulator  $M$ , we are given a set  $E^h \subseteq E(G[M])$ ; the solution is additionally required to contain at least one endpoint of every edge in  $E^h$ . (Note that the annotations  $E^h$  can be easily gadgeteered in the pure CHVD regime by adding, for every  $xy \in E^h$ , two new vertices  $x', y'$  and edges  $xx', yy'$ , and  $x'y'$ .)

For a vertex  $v \in M$ , we define  $G(\neg v) = G - (M \cup N(v))$ , that is, the subgraph of  $G - M$  induced by the vertices nonadjacent to  $v$  in  $G$ ; see Figure 1. By the previous Erdős–Pósa-type step, a boundary of a component of  $G(\neg v)$ , i.e., the set  $N_{G-M}(C)$  for a connected component  $C$  of  $G(\neg v)$ , is a clique. To achieve the desired set  $Q$ , we first reduce the graph in two ways.

- We shrink large cliques of  $G - M$ , bounding the size of a clique in  $G - M$  polynomially in  $k$  and  $|M|$  (section 7). The procedure is inspired by the clique reduction in the FPT algorithm by Marx [53].
- We reduce the number of connected components of  $G(\neg v)$  for every  $v \in M$  (section 8).

We start with  $Q$  being all boundaries of connected components of  $G(\neg v)$  for all  $v \in M$ . To ensure that the boundary of  $A$  consists of two cliques, we use lowest-common-

ancestor (LCA) closure of marked bags in a clique tree, similarly as in the case of the protrusion decomposition technique [11]. This yields a separator  $Q$  with the desired properties.

**Reduction phase 3.** The third phase is an easy application of the reduction template presented in section 6; the same reduction template is also used later in section 8. Since all holes pass through  $M \subseteq Q$ , a component  $A$  of  $G - Q$  can only be used on a hole to connect two vertices in  $Q$  by an induced path; it turns out that two different types of connections are relevant. For each of the  $|Q|^2$  pairs of vertices  $x, y$  of  $Q$ , we mark  $k + 2$  components of each type that provide a connection between  $x$  and  $y$  (or fewer, if the connection type is realized by fewer than  $k + 2$  components). Afterward, any component that is not marked can be safely discarded from the problem.

**Reduction phase 4.** The last reduction phase is technically involved and is described in section 9. Since a component  $A$  of  $G - Q$  is chordal and has two cliques as its boundary, and all its vertices have the same neighborhood in  $M$ , the only way  $A$  can be used to make a hole in the graph is to provide an induced path between its two boundary cliques. To break the holes in the graph, it might therefore be beneficial to have a separator between the two boundary cliques in a solution. By analyzing the local structure and exploiting the fact that an earlier reduction rule has reduced the sizes of all cliques in  $G - M$  to  $\text{poly}(k, |M|)$ , we can pinpoint a small number of relevant cliques in  $A$  such that there is always an optimal solution that does not delete vertices of  $A$  that fall outside the relevant cliques. Afterward we can shrink  $A$  by a “bypassing” operation on all the vertices that are not needed in an optimal solution.

**4. Erdős–Pósa for almost chordal graphs.** The main goal of this section is to prove Lemma 1.3. However, we first present some additional terminology and a subroutine that will be useful during the proof.

We use  $V(\mathcal{C})$  to denote the union of the vertices used on the holes in a flower  $\mathcal{C}$ . Observe that for any  $v$ -flower  $\mathcal{C} = \{C_1, \dots, C_\ell\}$  of order  $\ell$ , the structure  $\mathcal{C} - v$  consists of  $\ell$  pairwise vertex-disjoint chordless paths  $P_1, \dots, P_\ell$  that connect two nonadjacent neighbors of  $v$ , whose internal vertices belong to  $V(G) \setminus N_G[v]$ . Throughout our proofs we will switch between the interpretation of a  $v$ -flower as a set of intersecting holes in  $G$  and as a collection of induced paths in  $G - v$  between nonadjacent neighbors of  $v$ .

**PROPOSITION 4.1.** *There is a polynomial-time algorithm that, given a graph  $G$  and a vertex  $v$  such that  $G - v$  is chordal, computes a  $v$ -flower of order two or correctly determines that no such flower exists.*

*Proof.* Observe that  $\mathcal{C} = \{C_1, C_2\}$  is a  $v$ -flower of order two if and only if  $P_1 := C_1 - v$  and  $P_2 := C_2 - v$  are pairwise vertex-disjoint induced paths whose endpoints are nonadjacent neighbors of  $v$ , and whose internal vertices avoid  $N_G[v]$ . To search for a flower of order two, we try all  $\mathcal{O}(n^4)$  tuples  $(s_1, t_1, s_2, t_2) \in (N_G(v))^4$  for the endpoints of the paths  $P_1$  and  $P_2$  in the set  $N_G(v)$ . For all tuples consisting of distinct vertices for which  $s_1 t_1 \notin E(G)$  and  $s_2 t_2 \notin E(G)$ , we formulate an instance of the DISJOINT PATHS problem in  $G - (N_G[v] \setminus \{s_1, t_1, s_2, t_2\})$  in which we want to connect  $s_i$  to  $t_i$  for  $i \in \{1, 2\}$  by pairwise vertex-disjoint paths. Any FPT algorithm for the general  $k$ -DISJOINT PATHS problem suffices for this purpose as the parameter is constant. We may also use the linear-time algorithm for  $k$ -DISJOINT PATHS IN CHORDAL GRAPHS due to Kammer and Tholey [41] since  $G - v$  is chordal. If there are vertex-disjoint  $s_i t_i$ -paths  $P_1$  and  $P_2$ , then we can shortcut them to *induced* paths, which gives a flower of order two. Conversely, if there is a flower of order two, then

we will find such paths in the iteration where we guess the tuple containing their endpoints. Hence if the algorithm loops through all tuples without finding a solution to the DISJOINT PATHS problem, we may conclude that there is no flower of order two and terminate.  $\square$

We are now ready to prove Lemma 1.3, which we restate for completeness.

**LEMMA 1.3.** *There is a polynomial-time algorithm that, given a graph  $G$  and a vertex  $v$  such that  $G - v$  is chordal, outputs a  $v$ -flower  $C_1, \dots, C_\ell$  and a set  $S \subseteq V(G) \setminus \{v\}$  of size at most  $12\ell$  such that  $G - S$  is chordal.*

*Proof.* We first present an algorithm to compute a  $v$ -flower  $\mathcal{C}$ . Then we show how to compute a hitting set based on the structure of the flower, and argue that the constructed set is not too large. To initialize the procedure, we compute a clique tree  $(T, \beta)$  of  $G - v$  and root it at an arbitrary node.

**Computing a flower.** The algorithm to compute the flower is a local search procedure that maintains a  $v$ -flower  $\mathcal{C}$ , initially empty, and iteratively improves this structure according to three rules that are described below.

- (I) If  $G - (V(\mathcal{C}) \setminus \{v\})$  contains a hole  $C$ , then add  $C$  to  $\mathcal{C}$ . Equivalently, if there are two distinct nonadjacent vertices  $x, y \in N_G(v) \setminus V(\mathcal{C})$  such that graph  $G - (V(\mathcal{C}) \cup N_G[v] \setminus \{x, y\})$  contains an induced  $xy$ -path  $P$ , then add the hole induced by  $\{v\} \cup V(P)$  to  $\mathcal{C}$ .
- (II) If the current flower  $\mathcal{C}$  contains a hole  $C_i$  such that the graph  $G - (V(\mathcal{C}) \setminus V(C_i))$  contains a  $v$ -flower  $\mathcal{C}' = \{C'_i, C''_i\}$  of order two, then remove the hole  $C_i$  from flower  $\mathcal{C}$  and add the holes  $C'_i$  and  $C''_i$  instead.
- (III) If the current flower  $\mathcal{C}$  contains a hole  $C_i$  such that the path  $P_i := C_i - v$  has endpoints  $s$  and  $t$ , and there exists a  $t' \in N_G(v) \setminus (V(\mathcal{C}) \cup N_G(s))$  such that
  - $d_T(\beta^{-1}(s), \beta^{-1}(t)) > d_T(\beta^{-1}(s), \beta^{-1}(t'))$ ; and
  - the graph  $(G - (N_G[v] \setminus \{s, t'\})) - (V(\mathcal{C}) \setminus V(P_i))$  contains an induced  $st'$ -path  $P'$ , i.e., there is an induced  $st'$  path  $P'$  in  $G$  whose interior vertices avoid the closed neighborhood of  $v$  and avoid all vertices used on other holes of the flower,

then replace  $C_i$  in the flower by the hole induced by  $\{v\} \cup V(P')$ .

Figure 2 illustrates these concepts. The applicability of each of the three improvement steps can be tested in polynomial time. For (I) and (III) this follows from the fact that induced paths can be found using breadth-first search; for (II) this follows from Proposition 4.1. The number of improvement steps of types (I) and (II) is bounded by the maximum order of the flower, which is at most  $|V(G)|$ . Step (III) replaces a path in  $\mathcal{C} - v$  by one whose endpoints are closer together in  $T$ . As there are only  $|V(G)|^2$  pairs of possible endpoints, a path cannot be replaced more than  $|V(G)|^2$  times by a better path. Since the flower contains at most  $|V(G)|$  paths at any time, there can be no more than  $|V(G)|^3$  successive applications of step (III) before the algorithm either terminates or applies a step that increases the order of the flower. The algorithm therefore terminates after a polynomial number of steps. It is easy to verify that the stated conditions ensure that the structure  $\mathcal{C}$  is a  $v$ -flower at all times. Hence we can compute a  $v$ -flower  $\mathcal{C}$  that is maximal with respect to the three improvement rules in polynomial time.

**Computing a hitting set.** Based on a maximal flower  $\mathcal{C} = \{C_1, \dots, C_\ell\}$  and the associated set of paths  $P_1, \dots, P_\ell$  in  $G - v$ , we compute a set  $S \subseteq V(G) \setminus \{v\}$  that hits all holes. To every vertex of  $G - v$  we associate an edge of the decomposition tree  $T$ . For  $u \in V(G) \setminus \{v\}$ , define the *cutpoint above  $u$* , denoted  $\pi(u)$ , as follows. Let  $\pi(u)$

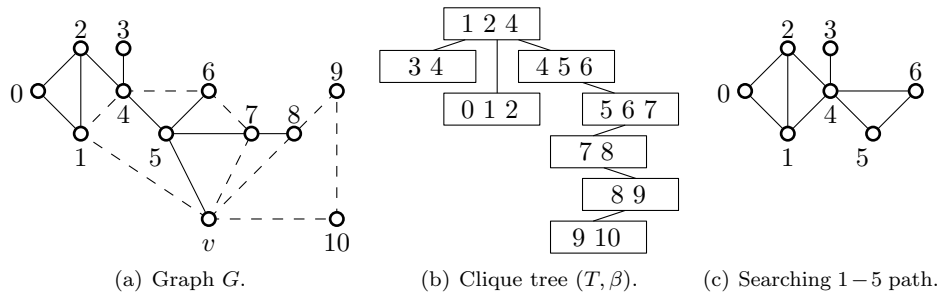


FIG. 2. (a) A graph  $G$  such that  $G - v$  is chordal. A  $v$ -flower of order two is highlighted by dashed edges. (b) A clique tree of  $G - v$ , visualized by drawing the contents of bags inside the nodes. With respect to this clique tree, the  $v$ -flower can be improved by step (III) by choosing  $P_i := (s = 1, 4, 6, 7 = t)$  and  $t' := 5$ . Note that  $d_T(\beta^{-1}(s), \beta^{-1}(t)) = 2$ , while  $d_T(\beta^{-1}(s), \beta^{-1}(t')) = 1$ . (c) The graph  $(G - (N_G[v] \setminus \{s, t'\})) - (V(C) \setminus V(P_i))$  contains an induced  $st'$ -path  $P' := (1, 4, 5)$ .

be the first edge  $e \in E(T)$  on the path from  $\text{TOP}(u)$  to the root for which  $\text{ADH}(e) \subseteq N_G(v) \cup V(C)$ , or NIL if no such edge exists. The hitting set  $S$  is constructed as follows:

- (a) For each path  $P_i$  with  $i \in \{1, 2, \dots, \ell\}$ , add the endpoints of  $P_i$  to  $S$ .
  - (b) For each vertex  $u \in N_G(v) \setminus V(C)$  with  $\pi(u) \neq \text{NIL}$ , add  $\text{ADH}(\pi(u)) \setminus N_G(v)$  to  $S$ .
- It is easy to carry out this construction in polynomial time.

CLAIM 4.1. *The set  $S$  is a subset of  $V(C) \setminus \{v\}$ .*

*Proof.* The endpoints of the paths  $P_i$  that are added to  $S$  in the first step clearly belong to the flower. The vertices added to  $S$  in the second step belong to  $\text{ADH}(\pi(u)) \setminus N_G(v)$ . By definition of cutpoint we have  $\text{ADH}(\pi(u)) \subseteq N_G(v) \cup V(C)$ , hence  $\text{ADH}(\pi(u)) \setminus N_G(v) \subseteq V(C)$ . As  $(T, \beta)$  is a clique tree of  $G - v$ , vertex  $v$  does not occur in any set  $\text{ADH}(\pi(u))$  and is not added to  $S$ .  $\square$

CLAIM 4.2. *If  $\mathcal{C}$  is a maximal  $v$ -flower, then for any distinct  $s, t \in N_G(v) \setminus V(C)$  with  $st \notin E(G)$ , the path  $\Pi_T(s, t)$  contains  $\pi(s)$  or  $\pi(t)$ .*

*Proof.* Let  $s, t \in N_G(v) \setminus V(C)$  be nonadjacent in  $G$ . Assume for a contradiction that  $\text{ADH}(e) \setminus (N_G(v) \cup V(C)) \neq \emptyset$  for all  $e$  on  $\Pi_T(s, t)$ . Let  $U := (N_G(v) \cup V(C)) \setminus \{s, t\}$ , so  $\text{ADH}(e) \setminus U \neq \emptyset$  for all edges  $e$  on  $\Pi_T(s, t)$ . By applying Proposition 2.2 to the chordal graph  $G - v$  we find an induced  $st$ -path in  $(G - v) - U$ . Such a path forms a hole together with  $v$ , since  $st \notin E(G)$ . So step (I) can be applied to increase the order of the  $v$ -flower, contradicting the maximality of  $\mathcal{C}$ .

Hence there is an edge  $e^*$  on  $\Pi_T(s, t)$  for which  $\text{ADH}(e^*) \subseteq N_G(v) \cup V(C)$ . Observe that the trees  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  are vertex-disjoint as each bag forms a clique in  $G$  and  $st \notin E(G)$ . We conclude with a case distinction; see Figure 3 for an illustration.

1. If  $\beta^{-1}(s)$  is contained in  $T_{\text{TOP}(t)}$ , then some vertex of  $\beta^{-1}(t)$  is an ancestor to all vertices of  $\beta^{-1}(s)$ . Hence  $\Pi_T(s, t)$  is the path from  $\text{TOP}(s)$  to its first ancestor in  $\beta^{-1}(t)$ , so  $\Pi_T(s, t) \subseteq \Pi_T(s, \text{ROOT})$ . As  $e^*$  shows that at least one edge on  $\Pi_T(s, \text{ROOT})$  satisfies the defining condition for a cutpoint,  $\pi(s)$  is well-defined. As  $\pi(s)$  is not higher than  $e^*$ , which lies on  $\Pi_T(s, t)$ , the edge  $\pi(s)$  lies on  $\Pi_T(s, t)$ .
2. If  $\beta^{-1}(t)$  is contained in  $T_{\text{TOP}(s)}$ , then reversing the roles of  $s$  and  $t$  in the previous argument shows that  $\pi(t)$  lies on  $\Pi_T(s, t)$ .
3. If neither case holds, then  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  are contained in different child

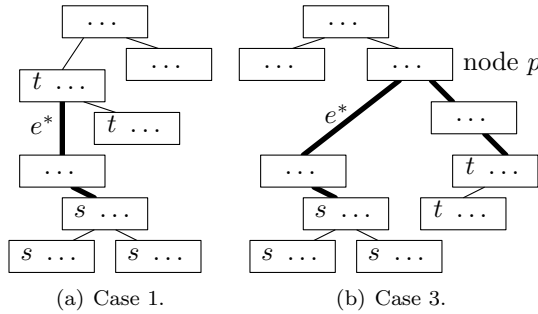


FIG. 3. Illustration of the proof of Claim 4.2. The figure shows a clique tree  $(T, \beta)$  of  $G - v$ . The minimal path  $\Pi_T(s, t)$  in  $T$  is drawn with thick edges. (a) All bags containing  $s$  are contained in the subtree of  $T$  rooted at the highest bag containing  $t$ . (b) If the first two cases do not hold, then the bags containing  $s$  are contained in a different child subtree of the lowest common ancestor  $p$  than the bags containing  $t$ .

subtrees of  $p := \text{LCA}(\text{TOP}(s), \text{TOP}(t))$ , implying that  $\Pi_T(s, t)$  is the concatenation of a path  $\Pi_s$  from  $\text{TOP}(s)$  to its ancestor  $p$ , and a path  $\Pi_t$  from  $\text{TOP}(t)$  to its ancestor  $p$ . Since  $e^*$  lies on  $\Pi_T(s, t)$ , it lies on  $\Pi_s$  or  $\Pi_t$ . Suppose that  $e^*$  lies on  $\Pi_s$ . Then  $\pi(s)$  is well-defined, lies above  $\text{TOP}(s)$  and no higher than  $e^*$ , and therefore lies on  $\Pi_T(s, t)$ . The case when  $e^*$  lies on  $\Pi_t$  is symmetric.  $\lrcorner$

CLAIM 4.3. *The graph  $G - S$  is chordal.*

*Proof.* Assume for a contradiction that  $G - S$  contains a hole  $C^*$ . Since  $G - v$  is chordal,  $C^*$  passes through  $v$  so that  $C^* - v$  is an induced  $st$ -path in  $G - S - v$  for nonadjacent  $s, t \in N_G(v)$ . By step (a) we know that  $s$  and  $t$  are not endpoints of paths in  $\mathcal{C} - v$ , with respect to the computed maximal flower  $\mathcal{C}$ . Since paths in  $\mathcal{C} - v$  cannot contain members of  $N_G(v)$  as interior vertices either, we find  $s, t \notin V(\mathcal{C})$ .

Since  $s$  and  $t$  are nonadjacent vertices of  $N_G(v) \setminus V(\mathcal{C})$ , by Claim 4.2 we know that  $\Pi_T(s, t)$  contains  $\pi(s)$  or  $\pi(t)$ . Let  $e^* \in \{\pi(s), \pi(t)\}$  be a well-defined edge on  $\Pi_T(s, t)$ . As  $s, t \in N_G(v) \setminus V(\mathcal{C})$ , by step (b) of the construction of  $S$  we know that  $\text{ADH}(e^*) \setminus S \subseteq N_G(v)$ . As only one bag of  $\Pi_T(s, t)$  contains  $s$  by minimality of  $\Pi_T(s, t)$ , and only one bag of  $\Pi_T(s, t)$  contains  $t$ , it follows that  $s, t \notin \text{ADH}(e^*)$ . On the other hand, since  $\Pi_T(s, t)$  connects  $\beta^{-1}(s)$  to  $\beta^{-1}(t)$  in  $T$  and  $e^*$  lies on  $\Pi_T(s, t)$ , by Observation 2.2 we know that the  $st$ -path  $P_v := C^* - v$  in  $G - S - v$  contains a vertex  $u \in \text{ADH}(e^*) \setminus S \subseteq N_G(v) \setminus \{s, t\}$ . This shows that  $P_v$  contains a vertex from  $N_G(v)$  in its interior, and hence has a chord, a contradiction to the assumption that  $C^*$  is a hole.  $\lrcorner$

The claims so far have established that we can compute a maximal  $v$ -flower  $\mathcal{C}$  and a corresponding chordal deletion set  $S$  that avoids  $v$  in polynomial time. To complete the proof, it remains to show that the size of  $S$  is bounded by twelve times the order of the computed flower. This is the most technical part. It follows from the next claim.

CLAIM 4.4. *If  $C_i$  is a hole of the maximal flower  $\mathcal{C}$  and  $P_i := C_i - v$ , then  $S$  contains at most 12 vertices from  $P_i$ .*

*Proof.* Assume for a contradiction that  $C_i$  is a hole in  $\mathcal{C}$  such that  $S$  contains at least 13 vertices from the path  $P_i := C_i - v$ . Let  $s, t \in N_G(v)$  be the endpoints of  $P_i$ . We identify two special vertices in  $S \cap V(P_i)$  that allow us to derive a contradiction. Define  $p := \text{LCA}(\text{TOP}(s), \text{TOP}(t))$  and note that  $p$  may coincide with  $\text{TOP}(s)$  or  $\text{TOP}(t)$ .

Since  $\beta(p)$  is a clique in  $G - v$  and  $P_i$  is an induced path, we know  $|\beta(p) \cap V(P_i)| \leq 2$ . Similarly,  $P_i$  contains at most two vertices of each of the sets  $\text{ADH}(\pi(s))$  and  $\text{ADH}(\pi(t))$ .

Let  $X$  be the set containing the first two and last two vertices on  $P_i$ , together with  $\beta(p) \cap V(P_i)$ ,  $\text{ADH}(\pi(s)) \cap V(P_i)$ , and  $\text{ADH}(\pi(t)) \cap V(P_i)$ . Since  $|X| \leq 10$  and  $|S \cap V(P_i)| \geq 13$ , there are at least three vertices in  $(S \cap V(P_i)) \setminus X$ . Since they lie on the induced path  $P_i$ , these three vertices do not form a clique, implying that there are two distinct vertices in  $(S \cap V(P_i)) \setminus X$  that are not adjacent in  $G$ . Call these  $x$  and  $y$  and choose their labels such that when traversing the path  $P_i$  from  $s$  to  $t$ , we visit  $x$  before visiting  $y$ . We will use the pair  $\{x, y\}$  to derive a contradiction to the maximality of  $\mathcal{C}$  with respect to the three presented improvement steps.

Since  $x$  and  $y$  are internal vertices of  $P_i$  and all paths in  $\mathcal{C} - v$  are vertex-disjoint,  $x$  and  $y$  are not endpoints of any path in  $\mathcal{C} - v$ , and therefore step (a) of the hitting set construction is not responsible for adding them to  $S$ . Consequently,  $x$  and  $y$  are in  $S$  because there are vertices  $x', y' \in N_G(v) \setminus V(\mathcal{C})$  with well-defined cutpoints  $\pi(x'), \pi(y')$  such that  $x \in \text{ADH}(\pi(x')) \setminus N_G(v)$  and  $y \in \text{ADH}(\pi(y')) \setminus N_G(v)$  that caused  $x$  and  $y$  to be added to  $S$  in step (b); we will argue later that  $x'$  and  $y'$  are distinct.

Since no internal vertex of  $\Pi_T(s, t)$  lies in  $\beta^{-1}(s)$  or  $\beta^{-1}(t)$ , there is a unique tree  $T^*$  in the forest  $T - (\beta^{-1}(s) \cup \beta^{-1}(t))$  that contains the internal nodes of  $\Pi_T(s, t)$ . To derive a contradiction we will argue for the following properties; note that (vii) contradicts the maximality of  $\mathcal{C}$  with respect to step (II).

