Three problems on well-partitioned chordal graphs*

Jungho Ahn[1,2], Lars Jaffke[3], O-joung Kwon[4,2], and Paloma T. Lima[3]

[1]Department of Mathematical Sciences, KAIST, Daejeon, South Korea
[2]Discrete Mathematics Group, Institute for Basic Science (IBS), Daejeon, South Korea
[3]Department of Informatics, University of Bergen, Norway
[4]Department of Mathematics, Incheon National University, South Korea
junghoahn@kaist.ac.kr,{lars.jaffke,paloma.lima}@uib.no,ojoungkwon@gmail.com

May 26, 2021

Abstract

In this work, we solve three problems on well-partitioned chordal graphs. First, we show that every connected (resp., 2-connected) well-partitioned chordal graph has a vertex that intersects all longest paths (resp., longest cycles). It is an open problem [Balister et al., Comb. Probab. Comput. 2004] whether the same holds for chordal graphs. Similarly, we show that every connected well-partitioned chordal graph admits a (polynomial-time constructible) tree 3-spanner, while the complexity status of the Tree 3-Spanner problem remains open on chordal graphs [Brandstädt et al., Theor. Comput. Sci. 2004]. Finally, we show that the problem of finding a minimum-size geodetic set is polynomial-time solvable on well-partitioned chordal graphs. This is the first example of a problem that is NP-hard on chordal graphs and polynomial-time solvable on well-partitioned chordal graphs. Altogether, these results reinforce the significance of this recently defined graph class as a tool to tackle problems that are hard or unsolved on chordal graphs.

1 Introduction

In this work, we deepen the structural and algorithmic understanding of the recently introduced class of well-partitioned chordal graphs [1]. This subclass of chordal graphs generalizes split graphs in two ways. Split graphs can be viewed as graphs whose vertices can be partitioned into cliques that are arranged in a star structure, the leaves of which are of size one. Well-partitioned chordal graphs are graphs whose vertex set can be partitioned into cliques that can be arranged in a tree structure, without any limitations on the size of any clique.

The star-like structure of split graphs is fairly restricted compared to the tree-like structure of chordal graphs. Questions in structural or algorithmic graph theory which are difficult to answer on chordal graphs may have an easy solution on split graphs thanks to their restricted structure. A natural path to a resolution of such questions on chordal graphs is to extend their solutions on split graphs to graph classes that are structurally closer to chordal graphs. Well-partitioned chordal graphs exhibit a tree-like structure, which makes them a natural target in such a scenario. We consider two such questions: We show that every well-partitioned chordal graph has a vertex that intersects all its longest paths (or cycles), while the corresponding question on chordal graphs

*An extended abstract of this work appeared in the proceedings of CIAC 2021.

has remained an open problem [4]. We also show that every well-partitioned chordal graph has a polynomial-time constructible tree 3-spanner, while the complexity of the TREE 3-SPANNER problem remains unresolved on chordal graphs [5]. We discuss these problems in more detail below.

There are several examples of algorithmic problems in the literature that are efficiently solvable in split graphs but hard on chordal graphs, see [1] and the references therein. In such cases it is worthwhile to narrow down the complexity gap between split and chordal graphs, especially due to the structural difference between the two classes. For several variants of vertex-coloring problems that are NP-hard on chordal graphs and polynomial-time solvable on split graphs, it was observed [1] that they remain NP-hard on well-partitioned chordal graphs. However, there was no example of such a problem that becomes polynomial-time solvable on well-partitioned chordal graphs. We give the first such example by showing that there is a polynomial-time algorithm that given a well-partitioned chordal graph, constructs a minimum-size geodetic set. This problem is known to be NP-hard on chordal graphs [16].

Transversals of longest paths and cycles. It is well-known that in a connected graph, every two longest paths always share a common vertex. In 1966, Gallai [19] asked whether every graph contains a vertex that belongs to all of its longest paths. This question, whose answer is already known to be negative in general [33, 34], was shown to have a positive answer on several well-known graph classes. It is not difficult to see that it holds for trees, and it has been shown for outerplanar graphs and 2-trees [14], which has later been generalized to series-parallel graphs, or equivalently, graphs of treewidth at most 2 [13]. (Interestingly, the couterexample for general graphs [33] has treewidth 3.) Besides that, Gallai's question has a positive answer on circular arc graphs [4, 23], P_4-sparse (which includes cographs) and $(P_5, K_{1,3})$-free graphs [10], dually chordal graphs [22], and $2K_2$-free graphs [20]. As alluded to above, it has a positive answer on split graphs [25], and this result has been generalized to starlike graphs [10]. Both split graphs and starlike graphs are subclasses of well-partitioned chordal graphs [1]. It remains a challenging open problem to determine whether all chordal graphs admit a longest path transversal of size one. As a step in the direction of answering this question for chordal graphs, we prove the following theorem.

Theorem 1. *Every connected well-partitioned chordal graph contains a vertex that intersects all its longest paths.*

A closely related question is whether a 2-*connected* graph has a vertex that intersects all its longest *cycles*. This question has also been studied extensively on graph classes, and several of the above mentioned references contain positive answers to this question on the corresponding graph classes. In some cases the results are not stated explicitly, but it is not too difficult to adapt the proofs for the case of longest paths to the case of longest cycles. We answer this question positively on 2-connected well-partitioned chordal graphs as well.

Theorem 2. *Every* 2*-connected well-partitioned chordal graph contains a vertex that intersects all its longest cycles.*

Tree 3-Spanner. For a connected graph G and a positive integer t, a spanning tree T of G is a *tree t-spanner* of G if for every pair (v, w) of vertices in G, $\mathrm{dist}_G(v, w) \leq t \cdot \mathrm{dist}_T(v, w)$, where $\mathrm{dist}_G(v, w)$ (resp., $\mathrm{dist}_T(v, w)$) denotes the length of shortest path in G (resp., T) from v to w. The TREE t-SPANNER problem asks whether a given graph G has a tree t-spanner. Tree t-spanners are motivated from applications including network research and computational geometry [2, 26]. Cai and Corneil [9] showed that Tree t-Spanner is linear-time solvable if $t \leq 2$, and is NP-complete if $t \geq 4$. For $t = 3$, the complexity of Tree 3-Spanner is not yet unveiled. Brandstädt et al. [5] investigated the complexity of Tree t-Spanner on chordal graphs of small diameter. They showed that for even $t \geq 4$ (resp., odd $t \geq 5$) it is NP-complete to decide if a chordal graph of diameter at most $t+1$ (resp., $t+2$) has a tree t-spanner. On the other hand, for any even t (resp., odd t), every chordal graph of diameter at most $t-1$ (resp., $t-2$) admits a tree t-spanner which can be found in linear time. Brandstädt et al. [5] also showed that Tree 3-Spanner is polynomial-time solvable on chordal graphs of diameter at most 2. On general chordal graphs, the complexity of Tree 3-Spanner is still open. Several subclasses of chordal graphs, such as split [32], very strongly chordal [5], and interval [27] graphs were shown to be *tree* 3*-spanner admissible*, meaning that each of its members admits a tree 3-spanner. In the above mentioned cases, such tree 3-spanners can always be computed in polynomial time. We show that the same holds for well-partitioned chordal graphs, generalizing the result for split graphs [32].

