We study the minimum interval deletion problem, which asks for the removal of a set of at most \( k \) vertices to make a graph of \( n \) vertices into an interval graph. We present a parameterized algorithm of runtime \( 10^k \cdot n^{O(1)} \) for this problem—that is, we show that the problem is fixed-parameter tractable.

Categories and Subject Descriptors: G.2.2 [Graph Theory]: Graph Algorithms; F.2.2 [Nonnumerical Algorithms and Problems]: Computations on Discrete Structures

General Terms: Algorithms

Additional Key Words and Phrases: Asteroidal triple, congenial hole, modular decomposition

1. INTRODUCTION

A graph is an interval graph if its vertices can be assigned to intervals of the real line such that there is an edge between two vertices if and only if their corresponding intervals intersect. Interval graphs are the natural models for DNA chains in biology and many other applications, among which the most cited ones include jobs scheduling in industrial engineering [Bar-Noy et al. 2001] and seriation in archeology [Kendall 1969]. Motivated by pure contemplation of combinatorics and practical problems of biology, respectively, Hajós [1957] and Benzer [1959] independently initiated the study of interval graphs.

Interval graphs are a proper subset of chordal graphs. After more than a half century of intensive investigation, the properties and the recognition of interval and chordal graphs are well understood [Booth and Lueker 1976]. More generally, many NP-hard problems (coloring, maximum independent set, etc.) are known to be polynomial time solvable when restricted to interval or chordal graphs. Therefore, one would like to generalize these results to graphs that do not belong to these classes, but close to them in the sense that they have only a few “erroneous”/“missing” edges or vertices. As a first step in understanding such generalizations, one would like to know how far the given graph is from the class and find the erroneous/missing elements. This leads

An extended abstract of the article was presented at the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA’14). This work is supported by the European Research Council (ERC) grant 289152 and the Hungarian Scientific Research Fund (OTKA) grant NK105645.

Authors’ addresses: Y. Cao, Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong, China; email: yixin.cao@polyu.edu.hk; D. Marx, Institute for Computer Science and Control, Hungarian Academy of Sciences, Kende u. 13-17, Budapest, H-1111, Hungary; email: dmarx@cs.bme.hu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2015 ACM 1549-6325/2015/01-ART21 $15.00 DOI: http://dx.doi.org/10.1145/2629595
us naturally to the area of graph modification problems, where given a graph $G$, the task is to apply a minimum number of operations on $G$ to make it a member of some prescribed graph class $\mathcal{F}$. Depending on the operations that we allow, we can consider versions of these problems such as completion (edge addition), edge deletion, and vertex deletion. Let us point out that when $\mathcal{F}$ is hereditary, the vertex deletion version can be considered as the most robust variant, which in some sense encompasses both edge addition and edge deletion: if $G$ can be made a member of $\mathcal{F}$ by $k_1$ edge additions and $k_2$ edge deletions, then it can also be made a member of $\mathcal{F}$ by deleting at most $k_1 + k_2$ vertices (e.g., by deleting one endpoint of each added/deleted edge).

Unfortunately, most of these graph modification problems are computationally hard: for example, a classical result of Lewis and Yannakakis [1980] shows that the vertex deletion problem is NP-hard for every nontrivial and hereditary class $\mathcal{F}$, and according to Lund and Yannakakis [1993], they are also MAX SNP-hard. Therefore, early work of Kaplan et al. [1999] and Cai [1996] focused on the fixed-parameter tractability of graph modification problems. Recall that a problem, parameterized by $k$, is fixed-parameter tractable if there is an algorithm with runtime $f(k) \cdot n^{O(1)}$, where $f$ is a computable function depending only on $k$ [Downey and Fellows 2013]. In the special case when the desired graph class $\mathcal{F}$ can be characterized by a finite number of forbidden (induced) subgraphs, fixed-parameter tractability of such a problem follows from a basic bounded search tree algorithm [Cai 1996]. However, many important graph classes, such as forests, bipartite graphs, and chordal graphs, have minimal obstructions of arbitrarily large size (cycles, odd cycles, and holes, respectively). It is much more challenging to obtain fixed-parameter tractability results for such classes; for example, see results on bipartite graphs [Reed et al. 2004; Kawarabayashi and Reed 2010], planar graphs [Marx and Schlotter 2012; Kawarabayashi 2009], acyclic graphs [Cao et al. 2010; Chen et al. 2008], and minor-closed classes [Adler et al. 2008; Fomin et al. 2012].

For interval graphs, the fixed-parameter tractability of the completion problem was raised as an open question by Kaplan et al. [1999] in 1994, to which a positive answer with a $k^{2k} \cdot n^{O(1)}$-time algorithm was given by Villanger et al. [2009] in 2007. In this article, we answer the complementary question on vertex deletion.

**Theorem 1.1 (Main Result).** There is a $10^k \cdot n^{O(1)}$-time algorithm for deciding whether or not there is a set of at most $k$ vertices whose deletion makes an $n$-vertex graph $G$ an interval graph.

**Related work.** Let us put our result into context. Interval graphs form a subclass of chordal graphs, which are graphs containing no induced cycle of length greater than 3 (also called holes). In other words, the minimal obstruction for being a chordal graph might be holes of arbitrary length, hence infinitely many of them. Even so, chordal completion (to make a graph chordal by the addition of at most $k$ edges) can still be solved by a bounded search tree algorithm by observing that a large hole immediately implies a negative answer to the problem [Kaplan et al. 1999; Cai 1996]. No such simple argument works for chordal deletion (to make the graph chordal by removing at most $k$ edges/vertices), and its fixed-parameter tractability was procured by a completely different and much more complicated approach [Marx 2010].

It is known that a graph is an interval graph if and only if it is chordal and does not contain a structure referred to as asteroidal triple (AT)—that is, three vertices such that each pair of them is connected by a path avoiding neighbors of the third one [Lekkerkerker and Boland 1962]. Therefore, in the graph modification problems related to interval graphs, one has to destroy not only all holes, but all ATs as well. The algorithm of Villanger et al. [2009] for the interval completion problem first destroys all holes by the same bounded search tree technique as in chordal completion. This
step is followed by a delicate analysis of the ATs and a complicated branching step to break them in the resulting chordal graph.

A subclass of interval graphs that received attention is the class of unit interval graphs: graphs that can be represented by intervals of unit length. Interestingly, this class coincides with proper interval graphs, which are those graphs that have a representation with no interval containing another one. It is known that unit interval graphs can be characterized as not having holes and three other specific forbidden subgraphs; thus, graph modification problems related to unit interval graphs [Kaplan et al. 1999; Van’t Hof and Villanger 2013] are very different from those related to interval graphs, where the minimal obstructions include an infinite family of ATs.

Our techniques. Even though both CHORDAL DELETION and INTERVAL COMPLETION seem related to INTERVAL DELETION, our algorithm is completely different from the published algorithms for these two problems. The algorithm of Marx [2010] for CHORDAL DELETION is based on iterative compression, identifying irrelevant vertices in large cliques, and the use of Courcelle’s theorem on a bounded treewidth graph; none of these techniques appears in the present article.

Villanger et al. [2009] used a simple bounded search tree algorithm to try every minimal way of completing all of the holes; therefore, one can assume that the input graph is chordal. ATs in a chordal graph are known to have the property of being shallow, and in a minimal witness of an AT, every vertex of the triple is simplicial. This means that the algorithm of Villanger et al. [2009] can focus on completing such ATs. On the other hand, there is no similar upper bound known on the number of minimal ways of breaking all holes by removing vertices, and it is unlikely to exist. Therefore, in a sense, INTERVAL DELETION is inherently harder than INTERVAL COMPLETION: in the former problem, we have to deal with two types of forbidden structures, holes and shallow ATs, whereas in the second problem, only shallow ATs concern us. Indeed, we spend significant effort in the present article to make the graph chordal; the main part of the proof is understanding how holes interact and identifying the minimal ways of breaking them.

The main technical idea to handle holes is developing a reduction rule based on the modular decomposition of the graph and analyzing the structural properties of reduced graphs. It turns out that the holes remaining in a reduced graph interact in a very special way (each hole is fully contained in the closed neighborhood of any other hole). This property allows us to prove that the number of minimal ways of breaking the holes is polynomially bounded, and thus a simple branching step can reduce the problem to the case when the graph is chordal. As another consequence of our reduction rule, we can prove that this chordal graph already has a structure close to interval graphs (it has a clique tree that is a caterpillar). We can show that in such a chordal graph, ATs interact in a well-behaved way, and we can find a set of 10 vertices such that there always exists a minimum solution that contains at least one of these 10 vertices. Therefore, we can complete our algorithm by branching on the deletion of one of these vertices.

Motivation. The motivation for the graph modification problem studied in this article is twofold: theoretical and coming from applications. Many classical graph-theoretic problems can be formulated as graph deletion to special graph classes. For instance, VERTEX COVER, FEEDBACK VERTEX SET, CLUSTER VERTEX DELETION, and ODD CYCLE TRANSVERSAL can be viewed as vertex deletion problems where the class $\mathcal{F}$ is the class of all empty graphs, forests, cluster graphs (i.e., disjoint union of cliques), and bipartite graphs, respectively. Thus, the study of graph modification problems related to important graph classes can be seen as a natural extension of the study of classical combinatorial problems. In light of the importance of interval graphs, it is not surprising that there are natural combinatorial problems that can be formulated as or computationally reduced
to INTERVAL DELETION, and then our algorithm for INTERVAL DELETION can be applied. For instance, Narayanaswamy and Subashini [2013] recently used Theorem 1.1 as a subroutine to solve the maximum CONSECUTIVE ONES SUB-MATRIX problem and the minimum CONVEX BIPARTITE DELETION problem.

As a historical coincidence, interval graph modification problems are motivated not only from the aforementioned theoretical studies but also because they have wide applications. One central problem in molecular biology is to reconstruct the relative positions of clones along the target DNA based on their pairwise overlap information obtained via experimental methods. These data are naturally formulated as a graph, where each clone is a vertex, and two clones are adjacent if and only if they overlap. The graph should be an interval graph provided that the relations are perfect, and the problem is then equivalent to the construction of its interval model, which can be done in linear time. However, real data are always inconsistent and contaminated by a few but crucial errors, which have to be detected and fixed. In particular, on the detection of false-positive errors that correspond to false edges, Goldberg et al. [1995] proposed the INTERVAL EDGE DELETION problem (to make the graph an interval graph by the deletion of at most \( k \) edges) and showed its NP-hardness. This problem is equivalent to the maximum spanning interval subgraph and is not known to be fixed-parameter tractable or not. Moreover, false-negative errors are also possible, which significantly complicates the situation.

In this regard, we turn to the clones (vertices) involved in erroneous relations (edges) instead of the relations themselves, and we try to identify them based on a similar assumption. More specifically, we study the INTERVAL (VERTEX) DELETION problem, which is equivalent to finding the maximum induced interval subgraph. Conceptually, this formulation is capable of dealing with both false negatives and false positives. Computationally, the number of clones involved in misobserved relations is never larger, and is believed to be significantly smaller, than the number of erroneous relations. It might thus provide better assistance to biologists by revealing more meaningful information in less time, as proclaimed by Karp [1993]:

Thus, optimization methods should be viewed not as vehicles for solving a problem, but for proposing a plausible hypothesis to be confirmed or disconfirmed by further experiments. The search for the correct solution of a reconstruction problem must inevitably be an iterative process involving a close interaction between experimentation and computation.

In a seriation problem of archeology, overlap information of a collection of artifacts is given, and we are asked to put them in chronological order. Again, we cannot expect the data to be consistent, and we have to deal with errors first. In particular, the famous Berge mystery story [Golumbic 2004] is essentially a seriation problem with false overlap information given by a cheater and can be viewed as INTERVAL DELETION with \( k = 1 \).

2. OUTLINE
The purpose of this section is to describe the main steps of our algorithm at a high level. We say that a set \( Q \subset V(G) \) is an interval deletion set to a graph \( G \) if \( G - Q \) is an interval graph. An interval deletion set \( Q \) is minimum if there is no interval deletion set strictly smaller than \( |Q| \), and it is minimal if no proper subset \( Q' \subset Q \) is an interval deletion set. A set \( X \) of vertices is called a minimal forbidden set if \( X \) does not induce an interval graph but every proper subset \( X' \subset X \) does; the subgraph \( G[X] \) is called a minimal forbidden induced subgraph. Clearly, set \( Q \) is an interval deletion set if and only if it intersects every minimal forbidden set. Our goal is to find an interval deletion
set of size at most \( k \). For technical reasons, it will be convenient to define the problem as follows:

**Interval Deletion:** Given a graph \( G \) and an integer parameter \( k \), return

- if an interval deletion set of size \( \leq k \) exists, a minimum interval deletion set \( Q \subset V(G) \);
- if no interval deletion set of size \( \leq k \) exists, “NO.”

**Phase 1: Preprocessing.** The first phase of the algorithm applies two reduction rules exhaustively. They either simplify the instance or branch into a constant number of instances with strictly smaller parameter value. The first reduction rule is straightforward: we destroy every forbidden set of size at most 10.

**Reduction 1 [Small Forbidden Sets].** Given an instance \( (G, k) \) and a minimal forbidden set \( X \) of no more than 10 vertices, we branch into \( |X| \) instances, \((G - v, k - 1)\) for each \( v \in X \).

A graph on which Reduction 1 cannot be applied is called prereduced.

The second reduction rule is less obvious and more involved. Recall that a subset \( M \) of vertices forms a module if every vertex in \( M \) has the same neighbors outside \( M \) [Gallai 1967]. A module \( M \) of \( G \) is nontrivial if \( 1 < |M| < |V(G)| \). We observe (see Section 4.2) that a minimal forbidden set \( X \) of at least five vertices is either fully contained in a module \( M \) or contains at most one vertex of \( M \). Moreover, if \( X \cap M = \{x\} \), then replacing \( x \) by any other vertex \( x' \in M \setminus \{x\} \) in \( X \) results in another minimal forbidden set. This permits us to branch on modules, as described in the following reduction rule.

**Reduction 2 [Main].** Let \( I = (G, k) \) be an instance where the graph \( G \) is prereduced and a nontrivial module \( M \) that does not induce a clique.

(1) If every minimal forbidden set is contained in \( M \), then return the instance \((G[M], k)\).
(2) If no minimal forbidden set is contained in \( M \), then return the instance \((G_M, k)\), where \( G_M \) is obtained from \( G \) by inserting edges to make \( G[M] \) a clique.
(3) Otherwise, we solve three instances: \( I_1 = (G - M, k - |M|) \), \( I_2 = (G[M], k - 1) \), and \( I_3 = (G', k - 1) \), where \( G' \) is obtained from \( G \) by adding a clique \( M' \) of \((k + 1)\) vertices, connecting every pair of vertices \( u \in M' \) and \( v \in N(M) \), and deleting \( M \); letting \( Q_1 \), \( Q_2 \), and \( Q_3 \) be the solutions of these instances, respectively, we return either \( Q_1 \cup M \) or \( Q_2 \cup Q_3 \) (“NO” when \( |Q_2 \cup Q_3| > k \)), whichever is smaller.

In other words, we branch into two directions: the solution is obtained either as the union of \( M \) and the solution of \( I_1 \) or as the union of solutions of \( I_2 \) and \( I_3 \). The two branches correspond to the two cases where the solution fully contains \( M \) or only a minimum interval deletion set to \( G[M] \) (i.e., \( Q_2 \), respectively. Note that in the second branch, it can be shown that \( Q_3 \) is disjoint from \( M' \); hence, \( Q_2 \cup Q_3 \) is indeed a subset of \( V(G) \). Moreover, we have to clarify what the behavior of the reduction is if one or more of \( Q_1 \), \( Q_2 \), and \( Q_3 \) are “NO.” If \( Q_2 \) or \( Q_3 \) is “NO,” then we define \( Q_2 \cup Q_3 \) to be “NO” as well. If one of \( Q_1 \) and \( Q_2 \cup Q_3 \) is “NO,” we return the other one; if both of them are “NO,” we return “NO” as well.

A graph on which neither reduction rule applies is called reduced; in such a graph, every nontrivial module induces a clique. In Section 4, we prove the correctness of the reductions rules and that it can be checked in polynomial time if a reduction rule is applicable. Hence, after exhaustive application of the reductions, we may assume that the graph is reduced.
The reductions are followed by a comprehensive study on reduced graphs that yields two crucial combinatorial statements. The first statement is on an AT $\{x, y, z\}$ that are witnessed by a minimal forbidden induced subgraph $W$ different from a hole. We say that $x$ is the shallow terminal if $W - N(x)$ is an induced path. We prove the shallow terminal $x$ is simplicial, i.e., $N(x)$ induces a clique.

**Theorem 2.1 [Shallow Terminals].** All shallow terminals in a reduced graph are simplicial.

We say that two holes are congenial to each other if each vertex of one hole is a neighbor of the other hole. It turns out that the holes are pairwise congenial in a reduced graph.

**Theorem 2.2 [Congenial Holes].** All holes in a reduced graph are congenial to each other.

We point out that circular-arc graphs form an important example of graphs where the holes are pairwise congenial. Indeed, all holes of a reduced graph induce a circular-arc graph, but such a proof will not be given in this article, as it is unnecessary for our purpose here. One may refer to Van’t Hof and Villanger [2013] for more intuition.

**Phase 2: Breaking holes.** A consequence of Theorem 2.2 is that if a vertex $v$ is in a hole, then $N[v]$ intersects every hole and thus makes a hole cover. Intuitively, this suggests that a minimal hole cover has to be very local in a certain sense. Indeed, by relating minimal hole covers in the reduced graph to minimal separators in the subgraph $G - N[v]$, we are able to establish a quadratic bound on the number of minimal hole covers and, more importantly, a cubic time algorithm to construct them.