- (i)  $\pi(x')$  and  $\pi(y')$  are distinct edges, and hence  $x' \neq y'$ .
- (ii)  $\pi(x')$  and  $\pi(y')$  both belong to  $T^*$ .
- (iii) The edges  $\pi(x')$  and  $\pi(y')$  are distinct from  $\pi(s)$  and  $\pi(t)$ .
- (iv) Neither one of  $\{x', y'\}$  is simultaneously adjacent to both  $s$  and  $t$ .
- (v) Neither one of  $\{x', y'\}$  is adjacent to  $s$  or  $t$ .
- (vi) There are paths from  $x'$  to  $x$  and from  $y'$  to  $y$  that are pairwise vertex-disjoint and whose interior avoids  $V(\mathcal{C}) \cup N_G(v)$ .
- (vii) The graph  $G - (V(\mathcal{C}) \setminus V(C_i))$  contains a  $v$ -flower of order two.

We continue by proving these properties one by one.

(i) If  $\pi(x') = \pi(y')$ , then  $x, y \in \text{ADH}(\pi(x')) = \text{ADH}(\pi(y'))$  are adjacent in  $G - v$  because they occur in a common bag of a clique tree. This contradicts our choice of  $x$  and  $y$  as nonadjacent vertices. Since  $\pi(x') \neq \pi(y')$  we must have  $x' \neq y'$ .

(ii) Since  $\pi(x')$  is an edge of  $T$  whose adhesion contains  $x$ , it follows that  $\pi(x')$  is an edge of the subtree  $\beta^{-1}(x)$ . Similarly,  $\pi(y')$  is an edge of  $\beta^{-1}(y)$ . It therefore suffices to prove that the subtrees  $\beta^{-1}(x)$  and  $\beta^{-1}(y)$  are entirely contained in  $T^*$ . This follows from Proposition 2.3, since both  $x$  and  $y$  are vertices on the induced  $st$ -path  $P_i$  in the chordal graph  $G - v$  that are not among the first two or last two vertices.

(iii) Suppose that  $\pi(x')$  equals  $\pi(s)$  or  $\pi(t)$ . Then  $x \in \text{ADH}(\pi(x'))$  is a vertex on  $P_i$  that is in  $\text{ADH}(\pi(s)) \cup \text{ADH}(\pi(t))$ . But this contradicts our choice of  $x$  to be a vertex of  $(S \cap V(P_i)) \setminus X$  above, since  $X$  contains  $\text{ADH}(\pi(s)) \cap V(P_i)$  and  $\text{ADH}(\pi(t)) \cap V(P_i)$ . The argument when  $\pi(y')$  equals  $\pi(s)$  or  $\pi(t)$  is symmetric.

(iv) Assume for a contradiction that  $x'$  is adjacent to both  $s$  and  $t$ . We make a case distinction on the structure of  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  in the clique tree  $T$ .

1. If  $\beta^{-1}(s)$  is contained in  $T_{\text{TOP}(t)}$ , then let  $p$  be the lowest node of  $\beta^{-1}(t)$  for which  $\beta^{-1}(s) \subseteq T_p$ . Then tree  $T^*$  as defined above is contained in  $T_p$ . As we assumed  $x'$  is adjacent to both  $s$  and  $t$ , vertices  $x'$  and  $t$  occur in a common bag, showing that  $\text{TOP}(x')$  is a proper ancestor to all nodes of  $T^*$ . Since  $\pi(x')$  is an edge on  $\Pi_T(x', \text{ROOT})$ , the edge  $\pi(x')$  lies above  $\text{TOP}(x')$ , while all nodes of  $T^*$  lie below  $\text{TOP}(x')$ . This implies that  $\pi(x')$  does not belong to  $T^*$ , contradicting (ii).

2. If  $\beta^{-1}(t)$  is contained in  $T_{\text{TOP}(s)}$ , then reversing the roles of  $s$  and  $t$  in the previous argument yields a contradiction.
3. If neither case holds, then  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$  are contained in different child subtrees of  $p := \text{LCA}(\text{TOP}(s), \text{TOP}(t))$ . It follows that  $\Pi_T(s, t)$  is the concatenation of a path from  $\text{TOP}(s)$  to  $p$  with a path from  $\text{TOP}(t)$  to  $p$ . Since  $x'$  is adjacent to both  $s$  and  $t$ , we know that  $\beta^{-1}(x')$  intersects both  $\beta^{-1}(s)$  and  $\beta^{-1}(t)$ , and therefore contains  $p$ . It follows that  $\text{TOP}(x')$  is an ancestor of  $p$ , and therefore that the edge  $\pi(x')$  connects two ancestors of  $p$ . By Proposition 2.4 we know that  $x$ , which is a vertex on the induced  $st$ -path  $P_i$ , occurs in a bag on  $\Pi_T(s, t)$ . Since  $x \in \text{ADH}(\pi(x'))$  and  $\pi(x')$  is an edge connecting two ancestors of  $p$ , while all paths from ancestors of  $p$  to nodes in  $\Pi_T(s, t)$  go through  $p$ , we have  $x \in V(P_i) \cap \beta(p)$ . This contradicts the choice of  $x$ , as we chose it from a set that excludes  $V(P_i) \cap \beta(p)$ .

The argument when  $y'$  is adjacent to both  $s$  and  $t$  is symmetric.

(v) Assume for a contradiction that  $z' \in \{x', y'\}$  is adjacent to  $r \in \{s, t\}$ ; by (iv) we then already know that  $z'$  is not adjacent to  $\hat{r} := \{s, t\} \setminus \{r\}$ . We distinguish two cases, depending on the relative positions of  $\beta^{-1}(r)$  and  $\beta^{-1}(\hat{r})$ .

1. If  $\beta^{-1}(\hat{r}) \subseteq T_{\text{TOP}(r)}$ , then let  $p$  be the lowest node of  $\beta^{-1}(r)$  for which  $\beta^{-1}(\hat{r}) \subseteq T_p$ . It follows that  $T^* \subseteq T_p$ . Now observe that since  $z'$  is adjacent to  $r$  in  $G$ , the trees  $\beta^{-1}(z')$  and  $\beta^{-1}(r)$  share at least one node, and hence  $\text{TOP}(z')$  is not a proper descendant of  $p$ . Since  $\pi(z')$  is an edge on  $\Pi_T(z', \text{ROOT})$ , it follows that  $\pi(z')$  does not lie in  $T^*$ . But this contradicts (ii).
2. If the previous case does not hold, then  $\Pi_T(r, \hat{r})$  does not enter  $\beta^{-1}(r)$  “from below,” which implies that  $\Pi_T(r, \hat{r})$  contains the edge from  $\text{TOP}(r)$  to its parent. We now distinguish two cases.
  - (a) If  $\text{TOP}(z')$  is a proper ancestor of  $\text{TOP}(r)$ , then since  $z'$  occurs in a common bag with  $r$  we must have that  $z'$  is contained in the bag of the parent of  $\text{TOP}(r)$ . We claim that, in this situation, step (III) is applicable to improve the structure of the flower, contradicting the maximality of  $\mathcal{C}$  with respect to the given rules. To see this, observe that the presence of  $z'$  in the parent bag of  $r$ , together with the fact that  $\Pi_T(r, \hat{r})$  uses the edge from  $\text{TOP}(r)$  to its parent, implies that  $d_T(\beta^{-1}(z'), \beta^{-1}(\hat{r}))$  is strictly smaller than  $d_T(\beta^{-1}(r), \beta^{-1}(\hat{r}))$ ; the latter quantity is exactly the number of edges on  $\Pi_T(r, \hat{r})$ . By Observation 2.2, the  $st$ -path  $P_i$  contains a vertex  $u$  of  $\text{ADH}(e)$ , where  $e$  is the edge from  $\text{TOP}(r)$  to its parent in  $T$ . Consequently,  $u$  is not  $r$  itself since  $r$  does not appear in the parent bag of  $\text{TOP}(r)$ . Since both  $u$  and  $z'$  appear in that parent bag, we know that  $z'$  is adjacent to  $u$  and therefore that  $z'$  together with the subpath of  $P_i$  from  $u$  to  $\hat{r}$  is a  $z'\hat{r}$ -path in  $G$ . Hence, there is an induced  $z'\hat{r}$ -path in  $G$  on a vertex subset of  $V(P_i) \cup \{z'\}$ . As  $z' \in \{x', y'\} \subseteq N_G(v) \setminus V(\mathcal{C})$  and  $z$  is not adjacent to  $\hat{r}$ , which is one of the endpoints of  $P_i$ , this shows that step (III) can be applied to improve the structure of the flower  $\mathcal{C}$ , a contradiction.
  - (b) If  $\text{TOP}(z')$  is not a proper ancestor of  $\text{TOP}(r)$ , then since  $z'$  and  $r$  occur in a common bag we know that  $\text{TOP}(z') \in T_{\text{TOP}(r)}$ . Since the cutpoint  $\pi(z')$  above  $z'$  lies on  $\Pi_T(z', \text{ROOT})$  by definition, and is contained in  $T^*$  (by (ii)) which lies “above”  $\text{TOP}(r)$  by the precondition to this case, it follows that  $\pi(z')$  is an edge between two ancestors of  $\text{TOP}(r)$ . As the adjacent vertices  $z'$  and  $r$  occur in a common bag and  $\text{TOP}(z')$  is not higher than  $\text{TOP}(r)$ , path  $\Pi_T(z', \text{ROOT})$  goes through  $\text{TOP}(r)$  and all bags on

the path from  $\text{TOP}(z')$  to  $\text{TOP}(r)$  contain  $r$ . Since no bag in  $T^*$  contains  $r$  and  $\pi(z') \in T^*$  it follows that  $\pi(z')$  is an edge connecting two ancestors of  $r$ , and must therefore be the first edge  $e$  on the path from  $\text{TOP}(r)$  to the root for which  $\text{ADH}(e) \subseteq N_G(v) \cup V(\mathcal{C})$ . But then  $\pi(z') = \pi(r)$ , which contradicts (iii).

(vi) We first construct the paths and then prove that they have interiors that avoid  $V(\mathcal{C}) \cup N_G(v)$  and are pairwise vertex-disjoint. Let  $z' \in \{x', y'\}$ , and let  $z$  be the corresponding vertex in  $\{x, y\}$ . Let  $\Pi'$  be the path in  $T$  from  $\text{TOP}(z')$  to the lower endpoint of  $\pi(z')$ . Since  $z \in \text{ADH}(\pi(z'))$  by definition of  $x, y, x', y'$ , we know that  $z$  appears in the bag of the bottom endpoint of  $\pi(z')$ , and so  $\Pi'$  connects  $\beta^{-1}(z)$  to  $\beta^{-1}(z')$  in  $T$ . Since none of the edges of  $\Pi'$  satisfied the condition for being the cutpoint above  $z'$ , for each edge  $e$  on  $\Pi'$  the set  $\text{ADH}(e) \setminus (N_G(v) \cup V(\mathcal{C}))$  is not empty. Define  $U := (N_G(v) \cup V(\mathcal{C})) \setminus \{z, z'\}$ . Applying Proposition 2.2 to the chordal graph  $G - v$  where  $z'$  plays the role of  $s$  and  $z$  plays the role of  $t$ , we establish that there is an induced  $zz'$ -path in the graph  $(G - v) - U$ . The interior vertices on this path avoid  $U \cup \{z, z'\}$  (since the latter are endpoints), and therefore the interior avoids  $N_G(v) \cup V(\mathcal{C})$  as demanded by (vi). As this holds for both choices of  $z$ , we obtain an induced  $xx'$ -path  $P_{x'}$  and an induced  $yy'$ -path  $P_{y'}$ . It remains to prove that  $P_{x'}$  and  $P_{y'}$  are pairwise vertex-disjoint. By (i) we know that  $x' \neq y'$ . As we chose  $x$  and  $y$  to be distinct vertices, and  $x, y \in V(\mathcal{C})$  while  $x', y' \in N(v) \setminus V(\mathcal{C})$ , it is clear that the endpoints of  $P_{x'}$  and  $P_{y'}$  are four distinct vertices. It remains to prove that the interior of  $P_{x'}$  is vertex-disjoint from the interior of  $P_{y'}$ . If one of the two paths has no interior, then this is trivial. So assume for a contradiction that both paths have an interior, which includes a vertex common to both paths. Orient both paths from the prime to the nonprime endpoint.

Choose  $z' \in \{x', y'\}$  and  $\hat{z}' \in \{x', y'\} \setminus \{z'\}$  such that the bottom endpoint  $p$  of  $\pi(z')$  satisfies  $\pi(\hat{z}') \notin T_p$ . This is always possible; see Figure 4 for an illustration. Let  $z \in \{x, y\}$ , depending on whether  $z' = x'$  or  $z' = y'$ ; let  $\hat{z} \in \{x, y\} \setminus \{z\}$ . Since the two interiors of  $P_{z'}$  and  $P_{\hat{z}'}$  intersect, there is a path  $Q$  in  $G - v$  from the successor  $\text{SUCC}(z')$  of  $z'$  on  $P_{z'}$  to the predecessor  $\text{PRED}(\hat{z}')$  of  $\hat{z}'$  on  $P_{\hat{z}'}$ , such that all vertices on  $Q$  belong to the interior of  $P_{x'}$  or  $P_{y'}$ . So  $V(Q)$  avoids  $N_G(v) \cup V(\mathcal{C})$ . Vertex  $\text{SUCC}(z')$  occurs in a bag of  $T_p$ , since it is adjacent to  $z'$  and  $\text{TOP}(z')$  is a descendant of both endpoints of  $\pi(z')$ . To reach a contradiction we show that  $\text{PRED}(\hat{z}')$  does not occur in a bag in  $T_p$ . To that end, we first show that  $\hat{z}$  does not occur in a bag of  $T_p$ . This follows from the fact that  $\hat{z}$  occurs in the adhesion of  $\pi(\hat{z}')$ , that  $\pi(\hat{z}')$  is not contained in  $T_p$ , and that if  $\hat{z}$  also occurred in a bag of  $T_p$  then this would force  $\hat{z}$  to be in  $\beta(p)$ , which contains  $z \in \text{ADH}(\pi(z'))$ ; but this would imply that  $z$  and  $\hat{z}$  are adjacent, contradicting our choice of  $x$  and  $y$  in the beginning of the proof. So  $\hat{z}$  does not occur in  $T_p$ , implying that  $\text{PRED}(\hat{z}')$  occurs in some bag outside  $T_p$ . Consider a path  $\Pi$  in  $T$  from a bag in  $T_p$  containing  $\text{SUCC}(z')$  to a bag outside  $T_p$  containing  $\text{PRED}(\hat{z}')$ ; we established that  $\Pi$  contains the edge from the root of  $T_p$  to its parent, which is  $\pi(z')$ . By Observation 2.2, the path  $Q$  connecting  $\text{SUCC}(z')$  to  $\text{PRED}(\hat{z}')$  contains a vertex of  $\text{ADH}(\pi(z'))$ . By definition of cutpoint, we have that  $\text{ADH}(\pi(z')) \subseteq N_G(v) \cup V(\mathcal{C})$ . But  $Q$  avoids  $N_G(v) \cup V(\mathcal{C})$ , a contradiction. It follows that  $P_{x'}$  and  $P_{y'}$  are indeed vertex-disjoint, proving (vi).

(vii) Consider the  $xx'$ -path  $P_{x'}$  and the  $yy'$ -path  $P_{y'}$ , whose existence we proved in (vi). The interiors of these paths avoid  $V(\mathcal{C}) \cup N_G(v)$ , the endpoints  $x'$  and  $y'$  belong to  $N_G(v)$ , and the endpoints  $x$  and  $y$  belong to  $V(\mathcal{C}) \setminus N_G(v)$ . Consider the walk in  $G - v$  that starts at  $s$ , traverses the  $st$ -path  $P_i = C_i - v$  until  $x$ , and then traverses  $P_{x'}$  to  $x'$ . Since  $s$  is not adjacent to  $x'$  by (v) and  $x' \in N_G(v)$ , this



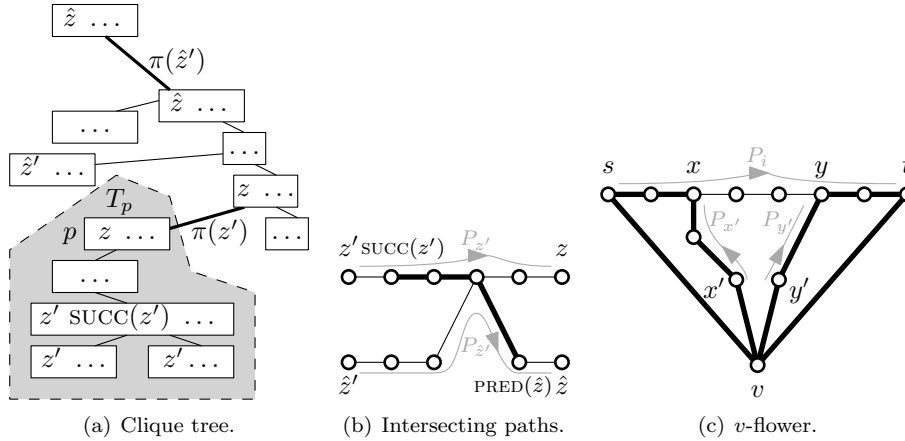


FIG. 4. (a) Illustration for part (vi) of the proof of Claim 4.4. Example of how the different entities can be arranged in the clique tree  $(T, \beta)$  of  $G - v$ . Subtree  $T_p$  is shaded, and the edges  $\pi(z')$  and  $\pi(\hat{z}')$  have thick lines. (b) The structure of a hypothetical intersection in the interiors of the paths  $P_{z'}$  and  $P_{\hat{z}'}$ . Path  $Q$  is shown by thick edges. (c) Illustration for part (vii) of Claim 4.4. Drawing of how the constructed paths  $P_{x'}$  and  $P_{y'}$  are combined with the original petal  $P_i := C_i - v$  of the flower, to obtain a flower of order two (visualized by thick edges).

walk contains an induced  $sx'$ -path between two neighbors of  $v$ , whose internal nodes avoid  $N_G(v)$ . Moreover, the only vertices of  $\mathcal{C}$  used on this path belong to  $P_i$ . Similarly, there is an induced  $ty'$ -path within the walk starting at  $t$ , traversing  $P_i$  from  $t$  to  $y$ , and then traversing  $P_{y'}$  from  $y$  to  $y'$ . Since we chose  $x$  and  $y$  such that  $x$  occurs before  $y$ , the pieces of  $P_i$  used by these two induced paths are vertex-disjoint. By (vi), the pieces of  $P_{x'}$  and  $P_{y'}$  used are also vertex-disjoint. Hence these two induced paths are vertex-disjoint and each connect two nonadjacent neighbors of  $v$ . Together with  $v$  these form a  $v$ -flower of order two in  $G$ . Since the only vertices of  $\mathcal{C}$  used on this order-2 flower are those of  $P_i$  and  $v$ , it follows that there is a  $v$ -flower of order two in  $G - (V(\mathcal{C}) \setminus V(C_i))$ , establishing (vii). This contradicts the fact that  $\mathcal{C}$  is maximal with respect to step (II) and completes the proof of Claim 4.4.  $\square$

Armed with these claims we finish the proof of Lemma 1.3. Claim 4.3 shows that  $S$  is a hitting set avoiding  $v$  for the holes in  $G$ . All vertices of  $S$  belong to  $V(\mathcal{C}) \setminus \{v\}$  by Claim 4.1, and therefore all lie on some path of the structure  $\mathcal{C} - v$ . Since the number of paths equals the order of the flower, while  $S$  contains at most 12 vertices from each path by Claim 4.4, it follows that the size of  $S$  is at most 12 times the order of the flower. This concludes the proof of Lemma 1.3.  $\square$

**5. ANNOTATED CHORDAL VERTEX DELETION.** In what follows it is more convenient to work with a variant of the CHVD problem with some annotations on the set  $M$ . More formally, the input to the problem ANNOTATED CHORDAL VERTEX DELETION (A-CHVD, for short) is a tuple  $(G, k, M, E^h)$  with the following properties:

1.  $G$  is a graph,  $k$  is an integer,  $M \subseteq V(G)$ , and  $E^h \subseteq E(G[M])$ .
2.  $G - M$  is a chordal graph, and, moreover, for every  $v \in M$  the graph  $G - (M \setminus \{v\})$  is a chordal graph.

The A-CHVD problem asks for the existence of a set  $X \subseteq V(G)$ , called henceforth a *solution*, such that  $|X| \leq k$ ,  $G - X$  is chordal, and for every pair  $\{x, y\} \in E^h$ , at least one of  $x$  and  $y$  belongs to  $X$ .

Lemma 1.3 can be used to transform an instance of the general CHVD problem

into an equivalent instance satisfying the requirements of the annotated problem, if a modulator  $M$  is known. For the final kernelization algorithm, such a modulator will be obtained from the approximation algorithm of Theorem 1.2.

LEMMA 5.1. *There is a polynomial-time algorithm that, given a graph  $G$ , an integer  $k$ , and a set  $M_0 \subseteq V(G)$  such that  $G - M_0$  is chordal, either correctly determines that  $(G, k)$  is a no-instance of CHVD or computes an instance  $(G', k', M, E^h)$  of A-CHVD with the same answer such that  $|M| = \mathcal{O}(k \cdot |M_0|)$  and  $k' \leq k$ .*

*Proof.* For every  $v \in M_0$ , we apply the algorithm of Lemma 1.3 to the nearly chordal graph  $G - (M_0 \setminus \{v\})$ , obtaining a  $v$ -flower  $\mathcal{C}_v$  and a hitting set  $S_v$ . If  $|\mathcal{C}_v| > k$  for some  $v$ , we delete the vertex  $v$  from  $G$ , decrease  $k$  by one, and restart the algorithm. If the budget  $k$  drops below zero, then we report that the original input was a no-instance of CHVD. Clearly, if  $|\mathcal{C}_v| > k$ , then every size- $k$  solution to CHVD on  $G$  contains the vertex  $v$ , so  $(G, k)$  is equivalent to  $(G - v, k - 1)$ . If  $|\mathcal{C}_v| \leq k$  for every  $v \in M_0$ , we define  $M := M_0 \cup \bigcup_{v \in M_0} S_v$ ; note that  $|M| \leq |M_0|(12k + 1)$ . Letting  $G'$  and  $k'$  denote the reduced graph and budget, we output the A-CHVD instance  $(G', k', M, E^h = \emptyset)$ . It follows from Lemma 1.3 that it is a valid A-CHVD instance, and clearly it is a yes-instance if and only if the original CHVD instance  $(G, k)$  is a yes-instance.  $\square$

For completeness, we remark that Lemma 1.3 can be circumvented in the reduction to the annotated problem given above (which means it can be bypassed for the entire kernelization). Instead of applying Lemma 1.3 to the graph  $G - (M_0 \setminus \{v\})$  to find a hitting set  $S_v$  for the holes through  $v$ , one can obtain such a hitting set by creating  $|V(G)|$  true-twin copies of vertex  $v$  and running the approximation algorithm for CHVD on the resulting inflated version of the graph  $G - (M_0 \setminus \{v\})$ . This increases the size of the kernel, though, so we prefer the linear bounds given by Lemma 1.3.