Theorem 3. *Every connected well-partitioned chordal graph admits a tree* 3*-spanner which can be constructed in polynomial time.*

A subclass of chordal graphs that is not tree 3-spanner admissible and yet has a polynomial-time algorithm for Tree 3-Spanner is that of 2-sep chordal graphs, as shown by Das and Panda [29]. Other (non-chordal) graph classes that are known to be tree 3-spanner admissible are bipartite ATE-free graphs [6] (which include convex graphs) and permutation graphs [27]; and there are polynomial-time algorithms for Tree 3-Spanner on cographs and co-bipartite graphs [8], as well as planar graphs [18].

Geodetic Sets. Given a graph G and a vertex set $S \subseteq V(G)$, the *geodetic closure of* S is the set of vertices that lie on a shortest path between a pair of distinct vertices in S. Such a set S is called a *geodetic set* if the geodetic closure of S is the entire vertex set of G. The Geodetic Set problem asks, given a graph G, for the smallest size of any geodetic set in G. The study of geodetic sets was initiated by Harary et al. [21] in 1986, and is related to convexity measures in graphs; we refer to [30] for an overview. Harary et al. [21] showed that the Geodetic Set problem is NP-hard on general graphs, see also [3]. Dourado et al. [16] showed that Geodetic Set remains NP-hard on chordal graphs, and that it is polynomial-time solvable on split graphs. We extend their ideas to give a polynomial-time algorithm for well-partitioned chordal graphs.

Theorem 4. *There is a polynomial-time algorithm that given a well-partitioned chordal graph G, computes a minimum-size geodetic set of G.*

The complexity of Geodetic Set has been deeply studied on graph classes. Besides the above mentioned results, it was shown to be NP-hard on chordal bipartite [16] and bipartite [15] graphs, as well as co-bipartite [17], subcubic [7], and planar graphs [12]. Very recently, Chakraborty et al. [11] showed NP-hardness on subcubic partial grids, which unifies hardness on subcubic, planar, and bipartite graphs. Interestingly, they showed that Geodetic Set is NP-hard even on interval graphs, while a polynomial-time algorithm for *proper* interval graphs is known due to Ekim et al. [17]. Other graph classes that are known to admit polynomial-time algorithms are cographs [16], outerplanar graphs [28], block-cactus graphs [17], and solid grid graphs [11]. Kellerhals and Koana [24] recently assessed the parameterized complexity of Geodetic Set, and proved it to be W[1]-hard parameterized by solution size plus pathwidth and feedback vertex set, while devising FPT-algorithms for the parameter feedback edge set as well as for tree-depth.

2 Preliminaries

In this paper, all graphs are simple and finite. For a graph G, we denote by $V(G)$ the vertex set of G, and by $E(G)$ the edge set of G. For graphs G and H, G is *isomorphic* to H if there is a bijection $\phi : V(G) \to V(H)$ such that for all vertices v and w of G, $vw \in E(G)$ if and only if $\phi(v)\phi(w) \in E(H)$. For graphs G and H, let $G \cup H := (V(G) \cup V(H), E(G) \cup E(H))$.

For a vertex v of G, let $N_G(v)$ be the set of *neighbors* of v in G, that is, $N_G(v) := \{w \in V(G) | vw \in E(G)\}$, and $N_G[v] := N_G(v) \cup \{v\}$. The *degree* of v in G, denoted by $\deg_G(v)$, is the number of neighbors of v in G. For a vertex set X of G, let $N_G(X) := \bigcup_{v \in X}(N_G(v) \setminus X)$, and $N_G[X] := N_G(X) \cup X$. We may omit the subscript G if it is clear what is the base graph.

For graphs G and H, H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For a vertex set X of G, the *subgraph induced by* X, denoted by $G[X]$, is a graph $(X, \{vw \in E(G) | v, w \in X\})$. If X consists of a singleton v, then we use the shorthand $G - v$ instead of $G - \{v\}$. For vertex sets X and Y of G, we denote by $G[X, Y]$ the graph $(X \cup Y, \{xy \in E(G) \mid x \in X, y \in Y\})$. For disjoint vertex sets X and Y of G, we say that X is *complete* to Y if each vertex in X is adjacent to every vertex in Y.

A graph G is *connected* if for every nonempty proper subset $X \subsetneq V(G)$, there are vertices $x \in X$ and $y \in V(G) \setminus X$ such that $xy \in E(G)$, and *disconnected*, otherwise. A *component* of G is a maximal connected subgraph of G, that is, an induced subgraph G' of G such that for any vertex $v \in V(G) \setminus V(G')$, $G[V(G') \cup \{v\}]$ is disconnected. A graph G is 2-*connected* if G is connected and has no vertex v such that $G - v$ is disconnected. A *cycle* is a connected graph where all vertices have degree exactly 2. A *forest* is a graph having no cycles, and a *tree* is a connected forest. Given a forest F, we sometimes call its vertex a node, to distinguish it from a vertex of other graphs. A *path* is a tree where all vertices have degree at most 2. A *hole* in a graph G is an induced subgraph of G isomorphic to a cycle of length at least 4. A graph is *chordal* if it has no holes.

Given a forest F, a *leaf* is a node of F having degree at most 1. A *rooted tree* is a tree where one node is singled out as the *root*. In a rooted tree, for a non-root node v, the *parent* of v is a neighbor of v toward the root, and the *children* of v are neighbors of v which are not the parent of v.

Well-partitioned chordal graphs

Ahn et al. [1] introduced the class of *well-partitioned chordal graphs*, which is a subclass of chordal graphs. A connected graph G is *well-partitioned chordal* if $V(G)$ admits a partition $\mathcal{P}$ and a tree $\mathcal{T}$ having $\mathcal{P}$ as a vertex set satisfying the following conditions.

(i) Each partite set $X \in \mathcal{P}$ is a clique in G.

(ii) For each edge XY of $\mathcal{T}$, there are subsets $X' \subseteq X$ and $Y' \subseteq Y$ such that

$$E(G[X, Y]) = \{xy \mid x \in X', y \in Y'\}.$$

(iii) For each pair of distinct nodes X, Y in $\mathcal{T}$ with $XY \notin E(\mathcal{T})$, $E(G[X, Y]) = \emptyset$.

We call the tree $\mathcal{T}$ a *partition tree* of G, and the elements in $\mathcal{P}$ the bags of G. A graph is well-partitioned chordal if each of its components is well-partitioned chordal. Given a partition tree $\mathcal{T}$ of a connected well-partitioned chordal graph G and distinct nodes X and Y of $\mathcal{T}$, the *boundary of X with respect to Y*, denoted by $\mathrm{bd}(X, Y)$, is the set of vertices in X having neighbors in Y. Namely, $\mathrm{bd}(X, Y) := \{x \in X | N_G(x) \cap Y \neq \emptyset\}$. Note that by the second item of the above definition, $\mathrm{bd}(X, Y)$ and $\mathrm{bd}(Y, X)$ are complete to each other.