**Theorem 2.3 [Hole Covers].** Every reduced graph of $n$ vertices contains at most $n^2$ minimal hole covers, and they can be enumerated in $O(n^3)$ time.

Any interval deletion set must be a hole cover and thus contains a minimal hole cover. This allows us to branch into at most $n^2$ instances, in each of which the input graph is chordal. Note that this branching step is applied only once; hence, only a polynomial factor will be induced in the running time.

**Phase 3: Breaking ATs.** As all of the holes have been broken, the graph is already chordal at the onset of the third phase. It should be noted that, however, that the graph might not be reduced, as new nontrivial nonclique modules can be introduced with the deletion of a hole cover in Phase 2. In principle, we could rerun the reductions of Phase 1 to obtain a reduced instance, but there is no need to do so at this point. The properties that we need in this phase are that the graph is prereduced and chordal, and every shallow terminal is simplicial (Theorem 2.1). We give a name to such graphs and compare it with previously defined notions here.

—A graph is prereduced if Reduction 1 does not apply.
—A prereduced graph is reduced if Reduction 2 does not apply.
—A prereduced graph is nice if it is chordal and every shallow terminal in it is simplicial.

Although both reduced graphs and nice graphs are prereduced, they are incomparable to each other. As only vertex deletions are applied after Phase 1, in the remainder of this algorithm the graph is an induced subgraph of that in a previous step. In other words, once a hereditary property is obtained after Phase 1, it remains true thereafter. It is easy to verify that the three defining properties of nice graphs are all hereditary. On the one hand, after the end of Phase 1, a reduced graph is prereduced by definition,
and according to Theorem 2.1, every shallow terminal in it is simplicial. On the other hand, Phase 2 destroys all holes and the chordal property is obtained. Therefore, the graph becomes nice after Phase 2 and will remain nice until the end of our algorithm.

The removal of all simplicial vertices from a nice graph breaks all ATs (Theorem 2.1), thereby yielding an interval graph. This implies that a nice graph has a very special structure: it has a clique tree decomposition where the tree is a caterpillar—that is, a path with degree-1 vertices attached to it. In other words, all vertices other than the shallow terminals can be arranged in a linear way, which greatly simplifies the examination of interactions between ATs. As a consequence, we can select an AT that is minimal in a certain sense and single out 10 vertices such that there must exist a minimum interval deletion set destroying this AT with one of these 10 vertices. We can therefore safely branch on removing one of these 10 vertices.

**Theorem 2.4 [Nice Graphs].** There is a \(10^k \cdot n^{O(1)}\)-time algorithm for interval deletion on nice graphs.

Putting together these steps, the fixed-parameter tractability of interval deletion follows (Figure 1).

**Theorem 1.1 (Restated).** There is a \(10^k \cdot n^{O(1)}\) time algorithm for deciding whether or not there is a set of at most \(k\) vertices whose deletion makes an \(n\)-vertex graph \(G\) an interval graph.

**Proof.** The algorithm described in Figure 1 solves the problem by making recursive calls to itself, or calling the algorithm of Theorem 2.4 \(O(n^2)\) times. In the former case, at most 10 recursive calls are made, all with parameter value at most \(k - 1\). In the latter case, the running time is \(10^k \cdot n^{O(1)}\). It follows that the total running time of the algorithm is \(10^k \cdot n^{O(1)}\).

The article is organized as follows. Section 3 sets the definitions and recalls some basic facts. Section 4 presents the details of the first phase. The next four sections are devoted to the proofs of Theorems 2.1 through 2.4. Sections 5 and 6 put shallow terminals and congenial holes under thorough examination and prove Theorems 2.1 and 2.2, respectively. Section 7 fully characterizes minimal hole covers in reduced graphs and proves Theorem 2.3. Section 8 presents the algorithm that destroys ATs in
nice graphs and proves Theorem 2.4. Section 9 closes this article by presenting some possible improvement and new directions.

3. PRELIMINARIES

All graphs discussed in this article shall always be undirected and simple. A graph $G$ is given by its vertex set $V(G)$ and edge set $E(G)$. If a pair of vertices $v_1$ and $v_2$ is connected by an edge, they are adjacent to each other and denoted by $v_1 \sim v_2$; otherwise, they are nonadjacent and denoted by $v_1 \not\sim v_2$. By $v \sim X$, we mean that $v$ is adjacent to at least one vertex of the set $X$. Two vertex sets $X$ and $Y$ are completely connected if $x \sim y$ for each pair of $x \in X$ and $y \in Y$. A graph is complete if each pair of vertices is adjacent. A clique is a graph is a subgraph that is complete, and a clique is maximal if it is not contained in another clique. A vertex is simplicial if its neighbors induce a clique. A neighbor of a vertex is another vertex that is adjacent to it, and the set of neighborhood of a vertex $v$ is denoted by $N(v)$. The closed neighborhood of $v$ is defined as $N[v] = N(v) \cup \{v\}$. This is generalized to a vertex set $U$, whose closed neighborhood and neighborhood are defined to be $N[U] = \bigcup_{v \in U} N(v)$ and $N(U) = N[U] \setminus U$. The notation $N_G(v)$ (or $N(v)$) stands for the neighbors of $v$ in the set $U$—that is, $N_G(v) = N(v) \cap U$ ($N_G[v] = N[v] \cap U$)—regardless of whether $v \in U$ or not. The subgraph of a graph $G$ induced by a subset of vertices $U$ is denoted by $G[U]$, and $G \setminus U$ is used as a shorthand for the subgraph induced by $V(G) \setminus U$.

A sequence of distinct vertices $(v_0, v_1, \ldots, v_\ell)$ such that $v_i \sim v_{i+1}$ for each $0 \leq i < \ell$ is called a $v_0 \cdots v_\ell$ path, whose length is defined to be $\ell$. Vertices $v_0$ and $v_\ell$ are the ends of the path, whereas others, $\{v_1, \ldots, v_{\ell-1}\}$, are called inner vertices. If the ends are distinct and adjacent (i.e., $\ell > 1$ and $v_0 \sim v_\ell$), then $(v_0v_1 \ldots v_\ell)$ is called a cycle, whose length is defined to be $\ell + 1$. As an abuse of notation, by $u \in P$ (resp. $u \in C$), we mean that the vertex $u$ appears in the path $P$ (resp. cycle $C$)—in other words, we use $P$ or $C$ as the set of vertices in the path (resp. cycle). A chord in a path or cycle is an edge between two nonconsecutive vertices in the path or cycle. It is worth noting that the edge $v_0v_\ell$, if it exists, is a chord in the path $(v_0v_1 \ldots v_\ell)$ but not in the cycle $(v_0v_1 \ldots v_\ell v_0)$. It is easy to verify that no shortest path can contain a chord, so between each pair of vertices of a connected graph there is a chordless path. A chordless cycle of length $\ell$, where $\ell \geq 4$, is called an $(\ell, 2)$-hole. A graph is chordal if it contains no hole—in other words, any cycle of length of at least 4 contains a chord.

Chordal graphs admit several important and related characterizations. A set $S$ of vertices separates $x$ and $y$, and is called an $x$-$y$ separator if there is no $x$-$y$ path in the subgraph $G \setminus S$, and a minimal $x$-$y$ separator if no proper subset of $S$ separates $x$ and $y$. For any pair of vertices $x$ and $y$, a minimal $x$-$y$ separator is also called a minimal separator. A graph is chordal if and only if each minimal separator in it induces a clique [Dirac 1961]. A nontrivial chordal graph contains at least two simplicial vertices, and there is at least one simplicial vertex in each component after the removal of any separator.

A tree $T$ whose nodes are the maximal cliques of a graph $G$ is a (maximal) clique tree of $G$ if it satisfies the following conditions: any pair of adjacent nodes $K_i$ and $K_j$ defines a minimal separator that is $K_i \cap K_j$; and for any vertex $x \in V$, the maximal cliques containing $x$ correspond to a subtree of $T$. A graph is chordal if and only if it has such a clique tree. A clique tree of a graph $G$ will be denoted by $T(G)$, or $T$ when the graph $G$ is clear from the context. Without distinguishing the node in a clique tree and the maximal clique in the graph $G$ corresponding to it, we use $K$ to denote both. A set of vertices is a minimal separator of $G$ if and only if it is the intersection of $K_i$ and $K_j$ for some edge $K_iK_j$ in $T$ [Buneman 1974]. This separator, $K_i \cap K_j$, is a minimal $x$-$y$ separator for any pair of vertices $x \in K \setminus K_j$ and $y \in K_j \setminus K_i$. 
As interval graphs are chordal, all aforementioned properties also apply to interval graphs. Moreover, by the following characterization of Fulkerson and Gross, each interval graph has a clique tree that is a path.

**Theorem 3.1 ([Fulkerson and Gross 1965]).** A graph $G$ is an interval graph if and only if the maximal cliques of $G$ can be linearly ordered such that for each vertex $v$, the maximal cliques containing $v$ occur consecutively.

For a comprehensive treatment and for references to the extensive literature on chordal graphs and interval graphs, one may refer to the monograph of Golumbic [2004] and the survey of Brandstädt et al. [1999].

### 4. REDUCTION RULES AND BRANCHING

This section discusses the reduction rules described in Section 2 in more detail.

#### 4.1. Forbidden Induced Subgraphs

Three vertices form an AT if each pair of them is connected by a path that avoids the neighborhood of the third one. We use the term *asteroidal witness* (AW) to refer to a minimal induced subgraph that is not a hole and contains an AT but none of its proper induced subgraphs does. It should be easy to check that an AW contains precisely one AT, and its vertices are the union of these three defining paths for this triple; the three defining vertices will be called *terminals* of this AW. It can be observed from Figure 2 that the three terminals are the only simplicial vertices of this AW and are nonadjacent to each other. Lekkerkerker and Boland [1962] observed that a graph is an interval graph if and only if it is chordal and contains no AW. Not stopping here, they rolled up their sleeves and got their hands dirty by checking each possible forbidden induced subgraph. Their effort brought the following less beautiful but more useful characterization; here a minimal noninterval graph refers to a graph whose every proper induced subgraph is an interval graph but itself is not.

**Theorem 4.1 ([Lekkerkerker and Boland 1962]).** A minimal noninterval graph is either a hole or an AW depicted in Figure 2.

Some remarks are in order. First, it is easy to verify that a hole of six or more vertices witnesses an AT (specifically, any three nonadjacent vertices from it) and is minimal, but following convention, we only refer to it as a hole while reserving the term AW for
graphs listed in Figure 2. Second, the set of AWs depicted in Figure 2 is not a literal copy of the original list in [Lekkerkerker and Boland 1962], which contains neither net nor tent; they are viewed as †-AW with \( d = 2 \) and ‡-AW with \( d = 1 \), respectively. We single them out for the convenience of later presentation. To avoid ambiguities, in this article we explicitly require a †-AW (resp., ‡-AW) to contain at least seven (resp., eight) vertices. Third, each of the four subgraphs in the first row of Figure 2 consists of a constant number of vertices (six or seven), and thus can be easily located and disposed of by standard enumeration. For the purpose of the current work, we are mainly concerned with the two kinds of AWs in the second row, whose sizes are unbounded. A †- or ‡-AW \( W \) contains a unique terminal \( s \), called the shallow terminal, such that \( W - N[s] \) is an induced path. The neighbor(s) of the shallow terminal are the center(s). The other two terminals are called base terminals, and other vertices are called base vertices. The whole set of base vertices is called the base. We use \((s : c : l, B, r)\) (resp., \((s : c_1, c_2 : l, B, r)\)) to denote the †-AW (resp., ‡-AW) with shallow terminal \( s \), center \( c \) (resp., centers \( c_1 \) and \( c_2 \)), base terminals \( l \) and \( r \), and base \( B = \{b_1, \ldots, b_d\} \). For the sake of notational convenience, we will also use \( b_0 \) and \( b_{d+1} \) to refer to the base terminals \( l \) and \( r \), respectively, even though they are not part of the base \( B \). The center(s) and base vertices are called nonterminal vertices.

Clearly, Reduction 1 can be applied in polynomial time: we can find a minimal forbidden set of size at most 10 in polynomial time (e.g., by complete enumeration). There are ways to improve this, but optimizing the exponent is not the focus of this article. After the exhaustive application of Reduction 1, the graph is prereduced. By definition, any AW in a prereduced graph contains at least 11 vertices, which rules out long claws, whipping tops, nets, and tents. Furthermore, the base of a †-AW (resp., ‡-AW) in a prereduced graph contains at least 7 (resp., 6) vertices.

The purpose of the following proposition and a detailed proof is twofold. These special structures arise frequently in this work, and we do not want to repeat the same argument again and again. The proof is exemplary in the sense that by and large, most proofs of this article exploit a similar contradictory arguments: they explicitly construct a forbidden induced subgraph, either a small AW or a short hole, assuming that the property under discussion does not hold; because all graphs discussed henceforth are prereduced, such a contradiction will suffice to prove the desired property.

**Proposition 4.2.** Let \( P = (v_0 \ldots v_p) \) be a chordless path of length \( p \) in a prereduced graph, and let \( u \) be adjacent to every inner vertex of \( P \).

1. If \( p \geq 4 \) and \( u \) is also adjacent to \( v_0 \) and \( v_p \), then \( N[v_i] \subseteq N[u] \) for every \( 2 \leq \ell \leq p - 2 \).
2. If \( p \geq 3 \) and \( u \) is also adjacent to \( v_0 \) and \( v_p \), then \( N[v_i] \cap N[v_{i+1}] \subseteq N[u] \) for every \( 1 \leq \ell \leq p - 2 \).
3. If \( p \geq 4 \), then \( N[v_i] \setminus (N[v_i] \cup N[v_{i-1}]) \subseteq N[u] \) for every \( 2 \leq \ell \leq p - 2 \).

**Proof.** Suppose to the contrary of statement (1) that there is a vertex \( x \in N[v_i] \setminus N[u] \), then we show the existence of a short hole or small AW in \( G \), thus contradicting the assumption that \( G \) is prereduced. Note that \( x \not\sim v_i \) for any \( i \leq \ell - 2 \) or \( i \geq \ell + 2 \), as otherwise there is a 4-hole \( (uv_{i-1}xv_{i+1}u) \) (here \( v_i \not\sim v_t \) because \( P \) is chordless). There is a 4-hole \( (uv_{i-1}xv_{i+1}u) \) when \( x \) is also adjacent to both \( v_{i-1} \) and \( v_{i+1} \); a tent \( \{u, v_{i-1}, v_{i-2}, x, v_{i-1}, v_{i+1}, v_{i+2}\} \) when \( x \) is adjacent to \( v_{i+1} \) but not \( v_{i-1} \); a tent \( \{u, v_{i-2}, v_{i-1}, x, v_{i-2}, v_{i-1}, v_{i+1}, v_{i+2}\} \) when \( x \) is adjacent to \( v_{i-1} \) but not \( v_{i+1} \); or a whipping top \( \{x, u, v_{i-2}, v_{i-1}, v_{i-1}, v_{i+1}, v_{i+2}\} \) otherwise (\( x \) is only adjacent to \( v_t \) in the path).

Suppose, for contradiction to statement (2), that \( x \in N[v_i] \cap N[v_{i+1}] \setminus N[u] \). If \( x \) is adjacent to \( v_{i-1} \) or \( v_{i+2} \), then there is a 4-hole; otherwise, there is a tent \( \{u, v_{i-1}, v_{i-2}, v_{i-1}, v_{i+1}, v_{i+2}\} \).
Statement (3) will follow from statement (1) if \( u \) is also adjacent to \( v_0 \) and \( v_p \); hence, we assume otherwise, and without loss of generality, \( u \not\sim v_0 \). Suppose to the contrary of statement (3), that there is a vertex \( x \in N(v_1) \setminus (N(v_1) \cup N(v_{p-1}) \cup N[u]) \). If \( v_2 \) is the only inner vertex of \( P \) that is adjacent to \( x \), then there is • a 4-hole \((x_0v_1v_2x)\) when \( x \sim v_0 \); • a 4-hole \((x_4v_3v_2x)\) when \( x \sim v_4 \); • a net \( \{v_0, v_1, x, v_2, u, v_4\} \) when \( x \not\sim v_0, v_4 \) and \( u \sim v_4 \); or • a \( \dag \)-AW \((x : v_2 : v_0, v_1v_3, v_4)\) when \( x \not\sim v_0, v_4 \) and \( u \not\sim v_4 \).

A symmetric argument proves the case when \( u \not\sim v_p \) and \( v_{p-2} \) is the only inner vertex of \( P \) that is adjacent to \( x \). Other cases follow from statements (1) and (2). \( \square \)

Let \( X \) be a nonempty set of vertices. A vertex \( v \) is a common neighbor of \( X \) if it is adjacent to every vertex \( x \in X \). We denote by \( \tilde{N}(X) \) the set of all common neighbors of \( X \). It is easy to verify that in a prereduced graph, at least one of \( X \) and \( \tilde{N}(X) \) induces a clique, as otherwise two nonadjacent vertices in \( \tilde{N}(X) \), together with two nonadjacent vertices in \( X \), will induce a 4-hole. In particular, we have the following proposition.

**Proposition 4.3.** Let \( X \) be a set of vertices of a prereduced graph that induces either a hole, an AW, or a path of length at least 2. Then \( \tilde{N}(X) \) induces a clique.

### 4.2. Modular Decomposition

A subset \( M \) of vertices forms a module of \( G \) if all vertices in \( M \) have the same neighborhood outside \( M \). In other words, for any pair of vertices \( u, v \in M \) and vertex \( x \not\in M \), \( u \sim x \) if and only if \( v \sim x \). The set \( V(G) \) and all singleton vertex sets are modules, called trivial. A brief inspection shows that no graph in Figure 2 has any nontrivial modules, and this is true as well for holes of length greater than 4.