The next few sections are devoted to the proof of the following theorem.

THEOREM 5.2. *A-CHVD admits a polynomial kernel when parameterized by  $|M|$ . More precisely, given an A-CHVD instance  $(G, k, M, E^h)$ , one can in polynomial time compute an equivalent instance  $(G', k, M, F^h)$  with  $E^h \subseteq F^h$  and  $|V(G')| = \mathcal{O}(k^{16}|M|^{29})$ .*

As typical for kernelization algorithms, the algorithm of Theorem 5.2 consists of a number of reduction rules; at every step, we apply the lowest-numbered reduction rule. Observe that any instance with  $k \geq |M|$  has a trivial yes-answer as the set  $M$  is a solution. For such instances we can output a constant-size yes-instance of A-CHVD. In the remainder we assume  $|M| > k$ .

Let  $(G, k, M, E^h)$  be an A-CHVD instance. We say that a vertex set  $S \subseteq V(G) \setminus M$  is *irrelevant* if the instances  $(G, k, M, E^h)$  and  $(G - S, k, M, E^h)$  are equivalent. We say that a pair  $xy \in \binom{M}{2}$  is a *new forced pair* if  $xy \notin E^h$ , but the instances  $(G, k, M, E^h)$  and  $((V(G), E(G) \cup \{xy\}), k, M, E^h \cup \{xy\})$  are equivalent. Note that adding an edge  $xy$  to  $G$  is meaningless, as a solution is required to delete  $x$  or  $y$  anyway, but it makes later notation cleaner.

A hole  $H$  in the graph  $G$  is a *permitted hole* if  $V(H)$  does not contain any pair of  $E^h$ . Note that a hole that is not permitted is hit by any solution to  $(G, k, M, E^h)$  due to the constraints imposed by  $E^h$ , and, informally speaking, we can ignore such holes in the further argumentation.

Recall that  $G(-v)$  was defined as  $G - (M \cup N(v))$ , the subgraph of  $G - M$  induced by the vertices nonadjacent to  $v$  in  $G$ . In what follows, we will use the following generalization of this shorthand notation.

DEFINITION 5.3. Let  $x_1, x_2, \dots, x_a, y_1, y_2, \dots, y_b \in M$  be (not necessarily distinct) vertices for some  $a, b \geq 0$ . Let us define

$$V(x_1, x_2, \dots, x_a, \neg y_1, \neg y_2, \dots, \neg y_b) = \left( (V(G) \setminus M) \cap \bigcap_{i=1}^a N(x_i) \right) \setminus \left( \bigcup_{j=1}^b N(y_j) \right),$$

$$G(x_1, x_2, \dots, x_a, \neg y_1, \neg y_2, \dots, \neg y_b) = G[V(x_1, x_2, \dots, x_a, \neg y_1, \neg y_2, \dots, \neg y_b)].$$

Consider the following reduction rule.

REDUCTION RULE 1. For every  $x, y \in M$  with  $x \neq y$ ,  $xy \notin E(G)$ , if the size of the largest independent set in  $G(x, y)$  is at least  $k + 2$ , then add  $xy$  to  $E^h$  and to  $E(G)$ .

The correctness is straightforward: any solution to  $(G, k, M, E^h)$  can delete at most  $k$  out of  $k + 2$  vertices of any maximum independent set in  $G(x, y)$ ; the remaining vertices form a  $C_4$  with  $x$  and  $y$ , forcing the solution to delete either  $x$  or  $y$ . We add the edge  $xy$  to  $E(G)$  to preserve the properties of an A-CHVD instance. Furthermore, note that as  $G(x, y)$  is chordal, maximum independent sets can be computed in polynomial time [36].

In the remainder we will often use the following simple observation to find holes, in proofs by contraposition.

OBSERVATION 5.1 (see [53, Proposition 3]). If a graph  $G$  contains a vertex  $v$  with two nonadjacent neighbors  $u_1, u_2 \in N(v)$  and contains a walk from  $u_1$  to  $u_2$  with all internal vertices in  $V(G) \setminus N[v]$ , then  $G$  contains a hole passing through  $v$ .

We conclude this section with an important immediate observation about A-CHVD.

PROPOSITION 5.4. Let  $(G, k, M, E^h)$  be an instance of A-CHVD. Then, for every  $v \in M$  and every connected component  $A$  of  $G(\neg v)$ , the set  $N_G(A) \setminus M$  is a clique.

*Proof.* Suppose there exist two nonadjacent vertices  $u_1, u_2 \in N(A) \setminus M$ . Apply Observation 5.1 to the graph  $G - (M \setminus \{v\})$  with the vertex  $v$  and a path between  $u_1$  and  $u_2$  with internal vertices in  $A$ . This yields a hole in  $G - (M \setminus \{v\})$ , which contradicts the requirement of A-CHVD instances that  $G - (M \setminus \{v\})$  is chordal for all  $v \in M$ .  $\square$

## 6. Reducing the number of components with respect to a separator.

The next procedure reduces an instance  $(G, k, M, E^h)$  of A-CHVD if we have access to a small vertex set whose removal splits the graph into many components. In general, it is not easy to find such a vertex set since it is related to computing the toughness of a graph, which is NP-hard [6]. The reduction can therefore only be applied for a concrete separator that is supplied to the procedure. For that reason, we call it a *reduction template*, instead of a concrete reduction rule. We will apply the template in situations where such a vertex set can be identified efficiently.

In the following, we will say “mark up to  $f(k)$  objects of type  $X$ ” to mean the following. If there are more than  $f(k)$  objects of type  $X$ , then choose  $f(k)$  of them arbitrarily and mark them. If there are fewer, then mark all of them.

REDUCTION TEMPLATE 1. Given a vertex subset  $S \supseteq M$  of  $G$ , do the following:

1. For every pair  $xy \in \binom{S}{2} \setminus E(G)$ , mark up to  $k + 2$  connected components  $C$  of  $G - S$  that contain both a neighbor of  $x$  and a neighbor of  $y$ .
2. For every pair  $xy \in \binom{S}{2} \cap E(G)$ , mark up to  $k + 1$  connected components  $C$  of  $G - S$  that contain a path  $P$  between a vertex in  $N_G(x) \cap V(C)$  and a vertex in  $N_G(y) \cap V(C)$  such that  $V(P)$  avoids  $N_G(x) \cap N_G(y)$ .

Delete the vertices in an unmarked component  $C$  from the graph.

Template 1 yields a stand-alone reduction that does not rely on any other reduction rules being exhaustively applied. Observe that if  $G - S$  has more than  $(k + 2) \binom{|S|}{2}$  components, then the marking scheme will leave at least one component unmarked, and therefore the reduction will shrink the graph. Given  $S$ , it is easy to apply the procedure in polynomial time. Observe that the existence of a path  $P$  through a component  $C$  satisfying the conditions of the second item can be tested by trying for each  $v \in N_G(x) \cap V(C)$  and  $u \in N_G(y) \cap V(C)$  whether  $u$  and  $v$  belong to the same connected component of  $G[V(C)] - (N_G(x) \cap N_G(y))$ . Correctness follows from the next lemma.

**LEMMA 6.1.** *If a component  $C$  of  $G - S$  is not marked by Reduction Template 1, then  $V(C)$  is irrelevant.*

*Proof.* Since  $S \supseteq M$ , no forced pairs involve  $V(C)$ . It therefore suffices to prove that if  $(G - V(C)) - X$  is chordal for some set  $X$  of size at most  $k$ , then  $G - X$  is chordal as well. Assume for a contradiction that  $(G - V(C)) - X$  is chordal but  $G - X$  is not, and consider some hole  $H$  in  $G - X$ . It uses at least one vertex  $v \in V(C)$ . Let  $x$  and  $y$  be the first vertices of  $S$  that you encounter when traversing the hole  $H$  in both directions when starting in  $v$ . By definition of A-CHVD and the fact that  $M \subseteq S$  such vertices exist and  $x \neq y$ .

If  $xy \notin E(G)$ , then  $C$  is unmarked and contains a path from a neighbor of  $x$  to a neighbor of  $y$ . We marked at least  $k + 2$  components that provide such a path and  $X$  intersects at most  $k$  of them, leaving two marked components  $C_1, C_2$  free. As each component  $C_i$  contains a path between a neighbor of  $x$  and a neighbor of  $y$ , by shortcutting these paths we find induced  $xy$ -paths whose interior vertices belong to  $C_i$  for each  $i \in \{1, 2\}$ . As  $C_1$  and  $C_2$  are distinct components of  $G - S$  and the only vertices from  $S$  used on these paths are  $x$  and  $y$ , the interiors of the path are vertex-disjoint and the interior vertices of one path are not adjacent to the interior of the other. Hence the concatenation of these paths is a hole in  $(G - V(C)) - X$ , a contradiction.

Now consider the case that  $xy \in E(G)$ . Since  $x$  and  $y$  are the first vertices of  $S$  encountered when traversing  $H$  starting at  $v$ , it follows that  $H \cap V(C)$  is an induced path in  $C$  between a neighbor of  $x$  and a neighbor of  $y$  that avoids  $N_G(x) \cap N_G(y)$ . The hole  $H$  consists of  $xy$  together with this single segment in  $H \cap V(C)$ . Since  $C$  was not marked, there are  $k + 1$  marked components containing an  $N_G(x) \cap V(C)$  to  $N_G(y) \cap V(C)$  path avoiding  $N_G(x) \cap N_G(y)$ . Consequently, one of these components  $C'$  is not intersected by  $X$ . Let  $P$  be an induced path in  $C'$  from  $u_1 \in N_G(x) \cap V(C')$  to  $u_2 \in N_G(y) \cap V(C')$  that avoids  $N_G(x) \cap N_G(y)$ . Let  $u'_1$  be the last vertex on  $P$  that is adjacent to  $x$ , and let  $u'_2$  be the first vertex after  $u'_1$  that is adjacent to  $y$ . The subpath of  $P$  from  $u'_1$  to  $u'_2$  is an induced path with at least two vertices from a neighbor of  $x$  to a neighbor of  $y$ , and no interior vertex of this subpath is adjacent to  $x$  or  $y$ . Moreover,  $u'_1 y, u'_2 x \notin E(G)$  since  $P$  avoids  $N_G(x) \cap N_G(y)$ . Hence  $P$  forms a hole together with  $x$  and  $y$ , and this hole avoids  $X$  and  $V(C)$ . So  $(G - V(C)) - X$  is not chordal, a contradiction.  $\square$

**7. Reducing cliques.** Our goal for this section is to prove the following statement.

**LEMMA 7.1.** *Given an A-CHVD instance  $(G, k, M, E^h)$  on which Reduction Rule 1 is not applicable and a clique  $K$  in  $G - M$  of size  $|K| > (k + 1)(|M|^3 + (k + 3)|M|^2)$ , one can in polynomial time find either an irrelevant vertex or a new forced pair.*

Let  $G^\circ := G - M$ . Fix a clique tree of  $G^\circ$  and root it in a node  $p$  such that  $K \subseteq \beta(p)$ .

**7.1. Sets  $Q^{xy}$  and forced pairs.** We first look for a new forced pair in the following manner. Let  $x$  and  $y$  be two distinct vertices of  $M$  with  $xy \notin E(G)$ . We say that a node  $q \in V(T)$  is *xy-good* if there exists an  $xy$ -path  $P_q$  in  $G$  whose every internal vertex  $v$  belongs to  $G^\circ$  and satisfies  $\beta^{-1}(v) \subseteq V(T_q)$ . Note that if  $q$  is *xy-good*, then all its ancestors in the tree  $T$  are also *xy-good*. Let  $Q^{xy}$  be the set of all lowest *xy-good* nodes in  $T$ , that is,  $q \in Q^{xy}$  if and only if  $q$  is *xy-good* but none of the children of  $q$  are *xy-good*. We claim the following.

**CLAIM 7.1.** *If  $|Q^{xy}| \geq k + 2$  for some pair  $xy \in \binom{M}{2} \setminus E(G)$ , then every solution to the instance  $(G, k, M, E^h)$  contains at least one vertex from the pair  $xy$ .*

*Proof.* For every  $q \in Q^{xy}$ , let  $P_q$  be a path witnessing that  $q$  is *xy-good*; without loss of generality, assume that  $P_q$  is an induced path. Consider two distinct  $q, q' \in Q^{xy}$ . Observe that, by the definition of  $Q^{xy}$ , the nodes  $q$  and  $q'$  are not in an ancestor-descendant relationship in  $T$ , and thus the subtrees  $T_q$  and  $T_{q'}$  are node-disjoint. Consequently, no internal vertex of  $P_q$  is equal or adjacent to any internal vertex of  $P_{q'}$ . As  $xy \notin E(G)$ , the union of the paths  $P_q$  and  $P_{q'}$  is a hole in  $G$ . As there are at least  $k + 2$  elements of  $Q^{xy}$ , any solution to  $(G, k, M, E^h)$  avoids the set of internal vertices of at least two such paths, and therefore contains  $x$  or  $y$ .  $\square$

Claim 7.1 justifies the following reduction rule.

**REDUCTION RULE 2.** *If  $|Q^{xy}| \geq k + 2$  for some pair  $xy \in \binom{M}{2} \setminus E(G)$ , then add  $xy$  to  $E^h$  and to  $E(G)$ .*

Observe that it is straightforward to check directly from the definition if a node  $q \in V(T)$  is *xy-good* for a given pair  $xy$ . Thus, Reduction Rule 2 can be applied in polynomial time, and henceforth we assume  $|Q^{xy}| \leq k + 1$  for every pair  $xy \in \binom{M}{2} \setminus E(G)$ .

**7.2. Marking relevant parts of a clique.** Having bounded the size of the sets  $Q^{xy}$ , we now mark a number of vertices of the clique  $K$ , and then argue that any unmarked vertex is irrelevant. The argumentation here was loosely inspired by the analogous part of the work of Marx [53, section 5].

Note that for every  $x \in M$ , there is at most one connected component of  $G(-x)$  that contains a vertex of  $\beta(p)$ , since  $\beta(p)$  is a clique in  $G^\circ$ . By  $A^x$  we denote the vertex set of this connected component of  $G(-x)$  if it exists. We define  $A^x := \emptyset$  otherwise, which occurs when  $\beta(p) \subseteq N(x)$ . By Proposition 5.4, if  $A^x \neq \emptyset$ , then  $N(A^x) \setminus M$  is a clique in  $G^\circ$ ; let  $p^x$  be a node of  $T$  such that  $N(A^x) \setminus M \subseteq \beta(p^x)$ . For a vertex  $x \in M$  with  $A^x = \emptyset$ , we define  $p^x = p$ , which is the root of the clique tree.

The marking procedure for this rule works with additional constraints. By a statement of the form “mark up to  $\ell$  vertices of  $Z$ , preferring vertices maximizing/minimizing distance from  $q$ ” for some node  $q \in V(T)$ , we mean that we do not choose the vertices of  $Z$  to mark arbitrarily (in the case  $|Z| > \ell$ ), but we sort vertices  $u \in Z$  according to the distance between  $\beta^{-1}(u)$  and  $q$  in  $T$ , and mark  $\ell$  vertices with the largest/smallest distance. We are now ready to present our marking algorithm.

- (a) For every triple  $(x_1, x_2, y) \in M \times M \times M$ , mark up to  $k + 1$  vertices from  $K \cap V(x_1, x_2, \neg y)$ .
- (b) For every pair  $(x_1, x_2) \in \binom{M}{2} \setminus E(G)$  and every node  $q \in Q^{x_1 x_2}$ , mark up to  $k + 1$  vertices of  $K \cap V(x_1, x_2)$ , preferring vertices maximizing distance from  $q$ .

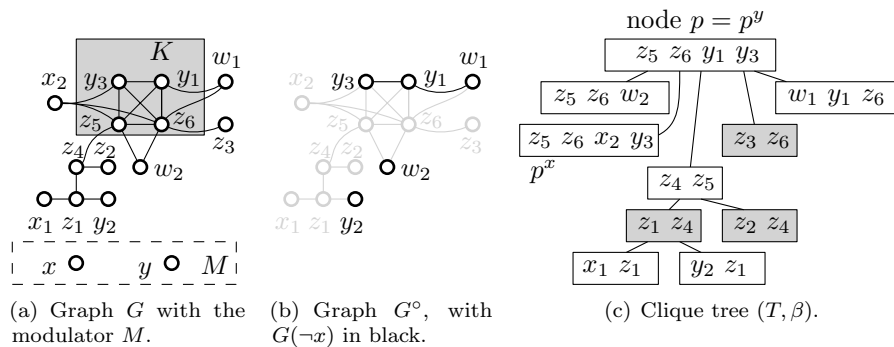


FIG. 5. Illustration for the clique marking procedure. (a) Instance of A-CHVD with graph  $G$  and modulator  $M$ , and a clique  $K$  in  $G^\circ$ . For readability, edges between  $G^\circ$  and  $M$  are not drawn, but encoded through vertex names. Vertex  $x$  is adjacent to the  $x_i$ 's and  $z_i$ 's; vertex  $y$  is adjacent to the  $y_i$ 's and  $z_i$ 's. (b) The graph  $G(\neg x)$  contains the vertices in black and has a component with vertex set  $A^x = \{y_1, y_3, w_1\}$  that intersects  $K$ . Since  $K \subseteq N_G(y)$  we have  $A^y = \emptyset$ . (c) A clique tree of  $G^\circ$  rooted at a node  $p$  whose bag contains the entire clique  $K$ . The three nodes of  $Q^{x^y}$  are colored gray. Contents of a bag are drawn within the rectangle representing its node. A node  $p^x$  containing all vertices of  $N(A^x) \setminus M$  is chosen.

- (c) For every pair  $(x, y) \in M \times M$ , mark up to  $k + 1$  vertices from  $K \cap V(x, \neg y)$ , preferring vertices minimizing distance from  $p^y$ .  
 (d) For every pair  $(x, y) \in M \times M$ , mark up to  $k + 1$  vertices from  $K \cap V(\neg x, \neg y)$ , preferring vertices minimizing distance from  $p^y$ .

A direct calculation shows that we mark at most  $(k + 1)(|M|^3 + (k + 3)|M|^2)$  vertices of  $K$ , and it is straightforward to implement the marking algorithm to run in polynomial time. Figure 5 illustrates the context of the marking procedure. Figure 6 illustrates the different types of holes that are preserved by this marking procedure.

**7.3. Every unmarked vertex is irrelevant.** The remainder of this section is devoted to the proof that any unmarked vertex of  $K$  is irrelevant. For that we need the following simple technical step.

**CLAIM 7.2.** *Let  $v \in K$  be an unmarked vertex, let  $y \in M$  be a vertex such that  $vy \notin E(G)$ , and let  $P$  be a path between  $v$  and  $y$  with all internal vertices in  $G^\circ$ . Then, for every  $x \in M$ , and every vertex  $v'$  marked for the pair  $(x, y)$ , either in point (c) if  $vx \in E(G)$  or in point (d) if  $vx \notin E(G)$ , the vertex  $v'$  is equal or adjacent to at least one vertex of  $V(P) \setminus \{v, y\}$ .*

*Proof.* First, note that since in points (c) and (d) we prefer vertices minimizing the distance to  $p^y$ , and the vertex  $v$  is a candidate for marking when  $v'$  has been marked, the distance from  $\beta^{-1}(v')$  to  $p^y$  is not larger than the distance from  $\beta^{-1}(v)$  to  $p^y$ .

Without loss of generality we may assume that  $P$  is an induced path: shortcutting the path  $P$  to an induced one would only make our task harder. Let  $u$  be the neighbor of  $y$  on the path  $P$ . We have  $u \in V(G^\circ)$ ; since  $vy \notin E(G)$ , we have also  $u \neq v$ . Furthermore, as  $P$  is an induced path, all vertices on  $P$  except for  $y$  and  $u$  lie outside  $N[y]$ , and thus they lie in  $A^y$  as  $v \in A^y$ . Consequently,  $u \in N(A^y)$ , and thus  $p^y \in \beta^{-1}(u)$ .

If  $p^y \in \beta^{-1}(v)$ , then also  $p^y \in \beta^{-1}(v')$  and  $v' = u$  or  $v'u \in E(G)$ . Otherwise, let  $e$  be the first edge of  $T$  on the unique path from  $p$  to  $p^y$  that is not contained in

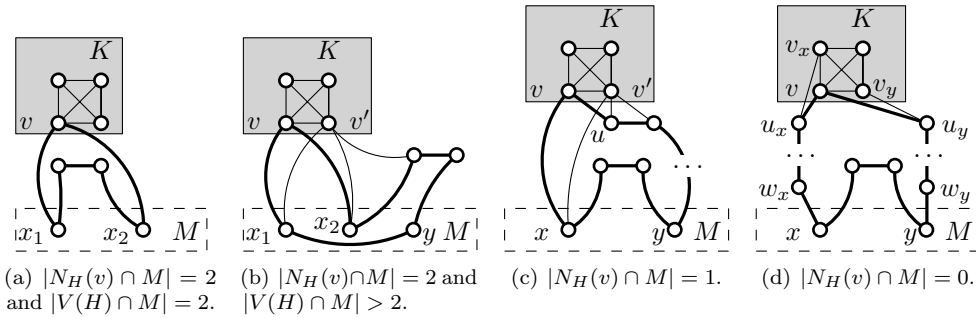


FIG. 6. Illustration for the correctness of the clique marking procedure (Claim 7.3). The four subfigures correspond to the four marking steps (a)–(d). Each subfigure shows a schematic view of an instance with a modulator  $M$  and clique  $K$  in  $G^\circ$  that is the target of the marking procedure. For each of the four types of holes highlighted by thick edges, the corresponding marking step ensures that the constraints they impose on size- $k$  solutions are preserved after deleting the unmarked vertices. Vertex names correspond to those used in the proof of Claim 7.3.

$T[\beta^{-1}(v)]$ , and let  $p'$  be the endpoint of  $e$  that belongs to  $\beta^{-1}(v)$ . Since  $V(P) \setminus \{y\}$  induces a connected subgraph of  $G^\circ$  containing vertices  $v \in \beta(p)$  and  $u \in \beta(p^y)$ , there exists a vertex  $w \in V(P) \setminus \{y\}$  with  $w \in \text{ADH}(e)$ . As  $v \notin \text{ADH}(e)$  we know  $w \neq v$ . Furthermore, we have  $p' \in \beta^{-1}(v')$ , since  $\beta^{-1}(v')$  is at least as close to  $p^y$  as  $\beta^{-1}(v)$ , and both  $\beta^{-1}(v)$  and  $\beta^{-1}(v')$  contain the root  $p$ . Hence,  $w = v'$  or  $wv' \in E(G)$ . This finishes the proof of the claim.  $\lrcorner$

We are now ready to prove the last claim, concluding the proof of Lemma 7.1.

CLAIM 7.3. Every unmarked vertex of  $K$  is irrelevant.

