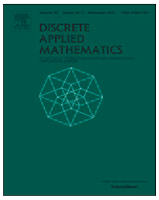
Discrete Applied Mathematics 389 (2026) 92–105

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam



Characterizing optimal monitoring edge-geodetic sets for some structured graph classes

Florent Foucaud [a], Arti Pandey [b], Kaustav Paul [b,*]

[a] *Université Clermont Auvergne, CNRS, Clermont Auvergne INP, Mines Saint-Étienne, LIMOS, 63000, Clermont-Ferrand, France*
[b] *Department of Mathematics, Indian Institute of Technology Ropar, Rupnagar, 140001, Punjab, India*

ARTICLE INFO

Article history:
Received 1 February 2025
Received in revised form 22 November 2025
Accepted 25 March 2026

Keywords:
Monitoring edge-geodetic set
Bipartite permutation graph
Distance-hereditary graph
P_4-sparse graph
Strongly chordal graph

ABSTRACT

Given a graph $G = (V, E)$, a set $S \subseteq V$ is said to be a monitoring edge-geodetic set if the deletion of any edge in the graph results in a change in the distance between at least one pair of vertices in S. The minimum size of such a set in G is called the monitoring edge-geodetic number of G.

In this work, we provide structural characterizations of the minimum monitoring edge-geodetic sets for several structured graph classes: distance-hereditary graphs, P_4-sparse graphs, bipartite permutation graphs, and strongly chordal graphs. These characterizations are in terms of the *mandatory vertices* of the graph (those that need to be in every solution). This extends previous results from the literature for cographs, interval graphs and block graphs, and leads to efficient algorithms to compute optimal monitoring edge-geodetic sets for graphs in these graph classes.

1. Introduction

The realm of network monitoring entails a range of applications focused on detecting and addressing periodic failures within a network. Within this framework, a network is considered a finite simple graph, where failures may manifest as connection losses between two nodes of the network. Such failures often lead to temporary increases in distances or the isolation of specific nodes from the broader network. Ideally, consistent monitoring of the entire graph would promptly reveal any instances of malfunction. Nevertheless, developing a methodology that achieves this objective with minimized cost is more advantageous.

In a recent study, Foucaud et al. [7] introduced a new graph parameter based on the distances between a specific group of vertices, motivated by applications in network monitoring. This concept arises from merging the classic concept of *geodetic sets* [16] and the more recent one of *distance edge-monitoring sets* [9]. An edge e of a graph $G = (V, E)$ is said to be *monitored* by a pair of vertices $a, b \in V$, if every shortest a-b path contains e. An edge e of a graph $G = (V, E)$ is said to be *monitored* by a set $S \subseteq V$, if e is monitored by at least one pair of vertices in S. A monitoring edge-geodetic set of G, abbreviated as MEG set of G, refers to a subset of the vertex set V, that monitors all the edges of the graph G. The *monitoring edge-geodetic number* (or MEG number) of G, denoted as $meg(G)$, represents the size of a minimum MEG set of a graph G. The corresponding algorithmic problem is defined as follows:

MINIMUM MONITORING EDGE-GEODETIC SET *problem (MIN-MEG)*

Instance: A graph $G = (V, E)$.

---

* Corresponding author.
E-mail addresses: florent.foucaud@uca.fr (F. Foucaud), arti@iitrpr.ac.in (A. Pandey), kaustav.20maz0010@iitrpr.ac.in (K. Paul).

https://doi.org/10.1016/j.dam.2026.03.048

Solution: An MEG set M of G with $|M| = meg(G)$.

Our goal is to continue the study of this computational problem on structured graph classes. To do so, we will use the following concept: Given a graph $G = (V, E)$, we define the notion of *mandatory vertex* as follows: a vertex $v \in V$ is *mandatory* in G if it is contained in every MEG set of G. We also define the set $Man(G) = \{v \in V \mid v \text{ is mandatory in } G\}$. In this manuscript, given a graph $G = (V, E)$, n and m are used to denote the cardinalities of V and E respectively.