Theorem 5 (Ahn et al. [1]). *Given a graph G, in polynomial time, one can either determine that G is not well-partitioned chordal, or find a partition tree for each component of G.*

3 Transversals of longest paths and cycles

In this section, we show that well-partitioned chordal graphs admit both longest path transversals and longest cycle transversals of size one. We start with the following lemma, the proof of which exploits the Helly property of subtrees of a tree to show the existence of a bag of the partition tree that intersects all longest paths of a well-partitioned chordal graph. The same proof strategy has been used by Rautenbach and Sereni [31] to show that for any graph G, there exists a set of size $\mathbf{tw}(G) + 1$ that intersects all the longest paths of G.

Lemma 1. *Let G be a well-partitioned chordal graph with partition tree $\mathcal{T}$. Then there exists $X \in V(\mathcal{T})$ such that every longest path of G contains a vertex of X.*

Proof. Let $P_1, \ldots, P_\ell$ be the longest paths of G. Since G is connected, for each $1 \leq i \leq \ell$, the set of bags of $\mathcal{T}$ containing at least one vertex from P_i forms a subtree of $\mathcal{T}$. Let T_i be such a subtree. Since in any connected graph every two longest paths have a vertex in common, we have that $V(T_i) \cap V(T_j) \neq \emptyset$ for every $i \neq j$. By the Helly property[1] of subtrees of a tree, there exists $X \in V(\mathcal{T})$ such that $X \in V(T_i)$ for every $1 \leq i \leq \ell$. That is, X is a bag of $\mathcal{T}$ that intersects every longest path of G.

We prove a similar lemma for longest cycles of 2-connected well-partitioned chordal graphs. The proof of this lemma follows the same lines as the one presented above, hence we omit it here.

Lemma 2. *Let G be a 2-connected well-partitioned chordal graph with partition tree $\mathcal{T}$. Then there exists $X \in V(\mathcal{T})$ such that every longest cycle of G contains a vertex of X.*

Restatement of Theorem 1. *Every connected well-partitioned chordal graph has a vertex that intersects all its longest paths.*

Proof. Let $P_1, \ldots, P_\ell$ be the longest paths of G. By Lemma 1, there exists a bag $B \in V(\mathcal{T})$ such that $V(P_i) \cap B \neq \emptyset$ for every i. Let $B_1, \ldots, B_k$ be the neighbors of B in $\mathcal{T}$. We define $\mathcal{T}_i$ to be the connected component of $\mathcal{T} - B$ containing B_i and G_i to be the subgraph of G induced by the vertices contained in the bags of $\mathcal{T}_i$. Let p_i be the length of a longest path in G_i with one endpoint in $\mathrm{bd}(B_i, B)$. We may assume without loss of generality that $p_1 \geq p_i$ for every $i > 1$. We will now show that every longest path of G contains all the vertices of $\mathrm{bd}(B, B_1)$. Let P be a longest path of G and suppose for a contradiction that there exists $v \in \mathrm{bd}(B, B_1)$ such that $v \notin V(P)$. Recall that $V(P) \cap B \neq \emptyset$. If there exists $x, y \in B$ such that $xy \in E(P)$, then we can obtain a path longer than P by inserting v between x and y in P, a contradiction with the fact that P is a longest path of G. Similarly, no endpoint of P belongs to B, otherwise we would also find a path longer than P in G. The same holds also if there exists $x \in \mathrm{bd}(B, B_1)$ and $y \in \mathrm{bd}(B_1, B)$ such that $xy \in E(P)$. Indeed, since $\mathrm{bd}(B, B_1) \cup \mathrm{bd}(B_1, B)$ is a clique, we would again find a path longer than P by inserting v between x and y in P. Therefore P contains no edge crossing from B to B_1, which implies that $V(P) \cap V(G_1) = \emptyset$. Let $P = x_1 x_2 \ldots x_t$ and let x_j be a vertex of $V(P) \cap B$ such that for every $i \geq 1$ we have $x_{j+i} \notin B$. Such a vertex exists since $x_t \notin B$. Assume without loss of generality that

[1]The Helly property of trees states that in every tree, every collection of pairwise intersecting subtrees has a common nonempty intersection.

$x_{j+1} \in \mathrm{bd}(B_j, B)$. Note that $x_{j+1}x_{j+2}\dots x_t$ is a path in G_j with an endpoint in $\mathrm{bd}(B_j, B)$. Hence the length of this path is at most p_1. Let $P' = x_1x_2\dots x_j$ and P'' be a longest path in G_1 with an endpoint in $\mathrm{bd}(B_1, B)$. Then $P' \cdot v \cdot P''$ is a path in G that is longer than P, a contradiction.

With a more careful argument, we can prove the analogous result for longest cycles.

Restatement of Theorem 2. *Every 2-connected well-partitioned chordal graph has a vertex that intersects all its longest cycles.*

Proof. We start as in the proof of Theorem 1. Let $C_1, \dots, C_\ell$ be the longest cycles of G. By Lemma 2, there is a bag $B \in V(\mathcal{T})$ such that $V(C_i) \cap B \neq \emptyset$ for every i. Note that we can assume B is not a leaf of $\mathcal{T}$, since if all the longest cycles intersect a bag that is a leaf, they also intersect the bag that is the neighbor of such a leaf. Let $B_1, \dots, B_k$ be the neighbors of B in $\mathcal{T}$. We define $\mathcal{T}_i$ to be a maximal subtree of $\mathcal{T}$ containing B_i and not containing B and G_i to be the subgraph of G induced by the vertices contained in the bags of $\mathcal{T}_i$.

Now, let p_i be the length of a longest path in G_i with *both* endpoints in $\mathrm{bd}(B_i, B)$. Note that this is well-defined, since $|\mathrm{bd}(B_i, B)| \geq 2$ for every i, as G is a 2-connected graph. We may assume without loss of generality that $p_1 \geq p_i$ for every $i > 1$. We will now show that every longest cycle of G contains all the vertices of $\mathrm{bd}(B, B_1)$. Let C be a longest cycle of G and suppose for a contradiction that there exists $v \in \mathrm{bd}(B, B_1)$ such that $v \notin V(C)$. We first point out the following.

Claim 1. $|V(C) \cap B| \geq 2$.