*Proof.* Let  $v \in K$  be an unmarked vertex. It suffices to show that if  $X$  is a solution to  $(G - v, k, M, E^h)$ , then it is also a solution to  $(G, k, M, E^h)$ . Assume the contrary; as  $v \notin M$ , it must hold that there is a hole  $H$  in  $G - X$ . Since  $G - v - X$  is chordal,  $v \in V(H)$ . By the properties of an A-CHVD instance  $(G, k, M, E^h)$ , the hole  $H$  contains at least two vertices of  $M$ . We make a case distinction, depending on whether the neighbors of  $v$  on the hole  $H$  belong to  $M$  or lie in  $G^\circ$  (see Figure 6).

**Case 1: Both neighbors of  $v$  on  $H$  lie in  $M$ .** Denote these two neighbors by  $x_1, x_2$ . As  $H$  is a hole,  $x_1x_2 \notin E(G)$ ; in particular,  $x_1x_2 \notin E^h$ .

Assume first that  $x_1$  and  $x_2$  are the only two vertices of  $V(H) \cap M$ . Then  $H$  consists of vertices  $v, x_1, x_2$  and a path  $P$  between  $x_1$  and  $x_2$  with all internal vertices in  $G^\circ$ . Consider the set  $P_T$  of all nodes of  $T$  whose bag contains at least one internal vertex of  $P$ . Since  $v \in K \subseteq \beta(p)$ , and  $H$  is a hole, we have  $p \notin P_T$ . Furthermore, by the connectivity of  $P$ ,  $P_T$  induces a subtree of  $T$ . Let  $q_0$  be the topmost (closest to  $p$ ) vertex of  $P_T$ ; note that  $q_0 \neq p$ .

By the existence of the path  $P$ , the node  $q_0$  is  $x_1x_2$ -good. By the definition of  $Q^{x_1x_2}$  there exists a node  $q \in Q^{x_1x_2}$  that is a descendant of  $q_0$ . Consider a vertex  $v'$  marked at point (b) for the pair  $x_1x_2$  and the node  $q$  that does not belong to the solution  $X$ . Since in point (b) we mark vertices that maximize the distance from  $q$ , the distance between  $\beta^{-1}(v')$  and  $q$  is at least as large as the distance between  $\beta^{-1}(v)$  and  $q$ . Consequently, since the root  $p$  belongs to  $\beta^{-1}(v')$  and to  $\beta^{-1}(v)$ , and  $q_0$  is an ancestor of  $q$ , we have  $q_0 \notin \beta^{-1}(v')$ . We infer that  $v'$  is not adjacent to any internal vertex of  $P_T$ , and thus if we replace  $v$  with  $v'$  on the hole  $H$ , we obtain a hole as well, contradicting the fact that  $G - v - X$  is a chordal graph.

In the remaining case, there exists a third node  $y \in V(H) \cap M$ , different from  $x_1$  and  $x_2$ ; in particular,  $vy \notin E(G)$ . Consider a vertex  $v'$  marked at point (a) for the triple  $(x_1, x_2, y)$  that does not belong to  $X$ . Consider the path  $P := H - y$  between the neighbors of  $y$  on the hole  $H$ , and let  $P'$  be the walk created from  $P$  by replacing the vertex  $v$  with  $v'$ . Since  $v' \in N(x_1) \cap N(x_2)$ ,  $P'$  is indeed a walk. Since  $v' \notin N(y)$ , all internal vertices of  $P'$  avoid  $N[y]$ . Since no vertex of  $P'$  belongs to  $X$  or equals  $v$ , by Observation 5.1 the graph  $G - v - X$  contains a hole, a contradiction.

**Case 2: One neighbor of  $v$  on  $H$  lies in  $M$ .** Let  $u$  and  $x$  be the two neighbors of  $v$  on  $H$ , with  $x \in M$  and  $u \in V(G^\circ)$ . Let  $y$  be the first vertex of  $V(H) \cap M$  that we encounter if we traverse  $H$  starting from  $v$  in the direction towards  $u$ . Since  $V(H) \cap M$  contains at least two vertices,  $x \neq y$ , in particular,  $vy \notin E(G)$ . Let  $P_u, P_x$  be the subpaths of  $H$  between  $v$  and  $y$ , with  $u \in V(P_u)$  and  $x \in V(P_x)$ .

Consider a vertex  $v'$  marked at point (c) for the pair  $(x, y)$  that does not belong to the solution  $X$ . By Claim 7.2,  $(V(P_u) \setminus \{v\}) \cup \{v'\}$  induces a connected subgraph of  $G$ . Since  $v'y \notin E(G)$  and  $v'x \in E(G)$ ,  $(V(P_x) \cup V(P_u) \setminus \{v\}) \cup \{v'\}$  contains a path from the two neighbors of  $y$  on the hole  $H$  whose internal vertices belong to  $V(G) \setminus (N[y] \cup \{v\})$ . By Observation 5.1, the graph  $G - v - X$  contains a hole, a contradiction.

**Case 3: None of the neighbors of  $v$  on  $H$  lies in  $M$ .** Let  $u_x$  and  $u_y$  be the two neighbors of  $v$  on  $H$ . Let  $x$  be the first vertex of  $M$  that we encounter if we traverse  $H$  starting from  $v$  in the direction towards  $u_x$ , and similarly define  $y$  for  $u_y$ . Since  $V(H) \cap M$  contains at least two vertices,  $x \neq y$ . Clearly, we have also  $v_x, v_y \notin E(G)$ . Let  $P_x$  be the subpath of  $H$  between  $v$  and  $x$  containing  $u_x$ , and let  $w_x$  be the neighbor of  $x$  on  $P_x$ . Similarly define  $P_y$  and  $w_y$ . Note that it is possible that  $w_x = u_x$  or  $w_y = u_y$ .

Consider vertices  $v_y$  and  $v_x$  marked at point (d) for the pairs  $(x, y)$  and  $(y, x)$ , respectively, that do not belong to the solution  $X$ . Note that it is possible that  $v_y = v_x$ . By Claim 7.2, both  $(V(P_x) \setminus \{v\}) \cup \{v_x\}$  and  $(V(P_y) \setminus \{v\}) \cup \{v_y\}$  induce connected graphs. As both  $v_x$  and  $v_y$  belong to the clique  $K$ , it follows that  $(V(P_x) \cup V(P_y) \cup \{v_x, v_y\}) \setminus \{v\}$  induces a connected subgraph of  $G$ ; in particular, it contains a path from  $w_x$  to  $w_y$  whose internal vertices belong to  $V(-x, -y)$ . This path, together with the subpath of  $H$  between  $x$  and  $y$  not containing  $v$ , witnesses that there exists a walk in  $G - v - X$  between two neighbors of  $y$  on  $H$  whose internal vertices do not belong to  $N[y]$ . By Observation 5.1,  $G - v - X$  contains a hole, a contradiction.  $\square$

Define  $\omega_{k,|M|} := (k+1)(|M|^3 + (k+3)|M|^2) \leq \mathcal{O}(k|M|^3)$  as a shorthand for the bound of Lemma 7.1; note that instances with  $k \geq |M|$  trivially have a solution consisting of the entire set  $M$ . Having proven Lemma 7.1, we may state the following reduction rule.

**REDUCTION RULE 3.** *Apply the algorithm of Lemma 7.1 to any maximal clique of  $G^\circ$  whose size is larger than  $\omega_{k,|M|}$ . Depending on the outcome, either add the new forced pair to  $E^h$  and to  $E(G)$  or delete an irrelevant vertex from  $G$ .*

Exhaustive application of Reduction Rule 3 ensures that the clique number  $\omega(G^\circ)$  of the chordal graph  $G^\circ$  is bounded by  $\omega_{k,|M|}$ . Recall that the number of maximal cliques in a chordal graph is linear in the number of vertices (cf. [13, section 1.2]). The maximal cliques can be identified efficiently from a clique tree, implying that Reduction Rule 3 can be applied exhaustively in polynomial time.



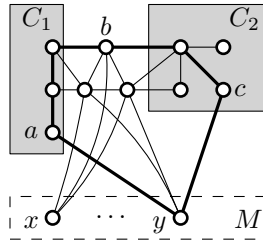


FIG. 7. Illustration for Observation 8.1. The figure shows an instance of A-CHVD with a graph  $G$  and modulator  $M$ . Gray boxes highlight components of  $G(\neg x)$ . Solid lines indicate edges in the graph. Thick edges are used to highlight a hole. Components  $C_1$  and  $C_2$  of  $G(\neg x)$  contain neighbors  $a$  and  $c$  of  $y$ . The nonneighbor  $b$  of  $y$  that belongs to  $N_{G^\circ}(C_1) \cap N_{G^\circ}(C_2)$  implies the existence of the hole drawn with thick lines, using only one vertex from  $M$ .

**8. Reducing the number of components of nonneighbors.** The main goal in this section is to reduce the number of connected components of  $G(\neg x)$  for each  $x \in M$ . Rather than reducing this number by deleting vertices, we will get rid of components of  $G(\neg x)$  by making them adjacent to  $x$ . To ensure that this process does not change the answer of the instance, we have to understand how holes in  $G$  use such components. We therefore need to introduce some notation.

**DEFINITION 8.1.** Let  $x, y \in M$  with  $x \neq y$ . Let  $\mathcal{C}_{\neg x}(y)$  be the connected components  $C$  of  $G(\neg x)$  such that  $V(C) \cap N_G(y) \neq \emptyset$  and  $N_{G^\circ}(C) \setminus N_G(y) \neq \emptyset$ .

A component  $C$  of  $\mathcal{C}_{\neg x}(y)$  contains the interior vertices of an induced path  $P_C$  between  $y$  and a vertex  $v$  of  $N_{G^\circ}(C) \setminus N_G(y)$ , which has at least two edges. Observe that if we have two components  $C_1, C_2 \in \mathcal{C}_{\neg x}(y)$  for which  $v \in (N_{G^\circ}(C_1) \cap N_{G^\circ}(C_2)) \setminus N_G(y)$  is a common neighbor, then the concatenation of the two paths  $P_{C_1}$  and  $P_{C_2}$  is a hole through  $y$  and  $v$  (see Figure 7). The key point here is that since  $C_1$  and  $C_2$  are different components of  $G(\neg x)$ , no vertex of  $C_1$  is adjacent to a vertex of  $C_2$ . The role of  $x$  in Definition 8.1 is to ensure this property.

Furthermore, we observe that such a hole, created from a concatenation of  $P_{C_1}$  and  $P_{C_2}$ , contains only one vertex of  $M$ , namely,  $y$ . This is a contradiction of the definition of an A-CHVD instance. Consequently, we have the following observation, which is crucial to our goal of reducing the number of components of  $G(\neg x)$  for all  $x \in M$ .

**OBSERVATION 8.1.** For fixed vertices  $x, y \in M$ , the sets  $\{N_{G^\circ}(C) \setminus N_G(y) \mid C \in \mathcal{C}_{\neg x}(y)\}$  are pairwise disjoint.

We are now ready to state our reduction rule.

**REDUCTION RULE 4.** Let  $(G, k, M, E^h)$  be exhaustively reduced under Reduction Rule 3. First, exhaustively apply Reduction Template 1 to sets

$$S_{\neg x, y} := M \cup \bigcup \{N_G(C) \mid C \text{ is a component of } G(\neg x) \text{ that contains a neighbor of } y\}$$

for all  $x, y \in M$ . Second, for every  $x \in M$  do the following marking process:

1. For all  $y \in M$  with  $xy \notin E(G)$ , mark up to  $(k + 1)((k + 2) \cdot \omega_{k, |M|} + |M|)^2 + 1$  components  $C$  of  $G(\neg x)$  that contain a neighbor of  $y$ .
2. For each  $y \in M$  with  $xy \in E(G)$ , mark up to  $k + 1$  components  $C$  of  $\mathcal{C}_{\neg x}(y)$ .
3. For each  $y_1, y_2 \in M$  with  $y_1 y_2 \notin E(G)$ , mark up to  $k + 1$  components  $C$  of  $G(\neg x)$  for which  $V(C) \cap N_G(y_1) \neq \emptyset$  and  $V(C) \cap N_G(y_2) \neq \emptyset$ .

If some component  $C$  of  $G(\neg x)$  is unmarked when the process finishes, then make  $x$  adjacent to all vertices of  $C$ .

If  $G(\neg x)$  has at least  $(k+1)((k+2) \cdot \omega_{k,|M|} + |M|)^2 + (k+1)(|M| + |M|^2) + 2$  components, then Reduction Rule 4 will be triggered to make  $x$  adjacent to all vertices of one such component. Hence exhaustively applying the rule bounds the number of components of  $G(\neg x)$  by  $\mathcal{O}(k^5|M|^6)$ .

Before proving the correctness of the reduction step, we give some intuition. The goal is to reduce the number of components of  $G(\neg x)$  for each  $x \in M$ . A simple way to achieve this reduction would be the following: if  $C$  is a component of  $G(\neg x)$  that remains unmarked at the end of the procedure, then remove all vertices from  $V(C)$  from the graph. However, this simpler operation would not always preserve the answer to the instance. Roughly speaking, the marking procedure ensures that if  $C$  remains unmarked, then there are no interesting holes that simultaneously pass through  $x$  and a subset of  $V(C)$  (each such hole can be rerouted through marked components), and therefore the presence of edges between  $V(C)$  and  $x$  can be flipped. There can, however, be interesting holes through vertices of  $M \setminus \{x\}$  and  $V(C)$ , which would be lost from the instance when removing all vertices of  $C$ . The removal of those vertices can therefore turn a *no*-instance into a *yes*-instance. This is why the more subtle reduction operation of making  $x$  adjacent to  $V(C)$  is needed.

**LEMMA 8.2.** *If Reduction Rule 4 reduces an instance  $(G, k, M, E^h)$  of A-CHVD to instance  $(G', k, M, E^h)$  by making  $x$  adjacent to all vertices of  $C$ , then these instances are equivalent.*

*Proof.* As adding edges to a nonchordal graph can make it chordal, and adding edges to a chordal graph can make it nonchordal, both directions of the equivalence are nontrivial.

( $\Rightarrow$ ) Assume that  $(G, k, M, E^h)$  has a solution  $X$  of size at most  $k$ , and assume for a contradiction that  $X$  is not a solution for the reduced instance. Since the instances have the same forced pairs, graph  $G - X$  is chordal but  $G' - X$  is not. Consider a hole  $H$  in  $G' - X$ . Since  $H$  is not a hole in  $G - X$ , while adding edges to a graph cannot make an existing cycle chordless, one of the edges that was added between  $x$  and  $C$  lies on  $H$ . Let  $xv \in E(H)$  with  $v \in V(C)$  be such an edge. Orient  $H$  such that  $x$  is the predecessor of  $v$  on  $H$ , and consider the successor  $y$  of  $v$ . Since  $H$  is a hole,  $xy \notin E(G')$ , and therefore  $y \notin V(C)$  since  $G'$  has all edges between  $x$  and  $V(C)$ . Moreover,  $y \notin N_G(C) \setminus M$  since  $N_G(C) \setminus M \subseteq N_G(x)$  as  $C$  is a component of  $G(\neg x)$ . Hence  $y \in M$ . Since  $C$  is an unmarked component of  $G(\neg x)$  that contains  $v \in N_G(y)$ , in point (1) we marked  $(k+1)((k+2) \cdot \omega_{k,|M|} + |M|)^2 + 1$  other components of  $G(\neg x)$  that contain a neighbor of  $y$ . Consider the set  $\mathcal{C}'$  of components of  $G(\neg x)$  that contain a  $G$ -neighbor of  $y$  and define  $S := \bigcup_{C \in \mathcal{C}'} N_{G^\circ}(C)$ .

**CLAIM 8.1.** *There is a subset  $S' \subseteq S$  of size  $k+2$  that is independent in  $G$ .*

*Proof.* Proposition 2.1 implies  $\alpha(G[S]) \geq |S|/\omega(G[S])$ . By the precondition that Reduction Rule 3 cannot be applied, we know  $\omega(G[S]) \leq \omega_{k,|M|}$ . It follows that if  $|S| \geq (k+2) \cdot \omega_{k,|M|}$ , an independent subset of size  $k+2$  exists.

Assume for a contradiction that  $|S| < (k+2) \cdot \omega_{k,|M|}$ . Let  $S^* := S \cup M$  and observe that each component of  $\mathcal{C}'$  occurs as a connected component of  $G - S^*$ . Hence  $G - S^*$  has at least  $|\mathcal{C}'| \geq (k+1)((k+2) \cdot \omega_{k,|M|} + |M|)^2 + 1 > (k+1)|S^*|^2$  connected components. But this implies that applying Template 1 with the separator  $S^* = S_{\neg x, y}$  will remove an irrelevant component, contradicting the precondition that this template is exhaustively applied for  $S_{\neg x, y}$ .  $\lrcorner$

**CLAIM 8.2.** *There are two distinct marked components  $C_1, C_2 \in \mathcal{C}'$  that avoid  $X$ , along with distinct vertices  $u_1 \in N_G(C_1) \setminus (M \cup X)$  and  $u_2 \in N_G(C_2) \setminus (M \cup X)$  such*

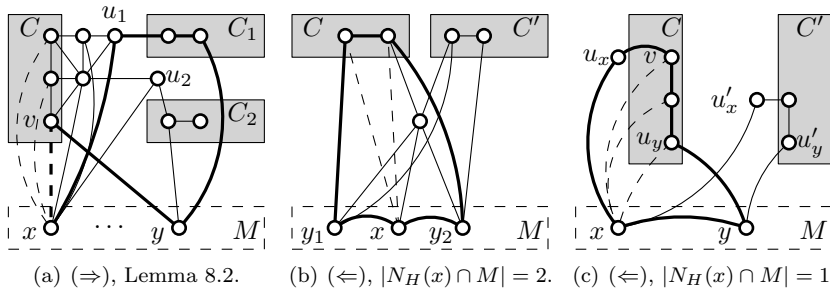


FIG. 8. Illustrations for Lemma 8.2, in the same style as Figure 7. Dotted lines represent edges between  $x$  and  $C$  added by the reduction rule to obtain  $G'$ . (a) If adding edges between  $x$  and  $C$  creates a hole in  $G' - X$  (thick edges) that was not in  $G - X$ , then a hole in  $G - X$  can be found through the marked components. (b) Case 1 of Lemma 8.2. If  $G' - X$  is chordal but  $G - X$  contains a hole through  $x$  (thick edges), then another hole can be found in  $G' - X$  through the marked component  $C'$ . (c) Case 2 of Lemma 8.2.

that there are no edges in  $G$  between  $V(C_1) \cup \{u_1\}$  and  $V(C_2) \cup \{u_2\}$ .

*Proof.* Consider an independent subset  $S'$  of  $S$  of size  $k + 2$ , whose existence is guaranteed by Claim 8.1. By definition of  $S$ , for each  $v \in S'$  there exists a component  $C_v \in \mathcal{C}'$  whose  $G$ -neighborhood contains  $v$ . For  $v \neq v' \in S'$ , the corresponding components  $C_v$  and  $C_{v'}$  are distinct. This follows from the fact that  $vv' \notin E(G)$  as  $S'$  is independent in  $G$ , together with the fact that for each component  $C$  of  $G(-x)$  the set  $N_{G^\circ}(C)$  is a clique by Proposition 5.4. Moreover,  $v$  is not adjacent to any vertex in  $C_{v'}$ , as otherwise  $v \in N_{G^\circ}(C_{v'})$  would be adjacent to  $v' \in N_{G^\circ}(C_{v'})$  as the neighborhood is a clique. Similarly,  $v'$  is not adjacent to any vertex in  $N_{G^\circ}(C_v)$ .

Consider the set  $S'$  of size  $k+2$ . For each vertex  $u \in X$ , discard  $v$  from  $S'$  if  $v = u$  or if  $u \in C_v \cap X$ . As each vertex of  $X$  discards at most one vertex, two vertices  $u_1, u_2 \in S'$  exist that are not discarded. Let  $C_1$  and  $C_2$  in  $\mathcal{C}'$  be the corresponding components of  $G(-x)$  that contain a  $G$ -neighbor of  $y$ . By the discarding process we know that  $X$  contains no vertex of  $\{u_1, u_2\} \cup V(C_1) \cup V(C_2)$ . The argumentation above shows that no vertex of  $\{u_1\} \cup V(C_1)$  is adjacent in  $G$  to a vertex of  $\{u_2\} \cup V(C_2)$ .  $\square$

Claim 8.2 allows us to derive a contradiction. Consider the marked components  $C_1, C_2$  with the corresponding vertices  $u_1$  and  $u_2$ , whose existence is guaranteed by that claim. Since each  $C_i$  contains a neighbor of  $y$ , while each  $u_i$  is in the  $G$ -neighborhood of a component of  $G(-x)$  and is therefore adjacent to  $x$ , it follows that each  $C_i \cup \{u_i\}$  contains the interior vertices of an induced  $xy$ -path in  $G$ . As the claim guarantees that these paths are vertex-disjoint, not adjacent to each other, and avoid  $X$ , their concatenation forms a hole in  $G - X$  (see Figure 8(a)). The hole avoids  $C$ , since its interior uses marked components and  $C$  is unmarked. But then this is also a hole in  $G' - X$ , a contradiction. This concludes the proof of the forward direction of the equivalence.

( $\Leftarrow$ ) Assume that  $G' - X$  is chordal but  $G - X$  is not, and let  $H$  be a hole in  $G - X$ . Since  $G - X$  is obtained from  $G' - X$  by removing all edges between  $x$  and  $V(C) \setminus X$ , hole  $H$  goes through  $x$  and the edges from  $x$  to  $C$  form chords on  $H$  in  $G'$ . Consider  $N_H[x]$  containing  $x$  and its two neighbors on  $H$ . As  $H$  is a hole in  $G - X$ , for all  $v \in V(H) \setminus N_H[x]$  we have  $xv \notin E(G)$ .

CLAIM 8.3. All vertices of  $V(H) \setminus N_H[x]$  belong to  $C$ .

*Proof.* If  $v \in V(H) \setminus (N_H[x] \cup V(C))$ , then  $xv \notin E(G)$  implies  $xv \notin E(G')$ . The

subpath of  $H$  connecting the neighbors of  $v$  on  $H$  forms a walk connecting  $N_{G'}(v)$  in  $G' - X$  whose interior avoids  $N_{G'}[v]$ , implying by Observation 5.1 that  $G' - X$  has a hole.  $\square$

We do a case distinction on where the neighbors of  $x$  on  $H$  appear in the graph.

**Case 1: Both neighbors of  $x$  on  $H$  lie in  $M$ .** Denote these two neighbors by  $y_1, y_2$ . As  $H$  is a hole in  $G - X$  we have  $y_1y_2 \notin E(G)$ , and as  $G'$  and  $G$  differ only in edges between  $x$  and  $V(C)$  we have  $y_1y_2 \notin E(G')$ . By Claim 8.3, all vertices of  $V(H) \setminus \{y_1, y_2, x\}$  belong to  $C$ , showing that both  $y_1$  and  $y_2$  have a  $G$ -neighbor in  $C$ . Hence  $C$  was a candidate for marking in point (3), and as  $C$  was not marked there is a marked component  $C'$  of  $G(-x)$  containing a  $G$ -neighbor of  $y_1$  as well as a neighbor of  $y_2$ , such that  $X$  avoids  $C'$ . There is an induced path  $P$  connecting  $y_1$  to  $y_2$  such that the interior of  $P$  is contained in  $C'$ . As the interior of  $P$  belongs to a component of  $G(-x)$ , the interior is not adjacent to  $x$  in  $G$ . The concatenation of  $P$  with  $(y_1, x, y_2)$  is therefore a hole in  $G - X$  (see Figure 8(b)). Since no vertex belongs to  $C$ , this is also a hole in  $G' - X$ , a contradiction.