**Proposition 4.4.** Let \( M \) be a module and \( X \) be a minimal forbidden set. If \( |X| > 4 \), then either \( X \subseteq M \) or \( |M \cap X| \leq 1 \).

Indeed, the only minimal forbidden induced subgraph of no more than four vertices is a 4-hole, of which the pair of nonadjacent vertices might belong to a module. This observation allows us to prove the following statement, which is the main combinatorial reason behind the correctness of the branching in Reduction 2.

**Theorem 4.5.** Let \( G \) be a graph that contains no 4-hole and \( M \) be a module of \( G \). A minimum interval deletion set to \( G \) contains either all vertices of \( M \) or only a minimum interval deletion set to \( G[M] \).

**Proof.** Let \( Q \) be a minimum interval deletion set to \( G \) such that \( M \not\subseteq Q \); otherwise, we are already done. To show that \( Q_M = Q \cap M \) is precisely a minimum interval deletion set to \( G[M] \), it suffices to show that for any minimum interval deletion set \( Q_M \) to \( G[M] \), the set \( Q = (Q \setminus Q_M) \cup Q_M \) is an interval deletion set to \( G \). Trivially, \( Q_M \) is an interval deletion set to \( G[M] \); if it is not minimum, then \( |Q_M| > |Q_M| \) and \( |Q| > |Q| \), which contradicts the fact that \( Q \) is minimum.

Suppose that the contrary and \( X \) is a minimal forbidden set in \( G - Q \). By construction, \( Q_M \) intersects every minimal forbidden set in \( G[M] \), whereas \( Q \setminus Q_M \) intersects every minimal forbidden set in \( G - M \). Thus, \( X \) intersects both \( M \) and \( V(G) \setminus M \). On the other hand, \( |X| > 4 \) as the graph is 4-hole free. According to Proposition 4.4, \( X \cap M \) contains exactly one vertex; let it be \( x \). Let \( x' \) be a vertex in \( M \setminus Q \), which is nonempty by the assumption \( M \not\subseteq Q \), and let \( X' = X \setminus \{x\} \cup \{x'\} \); it is immaterial whether \( x' = x \) or not. The set \( X' \) is disjoint from \( Q \), and by definition of modules, \( G[X'] \) and \( G[X] \) are isomorphic. In other words, \( X \) is a minimal forbidden set in \( G - Q \), which is impossible. Therefore, \( Q \) is an interval deletion set to \( G \), and this finishes this proof. \( \square \)
To apply Reduction 2, we have to first find a nontrivial module that is not a clique. For this purpose, we do not need to compute a modular decomposition tree of the graph. The simple algorithm described in Figure 3 is sufficient.

**Lemma 4.6.** We can find in polynomial time a nontrivial module $M$ such that $G[M]$ is not a clique or report that no such a module exists.

**Proof.** The algorithm described in Figure 3 finds such a module in a greedy manner. It starts from a pair of nonadjacent vertices $u$ and $v$, and it generates the module by adding vertices. Note that each vertex in the set $X$ defined at step 2.1 is a witness for the fact that $M$ is not a module—in other words, $M$ is a module only if $X = \emptyset$. When a nonempty vertex set $M$ is returned at step 2.2, from the algorithm we can derive that $X = \emptyset$ and $M \neq V(G)$; hence, $M$ must be a nontrivial module. Now it remains to show that as long as there is a nontrivial nonclique module $U$ in the graph, the algorithm is guaranteed to return a nonempty set (not necessarily $U$ itself). As $U$ does not induce a clique, it contains a pair of nonadjacent vertices $u$ and $v$, which shall be considered in some iteration of the for-loop. In this iteration, initially $M \subseteq U$ and by induction, we are able to show that no vertex of $V(G) \setminus U$ can be included in $X$ during this iteration; therefore, $M \subseteq U$ will remain an invariant. As a consequence, a subset $M$ that satisfies $\{u, v\} \subseteq M \subseteq U$ is returned. Indeed, one can easily verify that the module found (as earlier) is the inclusive-wise minimal one containing both $u$ and $v$. We are now ready to explain the application of Reduction 2 and prove its correctness.

**Lemma 4.7.** Reduction 2 is correct, and it can be checked in polynomial time whether Reduction 2 (and which case of it) is applicable.

**Proof.** The correctness of the reduction is clear in case 1: removing the vertices of $V(G) \setminus M$ does not make the problem any easier, as these vertices do not participate in any minimal forbidden set.

In case 2, the correctness of the reduction follows from the fact that $G$ and $G_M$ have the same set of minimal forbidden sets. Note that a clique is an interval graph, and more importantly, the insertion of edges to make $M$ a clique neither breaks the modularity of $M$ nor introduces any new 4-hole; thus, Proposition 4.4 is applicable to $G_M$. As $M$ induces an interval graph in both $G$ and $G_M$, if $X$ is a minimal forbidden set of $G$ or $G_M$, then Proposition 4.4 implies that $X$ contains at most one vertex of $M$. In other words, the insertion of edges has no effect on any minimal forbidden set, which means that $Q$ is an interval deletion set to $G$ if and only if it is an interval deletion set to $G_M$.

The correctness of case 3 can be argued using Theorem 4.5, which states the two possibilities of any interval deletion set to $G$ with respect to $M$. In particular, the two branches of case 3 correspond to these two cases. The first branch is straightforward: we simply remove all vertices of $M$ from the graph and solve the instance $I_1 = (G - M, k - |M|)$. It is the second branch (where we assume $M \not\subseteq Q$) that needs more
explanation. Recall that by construction of $I_3$, the set $M'$ is a module of $G'$ and induces an interval graph. It is clear that either solution $Q_2$ or $Q_3$ being “NO” will rule out the existence of an interval deletion set of $G$ that does not fully contain $M$. Hence, we may assume that $Q_2$ and $Q_3$ are minimum interval deletion sets of $I_2$ and $I_3$, respectively, and that $Q = Q_2 \cup Q_3$. Note that both $|Q_2|$ and $|Q_3|$ are upper bounded by $k - 1$.

**Claim 1.** Set $Q$ is an interval deletion set of $G$.

**Proof.** According to Theorem 4.5, if $Q_3$ intersects $M'$, which is a module of $G'$, then it must contain all $(k + 1)$ vertices in $M'$ (i.e., $|Q_3| > k$); a contradiction. Therefore, $Q_3 \cap M' = \emptyset$, which means that $Q \subset V(G)$. Suppose that there is a minimal forbidden set $X$ of $G$ disjoint from $Q$. It cannot be fully contained in $M$, as $Q_2 \subseteq Q$ is an interval deletion set of $G[M]$. Then, by Proposition 4.4, $X$ contains exactly one vertex $x$ of $M$, and $X = X \setminus \{x\}$ is also a minimal forbidden set of $G'$ for any $x' \in M$. Since $Q_3$ is an interval deletion set of $G'$ disjoint from $M'$, it has to contain a vertex of $X \setminus \{x'\} = X \setminus \{x\}$; a contradiction.

**Claim 2.** Set $Q$ is not larger than the smallest interval deletion set $Q'$ satisfying $M \subseteq Q$.

**Proof.** Suppose that $Q$ is an interval deletion set of $G$ of size at most $k$ with $M \subseteq Q$; let $Q_2 = Q \cap M$ and $Q_3 = Q \setminus M$. We claim that $Q_2$ and $Q_3$ are interval deletion sets of $I_2$ and $I_3$, respectively. First, we argue that $Q_2$ and $Q_3$ are not empty; hence, both of them have sizes at most $k - 1$. The assumption that $G[M]$ is not an interval graph implies $Q_3 \neq \emptyset$. By assumption, $M \not\subseteq Q$, and thus there is a vertex $x \in M \setminus Q$. Now, $Q_3 = \emptyset$ would imply that $G - (M \setminus \{x\})$ is an interval graph—that is, there is no minimal forbidden set containing only one vertex of $M$—and it follows that we should have been in case 1. Since $|Q_3| \leq k - 1$, it is clear that $Q_3$ is a solution of instance $I_3 = (G[M], k - 1)$. The only way $Q_3$ is not a solution of $I_3$ is that there is a minimal forbidden set $X$ containing a vertex of the $(k + 1)$-clique introduced to replace $M$. As this $(k + 1)$-clique is a module, Proposition 4.4 implies that $X$ contains exactly one vertex $y$ of this clique. But in this case, $X = X \setminus \{y\} \cup \{x\}$ (where $x$ is a vertex of $M \setminus Q$) is a minimal forbidden set disjoint from $Q'$, a contradiction. Thus, $|Q| \leq |Q'|$ follows from the fact that both $Q_2$ and $Q_3$ are minimum.

As a consequence of Claim 2, if $|Q| > k$, then there cannot be an interval deletion set of size no more than $k$ that does not fully include $M$. This finishes the proof of the correctness of Reduction 2.

On the applicability of Reduction 2, we first use Lemma 4.6 to find a nontrivial module that does not induce a clique. If such a module $M$ is found, then Reduction 2 is applicable, and it remains to figure out which case should apply by checking the conditions in order. To check whether case 1 holds, we need to check if there is a minimal forbidden set $X$ not contained in $M$. By Proposition 4.4, such an $X$, if it exists, contains at most one vertex $x$ from $M$, and $x$ can be replaced by any other vertex of $M$. Therefore, it suffices to pick any vertex $x \in M$ and test in linear time whether $G - (M \setminus \{x\})$ is an interval graph. If it is not an interval graph, then there is a minimal forbidden set $X$ not contained in $M$ (as it contains at most one vertex of $M$). Otherwise, $G - (M \setminus \{x\})$ is an interval graph for every $x \in M$, and there is no such $X$; hence, case 1 holds. To check whether case 2 holds, observe that the condition “there is no minimal

\[\text{Indeed, min}(k + 1, |N(M)|) \text{ vertices will suffice for our bookkeeping purpose, and an alternative way to this is to add only one vertex but mark it as “forbidden.”}\]
forbidden set contained in $M''$ is equivalent to saying that $G[M]$ is an interval graph, which can be checked in linear time. In all remaining cases, we are in case 3. □

5. SHALLOW TERMINALS

This section proves Theorem 2.1 by showing that each shallow terminal is contained in a module whose neighborhood induces a clique. This module either is trivial (consisting of only this shallow terminal) or induces a clique (after the application of Reduction 2). Therefore, this shallow terminal is always simplicial.

**Lemma 5.1.** Let $W$ be an AW in a prereduced graph. Every common neighbor $x$ of the base $B$ is adjacent to the shallow terminal $s$.

**Proof.** The center(s) of $W$ are also common neighbors of $B$, and hence according to Proposition 4.3, they are adjacent to $x$. Suppose, for contradiction, that $x \notin N(B) \setminus N(s)$. If $W$ is a $\dagger$-AW, then there is (see the first row of Figure 4) • a whipping top $\{s, c, l, b_1, x, b_d, r\}$ centered at $c$ when $x \sim l, r$; • a net $\{s, c, l, b_1, r, x\}$ when $x \sim r$ but $x \not\sim l$ (similarly for $x \sim l$ but $x \not\sim r$); or • an $\dagger$-AW $\{s : c : l, b_1xb_d, r\}$ when $x \not\sim l, r$. If $W$ is a $\dashv$-AW, then there is (see the second row of Figure 4) • a tent $\{x, c_1, b_1, s, b_d, c_2\}$ when $x \sim l, r$; • an $\dagger$-AW $\{s : c_1, c_2 : l, b_1xb_d, r\}$ when $x \sim r$ but $x \not\sim l$ (similarly for $x \sim l$ but $x \not\sim r$); or • a $\dagger$-AW $\{s : c_1, c_2 : l, b_1xb_d, r\}$ when $x \not\sim l, r$.

As none of these structures can exist in a prereduced graph, this lemma is proved. □

**Lemma 5.2.** Let $W$ be an AW in a prereduced graph and $x$ be adjacent to the shallow terminal $s$.

(1) Then $x$ is also adjacent to the center(s) of $W$ (different from $x$).
(2) Classifying $x$ with respect to its adjacency to the base $B$, we have the following categories:
--- (full) $x$ is adjacent to every base vertex.
Then $x$ is also adjacent to every vertex in $N(s) \setminus \{x\}$.
—(partial) $x$ is adjacent to some but not all base vertices.

Then there is an AW whose shallow terminal is $s$, one center is $x$, and base is a proper subpath of $B$.

—(none) $x$ is adjacent to no base vertex.

Then $x$ is adjacent to neither base terminals and thus replacing the shallow terminal of $W$ by $x$ makes another AW.

**Proof.** Suppose to the contrary of statement (1), that $x \not\sim c$ if $W$ is a $\dagger$-AW or (without loss of generality) $x \not\sim c_2$ if $W$ is a $\ddagger$-AW. If $x \sim b_i$ for some $1 \leq i \leq d$, then there is a 4-hole ($xsb_i,x$) or ($xsc_2b_ix$) (Figure 5(a)). Hence, we may assume $x \not\sim B$ (Figure 5(b)–(e)).

There is • a 5-hole ($xscb_1lx$) or ($xsc_2b_1lx$) if $W$ is a $\dagger$-AW, and $x \sim l$ or $x \sim r$, respectively; • a 5-hole ($xsc_2b_1lx$) or 4-hole ($xsc_2b_2rx$) if $W$ is a $\dagger$-AW, and $x \sim l$ or $x \sim r$, respectively; • a long claw $\{x,s,c,b_1,l,b_d,r\}$ if $W$ is a $\dagger$-AW and $x \not\sim l,r$; • a net $\{x,s,l,c_1,r,c_2\}$ if $W$ is a $\dagger$-AW and $x \not\sim c_1,l,r$; or • a whipping top $\{r,c_2,s,x,c_1,l,b_1\}$ centered at $c_2$ if $W$ is a $\ddagger$-AW and $x \not\sim l,r$, but $x \sim c_1$. Neither of these cases is possible, and thus statement (1) is proved.

For statement (2), let us handle category “none” first. Note that $x$, nonadjacent to $B$, cannot be a center of $W$. If $x \sim l$, then there is a 4-hole ($xcb_1lx$) or ($xcb_2lx$) when $W$ is a $\dagger$-AW or $\ddagger$-AW, respectively. A symmetrical argument will rule out $x \sim r$. Now that $x$ is adjacent to the center(s) but neither base terminals nor base vertices of $W$, then $(x:c:l,B,r)$ (resp., $(x:c_1,c_2:l,B,r)$) makes another $\dagger$-AW (resp., $\ddagger$-AW).

Assume now that $x$ is in category “full.” Suppose for contradiction that $x \not\sim v$ for some $v \in N(s)\setminus \{x\}$. We have already proved in statement (1) that $v$ and $x$ are adjacent to the center(s) of $W$ (different from them). In particular, if one of $v$ and $x$ is a center, then they are adjacent. Therefore, we can assume that $v$ and $x$ are not centers. If $v \sim b_i$ for some $1 \leq i \leq d$, then there is a 4-hole ($xvb_i,x$). Otherwise, $v \not\sim B$, and it is in category “none.” Let $W'$ be the AW obtained by replacing $s$ in $W$ by $v$; then, $x \sim v$ follows from Lemma 5.1.

Finally, assume that $x$ is in category “partial”—that is, $x \sim B$, but $x \not\sim b_i$ for some $1 \leq i \leq d$. In this case, we construct the claimed AW as follows. As the case $x \not\sim l$ but $x \sim r$ is symmetric to $x \sim l$, but $x \not\sim r$, it is ignored in the following—that is, we assume that $x \sim r$ only if $x \sim l$. Let $p$ be the smallest index such that $x \sim b_p$ and $q$ be the
\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
 & $q = p + 1$ & $q = p + 2$ & $q > p + 2$ \\
\hline
$p = 0$ & 4-hole & tent & \dagger\text{-AW} \\
\hline
Fig. 6(a) & $(xcb_l(x))^b$ & $(l, x, s, c, b_2, b_1)$ & $(s : x, c : l, b_1 \ldots b_{q-1}, b_q)^b$ \\
\hline
$p = 1$ & whipping top & net & \dagger\text{-AW} \\
\hline
Fig. 6(b) and (c) & $(l, b_1, x, s, c, b_3, b_2)^c$ & $(l, b_1, s, x, b_3, b_2)$ & $(s : x : l, b_1 \ldots b_{q-1}, b_q)^b$ \\
\hline
$p > 1$ & long claw$^1$ & net & \dagger\text{-AW} \\
\hline
Fig. 6(d) and (e) & $[b_{p-2}, b_{p-1}, b_p, s, x, b_{p+2}, b_{p+1}]$ & $[b_{p-1}, b_p, s, x, b_{p+1}]$ & $(s : x : b_{p-1}, b_p \ldots b_{q-1}, b_q)$ \\
\hline
\end{tabular}
\caption{Structures Used in the Proof of Lemma 5.2 (Category “Partial”)}
\end{table}

The vertex $x$ is in category “none.”

The vertex $x$ would be in category “full” if $q = d + 1$.

A 4-hole $(xb_l b_{p+1} b_{p+2} x)$ would be introduced if $x \sim b_{q+2}$.

smallest index such that $p < q \leq d + 1$ and $x \not\sim b_q$ ($q$ exists by assumptions). Table I provides the structures for $\dagger\text{-AW}$ and $\dagger\text{-AW}$, respectively (Figure 6).$^2$

As the graph is prereduced and contains no small forbidden induced subgraph, it is immediate from Table I that the case $q > p + 2$ holds; otherwise, there always exists a small forbidden induced subgraph. This completes the categorization of vertices in $N(s) \setminus T$ and the proof. $\square$

$^1$ We omit the figure for $\dagger\text{-AW}$s: for a $\dagger\text{-AW}$ $(s : c_1, c_2 : l, B, r)$, we are only concerned with the relation between center $c_2$ and $B \cup \{l\}$, which is the same as the relation between $c$ and $B \cup \{l\}$ in a $\dagger\text{-AW}$.\
The proof of our main result of this section is an inductive application of Lemma 5.2. To avoid the repetition of the essentially same argument for \( + \)-AWs and \( - \)-AWs, especially for the interaction between AWs, we use a generalized notation to denote both. We will uniformly use \( c_1, c_2 \) to denote center(s) of an AW, and although the AW under discussion is a \( + \)-AW, both \( c_1 \) and \( c_2 \) refer to the only center \( c \). As long as we do not use the adjacency of \( c_1 \) and \( l, c_2 \) and \( r \), or \( c_1 \) and \( c_2 \) in any of the arguments, this unified (abused) notation will not introduce inconsistencies.