Proof. We already know that $|V(C) \cap B| \geq 1$. Suppose for a contradiction that $|V(C) \cap B| = 1$. Then there exists $x_1, x_2, x_3 \in V(C)$ such that x_1, x_2, and x_3 appear consecutively in the cycle, and $x_2 \in B$ and $x_1, x_3 \notin B$. In particular, x_2 belongs to the boundary between B and some neighboring bag B_i, and $x_1, x_3 \in \mathrm{bd}(B_i, B)$. Since G is 2-connected, there exists $u \in \mathrm{bd}(B, B_i)$, with $u \neq x_2$, such that $u \notin V(C)$. Thus, we can add u between x_2 and x_3 in C and obtain a cycle longer than C, a contradiction. ⌟

If there exists $x, y \in B$ such that $xy \in E(C)$, then we can obtain a cycle longer than C by inserting v between x and y in C, a contradiction with the fact that C is a longest cycle of G. The same holds if there exists $x \in \mathrm{bd}(B, B_1)$ and $y \in \mathrm{bd}(B_1, B)$ such that $xy \in E(C)$. Indeed, since $\mathrm{bd}(B, B_1) \cup \mathrm{bd}(B_1, B)$ is a clique, we would again find a cycle longer than C by inserting v between x and y in C. Therefore C contains no edge crossing from B to B_1, which implies that $V(C) \cap V(G_1) = \emptyset$. Consider $u \in \mathrm{bd}(B, B_1)$ such that $u \neq v$. We consider two cases.

If $u \in V(C)$, since C cannot have two consecutive vertices in B, then there exists $i \neq 1$ such that $u \in \mathrm{bd}(B, B_i)$, and there exists $u' \in \mathrm{bd}(B_i, B)$ such that $uu' \in E(C)$. Moreover, by the above claim, there exists $u'' \in V(C) \cap \mathrm{bd}(B, B_i)$ such that if P is the subpath of C starting in u, ending in u'' and containing u', then $(V(P) \setminus \{u, u''\}) \subseteq V(G_i)$. Note also that $|P| \leq p_i + 2$, since the neighbors of u and u'' in P belong to $\mathrm{bd}(B_i, B)$. Let P_1 be a longest path of G_1 with both endpoints in $\mathrm{bd}(B_1, B)$ and let $P' = u \cdot P_1 \cdot vu''$. Let C' be the cycle obtained from C by replacing P by P'. Since $|P'| = p_1 + 3$ and $p_1 \geq p_i$, we have that C' is a cycle longer than C, a contradiction.

Now we consider the case in which $u \notin V(C)$. Recall that C cannot have two consecutive vertices in B. By Claim 1, there exists $i \neq 1$ such that $V(C) \cap V(G_i) \neq \emptyset$. Let $x, x', y, y' \in V(C)$ be such that $x, y \in \mathrm{bd}(B, B_i)$, $x', y' \in \mathrm{bd}(B_i, B)$, $xx', yy' \in E(C)$ and the subpath P of C starting in x, ending in y and containing x' and y' is such that $(V(P) \setminus \{x, y\}) \subseteq V(G_i)$. Note that it can be the case that $x' = y'$. Moreover, $|P| \leq p_i + 2$. Let P_1 be a longest path of G_1 with both endpoints in $\mathrm{bd}(B_1, B)$ and let $P' = xu \cdot P_1 \cdot vy$. Let C' be the cycle obtained from C by replacing P by P'. Since $|P'| = p_1 + 4$ and $p_1 \geq p_i$, we have that C' is a cycle longer than C, a contradiction. This concludes the proof that all the vertices of $\mathrm{bd}(B, B_1)$ are contained in all longest cycles of G.

4 Tree 3-spanner problem

In this section, we show that Tree 3-Spanner on well-partitioned chordal graphs can be solved in polynomial time. More specifically, we show that given a connected well-partitioned chordal graph, one can always find a tree 3-spanner in polynomial time.

Restatement of Theorem 3. *Every connected well-partitioned chordal graph admits a tree* 3*-spanner, which one can find in polynomial time.*

Proof. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$. We choose a bag R of $\mathcal{T}$ and consider it as a root bag. For each non-root bag B, let $P(B)$ denote the parent bag of B. For each non-root bag B,

- let S_B^* be a star whose center is in $\mathrm{bd}(B, P(B))$ and all leaves are exactly the vertices in $V(B) \setminus \mathrm{bd}(B, P(B))$,
- let S_B^{**} be a star whose center is in $\mathrm{bd}(P(B), B)$ and all leaves are exactly the vertices in $\mathrm{bd}(B, P(B))$, and
- let $S_B := S_B^* \cup S_B^{**}$.

Observe that the vertex set of S_B consists of all vertices of B and one vertex in $\mathrm{bd}(P(B), B)$. Moreover, S_B is a tree. For the root bag R, let S_R be a star on $V(R)$. We claim that $U := \bigcup_{B \in V(\mathcal{T})} S_B$ is a tree 3-spanner of G. It is sufficient to show that U is a spanning tree, and for every edge vw in G, $\mathrm{dist}_U(v, w) \leq 3$.

We first verify that U is a tree. Note that for each non-root bag B, S_B is a tree containing all vertices of B and at least one edge between B and $P(B)$, and furthermore, S_R is a spanning tree of R. Therefore, U is a connected subgraph containing all vertices of G. Suppose that U contains a cycle C.

Observe that for each non-root bag B of $\mathcal{T}$, the center of S_B^{**} separates $V(B)$ and $V(P(B))$ in U. Let B' be the bag containing a vertex of C such that $\mathrm{dist}_{\mathcal{T}}(R, B')$ is minimum. Since $U[V(B')]$ has no cycle, there is a child bag B'' of B' containing a vertex of C. By the above observation, $V(B') \cap V(C)$ has only one vertex that is the center of $S_{B''}^{**}$. As $|V(B') \cap V(C)| = 1$, there is no other child bag of B' containing a vertex of C.

By a repeated argument, we can see that there is no child bag of B'' containing a vertex of C. Then C contains $S_{B'}$, but by the construction, $S_{B'}$ has no cycle. We conclude that U is a spanning tree.

Now, we claim that for every edge vw in G, $\mathrm{dist}_U(v, w) \leq 3$. Choose an edge vw of G. If vw is an edge in a bag B, then $\mathrm{dist}_U(v, w) = \mathrm{dist}_{S_B}(v, w) \leq 3$. Assume that vw is an edge between a bag B and its parent $P(B)$ so that $v \in V(B)$ and $w \in V(P(B))$. If $vw \in E(S_B)$, then it is trivial. Assume that $w \notin V(S_B)$. Let z be the vertex of S_B contained in P_B. Then $\mathrm{dist}_U(v, w) = \mathrm{dist}_{S_B}(v, z) + \mathrm{dist}_{S_{P(B)}}(z, w) \leq 3$.

Our construction of a tree 3-spanner for G immediately follows the partition tree $\mathcal{T}$ of G. By Theorem 5, a partition tree of a well-partitioned chordal graph can be obtained in polynomial time, and therefore one can find a tree 3-spanner for G in polynomial time.