**Case 2: One neighbor of  $x$  on  $H$  lies in  $M$ .** Let  $u_x$  and  $y$  be the two neighbors of  $x$  on  $H$ , with  $y \in M$  and  $u_x \in V(G^\circ)$ . Let  $v$  be the neighbor of  $u_x$  on  $H$  that is not  $x$ , and let  $u_y$  be the neighbor of  $y$  that is not  $x$ . Since the hole contains  $(y, x, u_x)$  we know  $yu_x \notin E(G)$ . By Claim 8.3 we have  $v \in V(C)$  and therefore  $u_x \in N_{G^\circ}(C)$ . Claim 8.3 also ensures  $u_y \in V(C)$ .

So  $C$  is a component of  $G(-x)$  that contains a neighbor  $u_y$  of  $y$ , and the  $G^\circ$ -neighborhood of  $C$  contains a nonneighbor  $u_x$  of  $y$ . It follows that  $C \in \mathcal{C}_{-x}(y)$ , and hence  $C$  was eligible for marking in point (2), but was not marked because some other  $k + 1$  components of  $G(-x)$  were marked. Consider the set  $\mathcal{C}_M$  of marked components, and discard all components  $C$  that contain a vertex of  $X$  either in  $C$  or in  $N_{G^\circ}(C) \setminus N_G(y)$ . By Observation 8.1, every vertex of  $X$  discards at most one component of  $\mathcal{C}_M$ , implying that some marked component  $C' \in \mathcal{C}_M$  is not discarded and  $C'$  contains a  $G$ -neighbor of  $y$ . Let  $u'_x$  be a vertex of  $N_{G^\circ}(C') \setminus (N_G(y) \cup X)$ , which exists by the discarding process, and note that this implies  $u'_xx \in E(G)$ . There is an induced path from  $y$  to  $u'_x$  whose interior belongs to  $C'$ , and therefore the path avoids  $X$ . Since the interior belongs to  $C'$ , which is a component of  $G(-x)$ , the interior is not adjacent to  $x$ . This path forms a hole in  $G - X$  together with  $x$  (see Figure 8(c)), and as the hole avoids  $C$  it is also a hole in  $G' - X$ , a contradiction.

**Case 3: None of the neighbors of  $x$  on  $H$  lie in  $M$ .** This case is the easiest. Claim 8.3 ensures that the vertices of  $H$  that are not  $x$  or its neighbors belong to  $C \subseteq G^\circ$ . If the neighbors of  $x$  on  $H$  do not lie in  $M$ , then they also lie in  $G^\circ$  so that  $x$  is the only vertex from  $H$  that belongs to  $M$ . But then  $G - (M \setminus \{x\})$  is not chordal. So  $(G, k, M, E^h)$  is not a valid instance of A-CHVD, a contradiction.  $\square$

**9. Reducing a single component.** In this section we assume that we are working with an A-CHVD instance  $(G, k, M, E^h)$  for which none of the so-far defined reduction rules apply. Recall that we denote  $G^\circ = G - M$ . We fix some clique tree  $(T, \beta)$  of  $G^\circ$ . We mark a set  $Q_0 \subseteq V(T)$  of important nodes as follows:

1. For every pair  $xy \in \binom{M}{2} \setminus E(G)$ , for every maximal clique of  $G(x, y)$  we mark a node of  $T$  that contains this clique in its bag.
2. For every  $x \in M$  and every connected component  $A$  of  $G(-x)$ , we mark a node of  $T$  whose bag contains  $N_{G^\circ}(A)$ ; recall that by Proposition 5.4 the set  $N_{G^\circ}(A)$  induces a clique in  $G^\circ$ .

Let  $Q$  be the set  $Q_0$  together with the root of  $T$  and all lowest common ancestors in  $T$  of every pair of nodes from  $Q_0$ . Standard arguments show that  $|Q| \leq 1 + 2|Q_0|$ . We define  $S_Q = \bigcup_{s \in Q} \beta(s)$ .

REDUCTION RULE 5. *Apply Reduction Template 1 with the separator  $S_Q \cup M$ .*

PROPOSITION 9.1. *If Reduction Rules 1, 2, 3, 4, and 5 have been applied exhaustively, then  $|S_Q \cup M| = \mathcal{O}(k^6|M|^{10})$  and  $G - (S_Q \cup M) = G^\circ - S_Q$  has  $\mathcal{O}(k^{13}|M|^{20})$  connected components.*

*Proof.* We first bound the size of  $Q_0$  and  $Q$ . For every pair  $xy \in \binom{M}{2} \setminus E(G)$ , we add a bag to  $Q_0$  in the first step for each maximal clique of  $G(x, y)$ . Since Reduction Rule 1 is not applicable and  $xy \notin E(G)$ , we have  $\alpha(G(x, y)) < k + 2$ . Proposition 2.1 shows that  $|V(x, y)| \leq \alpha(G(x, y)) \cdot \omega(G(x, y))$ , and since Reduction Rule 3 is not applicable, we know  $\omega(G(x, y)) \leq \omega(G^\circ) \leq \omega_{k,|M|}$ . Hence  $|V(x, y)| \leq (k + 2) \cdot \omega_{k,|M|}$ . Since the number of maximal cliques in the chordal graph  $G(x, y)$  does not exceed its order, for each nonadjacent pair  $x, y$  in  $M$  we add at most  $|V(x, y)|$  bags to  $Q_0$ . Hence the first step contributes at most  $|M|^2(k + 2) \cdot \omega_{k,|M|} = \mathcal{O}(k^2|M|^5)$  bags to  $|Q_0|$ .

In the second step, we add a bag to  $Q_0$  for each connected component of  $G(\neg x)$  for each  $x \in M$ . Since Reduction Rule 4 is not applicable, we know that each  $x \in M$  yields  $\mathcal{O}(k^5|M|^6)$  connected components in  $G(\neg x)$ . The contribution of the second step therefore dominates the size of  $Q_0$ , and we have  $|Q_0| = \mathcal{O}(k^5|M|^7)$ , which implies  $|Q| = \mathcal{O}(k^5|M|^7)$ .

Since each bag is a clique in  $G^\circ$ , it contains  $\omega_{k,|M|} = \mathcal{O}(k|M|^3)$  vertices since Reduction Rule 3 is exhaustively applied. It follows that  $|S_Q| \leq |Q| \cdot \omega_{k,|M|} = \mathcal{O}(k^6|M|^{10})$  vertices, which also forms a size bound for the entire separator  $S_Q \cup M$ . As Reduction Rule 5 is exhaustively applied, it follows that the number of components of  $G^\circ - S_Q$  is  $\mathcal{O}(k \cdot |S_Q \cup M|^2) = \mathcal{O}(k^{13}|M|^{20})$ .  $\square$

As Proposition 9.1 shows that  $|S_Q \cup M|$  and the number of components of  $G - (S_Q \cup M)$  is bounded polynomially in the parameter, it only remains to bound the size of a single connected component of  $G^\circ - S_Q$ . Let  $A$  be the vertex set of such a component, and let  $Q^A = \bigcup_{v \in A} \beta^{-1}(v)$ . Since  $G^\circ[A]$  is connected,  $T[Q^A]$  is connected as well. By the construction of  $Q$ , there exist at most two nodes  $q \in N_T(Q^A)$  such that if  $p$  is the unique neighbor of  $q$  in  $Q^A$ , then  $\beta(p) \cap \beta(q) \neq \emptyset$ . Furthermore, one of these nodes  $q$  is the parent of the topmost vertex of  $Q^A$ , and we denote it by  $q^\uparrow$ . If the second one exists, we denote it by  $q^\downarrow$ ; otherwise, we add a dummy node  $q^\downarrow$  with an empty bag as a child of one of the leaves of  $T[Q^A]$ . Consequently, we have that

$$(1) \quad N_G(A) \subseteq M \cup \beta(q^\uparrow) \cup \beta(q^\downarrow).$$

Let  $\Pi$  be the unique path from  $q^\uparrow$  to  $q^\downarrow$  in the tree  $T$ . Refer to Figure 9 for an illustration of these concepts. We start with the following simple observation.

OBSERVATION 9.1. *For every vertex  $v \in A$  and two distinct neighbors  $x \in N(v) \cap M$  and  $u \in N(v)$  we have  $xu \in E(G)$ . In particular, we have  $N(v) \cap M = N(A) \cap M$  (i.e., every vertex in  $A$  has exactly the same set of neighbors in  $M$ ), and  $N(A) \cap M$  is a clique in  $G$ .*

*Proof.* First, assume the contrary of the first claim. If  $u \in M$ , then  $v \in V(x, u)$ , but since  $xu \notin E(G)$ ,  $v$  belongs to  $\bigcup_{q \in Q_0} \beta(q)$  by the first marking step. Hence  $v \in S_Q$  and cannot belong to a component  $A$  of  $G^\circ - S_Q$ , a contradiction. If  $u \notin M$ , then  $v \in N_{G^\circ}(V(\neg x))$  since  $u \in V(\neg x)$ , and thus  $v \in \bigcup_{q \in Q_0} \beta(q)$  by the second marking step. The second claim follows from the connectivity of  $A$ .  $\square$

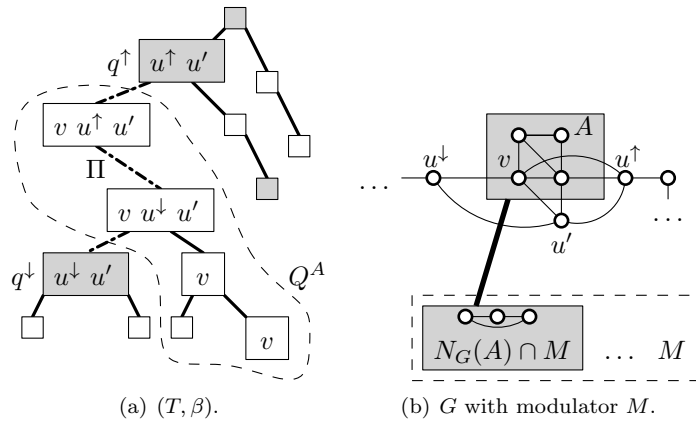


FIG. 9. Illustration of the structure of components  $A$  of  $G^\circ - S_Q$  analyzed in section 9. (a) Clique tree  $(T, \beta)$  of the graph  $G^\circ$ . The marked bags  $Q$  are drawn in gray; note that the set  $Q$  is closed with respect to taking lowest common ancestors. The bags  $Q^A$  that contain a vertex of  $A$  are highlighted by the dashed container. The subtree induced by  $Q^A$  has two neighbors in  $T$ :  $q^\uparrow$  and  $q^\downarrow$ . The unique simple  $q^\uparrow q^\downarrow$ -path  $\Pi$  in  $T$  is drawn dashed-dotted. (b) Local structure in the corresponding graph  $G$  with modulator  $M$ . The component  $A$  of  $G^\circ - S_Q$  is illustrated with a box. Its neighborhood in  $G^\circ$  is a subset of the vertices in  $\beta(q^\uparrow)$  and  $\beta(q^\downarrow)$ , both of which are cliques in  $G^\circ$ . These two cliques can overlap, as they do here in vertex  $u'$ . By Observation 9.1, all vertices in  $A$  have the same neighborhood in  $M$ , which is illustrated with a box. The thick edge between the two boxes represents all possible edges between the two sets. By Claim 9.1, any permitted hole  $H$  using a vertex of  $A$  uses  $A$  to form an induced path between some  $u^\uparrow \in \beta(q^\uparrow)$  and  $u^\downarrow \in \beta(q^\downarrow)$ . For any vertex  $v$  used on such a path, the subtree  $\beta^{-1}(v)$  uses a vertex of  $\Pi$ .

**9.1. Permitted holes through A.** In this section we analyze permitted holes in  $G$  that contain some vertices from  $A$ .

CLAIM 9.1. *Let  $H$  be a permitted hole in  $G$  that passes through  $A$ , and let  $v \in V(H) \cap A$  be arbitrary. Then*

1. both neighbors of  $v$  on  $H$  lie in  $G^\circ$ ;
2.  $H$  contains a subpath between a vertex of  $\beta(q^\uparrow)$  and a vertex of  $\beta(q^\downarrow)$  that contains  $v$  and whose interior vertices all lie in  $A$ ;
3.  $\beta^{-1}(v)$  contains at least one interior vertex of  $\Pi$ .

*Proof.* The first claim follows from Observation 9.1: if one of the neighbors of  $v$  on  $H$  lies in  $M$ , then it is adjacent to the second neighbor of  $v$  on  $H$ , a contradiction.

For the second claim, let  $P$  be a maximal subpath of  $H$  containing  $v$  such that all interior vertices lie in  $A$ ; recall that at least two vertices of  $H$  lie in  $M$ , so the endpoints of  $P$  are distinct. Furthermore, from the first claim, applied to all interior vertices of  $P$ , we infer that the endpoints of  $P$  lie in  $G^\circ$ . Since  $N(A) \subseteq M \cup \beta(q^\uparrow) \cup \beta(q^\downarrow)$  by (1), they lie in  $\beta(q^\uparrow) \cup \beta(q^\downarrow)$ . Since  $H$  needs to contain at least two vertices of  $M$ , and  $P$  does not contain any such vertices but contains  $v \in A$ , the two endpoints of  $P$  are nonadjacent, and thus one endpoint of  $P$  lies in  $\beta(q^\uparrow)$  and one lies in  $\beta(q^\downarrow)$  as both sets are cliques.

For the last claim, let  $P'$  be a subpath of  $H$  between a vertex  $u^\uparrow \in \beta(q^\uparrow)$  and  $u^\downarrow \in \beta(q^\downarrow)$  that contains  $v$ , whose existence we just established. The minimal path  $\Pi_T(u^\uparrow, u^\downarrow)$  connecting  $\beta^{-1}(u^\uparrow)$  and  $\beta^{-1}(u^\downarrow)$  in  $T$  is a subpath of  $\Pi$ . Since  $P'$  is an induced path in  $G^\circ$ , Proposition 2.4 shows that  $\beta^{-1}(v)$  contains a node of  $\Pi_T(u^\uparrow, u^\downarrow) \subseteq \Pi$ . This must be an interior node of  $\Pi$ , since  $v \in A$  does not occur in  $\beta(q^\uparrow)$  or  $\beta(q^\downarrow)$ . ┘

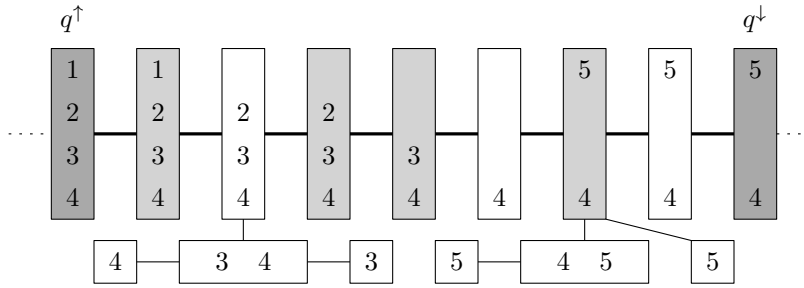


FIG. 10. Illustration of the process of marking nodes on  $\Pi$  to determine sets  $Q_i \subseteq V(\Pi)$  and  $Z_i \subseteq V(A) \subseteq V(G^\circ)$ . Path  $\Pi$  in  $(T, \beta)$  is laid out horizontally. Only the occurrence of the vertices in  $\beta(q^\uparrow) \cup \beta(q^\downarrow)$  in bags of  $T$  is shown. The bags also contain other vertices, which are hidden. Nodes in  $Q_0 = \{q^\uparrow, q^\downarrow\}$  are drawn in dark gray. The set  $Q_{i+1}$  contains  $Q_i$ , and for each vertex  $v$  in a bag  $q \in Q_i$  it contains the leftmost and rightmost occurrence of  $v$  on  $\Pi$ . The light-gray bags are added to  $Q_1$  by this process. For example, the first light-gray node from the left is added to  $Q_1$  because it has the rightmost occurrence of  $1 \in \beta(q^\uparrow \in Q_0)$ , i.e., the occurrence of  $1$  on  $\Pi$  that is closest to  $q^\downarrow$  in  $T$ .

**9.2. Important vertices in  $A$ .** We now define a set  $Z$  of important vertices in  $A$ . Intuitively, these are the only vertices that a “reasonable” solution may delete in  $A$ .

We define by induction sets  $Q_i \subseteq V(\Pi)$  for  $i \in \{0, 1, 2, \dots\}$ . Let  $Q_0 = \{q^\uparrow, q^\downarrow\}$ . Given  $Q_i$ , we first define  $Z_i = \bigcup_{q \in Q_i} \beta(q)$  and then  $Q_{i+1}$  as follows: we start with  $Q_{i+1} = Q_i$  and then, for every  $u \in Z_i$ , we add to  $Q_{i+1}$  the node on  $V(\Pi) \cap \beta^{-1}(u)$  that is closest to  $q^\uparrow$  and the node that is closest to  $q^\downarrow$ . In what follows we will work with sets  $Q_0, Q_1$ , and  $Q_2$  (see Figure 10).

Furthermore, we construct a set of edges  $R \subseteq E(\Pi)$  as follows: for every two nodes  $q_1, q_2 \in Q_2$  that are consecutive on  $\Pi$  (i.e., no node of  $\Pi$  between them belongs to  $Q_2$ ), we add to  $R$  an edge  $e$  of the subpath of  $\Pi$  between  $q_1$  and  $q_2$  that minimizes  $|\text{ADH}(e) \cap A|$  (see Figure 11).

We define the set

$$Z := A \cap \left( Z_2 \cup \bigcup_{e \in R} \text{ADH}(e) \right).$$

Since every clique in  $G^\circ$  has size  $\mathcal{O}(k|M|^3)$  (by Reduction Rule 3), we have  $|Q_i| = \mathcal{O}(k^i|M|^{3i})$  and  $|Z_i| = \mathcal{O}(k^{i+1}|M|^{3i+3})$  for every fixed  $i$ , and, consequently,  $|Z| = \mathcal{O}(k^3|M|^9)$ .

An important property of the set  $Q_1$  is the following.

**CLAIM 9.2.** *Let  $H$  be a permitted hole containing a vertex  $v \in A \setminus Z_1$ . Then both neighbors of the vertex  $v$  on the hole  $H$  lie in  $A$  as well. In particular, if  $P$  is the path obtained from Claim 9.1 for  $H$  and  $v$ , then  $v$  is not adjacent to either of the endpoints of the path  $P$ .*

*Proof.* We apply Claim 9.1 to the hole  $H$  with vertex  $v$ , obtaining a path  $P$  with endpoints  $u^\uparrow \in \beta(q^\uparrow)$  and  $u^\downarrow \in \beta(q^\downarrow)$ , and a node  $s \in V(\Pi) \cap \beta^{-1}(v)$ . Note that  $s \notin Q_1$ . Let  $s^\uparrow$  be the first node of  $Q_1$  encountered if we traverse  $\Pi$  from  $s$  in the direction of  $q^\uparrow$ , and similarly define  $s^\downarrow$  (recall that  $q^\uparrow, q^\downarrow \in Q_1$ , so these nodes are well-defined).

The crucial observation, stemming from the definition of  $Q_1$ , is that the bag of every node of  $V(\Pi) \cap \beta^{-1}(v)$  contains exactly the same subset of vertices of  $Z_0$ .

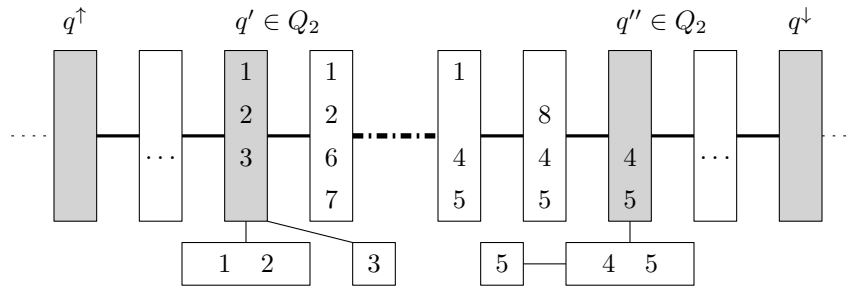


FIG. 11. The set  $Q_2$  on the  $q^\uparrow q^\downarrow$ -path  $\Pi$  is drawn in gray. For some bags on the path, the vertices from  $A$  they contain are drawn inside the bags. Between consecutive nodes from  $Q_2$  we find an edge  $e$  minimizing  $|\text{ADH}(e) \cap A|$  and add it to  $R$ . The dashed-dotted edge is added to  $R$ , since its adhesion within  $A$  is  $\{1, 2, 6, 7\} \cap \{1, 4, 5\} = \{1\} \in A$ , while the other edges between  $q'$  and  $q''$  all have at least two vertices from  $A$  in their adhesion.

Furthermore, this subset of vertices forms a clique in  $G^\circ$ , as they appear together in  $\beta(s)$ . By Claim 9.1, each neighbor  $w$  of  $v$  on  $P$  lies in  $A \cup Z_0$ . We have that  $\beta^{-1}(w)$  contains a node of  $V(\Pi)$ : this follows from Claim 9.1 if  $w \in A$ , and for  $w \in Z_0$  it follows from the facts that  $w \in \beta(q^\uparrow) \cup \beta(q^\downarrow)$  and  $w$  is adjacent to  $v \in A$ . Furthermore, since  $w$  is adjacent to  $v$ , we have that  $\beta^{-1}(w)$  actually needs to contain a vertex of  $V(\Pi) \cap \beta^{-1}(v)$ . Consequently, every neighbor  $w \in N(v) \cap Z_0$  is equal or adjacent to both neighbors of  $v$  on  $P$ , which implies that  $w$  cannot lie on  $P$ .  $\lrcorner$

**9.3. Vertices outside  $Z$  can be made undeletable.** Our goal now is to show the following lemma that says that the vertices of  $A \setminus Z$  can be made undeletable: the answer to the problem would not change if we would forbid the vertices of  $A \setminus Z$  to be part of a solution. Hence if a solution exists, there is a solution disjoint from  $A \setminus Z$ . Using this knowledge, we can later eliminate such vertices using a bypassing operation. In the following,  $\Delta$  denotes symmetric difference.

LEMMA 9.2. For every solution  $X$  to  $(G, k, M, E^h)$  there exists a solution  $X'$  with  $|X'| \leq |X|$ ,  $X \Delta X' \subseteq A$ , and  $X' \cap A \subseteq Z$ .

*Proof.* Assume the contrary, and let  $X$  be a counterexample with minimum number of vertices from  $A \setminus Z$ . Let  $v \in X \cap (A \setminus Z)$ . By the minimality of  $X$ , we have that  $G - (X \setminus \{v\})$  contains a hole  $H$  passing through  $v$ . Since  $X$  is a solution to  $(G, k, M, E^h)$  and  $v \notin M$ , this hole is permitted. We apply Claim 9.1: let  $P$  be a subpath of  $H$  whose existence is asserted in the second claim, and let  $r \in \beta^{-1}(v) \cap V(\Pi)$ .