**Theorem 5.3.** Let \( W \) be a \( + \) or \( - \)-AW in a prereduced graph \( G \) with shallow terminal \( s \) and base \( B \). Let \( C = N(s) \cap N(B) \), and let \( M \) be the vertex set of the component of \( G - C \) containing \( s \). Then \( M \) is completely connected to \( C \), and \( G[C] \) is a clique.

**Proof.** Let \( W = (s : c_1, c_2 : l, B, r) \), where \( c_1 = c_2 \) when \( W \) is a \( + \)-AW. Let \( x \) and \( y \) be any pair of vertices such that \( x \in C \) and \( y \in M \). By definition, \( G[M] \) is connected, and there is a chordless path \( P = (v_0, \ldots, v_p) \) from \( v_0 = s \) to \( v_p = y \) in \( G[M] \). We claim that no vertex in \( P \) is adjacent to \( B \). It holds vacuously if \( p = 1 \) and then \( y \sim s \); hence, we assume that \( p > 1 \). Suppose the contrary, and let \( q \) be the smallest index such that \( v_q \sim B \). This means that every \( v_i \) with \( i < q \) is in category “none” of Lemma 5.2(2). Therefore, applying Lemma 5.2(1,2) on \( v_q \) and \( v_{q-1} \) implies for \( k \in B \) and \( v_q \sim B \) and thus cannot be in category “none.” If \( v_q \) is adjacent to \( B \)—that is, in the category “full” with respect to every \( W \), then Lemma 5.2(2) on \( v_q \) and \( W \) implies that \( v_q \) is adjacent to \( v_{q-1} \), contradicting the assumption that \( P \) is chordless. Otherwise (the category “partial”), according to Lemma 5.2(2), there is another AW \( W' = (v_q : c_1', c_2' : l', B', r') \), where \( B \subset B \), and \( v_q \in \{ c_1', c_2' \} \). Now an application of Lemma 5.2(1) on \( v_q \) and \( W' \) shows that \( v_q \) is adjacent to \( v_{q-1} \), again a contradiction. From these contradictions, we can conclude that \( P \) is disjoint from \( N(B) \). Applying Lemma 5.2 inductively on \( v_{q+1} \) and \( W = (v : c_1, c_2 : l, B, r) \), we get an AW with the same centers for every \( 0 \leq i \leq p \).

As \( x \) is adjacent to both \( s \) and \( B \), it cannot be in category “none” with respect to \( W \). We now separate the discussion based on whether \( x \) is in the category “full” or “partial.” Suppose first that \( x \) is in the category “full”; as \( x \in N(s) \), Lemma 5.2(1) implies that \( x \sim c_1, c_2 \). Then applying Lemma 5.2(2) inductively, where \( i = 1, \ldots, p \), on vertex \( x \) and \( W \) we get that \( x \sim v_i \) for every \( i \leq p \); in particular, \( x \sim v_p \) (\( y \)). Suppose now that \( x \) is in in category “partial.” Then by Lemma 5.2(2), there is an AW \( W_0 = (v_0 : c_1, c_2 : l, B, r) \), where \( B \subset B \), and \( x \in \{ c_1, c_2 \} \). For any \( 0 \leq i \leq p \), as \( v_i \not\sim B \) as well—that is, \( v_i \) is in category “none” with respect to \( W_0 \). Therefore, by an inductive application of Lemma 5.2(2) on the vertex \( v_i \) and \( W_0 \) we get that there is an AW \( W_p = (v_p : c_1, c_2 : l, B, r) \), from which \( x \sim y \) follows immediately.

Now we show the second assertion. For any pair of vertices \( x \) and \( y \) in \( C \), we apply Lemma 5.2 on \( x \) and \( W \); by definition, \( x \sim B \) and thus cannot be in category “none.” If \( x \) is in category “full” with respect to \( W \), then Lemma 5.2(2) and the fact \( y \in N(s) \) imply that \( x \sim y \). Otherwise, if \( x \) is in category “partial” with respect to \( W \), then Lemma 5.2(1) implies that there is an AW \( W = (s : c_1', c_2' : l, B, r) \), where \( B \subset B \) and \( x \in \{ c_1', c_2' \} \). Therefore, by Lemma 5.2(1) on the vertex \( y \in N(s) \) and \( W' \), we get that \( y \sim c_1', c_2' \) and hence \( x \sim y \).  

Now, Theorem 2.1 follows from Theorem 5.3: the set \( M \) containing \( s \) is in a module whose neighborhood is a clique, and hence every vertex in \( M \) is simplicial. We point out that the set \( C \) is the minimal \( M-B \) separator.

ACM Transactions on Algorithms, Vol. 11, No. 3, Article 21, Publication date: January 2015.
6. LONG HOLES

This section proves Theorem 2.2 by showing that the holes in a reduced graph are pairwise congenial. During the study of vertices of a hole, their indices become very subtle. To simplify the presentation, we will frequently apply a common technique—that is, to number the vertices of a hole starting from a vertex of special interest for the property at hand. Needless to say, indexing two adjacent vertices in a hole will determine the indices of all vertices in the hole, as well as the ordering used to traverse the hole. All indices of vertices in a hole \( H \) should be understood as modulo \(|H|\) (e.g., \( h_0 = h_H \)).

We start from two simple facts on the relations between vertices and holes, from which we derive the relations between two holes, and finally generalize them to multiple holes.

**Proposition 6.1.** For any vertex \( v \) and hole \( H \) of a prereduced graph, \( N_H[v] \) are consecutive in \( H \). Moreover, either \( N_H[v] = H \) or \( |N_H[v]| < |H| - 7 \).

**Proof.** Both assertions are trivially true when \( v \not\sim H \), \( N_H[v] = H \) or \( v \in H \) (then \(|N_H[v]| = 3\)); it is hence assumed that none of them holds true. We number the vertices of \( H \) in a way that \( h_0 \sim v \) but \( h_1 \not\sim v \). Suppose that the first assertion is not true; then, we can find the following three vertices of \( H \), whose existence is clear from assumptions: (a) \( p_1 \) is the smallest index such that \( p_1 > 1 \) and \( h_{p_1} \sim v \), (b) \( p_2 \) is the smallest index such that \( p_2 > p_1 \) and \( h_{p_2} \not\sim v \), and (c) \( p_3 \) is the smallest index such that \( p_2 < p_3 < |H| \) and \( h_{p_1} \sim v \), or \( p_3 = |H| \) if \( h_i \not\sim v \) for each \( i > p_2 \) (then \( h_{p_1} = h_0 \)). By the selection of \( p_1 \), \( p_2 \), and \( p_3 \), we have \( v \sim h_i \) for each \( p_1 \leq i < p_2 \) and \( i = 0, p_3 \), but \( v \not\sim h_j \) for \( 0 < j < p_1 \) or \( p_2 \leq j < p_3 \) (relations between \( v \) and \( h_i \), where \( p_0 < i < |H| \) are immaterial).

Now we examine the distances between the three indices. If \( p_1 < 4 \) or \( p_3 < p_2 + 4 \), then there is a small hole, \((v h_0 h_{p_1} v) \) or \((v h_{p_2-1} h_{p_1} \ldots h_{p_0} v) \), respectively, of length at most 6. Thus, \( 4 < p_1 < p_2 < p_2 + 4 < p_3 \). Nonetheless, there is \( \circ \) a long claw \((h_0, v, h_{p_1}, h_{p_1-2}, h_{p_1-1}, h_{p_2}, h_{p_2+1}) \) if \( p_2 = p_1 + 1 \); \( \bullet \) a net \((h_0, v, h_{p_1-1}, h_{p_1}, h_{p_2}, h_{p_2-1}) \) if \( p_2 > p_1 + 2 \); or \( \circ \) a long claw \((h_1, h_0, v, h_{p_1-1}, h_{p_1}, h_{p_2}, h_{p_2-1}) \) if \( p_2 > p_1 + 2 \). None of these forbidden induced subgraphs involves both \( h_0 \) and \( h_{p_1} \), and thus they exist regardless of \( h_{p_0} = h_0 \) or not. This contradiction ensures that \( \tilde{N}_H[v] \) are consecutive, and so are the vertices of \( H \setminus N_H[v] \). On the second assertion, note that there is a hole of length at most 10 if \( 1 < |H| \setminus |N_H[v]| \) \( \leq 7 \).

Recall that \( \tilde{N}(H) \) is the set of all common neighbors of the hole \( H \). If a vertex \( v \not\in \tilde{N}(H) \) is adjacent to more than three vertices of \( H \), then we can use \( v \) as a shortcut for the inner vertices of the path induced by \( \tilde{N}_H[v] \) to obtain another hole that is strictly shorter than \( H \).

**Corollary 6.2.** Let \( H \) be a shortest hole. If \( v \not\in \tilde{N}(H) \), then \( N_H[v] \leq 3 \).

Note that each hole \( H \) in a prereduced graph contains at least 11 vertices. If \( v \in \tilde{N}(H) \), then on any 5 consecutive vertices of the hole \( H \) and \( v \), Proposition 4.2(1) applies, which implies that \( v \) is dominating in the closed neighborhood of \( H \).

**Corollary 6.3.** Let \( H \) be a hole in a prereduced graph. If \( v \in \tilde{N}(H) \), then \( v \) is adjacent to all vertices in \( N[H] \setminus \{v\} \).

So far, we characterized neighbors of holes in a prereduced graph: any vertex \( v \) is adjacent to a (possibly empty) set of consecutive vertices of a hole \( H \); if \( v \) is adjacent to all vertices of \( H \), then it is also adjacent to every neighbor of \( H \). From these facts, we now derive the relations between holes. Following is the most crucial concept of the section.
Definition 6.4. Two holes $H_1$ and $H_2$ are called congenial (to each other) if each vertex of one hole is a neighbor of the other hole—that is, $H_1 \subseteq N[H_2]$ and $H_2 \subseteq N[H_1]$.

We remark that every hole is congenial to itself by definition. The definition is partially motivated by the following proposition.

Proposition 6.5. Let $\mathcal{H}$ be a set of holes all congenial to $H$. For each $v \in H$, every hole in $\mathcal{H}$ intersects $N[v]$.

Since a vertex in a hole cannot be a common neighbor of it, Corollary 6.3 and the definition of congenial holes immediately imply the following result.

Corollary 6.6. For any pair of congenial holes $H_1$ and $H_2$ in a prereduced graph, $\bar{N}(H_1) = \bar{N}(H_2)$. Moreover, no vertex of $H_1$ (resp., $H_2$) is a common neighbor of $H_2$ (resp., $H_1$).

We analyze next the relation between two noncongenial holes. It turns out that if not all vertices of a hole $H_1$ are adjacent to another hole $H_2$, then, as shown in the following lemma, every vertex of $H_1$ is adjacent to either all or none of the vertices of $H_2$.

Lemma 6.7. Let $H_1$ and $H_2$ be two adjacent holes in a prereduced graph such that $H_1 \not\subseteq N[H_2]$. Each neighbor of $H_2$ in $H_1$ is a common neighbor of $H_2$—that is, $N_{H_1}[H_2] \subseteq \bar{N}(H_2)$. In particular, $H_1$ and $H_2$ are disjoint.

Proof. Let $u$ be any vertex in $N_{H_1}[H_2]$, which is nonempty by assumption, and let $P$ be the maximal path in $H_1$ with the property that $u \in P \subseteq N_{H_1}[H_2]$; denote by $p$ the number of vertices of $P$. Note that some vertices of $P$ can belong to $H_2$ (in particular, $u$ can be in $H_2$). Observe that $p < |H_1|$, as by assumption, $H_2$ is not contained in $N[H_2]$. Numbering the vertices in $H_1$ such that $P = u_0 \ldots u_{p-1}$ (the ordering of $H_1$ is immaterial when $p = 1$ and then $u_1$ can be either neighbor of $u_0$ in $H_1$), the selection of $P$ means $u_i \sim H_2$ for each $0 \leq i < p$, and $u_0, u_{p-1} \not\sim H_2$ (it is immaterial whether $u_0 = u_p$ or not). In the following, we show that both ends of $P$ belong to $\bar{N}(H_2)$, which induces a clique (Proposition 4.3). Thus, either $u_0 = u_{p-1}$ (i.e., $p = 1$) or $u_0$ and $u_{p-1}$ are adjacent (i.e., $p = 2$); in either case, we have $u \in \{u_0, u_{p-1}\} \subseteq \bar{N}(H_2)$. This proves the first assertion, and the second assertion ensues, as otherwise their common vertices will be common neighbors of $H_2$, which is not possible.

Note that $u_0 \not\in H_2$, as otherwise $u_{-1}$ is also adjacent to $H_2$, contradicting the maximality of $P$. Similarly, $u_{-1}, u_{-2} \not\in H_2$. If $u_0$ has a unique neighbor $v$ in $H_2$, then the subgraph induced by $u_1, u_0$, and five consecutive $H_2$ vertices centered at $v$ is a long claw (Figure 7(a)). Now we consider the case $2 \leq |N_{H_1}[u_0]| \leq |H_2| - 7$ (Proposition 6.1), and number the vertices of $H_2$ such that $N_{H_2}[u_0] = \{v_1, v_2, \ldots, v_q\}$. Note that $|N_{H_1}[u_0]| \leq |H_2| - 7$ implies that $v_0 \neq v_{q+1}$. If $u_{-2}$ is adjacent to $v_0$, $v_1$, $v_q$, or $v_{q+1}$, then there is a hole $(u_{-2}u_{-1}u_0v_1v_0u_{-2})$, $(u_{-2}u_{-1}u_0v_1u_{-2})$, $(u_{-2}u_{-1}u_0v_qu_{-2})$, or
(u−2u−1u0vqvq+1u−2), respectively. Otherwise, \( u−2 \neq \{v0, v1, vq, vq+1\} \), then there is a net \( \{u−1, u0, v0, v1, vq+1, vq\} \) when \( |N_{H0}(u0)| = 2 \), or long claw \( \{u−2, u−1, u0, v0, v1, vq+1, vq\} \) when \( |N_{H0}(u0)| > 2 \) (see Figure 7(b)). This proves \( u0 \in \tilde{N}(H2) \), and with a symmetrical argument we can prove \( u_{p−1} \in \tilde{N}(H2) \). □

We are now ready to establish the transitivity of the congenial relation. The reflexivity and symmetry of this relation are clear from definition; therefore, congenial holes form an equivalence class.

**Lemma 6.8.** Let \( H, H1, \) and \( H2 \) be three holes in a prereduced graph \( G \). If both \( H1 \) and \( H2 \) are congenial to \( H \), then \( H1 \) and \( H2 \) are congenial.

**Proof.** According to Corollary 6.6, \( \tilde{N}(H1) = \tilde{N}(H) = \tilde{N}(H2) \). If \( H1 \) and \( H2 \) are adjacent, then they have to be congenial, as otherwise Lemma 6.7 implies that one of them contains a common neighbor of the other, and hence a common neighbor of all three holes, which is impossible. Assume therefore that there is no edge between \( H1 \) and \( H2 \); in particular, they are disjoint. Let \( h \) be any vertex in \( H \), and we number the vertices of \( H1 \) and \( H2 \) such that \( N_{H1} \{h\} = \{u1, \ldots, u:\} \) and \( N_{H2} \{h\} = \{v1, \ldots, vq\} \). Proposition 6.1 implies that \( u0 \neq u_{p−1} \) and \( v0 \neq v_{p+1} \). Note that \( h \) is adjacent to some but not all vertices of both \( H1 \) and \( H2 \). There is • a long claw \( \{v1, h, u−1, u0, u1, u2, u3\} \) when \( p = 1 \); • a net \( \{v1, h, q, u0, u1, u2\} \) when \( p = 2 \); or • a long claw \( \{v0, v1, u0, u1, h, u_{p−1}, u_{p+1}\} \) when \( p \geq 3 \). □

To prove Theorem 2.2, we show that if there are two holes that are not congenial, then one of them is contained in a prereduced graph, which is impossible in a reduced graph, where every nontrivial module induces a clique. We construct this nontrivial module with the help of the following lemma, which shows that the common neighbors form a separator.

**Lemma 6.9.** Let \( H \) be a hole that is the shortest among all holes congenial to it in a prereduced graph \( G \). Then the set \( \tilde{N}(H) \) of common neighbors of \( H \) separates \( N[H] \setminus \tilde{N}(H) \) from \( V(G) \setminus N[H] \).

**Proof.** Suppose to the contrary that \( N[H] \setminus \tilde{N}(H) \) and \( V(G) \setminus N[H] \) are still connected in \( G \setminus \tilde{N}(H) \), then there is a pair of adjacent vertices \( u \in N[H] \setminus \tilde{N}(H) \) and \( v \in V(G) \setminus N[H] \). Note that \( u \notin H \), and we have two adjacent vertices, only one of which is adjacent to part of the hole \( H \). Depending on the number of neighbors of \( u \) in \( H \), we have either a long claw (when \( |N_{H} \{u\}| = 1 \)), a net (when \( |N_{H} \{u\}| = 2 \)), or a ♯AW of size 7 (when \( |N_{H} \{u\}| = 3 \)), none of which can exist in a prereduced graph. On the other hand, if \( |N_{H} \{u\}| > 3 \), then we can use \( u \) to find another hole \( H' \) that is strictly shorter than \( H \); it is surely congenial to \( H \), which contradicts the assumption. □

We are now ready to prove Theorem 2.2.