5 Geodetic Sets

We now give a polynomial-time algorithm for the Geodetic Set problem on well-partitioned chordal graphs. Recall that a geodetic set of a graph G is a subset S of its vertices such that each vertex that is not in S lies on a shortest path between some pair of vertices in S, and that the Geodetic Set problem asks, given a graph G, for a smallest-size geodetic set of G. Throughout the following, given a vertex set $S \subseteq V(G)$, we denote by $I[S]$ the *interval* of S in G, which is the set of all vertices lying on a shortest path between a pair of vertices in S. Note that $S \subseteq I[S]$.

We first observe that any geodetic set of a graph contains all its simplicial vertices. Since the neighborhood of a simplicial vertex v is a clique, v is never an internal vertex of any shortest path: Suppose v is an internal vertex of a path P, and let u_1 and u_2 be the two neighbors of v in P. Since u_1 and u_2 are adjacent, we can obtain a shorter path P' from P by replacing u_1vu_2 with u_1u_2 such that P' has the same endpoints as P.

Observation 1. Let G be a graph and let $v \in V(G)$ be a simplicial vertex in G. Then, every geodetic set of G contains v.

From now on we assume that we are given a connected well-partitioned chordal graph G with partition tree $\mathcal{T}$, such that $\mathcal{T}$ has at least two nodes (otherwise, G is simply a complete graph). If G is not connected, we can apply the procedure described below to each of its connected components. As a consequence of Observation 1, we have that each leaf bag of $\mathcal{T}$ has a vertex that is contained in every geodetic set of G. Let $B \in V(\mathcal{T})$ be a leaf with neighbor C. If $\mathrm{bd}(B, C) \neq B$, then each vertex in $B \setminus \mathrm{bd}(B, C)$ is simplicial. If $\mathrm{bd}(B, C) = B$, then each vertex in B is simplicial. This also immediately implies that each non-simplicial vertex in a leaf bag is on some shortest path between two simplicial vertices: if we have a non-simplicial vertex in B, then $\mathrm{bd}(B, C) \neq B$ and the non-simplicial vertices are precisely the ones in $\mathrm{bd}(B, C)$. Since $\mathcal{T}$ has at least two nodes, there is some other leaf bag in $\mathcal{T}$ which again has some simplicial vertex, say x. Now, each shortest path from a simplicial vertex in B to x uses some vertex from $\mathrm{bd}(B, C)$, and since the vertices in $\mathrm{bd}(B, C)$ are twins in $G[B \cup C]$, each of them is on such a shortest path.

Observation 2. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, and let S be the set of simplicial vertices of G. Each leaf bag B of $\mathcal{T}$ contains a simplicial vertex, and $B \subseteq I[S]$.

Dourado et al. [16] showed that the geodetic number of split graphs can be computed in polynomial time. In the following, we adapt their construction to the case of internal bags of a partition tree in a well-partitioned chordal graph. First, we prove a small auxiliary lemma; for an illustration of its arguments see Figure 1.

Lemma 3. *Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, let S denote the set of simplicial vertices of G, and let $B \in V(\mathcal{T})$ be an internal bag.*

(i) Let $u \in B$. If there are two distinct $C_1, C_2 \in N_{\mathcal{T}}(B)$ such that $u \in \mathrm{bd}(B, C_1) \cap \mathrm{bd}(B, C_2)$, then $u \in I[S]$.

(ii) For all $C_1, C_2 \in N_{\mathcal{T}}(B)$ with $\mathrm{bd}(B, C_1) \cap \mathrm{bd}(B, C_2) = \emptyset$, we have that $\mathrm{bd}(B, C_1) \cup \mathrm{bd}(B, C_2) \subseteq I[S]$.

Proof. (i). There are leaves D_1, D_2 in $\mathcal{T}$ such that C_1BC_2 is on the path from D_1 to D_2 in $\mathcal{T}$. By Observation 2, for all $i \in [2]$, D_i contains a simplicial vertex, say x_i. Each shortest path from x_1 to x_2 is of the form $x_1 \dots y_1 z y_2 \dots x_2$, where $y_1 \in C_1$, $y_2 \in C_2$, and $z \in \mathrm{bd}(B, C_1) \cap \mathrm{bd}(B, C_2)$. Since the vertices in $\mathrm{bd}(B, C_1) \cap \mathrm{bd}(B, C_2)$ are twins in $G[B \cup C_1 \cup C_2]$, we may assume that $z = u$, therefore $u \in I[x_1, x_2] \subseteq I[S]$.

(ii). The proof can be done very similarly to the one of the previous item, with the difference that each shortest path between the (corresponding) vertices x_1 and x_2 uses both a vertex from $\mathrm{bd}(B, C_1)$ and one from $\mathrm{bd}(B, C_2)$.

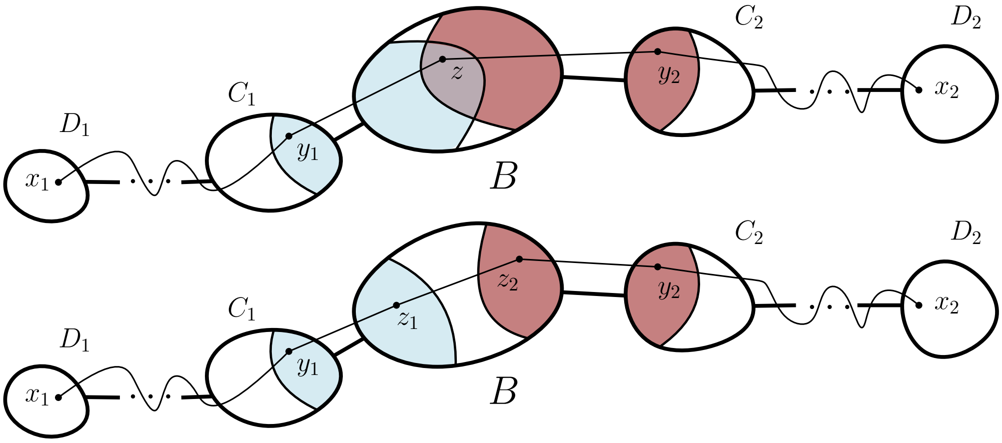


Figure 1: Illustration of the proof of Lemma 3. The top drawing shows item (i) and the bottom one item (ii).

Lemma 4. *Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, let S denote the set of simplicial vertices of G, and let $B \in V(\mathcal{T})$ be an internal bag. If B contains a simplicial vertex, then $B \subseteq I[S]$.*

Proof. Let $X \subseteq B$ be the set of vertices in B that are not contained in any boundary. Note that $X \subseteq S$. Then, we obtain $\mathcal{T}'$ from $\mathcal{T}$ by removing B, adding a bag $B' := B \setminus X$ and a bag X. We make all bags in $N_{\mathcal{T}}(B) \cup \{X\}$ adjacent to B' in $\mathcal{T}'$. Since B is a clique, X is a clique and complete to B', satisfying the requirements of the definition of a partition tree. Since no vertex in X was in any boundary, the boundaries from the other neighbors of B' in $\mathcal{T}'$ remain the same as the ones in $\mathcal{T}$ to B. We can conclude that $\mathcal{T}'$ is a partition tree of G. Moreover, each vertex $v \in B'$ is in $\mathrm{bd}(B', X)$ and at least one more boundary, since $v \notin X$. By Lemma 3(i), $B' \subseteq I[S]$, so $B' \cup X = B \subseteq I[S]$.