Let  $x$  be an arbitrary vertex of  $V(H) \cap M$ . Note that by Claim 9.1 we have  $vx \notin E(G)$  and, consequently, by Observation 9.1,  $A \cap N(x) = \emptyset$ .

Since  $v \notin Z$  we have  $\beta^{-1}(v) \cap Q_2 = \emptyset$ , and in particular  $r \notin Q_2$ . Let  $r^\uparrow$  be the first vertex of  $Q_2$  encountered if we traverse  $\Pi$  from  $r$  in the direction of  $q^\uparrow$ , and similarly define  $r^\downarrow$  (recall that  $q^\uparrow, q^\downarrow \in Q_2$ , so these nodes are well-defined). We claim the following.

CLAIM 9.3.  $X$  contains  $\text{ADH}(e_X) \cap A$  for some edge  $e_X$  that lies on  $\Pi$  between  $r^\uparrow$  and  $r^\downarrow$ .

*Proof.* Assume the contrary. Construct a set  $B \subseteq A \setminus X$  as follows: for every edge  $e$  that lies on  $\Pi$  between  $r^\uparrow$  and  $r^\downarrow$ , insert into  $B$  an arbitrary vertex of  $\text{ADH}(e) \cap (A \setminus X)$ . Since every bag of  $T$  is a clique,  $G[B]$  is connected.

Let  $P^\uparrow$  be the subpath of  $P$  between  $v$  and the endpoint of  $P$  in  $\beta(q^\uparrow)$ , and let



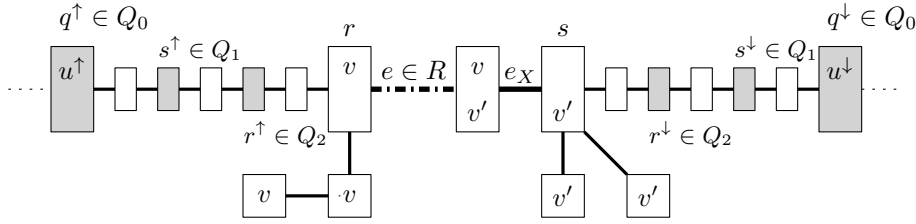


FIG. 12. Illustration of the situation for Claim 9.5. The set  $Q_2 \supseteq Q_1 \supseteq Q_0$  on the  $q^\uparrow q^\downarrow$ -path  $\Pi$  is drawn in light gray. A marked edge  $e \in R$  between  $r^\uparrow$  and  $r^\downarrow$  is drawn dashed-dotted. An edge  $e_X$  for which  $\text{ADH}(e_X) \cap A \subseteq X$  is also shown.

$P^\downarrow$  be the subpath of  $P$  between  $v$  and the endpoint of  $P$  in  $\beta(q^\downarrow)$ . Since  $r^\uparrow$  lies on the unique path in  $T$  between  $\beta^{-1}(v)$  and  $q^\uparrow$ ,  $P^\uparrow$  contains a vertex  $w^\uparrow \in \beta(r^\uparrow)$ ; note that  $w^\uparrow \neq v$  as  $v \notin \beta(r^\uparrow)$ . Similarly define  $w^\downarrow \in \beta(r^\downarrow)$ . By the construction of  $B$ ,  $w^\uparrow, w^\downarrow \in N[B]$ .

We infer that  $G[V(H) \setminus \{v\} \cup B]$  contains a walk between the neighbors of  $x$  on the hole  $H$  with interior vertices in  $V(G) \setminus N[x]$  (recall that  $B \subseteq A$  and  $A \cap N[x] = \emptyset$ ). By Observation 5.1,  $G - X$  contains a hole, which is easily seen to be permitted, a contradiction.  $\lrcorner$

By construction of  $R$  in the beginning of section 9.2, there is an edge  $e$  on  $R$  that lies between  $r^\uparrow, r^\downarrow \in Q_2$  on  $\Pi$ . Let  $X' = (X \setminus (\text{ADH}(e_X) \cap A)) \cup (\text{ADH}(e) \cap A)$ , where the edge  $e_X$  is given by Claim 9.3. By the choice of  $e$  in the process of constructing  $R$ , we have  $|X'| \leq |X|$ . Clearly, also  $X \Delta X' \subseteq A$ . Since  $\text{ADH}(e \in R) \cap A \subseteq Z$ , to prove that  $X'$  contains strictly fewer vertices of  $A \setminus Z$  than  $X$  it suffices to show the following.

CLAIM 9.4. *If  $e_X$  is an edge whose existence is asserted by Claim 9.3, then  $v \in \text{ADH}(e_X)$ .*

*Proof.* By Claim 9.2 for  $H$  and  $v$ , neither  $u^\uparrow$  nor  $u^\downarrow$  is adjacent to  $v$ ; in particular, they do not belong to  $\beta(r)$ . By the definition of  $Q_1$ , neither of them belongs to any bag on  $\Pi$  between  $r^\uparrow$  and  $r^\downarrow$ , exclusive.

Consequently,  $u^\uparrow, u^\downarrow \notin \text{ADH}(e_X)$ , and thus  $P$  contains a vertex of  $\text{ADH}(e_X) \cap A$ . Since  $H$  is a hole in  $G - (X \setminus \{v\})$ , and  $\text{ADH}(e_X) \cap A \subseteq X$ , this vertex is the vertex  $v$ .  $\lrcorner$

To reach a contradiction, it remains to show that  $X'$  is a solution to  $(G, k, M, E^h)$  as well. Since  $X \Delta X' \subseteq A$ ,  $X'$  satisfies all the constraints imposed by the set  $E^h$ . Thus, it remains to prove that  $G - X'$  is chordal. Assume the contrary, and let  $H'$  be a hole in  $G - X'$ .

Since  $G - X$  is chordal,  $H'$  needs to contain a vertex  $v' \in A \cap (\text{ADH}(e_X) \setminus \text{ADH}(e))$ . Since  $e$  and  $e_X$  are edges of  $\Pi$  between two consecutive vertices of  $Q_2$  (namely,  $r^\uparrow$  and  $r^\downarrow$ ), it follows from the definition of  $Q_2$  that  $\text{ADH}(e) \cap Z_1 = \text{ADH}(e_X) \cap Z_1$ . We infer that  $v' \notin Z_1$ . Refer to Figure 12 for an illustration.

We apply Claim 9.1 to the hole  $H'$  with vertex  $v'$ , obtaining a path  $P'$  with endpoints  $u^\uparrow \in \beta(q^\uparrow)$  and  $u^\downarrow \in \beta(q^\downarrow)$ , and a node  $s \in V(\Pi) \cap \beta^{-1}(v')$ . Note that  $s \notin Q_1$ . Let  $s^\uparrow$  be the first node of  $Q_1$  encountered if we traverse  $\Pi$  from  $s$  in the direction of  $q^\uparrow$ , and similarly define  $s^\downarrow$  (recall that  $q^\uparrow, q^\downarrow \in Q_1$ , so these nodes are well-defined). Observe that since  $v' \in \text{ADH}(e_X)$ , but  $s^\uparrow, s^\downarrow \notin \beta^{-1}(v')$ , we have that  $r^\uparrow, r^\downarrow, e$ , and  $e_X$  lie on the subpath of  $\Pi$  between  $s^\uparrow$  and  $s^\downarrow$ ; possibly  $r^\uparrow = s^\uparrow$  or  $r^\downarrow = s^\downarrow$ . We now show the following corollary of Claim 9.2.

CLAIM 9.5. *The vertex  $u^\uparrow$  is not present in any bag on  $\Pi$  between  $s^\uparrow$  (exclusive) and  $q^\downarrow$  (inclusive); i.e.,  $\beta^{-1}(u^\uparrow) \cap V(\Pi)$  is contained in the subpath of  $\Pi$  between  $s^\uparrow$  and  $q^\uparrow$ . Symmetrically, the vertex  $u^\downarrow$  is not present in any bag on  $\Pi$  between  $s^\downarrow$  (exclusive) and  $q^\uparrow$  (inclusive); i.e.,  $\beta^{-1}(u^\downarrow) \cap V(\Pi)$  is contained in the subpath of  $\Pi$  between  $s^\downarrow$  and  $q^\downarrow$ .*

*Proof.* By Claim 9.2 for  $H'$  and  $v' \notin Z_1$ , neither  $u^\uparrow$  nor  $u^\downarrow$  is adjacent to  $v'$ ; in particular, they do not belong to  $\beta(s)$ . By the definition of  $Q_1$ , neither of them belongs to any bag on  $\Pi$  between  $s^\uparrow$  and  $s^\downarrow$ , exclusive. The statement for  $u^\uparrow$  follows from the fact that  $\beta^{-1}(u^\uparrow) \cap V(\Pi)$  is connected and contains  $q^\uparrow$ , and similarly for  $u^\downarrow$ .  $\lrcorner$

Since all interior vertices of  $P'$  lie in  $A$ , from Claim 9.5 it follows that  $P'$  needs to contain a vertex of  $\text{ADH}(e) \cap A$ , as  $e$  lies between  $s^\uparrow$  and  $s^\downarrow$  on  $\Pi$ . Consequently,  $P'$  is hit by  $X'$ , a contradiction. This concludes the proof of the lemma.  $\square$

**9.4. Reducing  $A \setminus Z$ .** Our goal now is to reduce  $A \setminus Z$ . While by Lemma 9.2 we know that in a yes-instance there is a solution disjoint with  $A \setminus Z$ , we need to be careful, as some permitted holes may contain vertices from  $A \setminus Z$ . We first filter out a simple case.

OBSERVATION 9.2. *Any vertex of  $A \setminus \bigcup_{q \in V(\Pi)} \beta(q)$  is irrelevant.*

*Proof.* Let  $v \in A \setminus \bigcup_{q \in V(\Pi)} \beta(q)$ . By Claim 9.1, no permitted hole passes through  $v$ . Consequently, any solution  $X$  to  $(G - v, k, M, E^h)$  is a solution to  $(G, k, M, E^h)$  as well: as  $v \notin M$ ,  $X$  satisfies all constraints imposed by  $E^h$ , while  $G$  and  $G - v$  contain the same permitted holes.  $\square$

This allows us to formulate the following reduction rule.

REDUCTION RULE 6. *If  $A \setminus \bigcup_{q \in V(\Pi)} \beta(q)$  is nonempty, remove any vertex from this set.*

Other vertices of  $A \setminus Z$  need to be handled differently. For a vertex  $v \in V(G)$ , by *bypassing the vertex  $v$*  we mean the following operation: we first turn  $N_G(v)$  into a clique, and then delete  $v$  from  $G$ . We first observe that bypassing a vertex preserves being chordal.

OBSERVATION 9.3. *A graph constructed from a chordal graph by bypassing a vertex is chordal as well.*

*Proof.* Let  $G_1$  be a chordal graph with clique tree  $(T_1, \beta_1)$ , let  $v \in V(G_1)$ , and let  $G_2$  be constructed from  $G_1$  by bypassing  $v$ . Consider the following tree decomposition  $(T_2, \beta_2)$ : we take  $T_2 = T_1$  and then, for every  $p \in V(T_2)$ , we put  $\beta_2(p) = N(v)$  if  $v \in \beta_1(p)$ , and  $\beta_2(p) = \beta_1(p)$  otherwise. A direct check shows that  $(T_2, \beta_2)$  is a tree decomposition of  $G_2$  in which every bag induces a clique. Thus,  $G_2$  is chordal, as claimed.  $\square$

LEMMA 9.3. *Assume all previous reductions are inapplicable. Let  $v \in A \setminus Z$ , and let  $G'$  be constructed from  $G$  by first turning  $N_G(v)$  into a clique, and then removing the vertex  $v$ . Then the instances  $(G, k, M, E^h)$  and  $(G', k, M, E^h)$  are equivalent.*

*Proof.* In one direction, assume  $(G, k, M, E^h)$  is a yes-instance. Lemma 9.2 asserts that there exists a solution  $X$  that does not contain  $v$ . Observation 9.3, applied to  $G - X$  and vertex  $v$ , implies that  $G' - X$  is chordal as well. Consequently,  $X$  is a solution to  $(G', k, M, E^h)$  as well.

The other direction is significantly more involved. Intuitively, we want to rely on Claim 9.2 to show that  $v$  can only interact with permitted holes “locally,” and thus

any permitted hole in  $G$  has its counterpart in  $G'$ : possibly shorter but still being a hole. We start with showing that the new edges in  $G'$  are actually only in  $G[A]$ .

**CLAIM 9.6.** *If  $u, w$  are two distinct nonadjacent neighbors of  $v$  in  $G$ , then  $u, w \in A \setminus Z_1$ .*

*Proof.* Pick  $u, w$  as in the statement. Observation 9.1 implies that  $u, w \notin M$ . Since Reduction Rule 6 is not applicable to  $v$ , there exists  $r \in \beta^{-1}(v) \cap V(\Pi)$ . Since  $v \notin Z$ , we have  $r \notin Q_2$ . Let  $r^\uparrow$  be the first vertex of  $Q_2$  encountered if we traverse  $\Pi$  from  $r$  in the direction of  $q^\uparrow$ , and similarly define  $r^\downarrow$  (recall that  $q^\uparrow, q^\downarrow \in Q_2$ , so these nodes are well-defined).

By contradiction, assume  $u \in Z_1$ . Note that all nodes of  $\beta^{-1}(v) \cap V(\Pi)$  lie on  $\Pi$  between  $r^\uparrow$  and  $r^\downarrow$ , exclusive. By the definition of  $Q_2$ , every bag at every node of  $\beta^{-1}(v) \cap V(\Pi)$  contains exactly the same subset of  $Z_1$ ; in particular, it contains  $u$ . Since Reduction Rule 6 does not apply to  $w$ ,  $w$  is present in some bag on  $\Pi$ ; since  $vw \in E(G)$ ,  $w$  is present in some bag of  $\beta^{-1}(v) \cap V(\Pi)$ . We infer that  $u$  and  $w$  meet in some bag of  $\beta^{-1}(v) \cap V(\Pi)$ , a contradiction to the assumption that they are nonadjacent.  $\lrcorner$

Assume now that  $(G', k, M, E^h)$  is a yes-instance, and let  $X$  be a solution. We claim that  $X$  is a solution to  $(G, k, M, E^h)$  as well; clearly it satisfies all the constraints imposed by  $E^h$ . By contradiction, let  $H$  be a hole in  $G - X$ ; clearly,  $H$  is permitted. Since  $H$  is not a hole in  $G' - X$ , either it contains the vertex  $v$  or one of the edges of  $E(G') \setminus E(G)$  is a chord of  $H$  in  $G' - X$ . We infer that in both cases  $H$  contains a vertex  $w \in A \setminus Z_1$ : in the first case we take  $w = v$ , in the second case it follows from Claim 9.6.

We apply Claim 9.1 to the hole  $H$  and the vertex  $w$ , obtaining a path  $P$  with endpoints  $u^\uparrow \in \beta(q^\uparrow)$  and  $u^\downarrow \in \beta(q^\downarrow)$ . By the construction of  $G'$ , the subgraph  $G'[V(P) \setminus \{v\}]$  is connected. By Claim 9.2, the vertex  $w$  is nonadjacent to neither  $u^\uparrow$  nor  $u^\downarrow$ ; thus  $u^\uparrow u^\downarrow \notin E(G')$  and the shortest path  $P'$  between  $u^\uparrow$  and  $u^\downarrow$  in  $G'[V(P) \setminus \{v\}]$  is of length at least two. We infer that if we replace  $P$  with  $P'$  on the hole  $H$ , we obtain a hole in  $G' - X$ , a contradiction.  $\square$

Lemma 9.3 justifies the following reduction rule.

**REDUCTION RULE 7.** *If there exists a vertex  $v \in A \setminus Z$ , then bypass  $v$ .*

To conclude, observe that if the reductions introduced in this section are inapplicable, we have  $|A| = \mathcal{O}(k^3|M|^9)$  for every connected component  $A$  of  $G - (M \cup S_Q) = G^\circ - S_Q$ . Since Proposition 9.1 shows that the number of such components is bounded by  $\mathcal{O}(k^{13}|M|^{20})$  after exhaustive applications of all the reduction rules and  $|S_Q| = \mathcal{O}(k^6|M|^{10})$ , we obtain a final size bound of  $\mathcal{O}(k^{16}|M|^{29})$  vertices in an exhaustively reduced instance. All reduction rules can be applied in polynomial time. Each application either reduces the number of vertices, or adds an edge to the graph. It follows that an instance can be exhaustively reduced by  $\mathcal{O}(|V(G)| + |E(G)|)$  applications of a reduction rule, which means the entire kernelization runs in polynomial time. This concludes the proof of Theorem 5.2.

**10. Approximation algorithm.** This section is devoted to the proof of Theorem 1.2. We start with an informal overview. Let  $(G, k)$  be a CHVD instance. First, we can assume that  $\log n \leq k \log k$ , as otherwise the algorithm of Cao and Marx [19] solves  $(G, k)$  in polynomial time.

We observe that in a yes-instance there exists a balanced vertex cut<sup>1</sup> that consists of a clique and at most  $k$  vertices. To this end, it suffices to take a minimum solution and a

<sup>1</sup>That is, a vertex set  $S$  such that each connected component of  $G - S$  has at most  $c \cdot |V(G)|$  vertices for some constant  $c < 1$ .

“central” maximal clique of the remaining chordal graph  $G'$ , that is, a maximal clique  $K$  of  $G'$  such that every connected component of  $G' - K$  is of size at most  $|V(G')|/2$ . We can find an approximate variant of such a balanced vertex cut in the following fashion. Since we can assume that  $G$  is  $C_4$ -free (e.g., by greedily deleting whole  $C_4$ 's in  $G$ ), it has  $\mathcal{O}(n^2)$  maximal cliques [27, Proposition 2]. We iterate over all choices of a maximal clique  $K$  in  $G$  and apply the approximation algorithm for a balanced vertex cut of [28, section 6.2] to  $G - K$ . In this manner, we can always find a balanced vertex cut in  $G$  consisting of a clique and  $\mathcal{O}(k\sqrt{\log k})$  vertices (or conclude that we have a no-instance).

We iteratively apply the above partitioning algorithm to a connected component of  $G$  that is not chordal and remove the output cut. A simple charging argument shows that in a yes-instance such a procedure finishes after  $\ell = \mathcal{O}(k \log n)$  steps. In the end, we obtain a partition

$$V(G) = A_0 \uplus K_1 \uplus K_2 \uplus \cdots \uplus K_\ell \uplus X_0,$$

where  $G[A_0]$  is chordal, every  $K_i$  is a clique, and  $|X_0| = \mathcal{O}(\ell k \sqrt{\log k})$ . We put  $X_0$  into the solution we are constructing. To obtain the desired approximation algorithm, it is sufficient to tackle the special case where we are given a partition  $V(G) = A \uplus B$ , where  $G[A]$  is chordal and  $B$  is a clique in  $G$ . We can initially apply an algorithm for this special case to the graph  $G[A_0 \uplus K_1]$ , obtaining a solution  $X_1$ . We apply it again to  $G[((A_0 \uplus K_1) \setminus X_1) \uplus K_2]$  to obtain  $X_2$ , and then we repeat. After  $\ell$  iterations of finding an approximate solution and adding back the next clique, we have found an approximate solution to the original input.

To find an approximate solution in the mentioned special case, we again apply the balanced partitioning approach, but with a different toolbox. Let  $L$  be a “central” maximal clique of  $G[A]$ . If we find a small (bounded polynomially in the value of an LP relaxation) set of vertices that hits all holes passing through  $L$ , then we can delete such a set, discard  $L$  (as no holes now pass through  $L$ ), and recurse on connected components of remaining graph.

Thus, we can now focus only on holes passing through  $L$  (see Figure 13). We obtain an orientation  $G^\downarrow$  of  $G$  as follows: we root a clique tree of  $G[A]$  in  $L$  and orient every edge of  $G$  “downwards”; that is,  $uv \in E(G)$  becomes  $(u, v) \in E(G^\downarrow)$  if  $\text{TOP}(u)$  is an ancestor of  $\text{TOP}(v)$ , breaking ties arbitrarily for  $\text{TOP}(u) = \text{TOP}(v)$ . Then every hole  $H$  passing through  $L$  (and through  $B$  as  $G[A]$  is chordal) decomposes into two directed paths in  $G^\downarrow$ ,  $Q_1$  and  $Q_2$ , that (except for some corner cases when one of the paths  $Q_i$  consists of only a few vertices) go from  $L$  to  $N(B)$  through different components of  $G[A] - L$ . This fact allows us to claim that if we substitute  $Q_i$  with a different directed path in  $G^\downarrow$  with the same first few vertices and the same last vertex, we obtain again a hole in  $G$ . As a corollary, a (constant multiple of) the LP relaxation needs to separate (in  $G^\downarrow$ ) the first few vertices of  $Q_i$  from the last vertex of  $Q_i$  for some  $i \in \{1, 2\}$ . This observation allows us to reduce the task of hitting all holes passing through  $L$  to a MULTICUT instance in  $G^\downarrow$ , for which a constant multiple of the original LP relaxation is a feasible fractional solution. In this MULTICUT instance, we can apply an approximation algorithm of Gupta [39] to round the aforementioned fractional solution to an integral one.

The rest of this section is organized as follows. First, in section 10.1 we recall the definition of the MULTICUT problem and formally define LP relaxations of MULTICUT and CHVD. We also recall the approximation algorithm of Gupta [39] that we will use. In section 10.2 we focus on approximating CHVD in a special case of graphs  $G$  where  $V(G)$  is partitioned into  $A \uplus B$  such that  $G[A]$  is chordal and  $B$  is a clique. Finally, we wrap up the approximation algorithm in section 10.3.

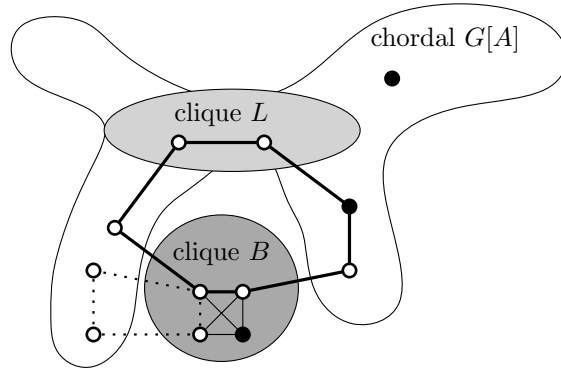


FIG. 13. Schematic overview for the key special case in the approximation algorithm: given a partition  $V(G) = A \uplus B$  such that  $G[A]$  is chordal and  $B$  is a clique, find a small vertex set (in black) intersecting all holes (thick lines) that pass through a “central” maximal clique  $L$  of  $G[A]$ . Except for easily handled special cases, the relevant holes through  $L$  split into two paths from  $L$  to  $N_G(B)$  through different components of  $G[A] - L$ . The holes passing through  $B$  but not  $L$  (dotted) are hit when recursing for the different components of  $G[A] - L$ . Since  $B$  is a clique, a hole in  $G$  avoiding  $L$  uses at most one component of  $G[A] - L$ .