**Theorem 2.2 (Restated).** All holes in a reduced graph are congenial to each other.

**Proof.** Suppose, for contradiction, that not all holes are congenial to each other. By Lemma 6.8, being congenial is an equivalence relation. Hence, there are two equivalence classes of holes, from each of which we pick a shortest one; let them be \( H1 \) and \( H2 \). Assume without loss of generality that \( H2 \) has a vertex \( v \) not in \( N[H1] \). Lemma 6.9 implies that \( \tilde{N}(H1) \) separates \( N[H1] \setminus \tilde{N}(H1) \) and \( V(G) \setminus N[H1] \). Either \( \tilde{N}(H1) = \emptyset \) and then \( G \) is disconnected where \( N[H1] \) induces a connected component \( v \notin N[H1] \), or \( \tilde{N}(H1) \) is the neighbor of \( N[H1] \) and they are completely connected (Corollary 6.3). In either case, the set \( N[H1] \setminus \tilde{N}(H1) \) is a nontrivial module that does not induce a clique. Thus, Reduction 2 is applicable, and the graph is not reduced. □
7. HOLES COVERS

A set of vertices is called a hole cover of a graph \( G \) if it intersects every hole in \( G \), and the removal of any hole cover makes the graph chordal. A hole cover is minimal if any proper subset of it is not a hole cover. Any interval deletion set makes a hole cover of the input graph and thus contains a minimal hole cover. The goal of this section is to prove Theorem 2.3—that is, to provide a polynomial bound on the number of minimal hole covers in a reduced graph and give a polynomial time algorithm to find all of them.

To simplify the task, observe that no minimal hole cover contains a vertex that is not in any hole.

**Proposition 7.1.** Let \( \mathcal{H} \) be the set of all holes in a reduced graph \( G \), and let \( G_0 \) be the subgraph induced by \( \bigcup_{H \in \mathcal{H}} H \). A set \( HC \) of vertices is a minimal hole cover of \( G \) if and only if it is a minimal hole cover of \( G_0 \).

In this section, we will focus on the subgraph \( G_0 \) induced by the union of all holes in the reduced graph \( G \). The subgraph \( G_0 \) has the same set of holes as \( G \), and they remain pairwise congenial. Moreover, each vertex of \( G_0 \) is in the closed neighborhood of each hole \( H \) of \( G_0 \), which means that \( G_0 \) is connected. As we have said earlier, circular-arc graphs form an important example of graphs of which all holes are pairwise congenial. Thinking of \( G_0 \) as a circular-arc graph gives the intuition behind most statements to follow. But since this fact is not directly used in this article, we are not giving a proof for it.

**Proposition 7.2.** The subgraph \( G_0 - HC \) is an interval graph for each hole cover \( HC \) of \( G_0 \).

**Proof.** Each vertex of \( G_0 \) belongs to some hole and thus cannot be simplicial. Therefore, by Theorem 2.1, \( G_0 \) contains no AW. By definition, \( G_0 - HC \) contains no hole; thus, \( G_0 - HC \) is an interval graph. \( \square \)

In what follows, we prove a series of claims on what the neighborhood of a vertex \( v \) of a hole \( H_1 \) looks like in another hole \( H_2 \). The first statement is a paraphrase of Corollary 6.6.

**Corollary 7.3.** No vertex \( v \) of \( G_0 \) can be a common neighbor of any hole in \( G_0 \).

Therefore, by definition of congenial holes and Proposition 6.1, we can assume that for every \( v \in \mathcal{V}(G_0) \) and hole \( H \), we have that \( N_H[v] \) is a proper nonempty subset of \( H \) and its vertices induce a path in \( H \). Fixing any ordering of the vertices in \( H \), we can denote two ends of the path as \( \begin{vertex} H[v] \) and \( \end{vertex} H[v] \), respectively; when \( N_H[v] \) contains both \( h_0 \) and \( h_{|H|-1} \), we number vertices of \( N_H[v] \) as \( \{h_{-p}, \ldots, h_0, \ldots, h_q\} \) where both \( p \) and \( q \) are nonnegative and then \( \begin{vertex} H[v] = -p \), \( \end{vertex} H[v] = q \).

**Proposition 7.4.** Let \( H \) be a hole of \( G_0 \). For any pair of adjacent vertices \( u, v \) of \( G_0 \), their closed neighborhoods in \( H \) satisfy the following properties.

1. \( N_H[u] \cap N_H[v] \neq \emptyset \) and \( N_H[u] \cup N_H[v] \neq H \).
2. If \( v \) is adjacent to neither \( \begin{vertex} H[u] \) nor \( \end{vertex} H[u] \), then \( N[v] \subset N[u] \).

**Proof.** We number vertices of \( H \) such that \( N_H[u] = \{h_0, \ldots, h_\ell\} \); the order can be either way if \( |N_H[u]| = 1 \) (i.e., \( \ell = 0 \)).

Statement (1) holds trivially if either or both of \( u \) and \( v \) belong to \( H \) (Proposition 6.1). Hence, we assume \( u, v \not\in H \). Suppose first, for contradiction, that \( N_H[u] \cap N_H[v] = \emptyset \); we may assume \( \{h_\ell, \ldots, h_{\ell_2}\} \neq N_H[v] \) where \( \ell < \ell_1 \leq \ell_2 < |H| \). If \( \ell_2 \geq |H| - 3 \), then \( (uvh_{\ell_1} \ldots h_{\ell_2}u) \) is a hole of length at most 6. Otherwise, \( (uvh_{\ell_1} \ldots h_{\ell_2}u) \) is a hole not
congenial to \( H \): in particular, the vertex \( h_{H-2} \) in \( H \) is nonadjacent to it. In either case, we end with a contradiction; hence, \( N_H[u] \) and \( N_H[v] \) must intersect. Suppose now, for contradiction, that \( N_H[u] \cap N_H[v] \neq \emptyset \). Then \( v \) is adjacent to every vertex in \( (h_{i-1}h_i \cdots h_{H-1}) \). Proposition 6.1 and Corollary 7.3 imply \( 6 < \ell < |H| - 6 \). If \( v \neq h_i \), then \( (uh_{i+1}v)H \) is a 4-hole. A symmetric argument applies when \( v \neq h_0 \). Now suppose that \( v \) is adjacent to both \( h_0 \) and \( h_\ell \), then \( (uh_{i+1} \cdots h_{H-1}u) \) is a hole and \( v \) is a common neighbor of it (contradicting Corollary 7.3). None of the cases is possible, and hence \( N_H[u] \cap N_H[v] \neq H \).

The condition of statement (2) means that \( v \neq h_0 \) and \( v \neq h_\ell \). According to statement (1), and since \( N_H[v] \) is consecutive in \( H \), we must have \( N_H[v] \subseteq \{h_1, \ldots, h_{\ell-1}\} \). Note that \( N_H[v] \) is nonempty, and thus \( \ell \geq 2 \). If \( u \in H \), then \( N_H[v] = \{u\} = \{h_1\} \), and statement (2) follows from statement (1); here we use the fact that every \( x \in N[v] \) is in the neighborhood of \( H \) and that \( u \) is the only neighbor of \( v \) in \( H \). Assume now that \( u \notin H \); the following argument holds regardless of whether \( v \in H \) or not. Let \( x \) be any vertex in \( N[v] \) different from \( u \), and we argue that \( x \sim u \). By statement (1), \( N_H[x] \) must intersect \( \{h_1, \ldots, h_{\ell-1}\} \). If \( x \) is nonadjacent to \( \{h_0, h_\ell\} \), then \( N_H[x] \) is also a subset of \( \{h_1, \ldots, h_{\ell-1}\} \), and \( x \sim u \) follows from Proposition 4.2(3) (taking \( \{h_1, h_{\ell-1}\} \) as the path). Otherwise, \( x \) is adjacent to at least one of \( \{h_0, h_\ell\} \), then \( x \sim u \) will imply a 4-hole \( (h_0uvxh_0) \) or \( (h_{\ell+1}uvxh_0) \), which is impossible. The proof is now completed.

Noting that the closed neighborhoods of two consecutive vertices in a hole are incomparable, Proposition 7.4(2) has the following corollary.

**Corollary 7.5.** Let \( H \) and \( H_1 \) be two holes in \( G_0 \). For each pair of consecutive vertex \( u_i, u_{i+1} \in H_1 \), at least one end of \( N_H[u_i] \) is in \( N_H[u_{i+1}] \).

The following lemmas characterize minimal hole covers of \( G_0 \).

**Lemma 7.6.** Any minimal hole cover of \( G_0 \) induces a clique.

**Proof.** Suppose to the contrary that there is a minimal hole cover \( HC \) that contains two nonadjacent vertices \( u \) and \( v \). By the minimality of \( HC \), there are two (unnecessarily disjoint) holes \( H_1 \) and \( H_2 \) such that \( HC \cap H_1 = \{u\} \) and \( HC \cap H_2 = \{v\} \). In particular, \( u \notin H_2 \) and \( v \notin H_1 \). We number the vertices of \( H_1 \) such that \( N_{H_1}[v] = \{u_1, u_2, \ldots, u_p\} \). The union of \( N_{H_1}[u_i] \) and \( N_{H_1}[u_p] \) is a consecutive set of vertices in \( H_2 \); they both contain \( v \) and, by Proposition 6.1, are consecutive in \( H_2 \). We number the vertices of \( H_2 \) such that \( u_1 \sim v_1 \) and \( N_{H_2}[u_1] \cup N_{H_2}[u_p] = \{v_1, \ldots, v_q\} \).

**Claim 3.** At least one vertex of \( H_2 \) is adjacent to neither \( u_1 \) nor \( u_p \).

**Proof.** The claim follows from Proposition 7.4(1) when \( p = 2 \); hence, we may assume \( p > 2 \), which means \( u_1 \neq u_p \) (note that \( u_0 \neq u_{p-1} \)). Suppose that \( N_{H_2}[u_1] \cap N_{H_2}[u_p] = \emptyset \), then by Proposition 6.1, we have \( 7 < |N_{H_2}[u_1]| < |H_2| - 7 \), which means at least one end of the path induced by \( N_{H_2}[u_1] \) is not adjacent to \( v \). Without loss of generality, let it be \( v_i \) where \( i = \text{end}_{H_2}[u_1] \); noting that by assumption \( v_{i+1} \sim u_p \), there is either a 4-hole \((uv_1v_{i+1}u_p) \) (if \( v_i \sim u_p \)) or a 5-hole \((uv_1v_{i+1}u_p) \) (if \( v_i \neq u_p \)).

In what follows, we show the existence of a hole in \( G - HC \), which contradicts the assumption that \( HC \) is a hole cover and thus proves this lemma. Let \( P_1 = (u_1u_2 \ldots u_p) \) and \( P_2 = (u_qu_{q+1} \ldots v_0v_1) \). By definition, \( u \notin P_1 \); to show \( v \notin P_2 \), it suffices to rule out the possibility that \( v \in \{v_1, v_2\} \), by the numbering of \( H_2 \); \( v \) is in \( \{v_1, v_2, \ldots, v_q\} \). According to Corollary 7.5, the two neighbors of \( v \) in \( H_2 \) are adjacent to either \( u_i \) or \( u_p \); however, Claim 3 implies that \( v_0 \) and \( v_{q+1} \) are inner vertices of \( P_2 \) and hence are not adjacent to \( u_i \) or \( u_p \). We now argue that each inner vertex \( v_j \) of \( P_2 \) is not adjacent to \( P_1 \) (see the thick edges in Figure 8). Suppose to the contrary that \( v_j \) is adjacent to \( P_1 \). Noting
that \( v_i \not\sim u_1, v_i \not\sim u_p, \) and \( u_1 \neq u_{p+1} \), Proposition 4.2(3) applies, and we can conclude \( v_i \sim v \), which is impossible. (It is immaterial whether \( v_i \in H_1 \) or not.) Now we construct the hole in \( G - HC \) as follows. Claim 3 implies that the length of \( P_2 \) is at least 2. If \( u_1 \sim v_q \), then \( (u_1P_2u_1) \) is such a hole. Otherwise, by assumption we have \( u_p \sim v_q \). Let \( \ell_1 = \max\{i : u_i \sim v_1 \text{ and } 0 \leq i \leq p\} \) and \( \ell_2 = \min\{i : u_i \sim v_q \text{ and } \ell_1 \leq i \leq p\} \). Then \( (u_{\ell_1}u_{\ell_2}P_2u_{\ell_1}) \) will be such a hole (see the solid hole in Figure 8).

**Lemma 7.7.** For any minimal hole cover \( HC \) of \( G_0 \) and any shortest hole \( H \), there is a vertex \( v \in H \) such that \( N_{G_0}[v] \not\sim HC \).

**Proof.** We show this by construction. By Corollaries 6.2 and 7.3, each vertex in \( G_0 \) has at most three neighbors in \( H \). By Lemma 7.6, \( HC \) is a clique and hence \( |H \cap HC| \leq 2 \). We number the vertices of \( H \) in a way that \( h_0 \in HC \) and \( h_1 \notin HC \), and claim that \( v = h_5 \) is the asserted vertex. Suppose to the contrary that \( N_{G_0}[h_5] \) and \( HC \) are adjacent, then there is a \( h_0-h_5 \) path \( P \) of length at most 3 and all its inner vertices belong to \( G_0 \). The case \( P = (h_0v_5h_5) \) is impossible, as by Proposition 6.1 and Corollary 6.2, \( v_5 \) is adjacent to at most three consecutive vertices in \( H \). Now we may assume that \( P = (h_0v_1v_2h_5) \) and examine the neighbors of \( v_1 \) and \( v_2 \) in \( H \). By Corollary 6.2, we have \( \text{end}_H(v_1) \leq 2 \) and \( \text{begin}_H(v_2) \geq 3 \). This means that there is a hole \( (v_1h_ih_{i+1} \ldots h_jv_2v_1) \), where \( i = \text{end}_H(v_1) \) and \( j = \text{begin}_H(v_2) \), of length at least 4 and at most 8.

We now relate minimal hole covers of \( G_0 \) to minimal separators in some interval subgraphs. In one direction of the proof, we need the following claim. Observe that in an interval representation of a connected interval graph, the union of all intervals also forms an interval. Similarly, if there is a point \( p \) in the real line such that there are intervals not containing \( p \) both to the left and to the right of \( p \), then the set of intervals containing \( p \) is a clique separator.

**Proposition 7.8.** Let \( v \) be a vertex in an interval graph \( G \). If \( v \) is not adjacent to any simplicial vertex, then \( N[v] \) is a separator of \( G \).
Proof. We consider an interval representation of $G$. Without loss of generality, we assume that no two intervals have the same ends. Denote by $x$ the interval with the smallest right end and by $y$ the interval with the largest left end. It is easy to see that $x$ and $y$ are simplicial. If $x \sim y$, then the graph is a complete graph (every interval contains the interval between the left end of $y$ and the right end of $x$); thus, every vertex is adjacent to a simplicial vertex, and the assertion is vacuously true. Therefore, we can assume that $x \not\sim y$, and let $p$ be an arbitrary point in interval $v$. By assumption, $v$ is not adjacent to $x$ or $y$, which means that $x$ is to the left of $p$ and $y$ is to the right of $p$. As every interval that contains $p$ is in $N[v]$, in the subgraph $G - N[v]$ that contains $x$ and $y$, no interval contains $p$; hence, $x$ and $y$ are disconnected. In other words, $N[v]$ is an $x$-$y$ separator.

According to Lemma 7.7, every minimal hole cover satisfies the condition in the following lemma; therefore, the lemma applies to all of them. Note that $G_0 - N_{G_0}[v]$ is the same as $G_0 - N[v]$.

Lemma 7.9. Let $v$ be a vertex in a shortest hole $H$ of $G_0$, and let $X$ induce a clique nonadjacent to $N_{G_0}[v]$. Set $X$ forms a minimal hole cover of $G_0$ if and only if $X$ is a minimal separator of $G_0 - N[v]$.

Proof. It suffices to show that $X$ is a hole cover of $G_0$ if and only if it is a separator of $G_0 - N[v]$.

$\Rightarrow$ Clearly, each component of $G_0 - N[v]$ contains a neighbor of $N[v]$. As $X$ is not adjacent to $N[v]$, the set $X$ cannot fully contain a component of $G_0 - N[v]$, which implies that the number of components of $G_0 - N[v] - X$ is no less than that of $G_0 - N[v]$. Therefore, if $G_0 - N[v]$ is not connected, then neither is $G_0 - N[v] - X$, and $X$ makes a trivial separator for $G_0 - N[v]$. In the following argument of this direction, we may assume that $G_0 - N[v]$ is connected, and it suffices to show that $G_0 - N[v] - X$ is not connected. By Proposition 7.2, $G_0 - X$ is an interval subgraph; as $G_0$ itself contains no simplicial vertex, any vertex $x$ that is simplicial in $G_0 - X$ must be a neighbor of $X$: otherwise, $N_{G_0 - X}(x) = N_{G_0}(x)$ and cannot be a clique. As $N[v]$ is not adjacent to $X$ by assumption, $v$ is not adjacent to any simplicial vertex of the interval graph $G_0 - X$. Therefore, according to Proposition 7.8, the removal of $N[v]$ disconnects $G_0 - X$. This finishes the proof of the “only if” direction.