We may assume that each simplicial vertex v of B is contained in a boundary. Clearly, a simplicial vertex can be contained in at most one boundary; let $C \in N_{\mathcal{T}}(B)$ be such that $v \in \mathrm{bd}(B, C)$. Since v is simplicial, we have that $\mathrm{bd}(B, C) = B$. Therefore, for each vertex $u \in B$ such that there is some neighbor $C' \neq C$ of B with $u \in \mathrm{bd}(B, C')$, so we have by Lemma 3(i) that $u \in I[S]$. On the other hand, each vertex in $B \setminus \bigcup_{C' \in N_{\mathcal{T}}(B), C' \neq C} \mathrm{bd}(B, C')$ is simplicial as well, so we can conclude that $B \subseteq I[S]$.

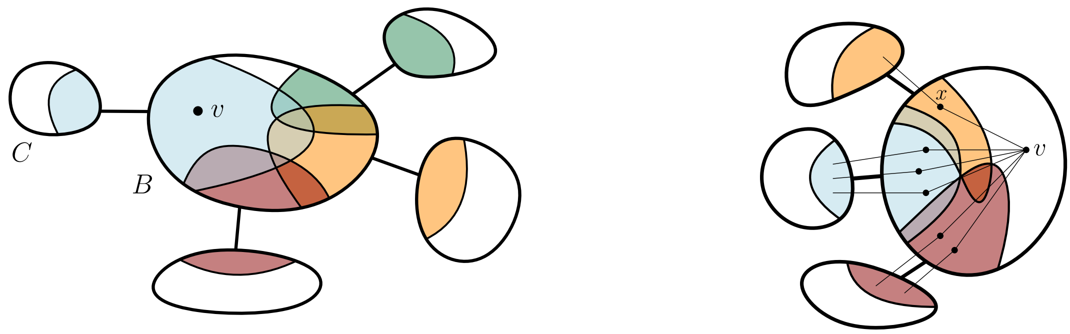
In the remainder, we show how to deal with vertices that are not on shortest paths between simplicial vertices. We call such vertices *problematic*, and they are the ones that are contained in internal bags without simplicial vertices and do not fall under one of the cases of Lemma 3. For an illustration of a problematic vertex, see Figure 2a.

Definition 1. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, and let $B \in V(\mathcal{T})$ be an internal bag that does not contain any simplicial vertex. A vertex $v \in B$ is called *problematic* if

(i) there is a unique $C \in N_{\mathcal{T}}(B)$ such that $v \in \mathrm{bd}(B, C)$, and

(ii) for each $C' \in N_{\mathcal{T}}(B) \setminus \{C\}$, $\mathrm{bd}(B, C) \cap \mathrm{bd}(B, C') \neq \emptyset$.

In this case we call C a *problematic neighbor bag.*



(a) Illustration of a problematic vertex v. The only boundary v is contained in is $\mathrm{bd}(B, C)$, and every other boundary in B intersects $\mathrm{bd}(B, C)$.

(b) Illustration of a problem solver v. Note that v may be in $I[S]$, and that x is a problem solver as well.

Figure 2: Problematic vertices and problem solvers.

Suppose that some bag B has no simplicial vertex. Then each shortest path in G between two simplicial vertices that uses a vertex from B passes through two neighbors of B. If a vertex is problematic, then it cannot be on any such shortest path, and if it is not problematic, then it falls under one of the cases of Lemma 3, which leads to the following observation.

Observation 3. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, let S denote the set of simplicial vertices of G, and let $B \in V(\mathcal{T})$ be an internal bag with $B \cap S = \emptyset$. Let P be the set of problematic vertices of B, then $P = B \setminus I[S]$.

By similar reasoning, we observe that if a problematic vertex in B is on some shortest path, then this shortest path has to have an endpoint in B.

Observation 4. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, and let $B \in V(\mathcal{T})$ be an internal bag. Let $v \in B$ be a problematic vertex. Any shortest path that has v as an internal vertex has one endpoint in B.

By Observations 3 and 4, we know that if a bag B has no simplicial vertex and it has at least one problematic vertex, then we need at least one more vertex from B in any geodetic set. The following notion captures in which situation a single additional vertex suffices. We illustrate the following definition in Figure 2b.

Definition 2. Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$ and let $B \in V(\mathcal{T})$. Let $P \subseteq B$ denote the set of problematic vertices in B and $C_1, \dots, C_\ell$ the problematic neighbor bags. A vertex $v \in B$ is called a *problem solver* if for each $i \in [\ell]$, either $v \notin \mathrm{bd}(B, C_i)$ or $\mathrm{bd}(B, C_i) \cap P = \{v\}$.

Lemma 5. *Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$ and let $S \subseteq V(G)$ be the simplicial vertices of G. Let $B \in V(\mathcal{T})$ be a bag with $B \cap S = \emptyset$. For each $v \in B$, $B \subseteq I[S \cup \{v\}]$ if and only if v is a problem solver.*

Proof. Throughout the proof, we denote by P the set of problematic vertices of B and by $C_1, \dots, C_\ell$ the problematic neighbor bags. Suppose that for all $i \in [\ell]$, either $v \notin \mathrm{bd}(B, C_i)$ or $\{v\} = \mathrm{bd}(B, C_i) \cap P$. By Observation 3, each vertex in $B \setminus I[S]$ is problematic, so we have to argue that each problematic vertex is on a shortest path from v to a simplicial vertex. Let $u \in P$ with problematic neighbor bag C. If $\mathrm{bd}(B, C) \cap P = \{u\}$ and $u = v$, then clearly $u \in I[S \cup \{v\}]$.

Otherwise we have that $v \notin \mathrm{bd}(B, C)$, so each shortest path from v that goes through C has a vertex from $\mathrm{bd}(B, C)$. Moreover, there is a leaf $D \in V(\mathcal{T})$ such that C is on the path from D to B in $\mathcal{T}$; by Observation 2, D has a simplicial vertex so the first direction follows.

For the other direction, suppose for a contradiction that there is some $v \in B$ such that $B \subseteq I[S \cup \{v\}]$, while for some $i \in [\ell]$, $v \in \mathrm{bd}(B, C_i)$ and there is some $u \in (\mathrm{bd}(B, C_i) \cap P) \setminus \{v\}$. Since $u \in I[S \cup \{v\}]$, u is on the shortest path between v and some simplicial vertex, denote that path by Q. (Note that since u is problematic, it is not on the shortest path between two simplicial vertices.) Moreover, by Observation 4, one of the endpoints of the path has to be in B. Since B has no simplicial vertex, we know that one of the endpoints of the path is v.