1.1. Related work

The monitoring edge-geodetic set problem was introduced by Foucaud et al. [13]. In their work, they computed the MEG number for some fundamental graph classes such as trees, cycles, unicyclic graphs, complete graphs, grids, and hypercubes. Additionally, using the feedback edge set number of a graph G, they derived an upper bound for $meg(G)$, which was subsequently refined in [6]. In the same paper, the authors presented several examples of graphs that satisfy $meg(G) = n$, and asked whether it is possible to characterize the class of graphs for which $meg(G) = n$. In [10,11], Foucaud et al. characterized the graphs of order n for which $meg(G) = n$, answering the question asked in [7], by characterizing mandatory vertices. Indeed, when $meg(G) = n$, all vertices must be mandatory. These graphs were termed as MEG-extremal graphs in [11]. Some examples of MEG-extremal graphs are complete graphs, hypercubes, corona products of two graphs [7] and 2-connected cographs, 2-connected proper interval graphs, 2-connected well-partitioned chordal graphs [11]. Haslegrave [17] also produced some bounds on the MEG number for certain product operations of two graphs, G and H, in terms of $meg(G)$ and $meg(H)$. See also [33] for further work on MEG on graph products, [24,25,31,34] for studies on the MEG number on various graph families, and [10] for an upper bound on the MEG number in terms of girth for sparse graphs.

Coming to the complexity of the problem, the first NP-hardness proof of the MIN-MEG problem for general graphs was proposed by Haslegrave [17]. The authors in [10] showed that the MIN-MEG problem remains NP-hard for 3-degenerate 2-apex graphs. Recently, Bilò et al. [3], have shown that there is no polynomial-time $(c \log n)$ (where $c < \frac{1}{2}$) factor approximation algorithm for the MIN-MEG problem, unless P = NP. Foucaud et al. [12] showed that the problem cannot be solved in subexponential time for 3-degenerate graphs unless the Exponential Time Hypothesis (ETH) fails, and that it is APX-hard for 4-degenerate graphs. Complementing these hardness results, they also prove that the problem admits a polynomial-time algorithm for interval graphs, a Fixed Parameter Tractable (FPT) algorithm for general graphs with clique-width plus diameter as the parameter, and an FPT algorithm for chordal graphs with treewidth as the parameter. They also provide an approximation algorithm with factor $\ln m \cdot meg(G)$ and $\sqrt{n \ln m}$ for the MIN-MEG problem.

1.2. Our results

Given a graph $G = (V, E)$, by definition, $|Man(G)|$ is a lower bound for $meg(G)$. Next we discuss an upper bound for $meg(G)$. As observed in [7] (see Theorem 2.3 in Section 2.2), the set of cut-vertices is not part of any minimum MEG set of G. We define $Cut(G) = \{v \in V \mid v \text{ is a cut-vertex of } G\}$. Then, for any minimum-size MEG set M, we have $M \subseteq V \setminus Cut(G)$. Hence, the following observation holds:

Observation 1.1. *Given a graph $G = (V, E)$, let M be a minimum MEG set of G. Then, $Man(G) \subseteq M \subseteq V \setminus Cut(G)$.*

By Observation 1.1 and existing literature, we can conclude that there are several graph classes (like block graphs, well-partitioned chordal graphs, interval graphs and cographs) which belong to the following two types (and there are many graph classes which are not of any of these two types, for example the class of cycles of order at least 5):

1. the graphs for which $Man(G)$ is a minimum-size MEG set;
2. even stronger, the graphs for which $Man(G) = V \setminus Cut(G)$, implying that $Man(G)$ is a minimum-size MEG set.

It is known that interval graphs are of type 1 [12], and block graphs, proper interval graphs, cographs, and well-partitioned chordal graphs are of type 2 [10,11].

Note that for a given graph $G = (V, E)$, if G is a graph of type 1, then $meg(G)$ can be computed in polynomial time, since the necessary and sufficient condition for a vertex to be a mandatory vertex, can be checked in polynomial time [11]. If G is of type 2, then we have an even simpler (linear-time) algorithm to compute $meg(G)$. Hence, if $meg(G)$ attains either of the upper and lower bound mentioned in Observation 1.1, it can be computed in polynomial time.

In this paper, we extend and strengthen some of these results by considering more general graph classes, as follows.

- In Section 3, we show that for any distance-hereditary graph G, $Man(G) = V \setminus Cut(G)$, which implies that $meg(G) = |Man(G)| = |V \setminus Cut(G)|$. This extends the results for cographs and block graphs in [10,11], as cographs and block graphs are distance-hereditary.
- In Section 4, we show that for any bipartite permutation graph G, $Man(G) = V \setminus Cut(G)$, which implies that $meg(G) = |Man(G)| = |V \setminus Cut(G)|$.
- In Section 5, we show that for any P_4-sparse graph G, $meg(G) = |Man(G)|$.