$\Leftarrow$ Let us start from a close scrutiny of $G_0 - N[v]$. According to Proposition 6.1, the removal of $N[v]$ transforms each hole into a path of length at least 7; in particular, let $P$ be the path induced by $H \setminus N_H[v]$. In the argument that follows, we show that ends of such path are connected to the ends of $P$, respectively. The further removal of $X$ separates each path into at most two subpaths; hence, if there is a hole disjoint from $X$, then the path left by it is able to connect every subpath and thereby every vertex, which is impossible.

We number the vertices of $H$ such that $v = v_0$. Then $N_H[v] = \{v_{-1}, v_0, v_1\}$ and the ends of $P$ are $v_{-2}$ and $v_2$. Let $H'$ be another hole of $G_0$, and let $P'$ be the path induced by $H' \setminus N_{H'}[v]$. We may number the vertices of $H'$ such that $N_{H'}[v] = \{h_1, \ldots, h_p\}$. As a result, the ends of $P'$ are $h_0$ and $h_{p+1}$.

Claim 4. The ends $h_0$ and $h_{p+1}$ of $P'$ are adjacent to $\{v_1, v_2\}$ and $\{v_{-1}, v_{-2}\}$, respectively.

Proof. By Corollary 6.2, $N_H[h_1] \subseteq \{v_{-2}, v_{-1}, v_0, v_1, v_2\}$, and according to Proposition 7.4(1), $h_0$ is adjacent to either $\{v_1, v_2\}$ or $\{v_{-1}, v_{-2}\}$; a symmetric argument works for $h_{p+1}$. Since the length of $H$ is at least 11, the sets $\{v_1, v_2\}$ and $\{v_{-1}, v_{-2}\}$ are disjoint. It remains to show that the ends of $P'$ cannot be adjacent to both $\{v_1, v_2\}$ or both $\{v_{-1}, v_{-2}\}$. Suppose, for contradiction, that both $h_0$ and $h_{p+1}$ are adjacent to $\{v_1, v_2\}$. We consider three cases.

ACM Transactions on Algorithms, Vol. 11, No. 3, Article 21, Publication date: January 2015.
Suppose first that both \( h_0 \) and \( h_{p+1} \) are adjacent to \( v_1 \). According to Corollary 7.5 (applied on the adjacent vertices \( v_0, v_1 \) and the hole \( H' \)), at least one end of \( N_H[v_1] \) is in \( N_H[v_0] \)—that is, \( (h_1, \ldots, h_p) \). As a result, \( N_H[v_1] \) must contain all of \( \{h_{p+1}, h_{p+2}, \ldots, h_0\} \), then \( N_H[v_0] \cup N_H[v_1] = H' \). This contradicts Proposition 7.4(1) and is impossible.

Suppose now that \( h_0 \) and \( h_{p+1} \) are both adjacent to \( v_2 \). Note that \( v_2 \not\in H' \); otherwise, since \( v_0 \not\sim v_2 \), we must have \( v_2 = h_{-1} \), then \( (v_0h_{-1}h_0v_2h_{p+1}v_0) \) is a 6-hole, which is impossible. We can apply Proposition 4.2(1) on \( v_2, v_0 \), and path \( (h_{-1}h_0h_1 \ldots h_ph_{p+1}h_{p+2}) \) to conclude \( v_0 \sim v_2 \), which is impossible.

In the remaining cases, \( h_0 \) and \( h_{p+1} \) are adjacent to \( v_1 \) and \( v_2 \), respectively. Without loss of generality, we consider \( h_0 \sim v_1 \) and \( h_{p+1} \sim v_2 \). Clearly, \( h_p \not\sim v_2 \), as they have different adjacencies to \( v_0 \); likewise, \( h_{p+1} \neq v_1 \) and \( h_{p+1} \neq v_2 \). We exclude \( h_p = v_1 \) and \( p = 1 \) by \( h_0 \sim v_1 \) and \( v_0 \sim h_0 \) by Corollary 7.5, which contradicts the numbering of \( H' \). Then \( h_p \sim v_2 \), as otherwise there is a hole \( (v_2v_1v_0h_ph_{p+1}v_2) \) or \( (v_2v_1h_ph_{p+1}v_2) \); likewise, \( h_{p+1} \sim v_1 \). It follows that by Corollary 6.2, \( h_p \not\sim v_1 \) and \( p > 1 \). On the other hand, \( h_0 \not\sim v_1 \), as otherwise there is a hole \( (h_0v_1v_0v_1h_0) \). We can apply Proposition 4.2(3) on \( v_1, v_{-1} \), and path \( (h_{-1}h_0h_1 \ldots h_ph_{p+1}) \) to conclude \( v_{-1} \sim v_1 \), which is impossible.

A symmetric argument applies to \( \{v_{-1}, v_2\} \).

We may assume without loss of generality that \( h_{p+1} \sim \{v_1, v_2\} \), then \( h_0 \sim \{v_{-1}, v_{-2}\} \). Let \( \ell_1 \) be the smallest index such that \( \ell_1 > p \) and \( h_{\ell_1} \in N[v_2] \); for its existence, observe that \( \ell_1 = p + 1 \) if \( h_{p+1} \in N[v_2] \); otherwise, by Corollary 7.5, \( h_\text{end}(v_1) \) must be in \( N[v_2] \).

For \( p + 1 \leq i \leq \ell_1 \), it holds that \( h_i \sim v_1 \), and since \( v_1 \in N[v] \), by assumption \( (X \not\sim X) \), we have \( h_1 \not\in X \). Symmetrically, we can find the largest index \( \ell_2 \) such that \( \ell_2 \leq X \) and \( h_{\ell_2} \in N[v_{-2}] \). For \( \ell_2 \leq i \leq |H| \), it holds that \( h_i \sim v_{-1} \) and \( h_i \not\in X \). On the other hand, as \( X \) is a clique, it contains at most two vertices of \( P' \), which are in \( \{h_{11}+1, \ldots, h_{h_1-1}\} \).

Therefore, the removal of \( X \) either leaves \( P' \) intact (when \( X \) is disjoint from \( P' \) or separates \( P' \) into two subpaths. In the latter case, the two subpaths, containing \( h_{\ell_1} \) and \( h_{\ell_2} \), respectively, are connected to \( v_{-1} \) and \( v_{-2} \), respectively.

To prove the “if” direction, we need to show that \( X \) intersects every hole. Suppose, for contradiction, that \( X \) is disjoint from some hole \( H_1 \). Then the path \( P_1 \) induced by \( H_1 \) remains a path of \( G_0 - N[v] - X \). Since \( P_1 \) is adjacent to both \( v_2 \) and \( v_{-2} \), we conclude that \( v_2 \) and \( v_{-2} \) are connected in \( G_0 - N[v] - X \). We have seen that for each hole \( H' \), the vertices left from \( H' \) after the removal of \( N[v] \) and \( X \) are connected to at least one of \( v_2 \) and \( v_{-2} \). Therefore, the subgraph \( G_0 - N[v] - X \) is connected, contradicting the assumption that \( X \) is a separator of \( G_0 - N[v] \). This finishes the proof of the “if” direction.

We are now ready to prove Theorem 2.3. We remark that the quadratic bound can be improved to linear with more careful analysis.

**Theorem 2.3 (Restated).** Every reduced graph of \( n \) vertices contains at most \( n^2 \) minimal hole covers, and they can be enumerated in \( O(n^3) \) time.

**Proof.** Let \( G_0 \) be induced by the union of the holes of \( G \). On the one hand, according to Lemmas 7.7 and 7.9, each minimal hole cover of \( G \) corresponds to a minimal separator of the interval subgraph \( G_0 - N[v] \) for some vertex \( v \) of a shortest hole \( H \). On the other hand, there are at most \( n \) minimal separators in \( G_0 - N[v] \) for each vertex \( v \in H \), which implies a quadratic bound for the total number of minimal hole covers of \( G \). To enumerate them, we try every vertex \( v \in H \) and enumerate all minimal separators of \( G_0 - N[v] \).

8. CATERPILLAR DECOMPOSITIONS

This section proves Theorem 2.4 by providing the claimed algorithm for INTERVAL DELETION on nice graphs. Recall that a nice graph is chordal and contains no small \( AW \).
and every shallow terminal in a nice graph is simplicial; nice graphs are hereditary. Our algorithm finds an AW satisfying a certain minimality condition, from which we can construct a set of 10 vertices that intersects some minimum interval deletion set. Hence, it branches on deleting one of these 10 vertices. The set of all shallow terminals, denoted by $ST(G)$, can be found in polynomial time as follows. For each triple of vertices, we check whether or not they form the terminals for an AW. If yes, then one of them is necessarily shallow. The following lemma ensures that all shallow terminals can be found as such.

**Proposition 8.1.** In a nice graph, all AWs with the same set of terminals have the same shallow terminal.

**Proof.** Of any AT $\{x, y, z\}$, there must be a vertex, say, $x$, such that the shortest $y$-$z$ path in $G - N[x]$ has length at least 4, as otherwise there is an AW of size at most 9, which contradicts the definition of nice graphs. Therefore, neither $y$ nor $z$ can be the shallow terminal in an AW with terminals $\{x, y, z\}$. \[\square\]

It should be noted that this does not rule out the possibility of a vertex being a base terminal of an AW and the shallow terminal of another AW. If this happens, these AWs necessarily have at least one different terminal. Recall that by Theorem 2.1, every vertex in $ST(G)$ is simplicial in $G$. For each $\uparrow$- or $\downarrow$-AW, its shallow terminal is in $ST(G)$ by definition, its base terminals might or might not be in $ST(G)$, and none of the non-terminal vertices can be in $ST(G)$ (as they are not simplicial). From Lemma 5.2, we can derive the following:

**Proposition 8.2.** Let $s$ be a shallow terminal in a nice graph. There is an AW of which every base vertex is adjacent to all vertices of $N(s) \setminus ST(G)$.

**Proof.** Let $W$ be an AW with shallow terminal $s$ and shortest possible base. Applying Lemma 5.2 on any vertex $x \in N(s) \setminus ST(G)$ and $W$, it cannot be in category “partial” by the minimality of $W$. Vertex $x$ cannot be in category “none” either; otherwise, $x$ is a shallow terminal, contradicting $x \in N(s) \setminus ST(G)$. Thus, every vertex in $N(s) \setminus ST(G)$ is in category “full.” \[\square\]

Now that the graph is chordal, it makes sense to discuss its clique tree, which shall be the main structure of this section. No generality will be lost by assuming that $G$ is connected. Since no inner vertex of a shortest path can be simplicial, the removal of simplicial vertices will not disconnect a connected graph; hence, $G - ST(G)$ is a connected interval graph. This observation suggests a clique tree of $G$ with a very nice structure. A caterpillar (tree) is a tree that consists of a central path and all other vertices are leaves connected to it.

**Proposition 8.3.** In polynomial time, we can construct a clique tree $T$ for a connected nice graph $G$ such that

—$T$ is a caterpillar,
—every shallow terminal of $G$ appears only in one leaf node of $T$, and
—every other vertex in $G$ appears in some nodes of the central path of $T$.

**Proof.** Let us inspect every maximal clique $K$ of $G$. If $K$ contains some shallow terminal, then $K$ must be $N[s]$: being a clique, $K \subseteq N[s]$; this containment cannot be proper, as $K$ is maximal and $N[s]$ induces a clique. Otherwise, $K \cap ST(G) = \emptyset$, then $K$ is also a maximal clique of $G - ST(G)$. On the other hand, a maximal clique $K'$ of $G - ST(G)$ has to be a maximal clique of $G$ as well; otherwise, it is a proper subset of some maximal clique $K$ of $G$ that must contain a shallow terminal $s$; hence, $K = N[s]$ and $K' \subseteq N[s] \setminus ST(G)$, which, however, according to Proposition 8.2, cannot be maximal.
in $G - ST(G)$. Therefore, a maximal clique of $G$ is either $N[s]$ for some vertex $s \in ST(G)$ or a maximal clique of $G - ST(G)$. We construct the claimed clique tree as follows: first use Theorem 3.1 to make a clique path $T'$ for the interval subgraph $G - ST(G)$, then for each $s \in ST(G)$, attach $N[s]$ as a leave to $T'$ at some maximal clique that properly contains $N[s] \setminus ST(G)$ (arbitrarily pick from multiple choices).

Within a caterpillar decomposition, we number the nodes in the central path as $K_0, K_1, \ldots$. By Proposition 8.3 and the definition of clique trees, each vertex not in $ST(G)$ is contained in some consecutive nodes of the central path. For each vertex $v \notin ST(G)$, we denote by $\text{first}(v)$ and $\text{last}(v)$ the smallest and, respectively, largest indices of nodes that contain $v$. In any $\uparrow$- or $\downarrow$-AW, every vertex of the base is nonsimplicial and hence belongs to the central path of the caterpillar decomposition. By assumption, $d = |B| \geq 3$ and $b_1 \neq b_d$; as a result, the nodes that contain $b_1$ and $b_d$ are disjoint. When numbering the vertices of the base, we follow the convention that $\text{last}(b_1) < \text{first}(b_2)$—that is, base $B$ goes “from left to right.” Given a numbering of the base, the base terminals $l$ and $r$ can be distinguished from each other based on their adjacency with $b_1$ and $b_d$. Similarly, in the case of a $\uparrow$-AW, the centers $c_1$ and $c_2$ can be distinguished from each other, as they have different adjacency relations with $l$ and $r$.

By observing the adjacencies and nonadjacencies between vertices of an AW and their possible positions in an interval representation of $G - ST(G)$, the following is straightforward and hence stated here without proof. To avoid pointless repetition, we are again using the same generalized notation for both $\uparrow$- and $\downarrow$-AW as stipulated in Section 4.

**Proposition 8.4.** Let $(s : c_1, c_2 : l, B, r)$ be an $\uparrow$- or $\downarrow$-AW in a nice graph $G$. In a caterpillar decomposition of $G$,

$$(\text{first}(b_1) \leq \text{last}(l) < \text{first}(c_2), \text{first}(b_2) \leq \text{last}(b_1) < \ldots \leq \text{first}(b_i) \leq \text{last}(b_i-1) < \text{first}(b_{i+1}) \leq \text{last}(b_i) < \text{first}(b_{i+2}) \leq \ldots \leq \text{first}(b_d) \leq \text{last}(b_d-1), \text{last}(c_1) (< \text{first}(r) \leq \text{last}(b_d)),$$

where relations in parentheses only hold when $l \notin ST(G)$ and $r \notin ST(G)$, respectively.

Nodes that contain nonterminal vertices of an AW appear consecutively in the central path of $T(G)$. We would like to identify a minimum set of consecutive nodes whose union contains all nonterminal vertices of the AW.

**Definition 8.5.** Let $T$ be a caterpillar decomposition of a nice graph $G$. We define $\Pi[p, q] = \bigcup_{p \leq i \leq q} K_i$ for a pair of indices $p \leq q$, and $\Pi(W) = \Pi[\text{last}(b_1), \text{first}(b_d)]$ for an AW $W$. Set $\Pi(W)$ will be referred to as the container of $W$, and we say that it is minimal if there exists no AW $W'$ such that $\Pi(W') \subset \Pi(W)$.

Let us observe that every base vertex of $W$ appears in $\Pi(W)$ and no shorter subsequence of nodes contains every base vertex. Moreover, the following proposition shows that the centers also appear in $\Pi(W)$ (recall that $\hat{N}(B)$ is the set of common neighbors of $B$ and every center is in $\hat{N}(B)$).

**Proposition 8.6.** $K_{\text{last}(b_1)} \cap K_{\text{first}(b_d)} = \hat{N}(B)$.

**Proof.** By definition, a vertex of the left side is in $K_i$ for every $\text{last}(b_1) \leq i < \text{first}(b_d)$ and thus belongs to $\hat{N}(B)$. On the other hand, if a vertex $v$ does not belong to the left

---

Note that we are not relying on the relation between $\text{first}(c_1)$ and $\text{first}(c_2)$ or that between $\text{last}(c_1)$ and $\text{last}(c_2)$, and they will not matter in the proofs to follow.
side, then either $\text{first}(v) > \text{last}(b_1)$ or $\text{last}(v) < \text{last}(b_2)$, which implies $v \not\in b_1$ or $v \not\in b_2$ respectively. In either case, we have $v \not\in \hat{N}(B)$. $\square$

In Section 6, we considered holes of the shortest length and observed that a vertex sees either all or at most three vertices in such a hole. Here, for an AW whose container is minimal and base consists of the inner vertices of a shortest $l$-$r$ path specified next, we can observe an analogous statement about the number of base vertices that a vertex can see.

**Definition 8.7.** Let $W = (s : c_1, c_2 : l, B, r)$ be an AW in a nice graph such that $\Pi(W)$ is minimal. We say that $B$ is a short base if $(lBr)$ is a shortest $l$-$r$ path in the subgraph induced by $(\Pi(W) \setminus \hat{N}(B)) \cup \{l, r\}$.

The following lemma shows that if the base is not short, then we can get an AW with a shorter base. In particular, this implies that a vertex of $\Pi(W) \setminus \hat{N}(B)$ can see at most three consecutive vertices of the base.

**Lemma 8.8.** Let $W = (s : c_1, c_2 : l, B, r)$ be an AW such that $\Pi(W)$ is minimal. Then there is an $W'$ such that $\Pi(W') = \Pi(W)$ and $W'$ has a short base.