If Q uses a vertex from C_i, in particular from $\mathrm{bd}(C_i, B)$, then we have a contradiction: We can remove u from Q and go from the vertex in $\mathrm{bd}(C_i, B)$ directly to v and obtain a shorter path. If Q does not use a vertex from C_i, then it must use a vertex from some other neighbor of B, say $D \in N_{\mathcal{T}}(B) \setminus \{C_i\}$. This is because the other endpoint of Q but v is not contained in B. Now, since u is problematic, we have that $u \notin \mathrm{bd}(B, D)$. However, Q contains a vertex in $\mathrm{bd}(B, D)$, so we can remove u from Q and obtain a shorter path with the same endpoints, again a contradiction.

Next we show that if there are at least two distinct problematic neighbor bags, then two additional vertices always suffice.

Lemma 6. *Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, let S denote the set of simplicial vertices of G, let $B \in V(\mathcal{T})$ be an internal bag with $B \cap S = \emptyset$. If there are two distinct problematic neighbor bags of B, then there are two vertices $v_1, v_2 \in B$ such that $B \subseteq I[S \cup \{v_1, v_2\}]$.*

Proof. Let $C_1, C_2 \in N_{\mathcal{T}}(B)$ be two distinct problematic neighbor bags of B, and for all $i \in [2]$, let v_i be a problematic vertex in $\mathrm{bd}(B, C_i)$. Again by Observation 3, we have to argue that each problematic vertex in B is on a shortest path from v_1 or v_2 to a simplicial vertex. Let u be a problematic vertex and let $C \in N_{\mathcal{T}}(B)$ be the corresponding problematic neighbor bag. Suppose $C = C_1$. There is a leaf D (containing a simplicial vertex by Observation 2) such that C is on the path from D to B in $\mathcal{T}$, and each shortest path from a vertex in D to a vertex in $B \setminus \mathrm{bd}(B, C)$ uses a vertex from $\mathrm{bd}(B, C)$. Since $v_2 \notin \mathrm{bd}(B, C)$ by the definition of a problematic vertex, it follows that $u \in I[S \cup \{v_2\}]$. Similarly, if $C = C_2$, then $u \in I[S \cup \{v_1\}]$. Finally, if $C \notin \{C_1, C_2\}$, then we have that $u \in I[S \cup \{v_1\}] \cap I[S \cup \{v_2\}]$. We can conclude that $B \subseteq I[S \cup \{v_1, v_2\}]$.

Finally we show that in the remaining case when there is only one problematic neighbor bag and no problem solver, then any geodetic set of G has to include all problematic vertices.

Lemma 7. *Let G be a connected well-partitioned chordal graph with partition tree $\mathcal{T}$, let S denote the set of simplicial vertices of G, let $B \in V(\mathcal{T})$ be an internal bag with $B \cap S = \emptyset$. Let $P \subseteq B$ be the set of problematic vertices of B, and suppose there is a neighbor $C \in N_{\mathcal{T}}(B)$ such that $P \subseteq \mathrm{bd}(B, C)$. If there is no problem solver, then every geodetic set of G contains P.*

Proof. Note that the condition that there is no problem solver is equivalent to the condition that $\mathrm{bd}(B, C) = B$; any vertex outside of $\mathrm{bd}(B, C)$ would be a problem solver. Suppose that some $v \in P$ is on a shortest path between two vertices x_1 and x_2. Since v is a problematic vertex, we may assume by Observation 4 that $x_1 \in B$ and $x_2 \notin B$. Let $D \in V(\mathcal{T})$ denote the bag containing x_2. Let $C^* \in N_{\mathcal{T}}(B)$ denote the neighbor of B that is on the path from D to B in $\mathcal{T}$. If $C^* \neq C$, then v cannot be in a shortest path from x_1 to x_2: since v is problematic, $v \notin \mathrm{bd}(B, C^*)$. We may assume that $C^* = C$. Since $x_1 \in B = \mathrm{bd}(B, C)$, v cannot be an internal vertex on the shortest path from x_1 to x_2. We can conclude that every geodetic set of G must contain v.

Input : A connected well-partitioned chordal graph G with partition tree $\mathcal{T}$.
Output: A minimum-size geodetic set of G.

1 Find the set S of simplicial vertices of G;
2 **foreach** *internal bag* $B \in V(\mathcal{T})$ **do**
3   **if** B *contains a simplicial vertex* **then** do nothing;
4   **else if** *there is a problem solver* $v \in B$ **then** $S \leftarrow S \cup \{v\}$;
5   **else if** B *has two distinct problematic neighbor bags* C_1 *and* C_2 **then**
6   Let $v_1 \in \mathrm{bd}(B, C_1)$ and $v_2 \in \mathrm{bd}(B, C_2)$ be problematic;
7         $S \leftarrow S \cup \{v_1, v_2\}$;
8   **else** $S \leftarrow S \cup P$, where P is the set of problematic vertices in B;
9 **return** S;

Algorithm 1: A polynomial-time algorithm for finding a minimum-size geodetic set of a well-partitioned chordal graph.

Now that we have covered all the cases, we can derive the algorithm to compute a minimum geodetic set of a well-partitioned chordal graph by properly prioritizing the cases. We describe the procedure in Algorithm 1.

We now argue the correctness of the algorithm. In line 1, it adds all simplicial vertices to the set it produces. This is safe by Observation 1. Moreover, by Observation 2, any vertex contained in any leaf of the partition tree is contained in the interval of the simplicial vertices. Let B be any internal bag in the partition tree. In line 3, the algorithm asserts that if B contains a simplicial vertex, then no additional vertex of B has to be added. Correctness of this decision is argued in Lemma 4. Suppose B has no simplicial vertex. By Observation 3, each vertex in B that is not in the interval of the simplicial vertices is problematic, and by Observation 4, a shortest path that has a problematic vertex as an internal vertex has one endpoint in B. Therefore, any geodetic set of G has to contain at least one vertex from B. Lemma 5 characterizes the situation in which one additional vertex (a problem solver) suffices, which is checked for next by the algorithm, in line 4. If no such vertex exists, then each geodetic set uses at least two vertices from B. If there are at least two distinct problematic neighbor bags, then two additional vertices suffice as shown in Lemma 6. The algorithm checks this next in line 5. Otherwise, there is precisely one problematic neighbor bag C, there is no problem solver, and $\mathrm{bd}(B, C)$ contains at least two problematic vertices. By Lemma 7, all these vertices are in any geodetic set of G, so the algorithm is correct in line 8.

It is easy to verify that each line in Algorithm 1 takes polynomial time, and that the main loop has a polynomial number of iterations. Since well-partitioned chordal graphs can be recognized in polynomial time by an algorithm that produces a partition tree if one exists, see Theorem 5, this proves Theorem 4.

Restatement of Theorem 4. *There is a polynomial-time algorithm that given a well-partitioned chordal graph G, computes a minimum-size geodetic set of G.*

References

[1] Jungho Ahn, Lars Jaffke, O-joung Kwon, and Paloma T. Lima. Well-partitioned chordal graphs: Obstruction set and disjoint paths. In Isolde Adler and Haiko Müller, editors, *Proceedings of the 46th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2020)*, volume 12301 of *LNCS*, pages 148–160, Cham, 2020. Springer.