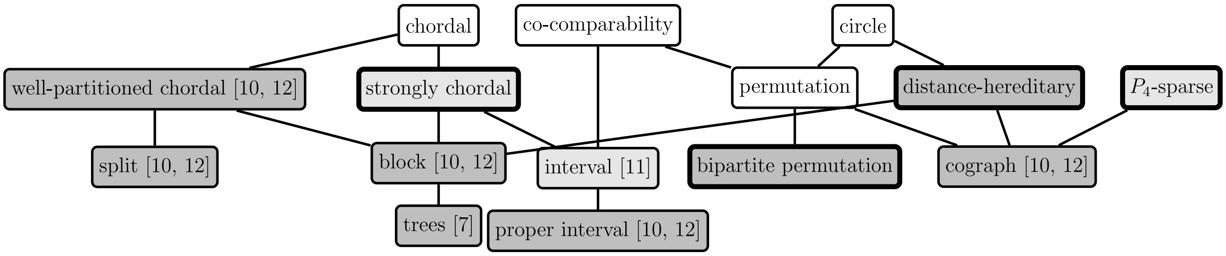


Fig. 1. Inclusion diagram for graph classes mentioned in this paper (and related ones). If a class A has an upward path to class B, then A is included in B. For any graph G in a dark gray class, we have $|Man(G)| = |V(G) \setminus Cut(G)| = meg(G)$. For any graph G in a light gray class, we have $|Man(G)| = meg(G)$. Results for boxes with a thick border are proved in this paper. For the graph classes which are not defined in this paper, their definition can be found in [19].

- In Section 6, we show that for any strongly chordal graph G, $meg(G) = |Man(G)|$. This extends the polynomial-time algorithm to solve MIN-MEG for interval graphs from [12], as interval graphs are also strongly chordal.
- Section 7 concludes our work.

Hence, for all the above-mentioned graph classes, the monitoring edge-geodetic number can be computed in polynomial time. Fig. 1 shows a hierarchy diagram of graph classes to signify the improvements achieved by our results over the existing literature. We remark that studying different graph optimization problems on these graph classes has been an active research direction for several decades (see [4,5,22,23,26–29,32] for some recent algorithmic results on these graph classes).

2. Preliminaries

2.1. Notations and definitions

This paper only considers simple, undirected, finite, and connected graphs with at least two vertices. Given a graph $G = (V, E)$, we use the notations V and $V(G)$ (resp. E and $E(G)$) alternatively. The *order* of the graph G is $|V|$. The notation $N_G(v)$ stands for the set of neighbors of a vertex v in G and $N_G[v] = N_G(v) \cup \{v\}$. When the context is clear, we use $N(v)$ (resp. $N[v]$) instead of $N_G(v)$ (resp. $N_G[v]$). The number of neighbors of a vertex $v \in V$ defines its *degree*, which is represented by the symbol $deg(v)$. The maximum degree of the graph G is denoted by $\Delta(G)$; when the context is clear, we use only Δ. Given a subset of vertices U of V, we use $deg_U(v)$ to denote the number of neighbors that a vertex v has within the set U. Additionally, we use $N_U(v)$ to refer to the set of neighbors of vertex v within U. A vertex $v \in V$ is said to be an *pendant vertex* (resp. *isolated vertex*) if $|N(v)| = 1$ (resp. $|N(v)| = 0$). Sometimes, a pendant vertex is also referred to as a *leaf*. The unique neighbor of a pendant vertex is called a *stem*. An edge $uv \in E$ is said to be an *isolated edge* if $deg(u) = deg(v) = 1$. A vertex v is said to be a *simplicial* vertex in G if $N[v]$ induces a clique of G. If a vertex v and the edges incident to v are deleted from G, then the resulting graph is denoted by $G \setminus \{v\}$. A vertex $v \in V$ is said to be a *cut-vertex* of G, if the graph $G \setminus \{v\}$ is not connected. Given a subset of vertices $S \subseteq V$, the subgraph of G induced by S is denoted by $G[S]$ and denoted with the notation $G[S] = (S, E_S)$, where $E_S = \{uv \in E \mid u \in S, v \in S\}$.