**Proof.** We show that if $(lPr)$ is a chordless $l$-$r$ path in the subgraph induced by $(\Pi(W) \setminus \hat{N}(B)) \cup \{l, r\}$, then we can replace the base $B$ of $W$ by $P$ to obtain another AW $W_P = (s : c_1, c_2 : l, P, r)$. Clearly, the center(s) of $W$ belong to $\hat{N}(B)$, thereby adjacent to every other vertex in $\Pi(W)$, and hence to $P$. It is also easy to verify that no vertex in $\Pi(W) \setminus \hat{N}(B)$ is adjacent to $s$: if such a vertex exists, then Lemma 5.2 classifies it as “partial” with respect to $W$; therefore, there is another AW $W'$ such that $B' \subseteq B$ and $\Pi(W') \subseteq \Pi(W)$, which contradicts the minimality of $\Pi(W)$. Thus, $W_P$ is indeed an AW. Letting $b'_1$ and $b'_2$ be the first and, respectively, last vertices of $P$, the selection of $P$ implies $\text{last}(b'_1) \geq \text{last}(b_1)$ and $\text{first}(b'_2) \leq \text{first}(b_2)$ and hence $\Pi(W_P) \subseteq \Pi(W)$; as the latter is already minimal, they must be equal. Therefore, if the base of $W$ is not short, then we can find another AW with the same container and shorter base. Applying this argument repeatedly will eventually procure an AW with the same container and having a short base. $\square$

With all pertinent definitions and observations, we are now ready to present the main lemma of this section, which justifies our branching rule. Without an upper bound on the number of vertices in an AW—in particular, the length of its base can be arbitrarily long—trying each vertex in it cannot be done in fixed-parameter tractable time. Thus, we have to avoid all but a (small) constant number of base vertices—those close to the base terminals—to procure the claimed algorithm. To further decrease the number of vertices that we need to consider, observing that the central path of the caterpillar decomposition has a linear structure, we start from the left-most minimal container. By definition, minimal containers cannot properly contain each other, and thus the one with smallest begin-index also has the smallest end-index. In particular, the left-most minimal container is unique, although it might be observed by more than one AW and can be identified in polynomial time. With this additional condition, if another AW intersects $\Pi(W)$, it has to come “from the right.”

Let $W$ be an AW of the left-most minimal container and having a short base. We claim that there is a minimum interval deletion set that breaks $W$ in a canonical way: it contains either one of a constant number of specific vertices of $W$ or a specific
minimum separator (details are given later) breaking the base of \( W \). Therefore, by branching into 10 directions, we can guess one vertex of this interval deletion set.\(^4\)

For each \( \text{last}(b_1) \leq i < \text{first}(b_{d-1}) \), let us define \( S_i = K_i \cap K_{i+1} \) to be the \( i \)th separator. Note that \( S_i \) contains \( \hat{N}(B) \) as a proper subset.

**Lemma 8.9.** Let \( T \) be a caterpillar decomposition of a nice graph \( G \), and let \( W = (s : c_1, c_2 : l, B, r) \) be an AW in \( G \) such that

- \( \text{first}(b_1) \) is the smallest among all AWs,
- \( \Pi(W) \) is minimal, and
- \( B \) is a short base.

Let \( \ell \) be the minimum index such that \( \text{last}(b_1) \leq \ell < \text{first}(b_{d-2}) \) and the cardinality of \( S_i \) is minimum among \( \{ S_i : \text{last}(b_1) \leq i < \text{first}(b_{d-2}) \} \). There is a minimum interval deletion set to \( G \) that either contains one of the nine vertices

\[
V_B = \{ s, c_1, c_2, l, b_1, b_{d-2}, b_{d-1}, b_d, r \},
\]

or the whole set \( X = S_i \setminus N \), where \( N = \hat{N}(B) \).

**Proof.** We prove by construction. Let \( Q \) be any minimum interval deletion set; we may assume that \( Q \cap V_B = \emptyset \) and \( X \not\subset Q \); otherwise \( Q \) satisfies the asserted condition and we are finished. We claim \( Q = (Q[V_I] \cup X, \text{where } V_I = \Pi[\text{last}(b_2), \text{first}(b_{d-3})] \setminus N \) is the desired interval deletion set, which fully contains \( X \) in particular.

As \( G \) is chordal, all minimal forbidden induced subgraphs in \( G \) are AWs. To show that \( Q \) makes an interval deletion set to \( G \), it suffices to argue that if there exists an AW \( W \) avoiding \( Q \), then we can also find an AW \( W' \), not necessarily the same as \( W \), avoiding \( Q \). Suppose that \( W = (s' : c'_1, c'_2 : l', B', r') \) is the AW in \( G - Q \). By the construction of \( Q \), this AW must intersect \( V_I \setminus X \); let \( u \in W' \cap (V_I \setminus X) \). Clearly, \( u \) can neither be \( s' \), as \( u \not\in ST(G) \), nor \( r' \), as otherwise according to Proposition 8.4, \( \text{first}(b_{d'}) < \text{first}(u) < \text{first}(b_d) \), contradicting the selection of \( W \). The following claim rules out the possibility that \( u \in \{ c'_1, c'_2 \} \).

**Claim 5.** For each vertex \( v \in \Pi[0, \text{first}(b_{d-2})] \setminus N \), we have \( \text{last}(v) < \text{first}(b_d) \), and \( v \not\in ST(G) \).

**Proof.** By definition, if \( v \) is adjacent to \( B \), then \( v \sim b_i \) for some \( i \leq d-3 \). If \( v \sim b_d \), then \( B \) is not a short base, as there is a shorter (not necessarily chordless) \( l-r \) path \((l, \ldots, b_i, v, b_d, r) \). Therefore, \( v \not\sim b_d \), and it follows that \( \text{last}(v) < \text{first}(b_d) \). Suppose to the contrary of the second assertion that \( v \) is adjacent to the shallow terminal \( x \) of some AW \( W_1 \). We apply Lemma 5.2(2) on \( v \) and \( W_1 \). As \( v \not\in ST(G) \), it has to be in categories “full” or “partial.” In either case, there exists an AW whose base is fully contained in \( \Pi[\text{first}(v), \text{last}(v)] \), contradicting the selection of \( W \).

Therefore, either \( u = l' \) or \( u \in B \). Now we focus on the chordless path \( l'Br' \), which we shall refer to by \( P' \), and how it reaches \( u \) when going from \( r' \) to \( l' \). Recall that every vertex of \( B \) appears in the central path of the caterpillar decomposition. Figure 9 depicts nonterminal vertices of \( W \) in an interval representation of the interval subgraph \( G - ST(G) \), where base terminals \( l \) and \( r \) are illustrated with dashed lines as they might belong to \( ST(G) \). The main observation here is this: for any vertex \( u \) in \( V_I \), if another

---

\(^4\)A slightly weaker version of Lemma 8.9 is given in the appendix. The proof of Lemma 8.9, trying to minimize the number of branching directions, has to consider many cases and is ponderous. In contrast, the proof of the weaker version uses only the fact that a vertex that is not a common neighbor of \( B \) sees at most three vertices in it; hence, the underlying ideas are easier to understand.
vertex $z \in N(u) \setminus N$ (the thick segment) reaches outside of $\Pi(W)$, then $z \sim b_d$, and $u$ and $z$ will make a short cut between $b_{d-4}$ and $b_d$, which is impossible.

**Claim 6.** $B \cap N = \emptyset$.

**Proof.** Suppose the contrary and let $x$ be a vertex in $B \cap N$ (see Figure 9). Then $s \sim x$ follows from Lemma 5.1. We claim that every neighbor $z$ of $u$ is adjacent to $x$. Note that $z \neq x \in N$ implies either $\text{last}(z) < \text{last}(b_1)$ or $\text{first}(z) > \text{first}(b_d)$, and the latter is ruled out by the definition of $u$ and Claim 5. Let $B_1$ be the subset of the inner vertices of the $z$-$r$ path $(x, u, \ldots, b_d, r)$. Then one of the following AWs contradicts the minimality of the choice of $W$: $(s : c_1, x : z, B_1, r)$ (when $c_1 \sim z$ and $x \sim r$), $(s : c_1 : z, B_1, r)$ (when $c_1 \neq z$), or $(s : x : z, B_1, r)$.

As $x$ and $u$ are both in the chordless path $P'$ and $x$ is adjacent to every neighbor of $u$, vertex $u$ has to be one end of $P'$. More specifically, $u = l'$ and $x = b_1'$. A further consequence is that $u$ is the only vertex in $W' \cap V_I$: the preceding argument applies to any vertex $u' \in W' \cap V_I$, and thus $u' = l' = u$. Now we show that, for any vertex $w$ in $X \setminus Q$, which is nonempty by assumption, it has the same neighbors as $u$ in $W'$, and hence $(s' : c_1', c_2' : w, B', r')$ is an AW in $G - Q$, contradicting the assumption that $Q$ is an interval deletion set to $G$.

First, if a vertex is in $N$, then it is adjacent to both $w$ and $u$. Vertex $b_1' (= x)$ is in $N$. We claim that $c_1'$ is also in $N$ when $w' = l'$ is a $\dagger$-AW. Otherwise, observe that $\text{first}(c_1') < \text{last}(u) < \text{first}(b_1)$, as $c_1' \sim w$ and $u$ satisfies the condition of Claim 5. Let $B_1$ be the path $(b_1, \ldots, b_1', u, c_1')$, where $b_1$ is the first base vertex of $W$ adjacent to $u$. Now one of the following AWs contradicts the minimality of $W$: $(s : x, c_2 : l, B_1, b_1')$ (when $x \sim l$ and $c_2 \sim b_1'$), $(s : x : l, B_1, b_1')$ (when $x \sim l$), or $(s : c_2 : l, B_1, b_1')$ (when $c_2 \neq b_1'$).

Second, $c_2' \neq u (= l')$ implies $c_2' \notin N$. We claim that $c_2' \notin N$. Otherwise, observe that $\text{first}(c_2') \leq \text{last}(w) < \text{first}(b_1')$, as $c_2' \sim w$ and $w$ satisfies the condition of Claim 5. Let $B_1$ be the path $(b_1, \ldots, b_1', w, c_2')$, where $b_1$ is the first base vertex of $W$ adjacent to $w$. Now one of the following AWs contradicts the minimality of $W$: $(s : x, c_2 : l, B_1, b_1')$ (when $x \sim l$ and $c_2 \sim b_1'$), $(s : x : l, B_1, b_1')$ (when $x \sim l$), or $(s : c_2 : l, B_1, b_1')$ (when $c_2 \neq b_1'$).

Finally, we claim that $w \neq b_1'$. Otherwise, observe that $\text{first}(b_1') \leq \text{last}(w) < \text{first}(b_1')$, as $b_1' \sim w$, $w$ satisfies the condition of Claim 5, and $b_1' \neq x (= b_1' \in N)$. Let $B_1$ be the path $(b_1, \ldots, w, b_1')$. Now one of the following AWs contradicts the minimality of $W$: $(s : x, c_2 : l, B_1, b_1')$ (when $x \sim l$ and $c_2 \sim b_1'$), $(s : c_2 : l, B_1, b_1')$ (when $c_2 \neq b_1'$), or $(s : x : l, B_1, b_1')$ (when $x \neq l$). Moreover, from $\text{last}(w) < \text{first}(b_1')$, it can be easily inferred that $w \neq b_1'$ for any $3 \leq i \leq d' + 1$.

Now that $P'$ reaches $u$ not through $N$, next we show that the center $c_2'$ has to be in $N$, as it is adjacent to all base vertices $b_i$ of $W$ for $3 \leq i \leq d$.

**Claim 7.** $c_2' \in N$. 

ACM Transactions on Algorithms, Vol. 11, No. 3, Article 21, Publication date: January 2015.
Suppose to the contrary that \( c'_2 \not\in N \), then \( c'_2 \) cannot be adjacent to \( b_1 \). From \( c'_2 \sim b'_1 \) and \( c'_2 \not\sim b_1 \), we can derive last\((b'_1) \geq first(c'_2) > last(b_1) \). On the other hand, as \( III(W) \) is minimal, it does not properly contain \( III(W') \), which implies \( first(b'_1) > first(b_2) \). Therefore, \( last(c'_2) > first(b'_1) > first(b_2) \) (see Figure 9). By the selection of \( W \) and \( B \), if a vertex \( z \) is adjacent to both \( u \) and \( b_d \), then \( z \in N \), as otherwise there exists a path \((l, b_1, \ldots, b_p, u, z, b_d, r)\), where \( p \leq d - 4 \), shorter than \( lBr \). In particular, the vertex next to \( u \) in the path \( l'Br' \) is not adjacent to \( b_d \); letting \( u = b'_i \) where \( i < d' \), it means that \( last(b'_i) < first(b_2) \). From \( c'_2 \sim b'_i \), we can conclude that \( first(c'_2) < first(b_2) \), and then \( c'_2 \) and \( b_d \) are adjacent, which further implies that \( u \) is not adjacent to \( c'_2 \). In other words, \( u \) has to be \( l' \). Let \( p \) be the index such that \( b_p \not\equiv c'_2 \) and \( b_{p+1} \sim c'_2 \), which exists by assumption. We now show \( b_{d-2} \not\equiv c'_2 \), and \( p \geq d - 2 \), by contradiction. • If \( s' \) is adjacent to every vertex in \( N \), then \((s': c_1, c'_2 : b_p, b_{p+1} \ldots b_d, r)\) or \((s': c'_2 : b_p, b_{p+1} \ldots b_d, r)\) would be an AW that contradicts the selection of \( W \). • If \( s' \) is not adjacent to \( x \in N \), then \((s: x, c_2 : l, b_1 \ldots b_{p+1}, c'_2, s'), (s: x : l, b_1 \ldots b_{p+1}, c_2, s')\), or \((s : c_2 : l, b_1 \ldots b_{p+1}, c'_2, s')\) would be an AW that contradicts the selection of \( W \): noting that \( first(c'_2) \leq last(b_{d-2}), as c'_2 \sim b_{d-2} \). However, \((s': c_1, c'_2 : b_p, b_q \ldots b_d, r'\), \((s': c_1 : b_p, b_q \ldots b_d, r')\), or \((s': c_2 : b_p, b_q \ldots b_d, r')\), where \( q \) is the largest index such that \( b_q \sim b_p \), will be an AW in \( G - Q \), which is impossible, as \( Q \) is an interval deletion set to \( G \).

An immediate consequence of Claim 7 is \( c'_2 \sim u \); hence, \( u \in B \). By Proposition 8.4, \( first(b'_1) \leq first(u) < first(b_{d-2}) \). Then from Claim 5 and the fact that \( l' \sim b'_1 \), it can be inferred that \( l' \not\in ST(G) \). Now \( first(l') \) is defined, and \( first(l') \leq first(c'_2) < last(b_1) \); the selection of \( W \) implies \( first(b'_1) > first(b_2) \). Therefore, the \( l'-b'_1 \) path \( l'B \) has to go through \( X \), and we end with a contradiction.

This verifies that \( Q \) is an interval deletion set to \( G \), and it remains to show that \( Q \) is minimum, from which the lemma follows.

**Claim 8.** \(|Q| \leq |Q| \).

**Proof.** It will suffice to show that \( Q \cap V_I \) makes a \( b_1-b_{d-2} \) separator in \( G - N \), and then the claim ensues as

\[
|Q| = |Q \cap V_I| + |X| \leq |Q \cap V_I| + |Q \cap V_I| = |Q|.
\]

Suppose to the contrary that there is a chordless \( b_1-b_{d-2} \) path \( P \). We can extend \( P \) into an \( l-r \) path \( P^+ = (lPb_{d-1}b_1r) \), which is disjoint from \( Q \) and \( N \). Within \( P^+ \), there is a chordless \( l-r \) path \( (B_1r) \). By assumption, \([s, c_1, c_2] \cap Q = \emptyset\); every vertex in \( B_1 \) satisfies the condition of Claim 9 and hence nonadjacent to \( s \). Moreover, \( c_1, c_2 \not\in N \), and therefore both \( c_1 \) and \( c_2 \) are adjacent to every vertex of \( B_1 \). Thus, \((s : c_1, c_2 : l, B_1, r)\) is an AW in \( G - Q \), which is impossible.

This completes the proof of the lemma. □

To prove Theorem 2.4, we need one last piece of the jigsaw—that is, to find the AW required by Lemma 8.9.

**Theorem 2.4 (Restated).** There is a \( 10^k \cdot n^{O(1)} \)-time algorithm for interval deletion on nice graphs.

**Proof.** Based on Lemma 8.9, it suffices to show how to find such an AW, and then the standard branching will deliver the claimed algorithm. For any triple of vertices \([x, y, z] \) and pair of indices \([p, q] \) for the nodes in the central path of the caterpillar decomposition, we can check whether or not there is an AW \( W \) whose terminals are \([x, y, z] \) and nonterminal vertices are fully contained in \( \Pi[p, q] \). Therefore, in \( O(n^6) \)
time, we are able to find the correct terminals and indices, from which the short base $B$ can also be easily constructed. This finishes the construction of the AW required by Lemma 8.9. □

9. CONCLUDING REMARKS

We have classified INTERVAL DELETION to be fixed-parameter tractable by presenting a $c^k \cdot n^{O(1)}$-time algorithm with $c = 10$. The constant $c$ might be improvable, and let us have a brief discussion on how to achieve this. The current constant 10 comes from Reduction 1 and Theorem 2.4. The constant in Reduction 1 is not tight, and it can be replaced by 8. We choose the current number for the convenience of later argument; for example, if we do not break AWs of size 9 in preprocessing, then we have to use a far more complicated proof for Proposition 8.1. In other words, the real dominating step is to break ATs in nice graphs, where we need to branch into 10 cases. As a nice graph exhibits a linear structure, it might help to apply dynamic programming here.

To further lower the constant $c$, we need to break small forbidden induced subgraphs in a better way than the brute force in our algorithm. So a natural question is this: can it be $c = 2$?