[2] Ingo Althöfer, Gautam Das, David Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993.

[3] Mustafa Atici. Computational complexity of geodetic set. *International Journal of Computer Mathematics*, 79(5):587–591, 2002.

[4] Paul N. Balister, Ervin Györi, Jenö Lehel, and Richard H. Schelp. Longest paths in circular arc graphs. *Combinatorics, Probability and Computing*, 13(3):311–317, 2004.

[5] Andreas Brandstädt, Feodor F. Dragan, Hoàng-Oanh Le, and Van Bang Le. Tree spanners on chordal graphs: complexity and algorithms. *Theoretical Computer Science*, 310(1-3):329–354, 2004.

[6] Andreas Brandstädt, Feodor F. Dragan, Hoàng-Oanh Le, Van Bang Le, and Ryuhei Uehara. Tree spanners for bipartite graphs and probe interval graphs. *Algorithmica*, 47(1):27–51, 2007.

[7] Letícia Rodrigues Bueno, Lucia Draque Penso, Fábio Protti, Victor R. Ramos, Dieter Rautenbach, and Uéverton S. Souza. On the hardness of finding the geodetic number of a subcubic graph. *Information Processing Letters*, 135:22–27, 2018.

[8] Leizhen Cai. *Tree spanners: spanning trees that approximate distances.* PhD thesis, University of Toronto, 1992.

[9] Leizhen Cai and Derek G. Corneil. Tree spanners. *SIAM Journal on Discrete Mathematics*, 8(3):359–387, 1995.

[10] Márcia R. Cerioli and Paloma T. Lima. Intersection of longest paths in graph classes. *Discrete Applied Mathematics*, 281:96–105, 2020.

[11] Dibyayan Chakraborty, Sandip Das, Florent Foucaud, Harmender Gahlawat, Dimitri Lajou, and Bodhayan Roy. Algorithms and complexity for geodetic sets on planar and chordal graphs. arXiv:2006.16511, 2020. To appear at ISAAC 2020.

[12] Dibyayan Chakraborty, Florent Foucaud, Harmender Gahlawat, Subir Kumar Ghosh, and Bodhayan Roy. Hardness and approximation for the geodetic set problem in some graph classes. In Manoj Changat and Sandip Das, editors, *Proceedings of the 6th International Conference on Algorithms and Discrete Applied Mathematics (CALDAM 2020)*, volume 12016 of *LNCS*, pages 102–115. Springer, 2020.

[13] Guantao Chen, Julia Ehrenmüller, Cristina G. Fernandes, Carl Georg Heise, Songling Shan, Ping Yang, and Amy N. Yates. Nonempty intersection of longest paths in series-parallel graphs. *Discrete Mathematics*, 340(3):287–304, 2017.

[14] Susanna F. de Rezende, Cristina G. Fernandes, Daniel M. Martin, and Yoshiko Wakabayashi. Intersecting longest paths. *Discrete Mathematics*, 313(11):1401–1408, 2013.

[15] Mitre C Dourado, Fábio Protti, and Jayme L Szwarcfiter. On the complexity of the geodetic and convexity numbers of a graph. *Lecture Notes of the Ramanujan Mathematical Society*, 7:497–500, 2006.

[16] Mitre Costa Dourado, Fábio Protti, Dieter Rautenbach, and Jayme Luiz Szwarcfiter. Some remarks on the geodetic number of a graph. *Discrete Mathematics*, 310(4):832–837, 2010.

[17] Tínaz Ekim, Aysel Erey, Pinar Heggernes, Pim van 't Hof, and Daniel Meister. Computing minimum geodetic sets of proper interval graphs. In David Fernández-Baca, editor, *Proceedings of the 10th Latin American Symposium on Theoretical Informatics (LATIN 2012)*, volume 7256 of *LNCS*, pages 279–290. Springer, 2012.

[18] Sándor P. Fekete and Jana Kremer. Tree spanners in planar graphs. *Discrete Applied Mathematics*, 108(1-2):85–103, 2001.

[19] Tibor Gallai. Problem 4. In Paul Erdős and Gyula O. H. Katona, editors, *Proceedings of the Colloquium on Theory of Graphs Held in Tihany, Hungary, 1966*, page 362, 1968.

[20] Gili Golan and Songling Shan. Nonempty intersection of longest paths in $2K_2$-free graphs. *The Electronic Journal of Combinatorics*, 25(2):P2.37, 2018.

[21] Frank Harary, Emmanuel Loukakis, and Constantine Tsouros. The geodetic number of a graph. *Mathematical and Computer Modelling*, 17(11):89–95, 1993.

[22] Adam S. Jobson, André E. Kézdy, Jenö Lehel, and Susan C. White. Detour trees. *Discrete Applied Mathematics*, 206:73–80, 2016.

[23] Felix Joos. A note on longest paths in circular arc graphs. *Discussiones Mathematicae Graph Theory*, 35(3):419–426, 2015.

[24] Leon Kellerhals and Tomohiro Koana. Parameterized complexity of geodetic set. arXiv:2001.03098, 2020. To appear at IPEC 2020.

[25] Sandi Klavžar and Marko Petkovšek. Graphs with nonempty intersection of longest paths. *Ars Combinatoria*, 29:43–52, 1990.

[26] Michael C. Loui and David R. Luginbuhl. Optimal on-line simulations of tree machines by random access machines. *SIAM Journal on Computing*, 21(5):959–971, 1992.

[27] M. S. Madanlal, G. Venkatesan, and C. Pandu Rangan. Tree 3-spanners on interval, permutation and regular bipartite graphs. *Information Processing Letters*, 59(2):97–102, 1996.

[28] Mauro Mezzini. Polynomial time algorithm for computing a minimum geodetic set in outerplanar graphs. *Theoretical Computer Science*, 745:63–74, 2018.

[29] B. S. Panda and Anita Das. Tree 3-spanners in 2-sep chordal graphs: characterization and algorithms. *Discrete Applied Mathematics*, 158(17):1913–1935, 2010.

[30] Ignacio M. Pelayo. *Geodesic Convexity in Graphs*. Springer, 2013.

[31] Dieter Rautenbach and Jean-Sébastien Sereni. Transversals of longest paths and cycles. *SIAM Journal on Discrete Mathematics*, 28(1):335–341, 2014.

[32] G. Venkatesan, Udi Rotics, M. S. Madanlal, Johann A. Makowsky, and C. Pandu Rangan. Restrictions of minimum spanner problems. *Information and Computation*, 136(2):143–164, 1997.

[33] Hansjoachim Walther and Heinz Jürgen Voss. *Über Kreise in Graphen*. Deutscher Verlag der Wissenschaften, 1974.

[34] Tudor Zamfirescu. On longest paths and circuits in graphs. *Mathematica Scandinavica*, 38(2):211–239, 1976.