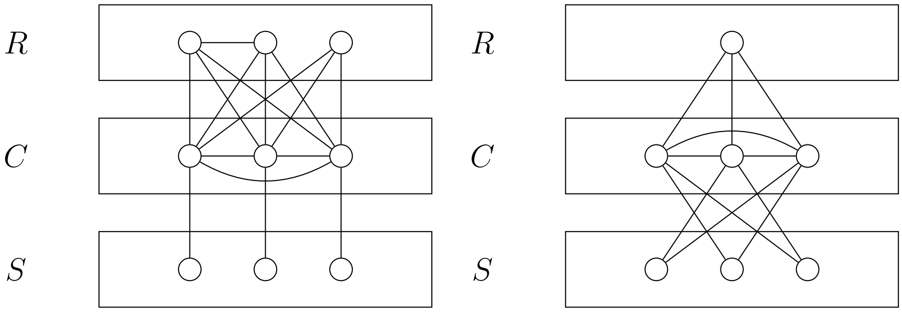
A *path* $P = v_0 v_1 \dots v_k$, is a sequence of distinct vertices, such that $v_{i-1}v_i \in E$, where $1 \leq i \leq k$ and $k \geq 1$. Such a path is called a path between v_0 and v_k. We denote $V(P) = \{v_0, v_1, \dots, v_k\}$. The length of the path P is $|V(P)| - 1$. A k-path (or P_k) is a path of length $k - 1$, for any integer $k > 1$. The *distance* between any two vertices a and b in G is the length of a shortest path between them and is denoted by $d_G(a, b)$ (simply $d(a, b)$ when the context is clear). Two vertices a and b are said to be at even (resp. odd) distance away from each other if $d(a, b)$ is even (resp. odd). Given a path P and two vertices x, y of P, the distance of x and y in P is denoted as $d_P(x, y)$ and the path P induced by x, y and all the intermediate vertices between x and y in P, is denoted as $P(x, y)$. A *cycle* $C = v_0 v_1 \dots v_k$, is a sequence of distinct vertices, such that $v_{i-1}v_i \in E$, where $1 \leq i \leq k$; and $k \geq 2$ and $v_k v_0 \in E$. The length of C is $k + 1$. A n-cycle (or C_n) is a cycle of length n, for $n \geq 3$.

The *join* (resp. *disjoint union*) of two graphs G_1 and G_2 is the graph $G = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2) \cup \{uv \mid u \in V(G_1)$ and $v \in V(G_2)\})$ (resp. $G = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2))$). The symbol $\oplus$ (resp. $\cup$) denotes the join operation (resp. disjoint union operation).

Two vertices u and v are called *true twins* if $N_G[u] = N_G[v]$. Similarly, u and v are called *false twins* if $N_G(u) = N_G(v)$. Vertices u and v are called *twins* if they are either true or false twins.

A vertex v of a graph $G = (V, E)$ is said to be a *universal vertex* if $N[v] = V$. A pair of nonadjacent vertices $\{u, u'\}$ of G is said to be a *universal pair* if every $x \in V \setminus \{u, u'\}$ is adjacent to both u and u'. A set $A \subseteq V$ is a *stable set* or an *independent set* of G if the graph induced by A contains no edge. A set $C \subseteq V$ is a *clique* if for every pair of vertices u, v in C, $uv \in E$.

A graph $G = (V, E)$ is said to be *P_4-sparse* if any induced subgraph with five vertices contains at most one P_4. A *spider* is a graph $G = (V, E)$, where V admits a partition into three subsets S, C, and R such that



(a) Thin spider. (b) Thick spider.

Fig. 2. Examples of spiders with spider partition (S, C, R).

- $C = \{c_1, \dots, c_l\}$ $(l \geq 2)$ is a clique.
- $S = \{s_1, \dots, s_l\}$ is a stable set.
- Every vertex in R is adjacent to every vertex in C and nonadjacent to any vertex of S. The edges in R are unrestricted.

More on P_4-sparse graphs and spider graphs can be found in [20]. From now on, we denote a spider with $G(S, C, R)$. A spider $G(S, C, R)$ is said to be a

- *thin spider* if for every $i \in \{1, \dots, l\}$, $N_C(s_i) = \{c_i\}$
- *thick spider* if for every $i \in \{1, \dots, l\}$, $N_C(s_i) = C \setminus \{c_i\}$ (see Fig. 2).

A graph $G = (V, E)$ is *bipartite* if $V(G)$ can be partitioned into two independent sets. A *permutation graph* is a graph whose vertices represent the elements of a permutation, and whose edges represent pairs of elements that are reversed by the permutation. A bipartite graph G is a *bipartite permutation* graph if G is a permutation graph as well.

A graph $G = (V, E)$ is said to be *chordal*, if there does not exist any induced cycle of length at least 4 in G. G is *strongly chordal* if it is a chordal graph and every cycle of even length (at least 6) in G has an odd chord, that is, an edge that connects two vertices which are an odd distance (>1) away from each other in the cycle. More on these graph classes can be found in the books [2,14].