It is known that CHORDAL COMPLETION can be solved in polynomial time if the input graph is a circular-arc graph [Kloks et al. 1998], whereas INTERVAL COMPLETION remains NP-hard on chordal graphs [Peng and Chen 2006]. It would be interesting to inquire as to the complexity of INTERVAL DELETION on chordal graphs and other graph classes. At least, we could ask if it can be solved in polynomial time if the input graph is nice, which, if positively answered, would suggest that all of the troubles are small forbidden subgraphs. We leave open the parameterized complexity of INTERVAL EDGE DELETION, which instead asks for a set of $k$ edges whose removal makes an interval graph [Goldberg et al. 1995; Bodlaender et al. 1995]. To adapt our approach to this problem, one needs a reasonable bound for the number of edge hole covers for congenial holes.

As having been explored by Narayanaswamy and Subashini [2013], we would also like to ask which other problems can be formulated as or reduced to INTERVAL DELETION and then solved with our algorithm. Both practical and theoretical consequences are worth further investigation.

APPENDIX. A SIMPLER AND WEAKER VERSION OF LEMMA 8.9

Lemma a. Let $T$ be a caterpillar decomposition of a nice graph $G$, and let $W = (s : c_1, c_2 : l, B, r)$ be an AW in $G$ such that

— $\text{first}(b_d)$ is the smallest among all AWs,
— $\Pi(W)$ is minimal, and
— $B$ is a short base.

Let $\ell$ be the minimum index such that $\text{last}(b_2) \leq \ell < \text{first}(b_{d-5})$ and the cardinality of $S_\ell$ is minimum among $|S_i : \text{last}(b_2) \leq i < \text{first}(b_{d-5})|$. There is a minimum interval deletion set to $G$ that either contains one of the 13 vertices

$$V_B = \{s, c_1, c_2, l, b_1, b_2, b_{d-5}, b_{d-4}, b_{d-3}, b_{d-2}, b_{d-1}, b_d, r\},$$

or the whole set $X = S_\ell \setminus N$, where $N = \hat{N}(B)$.

PROOF. We prove by construction. Let $Q$ be any minimum interval deletion set; we may assume that $Q \cap V_B = \emptyset$, and $X \subseteq Q$, as otherwise $Q$ satisfies the asserted condition and we are finished. We claim that $Q = (Q \setminus V_I) \cup X$, where $V_I = \Pi([\text{last}(b_2), \text{first}(b_{d-6})] \setminus N)$, is the desired interval deletion set, which fully contains $X$ in particular. By definition of $V_I$, any vertex $z \in V_I$ is adjacent to some vertex $b_i$ for
4 \leq i \leq d - 7$, then as $B$ is short and $z \notin N$, we have

$$\text{first}(b_2) \leq \text{last}(b_1) < \text{first}(z) \leq \text{last}(x) < \text{first}(b_{d-4}) \leq \text{last}(b_{d-5}).$$  \hfill (1)

As $G$ is chordal, all minimal forbidden induced subgraphs in $G$ are AWs. To show that $Q$ makes an interval deletion set to $G$, it suffices to argue that if there exists an AW $W'$ avoiding $Q$, then we can also find an AW, not necessarily the same as $W'$, avoiding $Q$. Suppose that $W' = (s' : c_1', c_2' : l', B, r')$ is an AW in $G - Q$. By the construction of $Q$, this AW must intersect $\mathcal{V}_1 \setminus X_i$; let $u \in W' \cap (\mathcal{V}_1 \setminus X_i)$. Clearly, $u$ can neither be $s'$, as $u \notin ST(G)$, nor $r'$, as otherwise according to Proposition 8.4, $\text{first}(b') < \text{first}(u) < \text{first}(b_2)$, contradicting the selection of $W$. The following claim further rules out the possibility that $u \in [c_1', c_2']$.

**Claim 9.** For each vertex $v \in \Pi[0, \text{first}(b_{d-2})] \setminus N$, we have $\text{last}(v) < \text{first}(b_2)$, and $v \notin ST(G)$.

**Proof.** By definition, if $v$ is adjacent to $B$, then $v \sim b_i$ for some $i \leq d - 3$. If $v \sim b_d$, then $B$ is not a short base, as there would be a shorter (not necessarily chordless) $l$-$r$ path $(\ldots, b_i, v, b_d, r)$. Therefore, $v \notin b_d$, and it follows that $\text{last}(v) < \text{first}(b_2)$. Suppose to the contrary that $v$ is adjacent to the shallow terminal $x$ of some AW $W_1$. We apply Lemma 5.2(2) on $v$ and $W_1$. As $v \notin ST(G)$, it has to be in categories “full” or “partial.” In either case, there exists an AW whose base is fully contained in $\Pi[\text{first}(v), \text{last}(v)]$, contradicting the selection of $W$. \hfill \_\_\_

Therefore, either $u = l'$ or $u \in B$. Now we focus on the chordless path $l' Br'$, which we shall refer to by $P'$, and how it reaches $u$ when going from $r'$ to $l'$. Recall that every vertex of $B'$ appears in the central path of the caterpillar decomposition.

**Claim 10.** $B \cap N = \emptyset$.

**Proof.** Suppose the contrary and let $x$ be a vertex in $B \cap N$. By definition of $N$ and (2), we have $\text{first}(x) < \text{first}(u) \leq \text{last}(u) < \text{last}(x)$. Then every neighbor of $u$, which is not in $ST(G)$ according to Claim 9, is thus adjacent to $x$. As $x$ and $u$ are both in the chordless path $P'$, vertex $u$ has to be one end of it. More specifically, $u = l'$ and $x = b_i'$. A further consequence is that $u$ is the only vertex in $W' \cap V_1$; the preceding argument applies to any vertex $u' \in W' \cap V_1$, and thus $u' = l' = u$.

Now we show that, for any vertex $w$ in $X \setminus Q$, which is nonempty by assumption, it has the same neighbors as $u$ in $W'$, and hence $(s' : c_1', c_2' : w, B, r')$ is an AW in $G - Q$, contradicting the assumption that $Q$ is an interval deletion set to $G$. Observe that any vertex in $N$ is adjacent to both $u$ and $w$:

—The assumption $w \notin N$ implies $w \notin s'$; otherwise, $w$ is adjacent to both $s$ and $B$ but not in $N$, and we can apply Lemma 5.2 to $W$ and $w$, which is in category “partial,” to obtain an AW with strictly smaller container.

—By the selection of $W$, we have $\text{last}(c_i') \geq \text{first}(b_i') \geq \text{first}(b_d)$ for both $i = 1, 2$. If $c_i'$, where $i = 1$ or 2, is adjacent to one of $u$ and $w$, then (2) implies $\text{first}(c_i') < \text{last}(b_{d-5})$; as $B$ is short, $c_i'$ must be in $N$, then adjacent to both $u$ and $w$.

—Vertex $b_i'$ (= $x$) is in $N$ and hence adjacent to $w$.

—By definition, $b_2' \sim b_2$ and $b_2' \not\sim b_i' (\in N)$ imply $\text{last}(b_2') \geq \text{first}(b_i') > \text{last}(b_1') \geq \text{first}(b_2)$. On the other hand, $b_2' \not\sim u$ implies $b_2' \not\sim N$. Then, as $B$ is short, $\text{first}(b_2') > \text{last}(b_{d-5})$. Therefore, from (2) we can conclude for $2 \leq i \leq d' + 1$, it holds that $\text{first}(b_i') > \text{last}(w)$ and thus $w \not\sim b_i'$. \hfill \_\_\_

**Claim 11.** $c_2' \in N$.  

ACM Transactions on Algorithms, Vol. 11, No. 3, Article 21, Publication date: January 2015.
PROOF. As \( u = b'_j \) for some \( 0 \leq i \leq d' \), Proposition 8.4 and (2) imply \( \text{first}(b'_j) (\leq \text{last}(l')) \leq \text{last}(u) < \text{last}(b_{d-3}) \). By Claim 10, \( b'_j \) is not in \( N \) and adjacent to at most three vertices of \( B \); thus, \( \text{last}(b'_j) < \text{first}(b_{d-2}) \leq \text{last}(b_{d-3}) \). On the other hand, by the selection of \( W \), we have \( \text{last}(c'_2) \geq \text{first}(b'_d) \geq \text{first}(b_4) \). Therefore, \( c'_2 \) is adjacent to at least four vertices of \( B \) and is in \( N \).

From Claim 11, we can conclude that \( c'_2 \sim u \) and then \( u \in B \). By Proposition 8.4, \( \text{first}(b'_j) < \text{first}(u) < \text{first}(b_{d-4}) \). Then, from Claim 9 and the fact that \( l' \sim b'_j \), it can be inferred that \( l' \notin ST(G) \). Now \( \text{last}(l') \) is defined, and \( \text{last}(l') < \text{first}(c'_2) \leq \text{last}(b_4) \); the selection of \( W \) implies \( \text{first}(b'_d) \geq \text{first}(b_4) \). Therefore, the \( l' - b'_d \) path \( l'B \) has to go through \( X \) and we end with a contradiction. This verifies that \( Q \) is an interval deletion set to \( G \), and it remains to show that \( Q \) is minimum, from which the lemma follows.

CLAIM 12. \( |Q| \leq |Q'| \).

PROOF. It will suffice to show that \( Q \cap V_I \) makes a \( b_2 - b_{d-5} \) separator in \( G - N \), and then the claim ensues as

\[
|Q'| = |Q \cap V_I| + |X| \leq |Q \setminus V_I| + |Q \cap V_I| = |Q|.
\]

Suppose to the contrary that there is a chordless \( b_2 - b_{d-5} \) path \( P \). We can extend \( P \) into an \( l \)-\( r \) path \( P^+ = (b_1Pb_2 \cdots b_{d-5}b_{d-1}b_dr) \), which is disjoint from \( Q \) and \( N \). Within \( P^+ \), there is a chordless \( l \)-\( r \) path \( (lB_1r) \). By assumption, \( \{s, c_1, c_2 \} \cap Q = \emptyset \); every vertex in \( B_1 \) satisfies the condition of Claim 9 and hence nonadjacent to \( s \). Moreover, \( c_1, c_2 \notin N \), and therefore both \( c_1 \) and \( c_2 \) are adjacent to every vertex of \( B_1 \). Thus, \( (s : c_1, c_2 : l, B_1, r) \) is an AW in \( G - Q \), which is impossible.

This completes the proof of the lemma.

REFERENCES
Isolde Adler, Martin Grohe, and Stephan Kreutzer. 2008. Computing excluded minors. In Proceedings of SODA 2008. 641–650. doi:10.1145/1347082.1347153
Amotz Bar-Noy, Reuven Bar-Yehuda, Ari Freund, Joseph Naor, and Baruch Schieber. 2001. A unified approach to approximating resource allocation and scheduling. Journal of the ACM 48, 5, 1069–1090. doi:10.1145/502102.502107
Seymour Benzer. 1959. On the topology of the genetic fine structure. Proceedings of the National Academy of Sciences 45, 11, 1607–1620.
Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, Michael T. Hallett, and Harold T. Wareham. 1995. Parameterized complexity analysis in computational biology. Computer Applications in the Biosciences 11, 1, 49–57. doi:10.1093/bioinformatics/11.1.49
Kellogg S. Booth and George S. Lueker. 1976. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. Journal of Computer and System Sciences 13, 3, 335–379. doi:10.1016/S0022-0000(76)80045-1 A preliminary version appeared in STOC 1975.
Andreas Brandstädt, Van Bang Le, and Jeremy P. Spinrad. 1999. Graph Classes: A Survey. SIAM, Philadelphia, PA.
Peter Buneman. 1974. A characterization of rigid circuit graphs. Discrete Mathematics 9, 4, 205–212. doi:10.1016/0012-365X(74)90007-3
Leizhen Cai. 1996. Fixed-parameter tractability of graph modification problems for hereditary properties. Information Processing Letters 58, 4, 171–176. doi:10.1016/0020-0190(96)00050-6
Yixin Cao, Jianer Chen, and Yang Liu. 2010. On feedback vertex set: New measure and new structures. In Algorithm Theory—SWAT 2010. Lecture Notes in Computer Science, Vol. 6139. Springer, 93–104. doi:10.1007/978-3-642-13731-0_10
Jianer Chen, Yang Liu, Songjian Lu, Barry O’Sullivan, and Igor Razgon. 2008. A fixed-parameter algorithm for the directed feedback vertex set problem. Journal of the ACM 55, 5, Article No. 21. doi:10.1145/1411509.1411511
Gabriel A. Dirac. 1961. On rigid circuit graphs. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg 25, 1, 71–76. doi:10.1007/BF02992776

ACM Transactions on Algorithms, Vol. 11, No. 3, Article 21, Publication date: January 2015.
Interval Deletion Is Fixed-Parameter Tractable

Rodney G. Downey and Michael R. Fellows. 2013. Fundamentals of Parameterized Complexity. Springer. DOI: http://dx.doi.org/10.1007/978-1-4471-5559-1

Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. 2012. Planar F-deletion: Approximation and optimal FPT algorithms. In Proceedings of FOCS 2012. IEEE, Los Alamitos, CA, 470–479. DOI: http://dx.doi.org/10.1109/FOCS.2012.62

Delbert R. Fulkerson and Oliver A. Gross. 1965. Incidence matrices and interval graphs. Pacific Journal of Mathematics 15, 3, 835–855.

Tibor Gallai. 1967. Transitiv orientierbare Graphen. Acta Mathematica Academiae Scientiarum Hungarica 18, 1–2, 25–66. DOI: http://dx.doi.org/10.1007/BF02020961 Translated by Frédéric Maffray and Myriam Preissmann in Perfect Graphs, Jorge L. Ramirez-Alfonsin and Bruce A. Reed (Eds.). Wiley, 2001, 25–66.

Paul W. Goldberg, Martin C. Golumbic, Haim Kaplan, and Ron Shamir. 1995. Four strikes against physical mapping of DNA. Journal of Computational Biology 2, 1, 139–152. DOI: http://dx.doi.org/10.1089/ cmb.1995.2.139

Martin C. Golumbic. 2004. Algorithmic Graph Theory and Perfect Graphs. Annals of Discrete Mathematics, Vol. 57. North-Holland Publishing, Amsterdam, The Netherlands.

György Hajós. 1957. Über eine Art von Graphen. Internationale Mathematische Nachrichten 11.

Haim Kaplan, Ron Shamir, and Robert E. Tarjan. 1999. Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs. SIAM Journal on Computing 28, 5, 1906–1922. DOI: http://dx.doi.org/10.1137/S0097539796303044 A preliminary version appeared in FOCS 1994.

Richard M. Karp. 1993. Mapping the genome: Some combinatorial problems arising in molecular biology. In Proceedings of STOC 1993 ACM, New York, NY, 278–285. DOI: http://dx.doi.org/10.1145/167088.167170

Ken-ichi Kawarabayashi. 2009. Planarity allowing few error vertices in linear time. In Proceedings of FOCS 2009. IEEE, Los Alamitos, CA, 639–648. DOI: http://dx.doi.org/10.1109/FOCS.2009.45

Ken-ichi Kawarabayashi and Bruce A. Reed. 2010. An (almost) linear time algorithm for odd cycles transversal. In Proceedings of SODA 2010.365–378.

David George Kendall. 1969. Incidence matrices, interval graphs and seriation in archaeology. Pacific Journal of Mathematics 28, 565–570.

Ton Kloks, Dieter Kratsch, and C. K. Wong. 1998. Minimum fill-in on circle and circular-arc graphs. Journal of Algorithms 28, 2, 272–289. DOI: http://dx.doi.org/10.1006/jagm.1998.0936 A preliminary version appeared in ICALP 1996.

C. G. Lekkerkerker and J. Ch. Boland. 1962. Representation of a finite graph by a set of intervals on the real line. Fundamenta Mathematicae 51, 45–64.

John M. Lewis and Mihalis Yannakakis. 1980. The node-deletion problem for hereditary properties is NP-complete. Journal of Computer and System Sciences 20, 2, 219–230. DOI: http://dx.doi.org/10.1016/0022-0000(80)90060-4 Preliminary versions were independently presented in STOC 1978.

Carsten Lund and Mihalis Yannakakis. 1993. The approximation of maximum subgraph problems. In Automata, Languages, and Programming. Lecture Notes in Computer Science, Vol. 700. Springer, 40–51. DOI: http://dx.doi.org/10.1007/3-540-56839-1_60

Dániel Marx. 2010. Chordal deletion is fixed-parameter tractable. Algorithmica 57, 4, 747–768. DOI: http://dx.doi.org/10.1007/s00453-008-9233-8

Dániel Marx and Ildikó Schlotter. 2012. Obtaining a planar graph by vertex deletion. Algorithmica 62, 3–4, 807–822. DOI: http://dx.doi.org/10.1007/s00453-010-9484-z

N. S. Narayanaswamy and R. Subashini. 2013. FPT algorithms for consecutive ones submatrix problems. In Parameterized and Exact Computation. Lecture Notes in Computer Science, Vol. 8246. Springer, 295–307. DOI: http://dx.doi.org/10.1007/978-3-319-03898-8_25

Sheng-Lung Peng and Chi-Kang Chen. 2006. On the interval completion of chordal graphs. Discrete Applied Mathematics 154, 6, 1003–1010. DOI: http://dx.doi.org/10.1016/j.dam.2005.09.010

Bruce A. Reed, Kaleigh Smith, and Adrian Vetta. 2004. Finding odd cycle transversals. Operations Research Letters 32, 4, 299–301. DOI: http://dx.doi.org/10.1016/j.orl.2003.10.009

Pim Van’t Hof and Yngve Villanger. 2013. Proper interval vertex deletion. Algorithmica 65, 4, 845–867. DOI: http://dx.doi.org/10.1007/s00453-012-9661-3

Yngve Villanger, Pinar Heggernes, Christophe Paul, and Jan Arne Telle. 2009. Interval completion is fixed parameter tractable. SIAM Journal on Computing 38, 5, 2007–2020. DOI: http://dx.doi.org/10.1137/ 070710913 A preliminary version appeared in STOC 2007.

Received May 2013; revised January 2014; accepted April 2014