**10.1. MULTICUT and LP relaxations.** An instance of the MULTICUT problem consists of a directed graph  $G$  and a family  $\mathcal{T} \subseteq V(G) \times V(G)$  of terminal pairs. A set  $X \subseteq V(G)$  is a feasible solution to a multicut instance  $(G, \mathcal{T})$  if, for every  $(s, t) \in \mathcal{T}$ , there is no path from  $s$  to  $t$  in  $G - X$ . The MULTICUT problem asks for a feasible solution of minimum cardinality. That is, we focus on the unweighted node deletion variant of MULTICUT and note that the set  $X$  may contain terminals.

Given a MULTICUT instance  $(G, \mathcal{T})$ , a fractional solution is a function  $\mathbf{x}^* : V(G) \rightarrow [0, +\infty)$  such that for every  $(s, t) \in \mathcal{T}$  and every path  $P$  from  $s$  to  $t$  in  $G$ , we have

$$\mathbf{x}^*(P) := \sum_{v \in V(P)} \mathbf{x}^*(v) \geq 1.$$

The equation above is a linear constraint on values of  $\mathbf{x}^*$  that defines the polytope of fractional solutions. Clearly, a feasible solution to the MULTICUT instance  $(G, \mathcal{T})$  induces a fractional solution that assigns 1 to the vertices of the solution. Furthermore, the separation oracle of the aforementioned polytope is straightforward to implement by a shortest-path algorithm such as Dijkstra’s. Consequently, using the ellipsoid method, in polynomial time we can find a fractional solution of total value arbitrarily close to the minimum possible.

Very similarly we define a fractional solution to CHVD on an undirected graph  $G$ : it is a function  $\mathbf{x}^* : V(G) \rightarrow [0, +\infty)$  such that for every hole  $H$  in  $G$  we have

$$\mathbf{x}^*(H) := \sum_{v \in V(H)} \mathbf{x}^*(v) \geq 1.$$

Again, the equation above defines a polytope containing all feasible solutions to CHVD on  $G$ , and there is a simple separation oracle for this polytope.<sup>2</sup>

<sup>2</sup>Iterate over all possibilities of three consecutive vertices  $v_1, v_2, v_3$  on the hole  $H$ , and compute the shortest path from  $v_1$  to  $v_3$  in  $G - N[v_2]$  with respect to the distances defined by the current values of  $\mathbf{x}^*$ .

For a fractional solution  $\mathbf{x}^*$  (either to MULTICUT or CHVD), we use the notation  $\mathbf{x}^*(X)$  for a vertex set  $X \subseteq V(G)$  for  $\sum_{v \in X} \mathbf{x}^*(v)$ . Furthermore, we use  $|\mathbf{x}^*|$  for  $\mathbf{x}^*(V(G))$  and  $\mathbf{x}^*(H) = \mathbf{x}^*(V(H))$  for a subgraph  $H$  of  $G$  (e.g., a path).

MULTICUT is NP-hard even in undirected graphs with three terminal pairs [23]. The best known approximation ratio for undirected graphs is  $\mathcal{O}(\log n)$  [34], but for general directed graphs only  $\mathcal{O}(\sqrt{n})$ -approximation is known [39]. We will need the following result from the work of Gupta [39].

**THEOREM 10.1** (see [39]). *Given a MULTICUT instance  $(G, \mathcal{T})$  and a fractional solution  $\mathbf{x}^*$ , one can compute in polynomial time an integral solution of size  $\mathcal{O}(|\mathbf{x}^*|^2)$ .*

Although [39] tackles the edge deletion variant of MULTICUT, note that in directed graphs the edge and node deletion variants are equivalent via standard reductions.

Finally, we will frequently need the following observation.

**THEOREM 10.2** (see [38, Theorem 1]). *For every graph  $G$ , its tree decomposition  $(T, \beta)$ , and a function  $f : V(G) \rightarrow \mathbb{R}_{\geq 0}$ , there exists a node  $t \in V(T)$  such that every connected component  $C$  of  $G - \beta(t)$  satisfies  $f(V(C)) \leq f(V(G))/2$ , where  $f(X) = \sum_{v \in X} f(v)$  for  $X \subseteq V(G)$ . In particular, for every chordal graph  $G$  and a function  $f : V(G) \rightarrow \mathbb{R}_{\geq 0}$  there exists a maximal clique  $X$  in  $G$  such that every connected component  $C$  of  $G - X$  satisfies  $f(V(C)) \leq f(V(G))/2$ .*

**10.2. Adding a clique to a chordal graph.** In this section we develop an approximation algorithm for the following special case.

**LEMMA 10.3.** *Assume we are given a graph  $G$ , a fractional solution  $\mathbf{x}^*$  to CHVD on  $G$ , and a partition  $V(G) = A \uplus B$ , such that  $G[A]$  is chordal and  $G[B]$  is complete. Then one can in polynomial time find an integral solution to CHVD on  $G$  of size  $\mathcal{O}(|\mathbf{x}^*|^2 \log |\mathbf{x}^*|)$ .*

As an intermediate step towards Lemma 10.3, we show the following.

**LEMMA 10.4.** *Let  $G$ ,  $\mathbf{x}^*$ ,  $A$ , and  $B$  be as in the statement of Lemma 10.3. Furthermore, let  $L$  be a maximal clique of  $G[A]$ . Furthermore, assume that  $\mathbf{x}^*(v) = 0$  for every  $v \in B$  and  $\mathbf{x}^*(v) < 1/10$  for every  $v \in A$ . Then one can in polynomial time find a set  $X \subseteq V(G)$  of size at most  $\mathcal{O}(|\mathbf{x}^*|^2)$  such that  $G - X$  does not contain a hole passing through  $L$ .*

*Proof.* We fix a clique tree  $(T, \beta)$  of  $G[A]$ , and root it in a node  $\text{ROOT}$  with  $\beta(\text{ROOT}) = L$ . We define a partial order  $\preceq_0$  on  $A$  as  $v \prec_0 u$  if  $\text{TOP}(v)$  is an ancestor of  $\text{TOP}(u)$ , and we pick  $\preceq$  to be an arbitrary total order extending  $\preceq_0$ . We obtain an orientation  $G^\downarrow$  of  $G[A]$  by orienting the edges of  $G$  so that  $\preceq$  becomes a topological ordering of  $G^\downarrow$  (i.e., we orient  $uv \in E(G)$  as  $(u, v)$  if  $u \preceq v$ ). In  $G^\downarrow$ , we define a distance measure  $\text{dist}_{G^\downarrow, \mathbf{x}^*}$  as distances with respect to the cost function  $\mathbf{x}^*$ ; that is,  $\text{dist}_{G^\downarrow, \mathbf{x}^*}(u, v)$  is the minimum value of  $\mathbf{x}^*(P)$  over all  $uv$ -paths  $P$  in  $G^\downarrow$ . Finally, we define

$$\mathcal{T} = \{(u, v) \in A \times A : \text{dist}_{G^\downarrow, \mathbf{x}^*}(u, v) \geq 1/10\}.$$

Clearly,  $10\mathbf{x}^*$  is a fractional solution to the MULTICUT instance  $(G^\downarrow, \mathcal{T})$ . By applying the algorithm of Theorem 10.1, we obtain a set  $X \subseteq V(G)$  of size  $\mathcal{O}(|\mathbf{x}^*|^2)$  that is a solution to MULTICUT on  $(G^\downarrow, \mathcal{T})$ . To finish the proof of the lemma, it suffices to show that  $X$  is also good for our purposes, that is,  $G - X$  does not contain any hole passing through  $L$ .

Assume the contrary, and let  $H$  be any such hole in  $G - X$ . Since  $G[A]$  is chordal,  $H$  contains at least one vertex of  $B$ . Since both  $L$  and  $B$  are cliques in  $G$ ,

$H$  contains at most two vertices of each of these cliques. Consequently, there exists a unique decomposition of  $H$  into two paths  $P^1, P^2$ , such that every path  $P^i$  has an endpoint  $x^i \in L$ , and endpoint  $y^i \in B$ , and all internal vertices in  $A \setminus L$ . We have  $V(P^1) \cup V(P^2) = V(H)$ . Note that it is possible that  $x^1 = x^2$  or  $y^1 = y^2$ .

Let  $u, v$  be two consecutive internal vertices on  $P^i$ , such that  $u$  is closer to  $x^i$  on  $P^i$  than  $v$ . Observe that  $\text{TOP}(u)$  is an ancestor of  $\text{TOP}(v)$ , as otherwise  $v$  is adjacent in  $G$  to the predecessor of  $u$  on  $P^i$ , a contradiction to the assumption that  $H$  is a hole. Consequently,  $P^i - \{x^i, y^i\}$  is a directed path in  $G^\downarrow$ .

Let  $u_j^i$  be the  $j$ th internal vertex on  $P^i$  starting from  $x^i$ , that is, the consecutive vertices on  $P^i$  are  $x^i, u_1^i, u_2^i, \dots, u_{\ell^i}^i, y^i$ , where  $\ell^i = |P^i| - 1$ .

Since  $X$  is a multicut for terminals  $\mathcal{T}$  and  $P^i - \{x^i, y^i\}$  is a directed path in  $G^\downarrow - X$  for  $i \in \{1, 2\}$ , for any  $1 \leq a < b \leq \ell^i$  the pair  $(u_a^i, u_b^i)$  is not a terminal pair. By definition of  $\mathcal{T}$ , this implies that there is a path  $P$  in  $G^\downarrow$  from  $u_a^i$  to  $u_b^i$  with  $\mathbf{x}^*(P) < 1/10$ . We define two alternative paths  $\hat{P}^i$  to derive a contradiction. If  $P^i$  has at most two internal vertices, then  $\hat{P}^i := P^i$ . Otherwise, define  $\hat{P}^i$  as  $(x^i, u_1^i, Q^i, y^i)$ , where  $Q^i$  is a path in  $G^\downarrow$  from  $u_2^i$  to  $u_{\ell^i}^i$  with  $\mathbf{x}^*(Q^i) < 1/10$ . No vertex on  $Q^i$  is adjacent to  $x^1$  in  $G$ : since  $u_2^i$  is not adjacent to  $x^1$  on the hole  $H$ , we know  $u_2^i$  and  $x^1$  do not occur in a common bag. This implies that  $x^1$ , which is in the root bag, does not belong to  $\text{TOP}(u_2^i)$  and does not occur in the subtree rooted there; all vertices that can be reached from  $u_2^i$  in  $G^\downarrow$  have root bags that are descendants of  $\text{TOP}(u_2^i)$ .

The concatenation of  $\hat{P}^1$  with  $P^2$  forms a (possibly nonsimple) walk in  $G$  that visits  $x^1$  exactly once. Consider the walk  $W$  obtained by removing  $x^1$  from this cycle. It connects the successor of  $x^1$  on  $H$  to the predecessor of  $x^1$  on  $H$ . All internal vertices of the walk are nonadjacent to  $x^1$ , which follows from our claims on  $Q^i$  and the fact that  $H$  is a hole. Hence we can shortcut  $W$  to an induced path  $P$  connecting the predecessor and successor of  $x^i$  on  $H$ , to obtain a hole  $H'$  in  $G$ . Since  $\mathbf{x}^*(v) < 1/10$  for each vertex  $v \in G$ , and  $\mathbf{x}^*(Q^i) < 1/10$ , it follows that  $\mathbf{x}^*(\hat{P}^i) < 3/10 + 1/10 = 4/10$ . Hence  $\mathbf{x}^*(H') < 8/10 < 1$ , a contradiction to the assumption that  $\mathbf{x}^*$  is a fractional solution to CHVD on  $G$ .  $\square$

*Proof of Lemma 10.3.* First, we take  $X_0 = \{v \in V(G) : \mathbf{x}^*(v) \geq 1/20\}$ , put  $X_0$  into the solution we are constructing, and delete  $X_0$  from the graph. Clearly,  $|X_0| \leq 20|\mathbf{x}^*|$ ; from this point, we can assume that  $\mathbf{x}^*(v) < 1/20$  for every  $v \in V(G)$ .

Second, we modify  $\mathbf{x}^*(v)$  as follows: we put  $\mathbf{x}_2^*(v) := 2\mathbf{x}^*(v)$  for every  $v \in A$  and  $\mathbf{x}_2^*(v) := 0$  for every  $v \in B$ . Since  $\mathbf{x}^*(v) < 1/20$  for every  $v \in V(G)$ , the length of the shortest hole in  $G$  is at least 21, while at most two vertices of such a hole can lie in the clique  $G[B]$ . Consequently,  $\mathbf{x}_2^*$  is a fractional solution to CHVD on  $G$  with the additional properties that  $\mathbf{x}_2^*(v) = 0$  for every  $v \in B$  and  $\mathbf{x}_2^*(v) < 1/10$  for every  $v \in A$ .

We now perform the following iterative procedure. Let  $\mathcal{C}$  be the set of connected components of  $G[A]$ . We pick a component  $C \in \mathcal{C}$  with largest value  $\mathbf{x}_2^*(C)$ . If  $\mathbf{x}_2^*(C) < 1$ , we infer that  $G[C \cup B]$  is chordal; by the choice of  $C$ , this implies that the entire graph  $G$  is chordal, and we are done. Otherwise, let  $L$  be a maximal clique of  $G[C]$  such that  $\mathbf{x}_2^*(D) \leq \mathbf{x}_2^*(C)/2$  for every connected component of  $G[C] - L$  (Theorem 10.2). We apply the algorithm of Lemma 10.4 to  $G[C \cup B]$  with partition  $C \uplus B$ , the clique  $L$ , and a fractional solution  $\mathbf{x}_2^*|_C$ , obtaining a solution  $X_C$  of size  $\mathcal{O}(\mathbf{x}_2^*(C)^2)$ . We put  $X_C$  into the solution we are constructing, and delete it from the graph. At this point, no hole in  $G$  passes through  $L$ ; we delete  $L$  from the graph as well.

Consider a vertex  $v \in C \setminus (L \cup X_C)$  after this step. Let  $D$  be a connected component of  $G[A]$  containing  $v$ . Since we deleted the entire clique  $L$ , we have

$\mathbf{x}_2^*(D) \leq \mathbf{x}_2^*(C)/2$ . Consequently, every vertex  $v$  can be contained in a component  $C$  chosen by the algorithm at most  $\lceil \log(1 + |\mathbf{x}_2^*|) \rceil$  times. Whenever an algorithm considers a component  $C$ , we charge a cost of  $\mathcal{O}(\mathbf{x}_2^*(v)|\mathbf{x}_2^*|)$  to every vertex  $v \in C$ . Since the set  $X_C$  is of size  $\mathcal{O}(\mathbf{x}_2^*(C)^2) \leq \mathcal{O}(\mathbf{x}_2^*(C)|\mathbf{x}_2^*|)$ , the total size of all sets  $X_C$  in the course of the algorithm is bounded by the total charge. On the other hand, by the previous argument, every vertex  $v$  is charged a value of  $\mathcal{O}(\mathbf{x}_2^*(v)|\mathbf{x}_2^*| \log |\mathbf{x}_2^*|)$ . Consequently, the total size of all sets  $X_C$  is  $\mathcal{O}(|\mathbf{x}_2^*|^2 \log |\mathbf{x}_2^*|)$ .  $\square$

**10.3. The approximation algorithm.** Let  $(G, k)$  be a CHVD instance. Our main goal in this section is to show that, after deleting a small number of vertices from  $G$ , we can decompose  $V(G)$  into a chordal graph and a small number of cliques. Then we can use Lemma 10.3 to add these cliques to the chordal graph one by one.

We start with a bit of preprocessing. As discussed before, it is straightforward to implement a separation oracle for the LP relaxation of CHVD. Using the ellipsoid method, we can compute a fractional solution  $\mathbf{x}^*$  to CHVD on  $G$  of cost arbitrarily close to the optimum. For our purposes, a 2-approximation  $\mathbf{x}^*$  suffices. If  $|\mathbf{x}^*| > 2k$ , we conclude that  $(G, k)$  is a no-instance. Otherwise, we proceed with  $\mathbf{x}^*$  further. We greedily take into the constructed solution and delete from  $G$  any vertex  $v$  with  $\mathbf{x}^*(v) \geq 1/4$ ; note that there are at most  $4|\mathbf{x}^*| \leq 8k$  such vertices. Thus, from this point we can assume that  $\mathbf{x}^*(v) < 1/4$  for every  $v \in V(G)$ ; in particular,  $G$  is  $C_4$ -free.

Our main partitioning tool is an  $\mathcal{O}(\sqrt{\log \text{opt}})$ -approximation algorithm for balanced vertex cut by Feige, Hajiaghayi, and Lee [28]. Given an  $n$ -vertex graph  $G$ , a *balanced vertex cut* is a set  $X \subseteq V(G)$  such that every connected component of  $G - X$  has at most  $2n/3$  vertices. We use the approximation algorithm of [28] in the following lemma.

**LEMMA 10.5.** *Given a CHVD instance  $(G, k)$ , we can in polynomial time find a set  $Z \subseteq V(G)$  and a set  $K \subseteq Z$  such that every connected component of  $G - Z$  has size at most  $3n/4$ ,  $G[K]$  is complete, and  $|Z \setminus K| = \mathcal{O}(k\sqrt{\log k})$ , or correctly conclude that  $(G, k)$  is a no-instance to CHVD.*

*Proof.* Since  $G$  is  $C_4$ -free,  $G$  has at most  $n^2$  maximal cliques [27, Proposition 2]. Furthermore, we can enumerate all maximal cliques in  $G$  with polynomial delay [57]. If there exists a maximal clique  $K$  of size at least  $n/4$ , we can simply return  $Z = K$ , as  $|V(G) \setminus K| \leq 3n/4$ . Otherwise, for every maximal clique  $K$  of  $G$ , we invoke the approximation algorithm for a balanced vertex cut of [28] on the graph  $G - K$ . If for some clique  $K$  the algorithm returns a cut  $Y$  of size  $\mathcal{O}(k\sqrt{\log k})$ , we output  $Z := K \cup Y$  and  $K$ . Clearly, this output satisfies the desired properties.

It remains to argue that if  $(G, k)$  is a yes-instance, then the algorithm finds a desired cut. Let  $X$  be a solution to CHVD on  $G$  of size at most  $k$ . By Theorem 10.2, the chordal graph  $G - X$  contains a maximal clique  $K_0$  such that every connected component of  $G - (X \cup K_0)$  has at most  $(n - |X|)/2 \leq n/2$  vertices. Consider the iteration of the algorithm for a maximal clique  $K \supseteq K_0$  in  $G$ . Note that  $X$  is a balanced cut of  $G - K$  of size at most  $k$ : every connected component of  $G - (X \cup K)$  is of size at most

$$\frac{n}{2} = \frac{2}{3} \cdot \frac{3}{4}n \leq \frac{2}{3}|V(G) \setminus K|.$$

Consequently, for this choice of  $K$ , the algorithm of [28] finds a balanced vertex cut of size  $\mathcal{O}(k\sqrt{\log k})$ . This finishes the proof of the lemma.  $\square$



By iteratively applying Lemma 10.5, we obtain the following.

LEMMA 10.6. *Given a CHVD instance  $(G, k)$ , we can in polynomial time either correctly conclude that it is a no-instance or find a partition  $V(G) = A_0 \uplus K_1 \uplus K_2 \uplus \dots \uplus K_\ell \uplus X_0$  such that  $G[A_0]$  is chordal,  $G[K_i]$  is complete for every  $1 \leq i \leq \ell$ ,  $\ell = \mathcal{O}(k \log n)$ , and  $|X| = \mathcal{O}(\ell k \sqrt{\log k}) = \mathcal{O}(k^2 \sqrt{\log k} \log n)$ .*

*Proof.* Consider the following iterative procedure. We start with  $H := G$ ,  $\ell := 0$ , and  $X_0 := \emptyset$ . As long as  $H$  contains a connected component  $C$  that is not chordal, we apply the algorithm of Lemma 10.5 to a CHVD instance  $(C, k)$ . In case the algorithm returns that  $(C, k)$  is a no-instance, we can return the same answer for  $(G, k)$ , as we will maintain the invariant that  $H$  is an induced subgraph of  $G$ . Otherwise, if a pair  $(Z, K)$  is returned, we take  $K$  as a next clique  $K_i$ , insert  $Z \setminus K$  into  $X_0$ , and delete  $Z$  from  $H$ . The algorithm terminates when  $H$  is a chordal graph; we then put  $A_0 := V(H)$ . Clearly, the algorithm outputs a partition  $V(G) = A_0 \uplus K_1 \uplus K_2 \uplus \dots \uplus K_\ell \uplus X_0$  as desired, and  $|X| = \mathcal{O}(\ell k \sqrt{\log k})$ . It remains to argue that if  $(G, k)$  is a yes-instance, then the algorithm always terminates after  $\mathcal{O}(k \log n)$  steps, that is,  $\ell = \mathcal{O}(k \log n)$ .

Consider a solution  $X$  to CHVD on  $(G, k)$ . Whenever the algorithm applies Lemma 10.5 to a component  $C$ , we have  $X \cap V(C) \neq \emptyset$ , since  $C$  is not chordal. We charge this step of the algorithm to an arbitrarily chosen vertex  $v \in X \cap V(C)$ . Note that after the algorithm finds  $Z \subseteq V(C)$ , we have either  $v \in Z$  or  $v$  lies in a connected component of  $C - Z$  of size at most  $2|V(C)|/3$ . Consequently, every vertex  $v \in X$  is charged at most  $\log_{3/2} n$  times, and in a yes-instance the algorithm always terminates after  $k \log_{3/2} n$  steps. (That is, if it runs longer than expected, we terminate the algorithm and claim that  $(G, k)$  is a no-instance.)  $\square$

We can now conclude the proof of Theorem 1.2. Let  $(G, k)$  be a CHVD instance. First, if  $\log n > k \log k$ , the exact FPT algorithm of Cao and Marx [19] with runtime  $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$  actually runs in polynomial time in  $n$ , and we can just solve the instance. Otherwise, we apply the algorithm of Lemma 10.6, either concluding that  $(G, k)$  is a no-instance, or finding the partition  $V(G) = A_0 \uplus K_1 \uplus K_2 \uplus \dots \uplus K_\ell \uplus X_0$  with  $\ell = \mathcal{O}(k \log n) = \mathcal{O}(k^2 \log k)$  and  $|X_0| = \mathcal{O}(k^2 \sqrt{\log k} \log n) = \mathcal{O}(k^3 \log^{3/2} k)$ . Then, for  $i = 1, 2, \dots, \ell$ , we construct sets  $A_i$  and  $X_i$  as follows. We apply the algorithm of Lemma 10.3 to a graph  $G[A_{i-1} \cup K_i]$  with the clique  $K_i$  and the fractional solution  $\mathbf{x}^*$  restricted to  $A_{i-1} \cup K_i$ . The algorithm returns a solution  $X_i$  to CHVD on  $G[A_{i-1} \cup K_i]$  of size  $\mathcal{O}(|\mathbf{x}^*|^2 \log |\mathbf{x}^*|) = \mathcal{O}(k^2 \log k)$ ; we put  $A_i := (A_{i-1} \cup K_i) \setminus X_i$ .

Note that every  $G[A_i]$  is chordal, and

$$V(G) \setminus A_\ell = X_0 \cup X_1 \cup \dots \cup X_\ell.$$

Thus, we can return a solution  $X := X_0 \cup X_1 \cup \dots \cup X_\ell$  which is of size

$$|X| = |X_0| + \sum_{i=1}^{\ell} |X_i| = \mathcal{O}(k^3 \log^{3/2} k) + \mathcal{O}(k^2 \log k) \cdot \mathcal{O}(k^2 \log k) = \mathcal{O}(k^4 \log^2 k).$$

This concludes the proof of Theorem 1.2.

**11. Kernelization algorithm.** We are now in position to repeat the statement of the main data reduction procedure for the unannotated problem, Theorem 1.1, and prove it. The proof shows that the annotations can be simulated by small gadgets.

THEOREM 1.1. *There is a polynomial-time algorithm that, given an instance  $(G, k)$  of CHORDAL VERTEX DELETION and a modulator  $M_0 \subseteq V(G)$  such that  $G - M_0$  is chordal, outputs an equivalent instance  $(G', k')$  with  $\mathcal{O}(k^{45} \cdot |M_0|^{29})$  vertices and  $k' \leq k$ .*

*Proof.* Given a CHVD instance  $(G, k)$  and a modulator  $M_0$ , apply Lemma 5.1. If it reports that  $(G, k)$  is a no-instance, output a constant-size no-instance of CHVD. Otherwise, we obtain an equivalent instance  $(G', k', M, E^h)$  of A-CHVD with  $k' \leq k$  and  $|M| = \mathcal{O}(k \cdot |M_0|)$ . Apply Theorem 5.2 to  $(G', k', M, E^h)$  to obtain an equivalent instance  $(G'', k', M, F^h)$  of A-CHVD with  $\mathcal{O}(k^{16}|M|^{29}) \leq \mathcal{O}(k^{45}|M_0|^{29})$  vertices. To turn this into an instance of the original CHVD problem, we simulate the annotations.

By definition of the annotated problem, we have  $xy \in E(G''[M])$  for each pair  $\{x, y\} \in F^h$  and therefore  $x, y \in M$ . For each pair  $\{x, y\} \in F^h$ , we simulate the annotation by adding two new vertices  $x', y'$  and the edges  $xx', x'y'$ , and  $y'y$  to  $G''$ , thereby creating a  $C_4$ . As all holes passing through  $x'$  or  $y'$  also pass through  $x$  and  $y$ , there is always a minimum chordal modulator in the resulting graph that does not contain any of the newly added vertices. Moreover, since we introduce a  $C_4$  containing  $x$  and  $y$  for each annotated pair  $\{x, y\} \in F^h$ , such solutions hit at least one vertex of each annotated pair. Letting  $G^*$  denote the graph resulting from adding such  $C_4$ 's for all annotated pairs, we therefore establish that if  $G^*$  has a chordal modulator of size  $k'$ , then  $G''$  has a solution of size  $k'$  that hits all annotated pairs. Conversely, any solution  $X$  to the annotated problem on  $G''$  is a chordal deletion set in  $G^*$ , since in  $G^* - X$  the newly added vertices form a pendant path and are therefore not part of any hole. Consequently, the CHVD instance  $(G^*, k')$  is equivalent to the annotated instance and therefore to the original input  $(G, k)$ . Since the definition of A-CHVD ensures that we only annotate pairs of vertices from  $M$ , we add at most  $|M|^2$  vertices. Hence the asymptotic size bound for  $G^*$  is dominated by the size of  $G''$ . The pair  $(G^*, k')$  is given as the output of the procedure.  $\square$

Combining the previous theorem with our approximation algorithm in Theorem 1.2, we obtain a polynomial kernel for CHVD. The size bound follows from plugging in  $\mathcal{O}(k^4 \log^2 k)$  for the modulator size.

**COROLLARY 11.1.** CHORDAL VERTEX DELETION *has a kernel with  $\mathcal{O}(k^{161} \log^{58} k)$  vertices.*

**12. Conclusion.** We presented a polynomial kernel for CHORDAL VERTEX DELETION based on new graph-theoretic insights and a poly(OPT)-approximation algorithm. Our work raises several questions for future research. The most obvious questions concern the kernel size and the approximation factor; both have room for significant improvements. Since there is a simple linear-parameter transformation from VERTEX COVER to CHVD, by the results of Dell and van Melkebeek [24] there is no kernel for CHVD that can be encoded in  $\mathcal{O}(k^{2-\epsilon})$  bits unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ . The bound of  $\tilde{\mathcal{O}}(k^{161})$  vertices is very far from this, and we expect the optimal kernel to have a size bound with degree at most ten. A better approximation ratio will directly translate into a better kernel size. While our current kernel is only of theoretical interest, it paves the way for future improvements. Indeed, building on an extended abstract of our work, Agrawal et al. [2] recently improved the kernel size for CHVD to  $\mathcal{O}(k^{25} \log^{14} k)$  vertices, aided by a new polynomial-time algorithm that outputs a solution of size  $\mathcal{O}(\text{OPT}^2 \log^2 n)$ . Further improvements for approximation algorithms for (weighted generalizations of) CHVD were later obtained by the same set of authors [3].

Our work also relates to purely graph-theoretical questions. Lemma 1.3 proves that the set of holes in nearly chordal graphs satisfies a form of Erdős-Pósa-type packing/covering duality. For a long time, it was open whether such packing/covering duality holds for holes in general graphs. Kim and Kwon [44] recently established that this is indeed the case: there a constant  $c$  such that for any  $k$ , an arbitrary graph  $G$

either has at least  $k + 1$  vertex-disjoint holes or has a set of  $ck^2 \log k$  vertices that intersect all holes.

We conclude with an open problem about kernelization for vertex deletion problems. The INTERVAL VERTEX DELETION problem, asking whether a graph  $G$  can be turned into an interval graph by removing at most  $k$  vertices, is fixed-parameter tractable [18, 20]. Does it have a polynomial kernel?

## REFERENCES

- [1] A. AGARWAL, M. CHARIKAR, K. MAKARYCHEV, AND Y. MAKARYCHEV,  *$O(\sqrt{\log n})$  approximation algorithms for min UnCut, min 2CNF deletion, and directed cut problems*, in Proc. 37th STOC, 2005, pp. 573–581, <https://doi.org/10.1145/1060590.1060675>.
- [2] A. AGRAWAL, D. LOKSHTANOV, P. MISRA, S. SAURABH, AND M. ZEHAZI, *Feedback vertex set inspired kernel for chordal vertex deletion*, in Proc. 28th Annual ACM-SIAM Symposium on Discrete Algorithms, P. N. Klein, ed., SIAM, ACM, 2017, pp. 1383–1398, <https://doi.org/10.1137/1.9781611974782.90>.
- [3] A. AGRAWAL, D. LOKSHTANOV, P. MISRA, S. SAURABH, AND M. ZEHAZI, *Polylogarithmic Approximation Algorithms for Weighted-F-Deletion Problems*, in Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2018, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2018, pp. 1:1–1:15.
- [4] V. BAFNA, P. BERMAN, AND T. FUJITO, *A 2-approximation algorithm for the undirected feedback vertex set problem*, SIAM J. Discrete Math., 12 (1999), pp. 289–297, <https://doi.org/10.1137/S0895480196305124>.
- [5] A. BAR-NOY, R. BAR-YEHUDA, A. FREUND, J. NAOR, AND B. SCHIEBER, *A unified approach to approximating resource allocation and scheduling*, J. ACM, 48 (2001), pp. 1069–1090, <https://doi.org/10.1145/502102.502107>.
- [6] D. BAUER, S. L. HAKIMI, AND E. F. SCHMEICHEL, *Recognizing tough graphs is NP-hard*, Discrete Appl. Math., 28 (1990), pp. 191–195, [https://doi.org/10.1016/0166-218X\(90\)90001-S](https://doi.org/10.1016/0166-218X(90)90001-S).
- [7] A. BECKER AND D. GEIGER, *Optimization of Pearl’s method of conditioning and greedy-like approximation algorithms for the vertex feedback set problem*, Artificial Intelligence, 83 (1996), pp. 167–188, [https://doi.org/10.1016/0004-3702\(95\)00004-6](https://doi.org/10.1016/0004-3702(95)00004-6).
- [8] I. BLIZNETS, F. V. FOMIN, M. PILIPCZUK, AND M. PILIPCZUK, *A subexponential parameterized algorithm for proper interval completion*, SIAM J. Discrete Math., 29 (2015), pp. 1961–1987, <https://doi.org/10.1137/140988565>.
- [9] I. BLIZNETS, F. V. FOMIN, M. PILIPCZUK, AND M. PILIPCZUK, *Subexponential parameterized algorithm for interval completion*, in Proc. 27th Annual ACM-SIAM Symposium on Discrete Algorithms, R. Krauthgamer, ed., SIAM, ACM, 2016, pp. 1116–1131, <https://doi.org/10.1137/1.9781611974331.ch78>.
- [10] H. L. BODLAENDER, *On linear time minor tests with depth-first search*, J. Algorithms, 14 (1993), pp. 1–23, <https://doi.org/10.1006/jagm.1993.1001>.
- [11] H. L. BODLAENDER, F. V. FOMIN, D. LOKSHTANOV, E. PENNINKX, S. SAURABH, AND D. M. THILIKOS, *(Meta) Kernelization*, J. ACM, 63 (2016), pp. 44:1–44:69, <https://doi.org/10.1145/2973749>.
- [12] H. L. BODLAENDER AND T. C. VAN DIJK, *A cubic kernel for feedback vertex set and loop cutset*, Theory Comput. Syst., 46 (2010), pp. 566–597, <https://doi.org/10.1007/s00224-009-9234-2>.
- [13] A. BRANDSTÄDT, V. B. LE, AND J. P. SPINRAD, *Graph Classes: A Survey*, SIAM, Philadelphia, 1999, <https://doi.org/10.1137/1.9780898719796>.
- [14] P. BUNEMAN, *A characterisation of rigid circuit graphs*, Discrete Math., 9 (1974), pp. 205–212, [https://doi.org/10.1016/0012-365X\(74\)90002-8](https://doi.org/10.1016/0012-365X(74)90002-8).
- [15] K. BURRAGE, V. ESTIVILL-CASTRO, M. R. FELLOWS, M. A. LANGSTON, S. MAC, AND F. A. ROSAMOND, *The undirected feedback vertex set problem has a poly(k) kernel*, in Proc. 2nd IWPEC, Lecture Notes in Comput. Sci. 4169, H. L. Bodlaender and M. A. Langston, eds., Springer, 2006, pp. 192–202, [https://doi.org/10.1007/11847250\\_18](https://doi.org/10.1007/11847250_18).
- [16] J. F. BUSS AND J. GOLDSMITH, *Nondeterminism within P*, SIAM J. Comput., 22 (1993), pp. 560–572, <https://doi.org/10.1137/0222038>.
- [17] L. CAI, *Parameterized complexity of vertex colouring*, Discrete Appl. Math., 127 (2003), pp. 415–429, [https://doi.org/10.1016/S0166-218X\(02\)00242-1](https://doi.org/10.1016/S0166-218X(02)00242-1).
- [18] Y. CAO, *Linear recognition of almost interval graphs*, in Proc. 27th Annual ACM-SIAM Symposium on Discrete Algorithms, R. Krauthgamer, ed., SIAM, ACM, 2016, pp. 1096–1115, <https://doi.org/10.1137/1.9781611974331.ch77>.

- [19] Y. CAO AND D. MARX, *Chordal editing is fixed-parameter tractable*, *Algorithmica*, 75 (2016), pp. 118–137, <https://doi.org/10.1007/s00453-015-0014-x>.
- [20] Y. CAO AND D. MARX, *Interval deletion is fixed-parameter tractable*, *ACM Trans. Algorithms*, 11 (2015), pp. 21:1–21:35, <https://doi.org/10.1145/2629595>.
- [21] M. CYGAN, F. V. FOMIN, L. KOWALIK, D. LOKSHTANOV, D. MARX, M. PILIPCZUK, M. PILIPCZUK, AND S. SAURABH, *Parameterized Algorithms*, Springer, 2015, <https://doi.org/10.1007/978-3-319-21275-3>.
- [22] M. CYGAN, L. KOWALIK, AND M. PILIPCZUK, *Open Problems from Workshop on Kernels*, 2013, <http://worker2013.mimuw.edu.pl/slides/worker-opl.pdf>.
- [23] E. DAHLHAUS, D. S. JOHNSON, C. H. PAPADIMITRIOU, P. D. SEYMOUR, AND M. YANNAKAKIS, *The complexity of multiterminal cuts*, *SIAM J. Comput.*, 23 (1994), pp. 864–894, <https://doi.org/10.1137/S009753979225297>.
- [24] H. DELL AND D. VAN MELKEBEEK, *Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses*, *J. ACM*, 61 (2014), pp. 23:1–23:27, <https://doi.org/10.1145/2629620>.
- [25] P. G. DRANGE, F. V. FOMIN, M. PILIPCZUK, AND Y. VILLANGER, *Exploring the subexponential complexity of completion problems*, *ACM Trans. Comput. Theory*, 7 (2015), 14, <https://doi.org/10.1145/2799640>.
- [26] P. ERDŐS AND L. PÓSA, *On independent circuits contained in a graph*, *Canad. J. Math.*, 17 (1965), pp. 347–352, <https://doi.org/10.4153/CJM-1965-035-8>.
- [27] M. FARBER, *On diameters and radii of bridged graphs*, *Discrete Math.*, 73 (1989), pp. 249–260, [https://doi.org/10.1016/0012-365X\(89\)90268-9](https://doi.org/10.1016/0012-365X(89)90268-9).
- [28] U. FEIGE, M. HAJIAGHAYI, AND J. R. LEE, *Improved approximation algorithms for minimum weight vertex separators*, *SIAM J. Comput.*, 38 (2008), pp. 629–657, <https://doi.org/10.1137/05064299X>.
- [29] J. FLUM AND M. GROHE, *Parameterized Complexity Theory*, Springer-Verlag, 2006.
- [30] F. V. FOMIN, B. M. P. JANSEN, AND M. PILIPCZUK, *Preprocessing subgraph and minor problems: When does a small vertex cover help?*, *J. Comput. System Sci.*, 80 (2014), pp. 468–495, <https://doi.org/10.1016/j.jcss.2013.09.004>.
- [31] F. V. FOMIN, D. LOKSHTANOV, N. MISRA, AND S. SAURABH, *Planar  $\mathcal{F}$ -deletion: Approximation, kernelization and optimal FPT algorithms*, in *Proc. 53rd FOCS*, 2012, pp. 470–479, <https://doi.org/10.1109/FOCS.2012.62>.
- [32] F. V. FOMIN, S. SAURABH, AND Y. VILLANGER, *A polynomial kernel for proper interval vertex deletion*, *SIAM J. Discrete Math.*, 27 (2013), pp. 1964–1976, <https://doi.org/10.1137/12089051X>.
- [33] J. GAJARSKÝ, P. HLINENÝ, J. OBDRZÁLEK, S. ORDYNIK, F. REIDL, P. ROSSMANITH, F. S. VILLAAMIL, AND S. SIKDAR, *Kernelization using structural parameters on sparse graph classes*, *J. Comput. System Sci.*, 84 (2017), pp. 219–242, <https://doi.org/10.1016/j.jcss.2016.09.002>.
- [34] N. GARG, V. V. VAZIRANI, AND M. YANNAKAKIS, *Approximate max-flow min-(multi)cut theorems and their applications*, *SIAM J. Comput.*, 25 (1996), pp. 235–251, <https://doi.org/10.1137/S0097539793243016>.
- [35] N. GARG, V. V. VAZIRANI, AND M. YANNAKAKIS, *Multiway cuts in node weighted graphs*, *J. Algorithms*, 50 (2004), pp. 49–61, [https://doi.org/10.1016/S0196-6774\(03\)00111-1](https://doi.org/10.1016/S0196-6774(03)00111-1).
- [36] F. GAVRIL, *Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph*, *SIAM J. Comput.*, 1 (1972), pp. 180–187, <https://doi.org/10.1137/0201013>.
- [37] A. C. GIANNOPOULOU, B. M. P. JANSEN, D. LOKSHTANOV, AND S. SAURABH, *Uniform kernelization complexity of hitting forbidden minors*, *ACM Trans. Algorithms*, 13 (2017), pp. 35:1–35:35, <https://doi.org/10.1145/3029051>.
- [38] J. R. GILBERT, D. J. ROSE, AND A. EDENBRANDT, *A separator theorem for chordal graphs*, *SIAM J. Algebraic Discrete Methods*, 5 (1984), pp. 306–313, <https://doi.org/10.1137/0605032>.
- [39] A. GUPTA, *Improved results for directed multicut*, in *Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, SIAM, ACM, 2003, pp. 454–455, <http://dl.acm.org/citation.cfm?id=644108.644181>.
- [40] P. HEGGERNES, P. VAN ’T HOF, B. M. P. JANSEN, S. KRATSCHE, AND Y. VILLANGER, *Parameterized complexity of vertex deletion into perfect graph classes*, *Theoret. Comput. Sci.*, 511 (2013), pp. 172–180, <https://doi.org/10.1016/j.tcs.2012.03.013>.
- [41] F. KAMMER AND T. THOLEY, *The  $k$ -disjoint paths problem on chordal graphs*, in *Proc. 35th International Workshop on Graph-Theoretic Concepts in Computer Science, Lecture Notes in Comput. Sci. 5911*, Springer, 2009, pp. 190–201, [https://doi.org/10.1007/978-3-642-11409-0\\_17](https://doi.org/10.1007/978-3-642-11409-0_17).

- [42] Y. KE, Y. CAO, X. OUYANG, W. LI, AND J. WANG, *Unit interval vertex deletion: Fewer vertices are relevant*, J. Comput. System Sci., 95 (2018), pp. 109–121, <https://doi.org/10.1016/j.jcss.2018.01.001>.
- [43] E. J. KIM AND O. KWON, *A polynomial kernel for distance-hereditary vertex deletion*, in Proc. 15th WADS, Lecture Notes in Comput. Sci. 10389, F. Ellen, A. Kolokolova, and J. Sack, eds., Springer, 2017, pp. 509–520, [https://doi.org/10.1007/978-3-319-62127-2\\_43](https://doi.org/10.1007/978-3-319-62127-2_43).
- [44] E. J. KIM AND O. KWON, *Erdős-Pósa property of chordless cycles and its applications*, in Proc. 29th Annual ACM-SIAM Symposium on Discrete Algorithms, A. Czumaj, ed., SIAM, ACM, 2018, pp. 1665–1684, <https://doi.org/10.1137/1.9781611975031.109>.
- [45] E. J. KIM, A. LANGER, C. PAUL, F. REIDL, P. ROSSMANITH, I. SAU, AND S. SIKDAR, *Linear kernels and single-exponential algorithms via protrusion decompositions*, ACM Trans. Algorithms, 12 (2016), 21, <https://doi.org/10.1145/2797140>.
- [46] S. KRATSCHE, *Recent developments in kernelization: A survey*, Bull. Eur. Assoc. Theor. Comput. Sci. EATCS, 113 (2014), pp. 58–97, <http://eatcs.org/beatcs/index.php/beatcs/article/view/285>.
- [47] S. KRATSCHE AND M. WAHLSTRÖM, *Representative sets and irrelevant vertices: New tools for kernelization*, in Proc. 53rd FOCS, 2012, pp. 450–459, <https://doi.org/10.1109/FOCS.2012.46>.
- [48] S. KRATSCHE AND M. WAHLSTRÖM, *Compression via matroids: A randomized polynomial kernel for odd cycle transversal*, ACM Trans. Algorithms, 10 (2014), pp. 20:1–20:15, <https://doi.org/10.1145/2635810>.
- [49] J. M. LEWIS AND M. YANNAKAKIS, *The node-deletion problem for hereditary properties is NP-complete*, J. Comput. System Sci., 20 (1980), pp. 219–230, [https://doi.org/10.1016/0022-0000\(80\)90060-4](https://doi.org/10.1016/0022-0000(80)90060-4).
- [50] D. LOKSHTANOV, *Kernelization: An overview*, in Proc. 18th FCT, Lecture Notes in Comput. Sci. 6914, O. Owe, M. Steffen, and J. A. Telle, eds., Springer, 2011, pp. 39–40, [https://doi.org/10.1007/978-3-642-22953-4\\_3](https://doi.org/10.1007/978-3-642-22953-4_3).
- [51] D. LOKSHTANOV, N. MISRA, AND S. SAURABH, *Kernelization—preprocessing with a guarantee*, in The Multivariate Algorithmic Revolution and Beyond, Lecture Notes in Comput. Sci. 7370, H. L. Bodlaender, R. Downey, F. V. Fomin, and D. Marx, eds., Springer, 2012, pp. 129–161, [https://doi.org/10.1007/978-3-642-30891-8\\_10](https://doi.org/10.1007/978-3-642-30891-8_10).
- [52] D. MARX, *Chordal deletion is fixed-parameter tractable*, in Graph-Theoretic Concepts in Computer Science, 32nd International Workshop, WG 2006, Bergen, Norway, 2006, Lecture Notes in Comput. Sci. 4271, F. V. Fomin, ed., Springer, 2006, pp. 37–48, [https://doi.org/10.1007/11917496\\_4](https://doi.org/10.1007/11917496_4).
- [53] D. MARX, *Chordal deletion is fixed-parameter tractable*, Algorithmica, 57 (2010), pp. 747–768, <https://doi.org/10.1007/s00453-008-9233-8>.
- [54] G. NEMHAUSER AND L. TROTTER, *Vertex packings: Structural properties and algorithms*, Math. Program., 8 (1975), pp. 232–248, <https://doi.org/10.1007/BF01580444>.
- [55] D. J. ROSE, *A graph-theoretic study of the numerical solution of sparse positive definite systems of linear equations*, in Graph Theory and Computing, Academic Press, 1972, pp. 183–217, <https://doi.org/10.1016/B978-1-4832-3187-7.50018-0>.
- [56] S. THOMASSÉ, *A  $4k^2$  kernel for feedback vertex set*, ACM Trans. Algorithms, 6 (2010), 32, <https://doi.org/10.1145/1721837.1721848>.
- [57] S. TSUKIYAMA, M. IDE, H. ARIYOSHI, AND I. SHIRAKAWA, *A new algorithm for generating all the maximal independent sets*, SIAM J. Comput., 6 (1977), pp. 505–517, <https://doi.org/10.1137/0206036>.
- [58] R. VAN BEVERN, H. MOSER, AND R. NIEDERMEIER, *Approximation and tidying: A problem kernel for  $s$ -plex cluster vertex deletion*, Algorithmica, 62 (2012), pp. 930–950, <https://doi.org/10.1007/s00453-011-9492-7>.
- [59] P. ZHANG, E. A. SCHON, S. G. FISCHER, E. CAYANIS, J. WEISS, S. KISTLER, AND P. E. BOURNE, *An algorithm based on graph theory for the assembly of contigs in physical mapping of DNA*, Comput. Appl. Biosci., 10 (1994), pp. 309–317, <https://doi.org/10.1093/bioinformatics/10.3.309>.