Distance-hereditary graphs are graphs in which the distance between any two vertices in any connected induced subgraph is the same as their distance in the original graph. More on distance-hereditary graphs can be found in the seminal paper [1].

2.2. Results from previous literature

The following results are from the existing literature and are used in the upcoming proofs.

Theorem 2.1 (*[10]*). *Given a graph $G = (V, E)$ and a vertex $v \in V$, v is a mandatory vertex of G if and only if one of the following two conditions hold:*

- *there exists $u \in N(v)$ such that there does not exist any induced 2-path of the form uvx in G.*
- *there exists $u \in N(v)$ such that every induced 2-path uvx is part of a 4-cycle.*

We also propose the following observation, which follows directly from Theorem 2.1.

Observation 2.1. *Let $u, v \in V$; if $N_G(u) \subseteq N_G(v)$ or $N_G[u] \subseteq N_G[v]$, then u is a mandatory vertex. In particular, every simplicial vertex of degree at least 1 is a mandatory vertex. If two vertices u and v are twins of degree at least 1 in a graph G, then they both are mandatory vertices of G.*

Theorem 2.2 (*[10]*). *Let $G = (V, E)$ be a graph with a path $v_0v_1 \dots v_{k-1}v_k$ whose internal vertices have degree 2, v_0 has degree at least 2, and v_k has degree 1. Then the vertices $v_0, v_1, \dots, v_{k-1}$ are never part of any minimum MEG set.*

Theorem 2.3 (*[7]*). *Let G be a graph with a cut-vertex v and $C_1, C_2, \dots, C_k$ be the k connected components obtained when removing v from G. If $S_1, S_2, \dots, S_k$ are MEG sets of the induced subgraphs $G[C_1 \cup \{v\}]$, $G[C_2 \cup \{v\}], \dots, G[C_k \cup \{v\}]$, then $S = (S_1 \cup S_2 \cup \dots \cup S_k) \setminus \{v\}$ is an MEG set of G.*

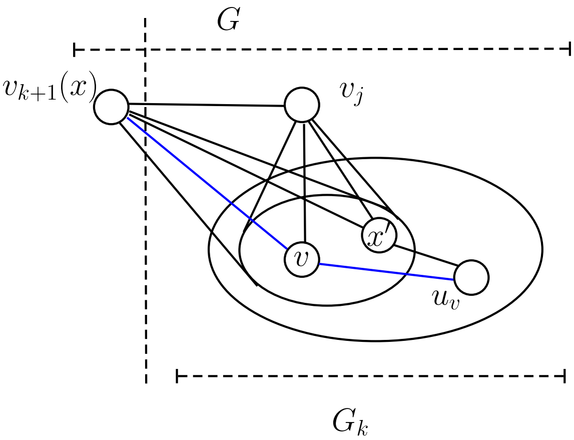


Fig. 3. The mandatory vertices in $V(G_k) \setminus Cut(G_k)$ remain mandatory in $V(G) \setminus Cut(G)$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. MEG for distance-hereditary graphs

In this section, we show that the MIN-MEG problem can be solved efficiently for distance-hereditary graphs. Throughout this section, we assume that $G = (V, E)$ is a connected distance-hereditary graph.

For any given ordering $(v_1, v_2, \ldots, v_n)$ of V, let the induced subgraph on $\{v_1, v_2, \ldots, v_i\}$ be denoted as G_i. A *one vertex-extension ordering* of G is an ordering $(v_1, v_2, \ldots, v_n)$ of V such that v_i is a leaf or a twin (true or false) of some vertex in G_i for each i, $2 \le i \le n$. Hammer et al. [15] proved that a graph is distance-hereditary if and only if it has a one vertex-extension ordering. We show that for a distance-hereditary graph G, $V \setminus Cut(G)$ forms a minimum MEG set of G.

Theorem 3.1. *Given a connected distance-hereditary graph $G = (V, E)$, $Man(G) = V \setminus Cut(G)$.*

Proof. Our objective is to prove that every vertex $v \in V \setminus Cut(G)$ is mandatory in G. To prove this, we employ induction on $|V|$. For the base case, $|V| = 3$; the only connected graphs of order 3 are P_3 and C_3 and for both of them, $V \setminus Cut(G) = Man(G)$. So, the statement is true for base cases. For the induction hypothesis, let us assume that every vertex $v \in V \setminus Cut(G)$ is contained in $Man(G)$, whenever $|V| \le k$.

Let us consider a connected distance-hereditary graph $G = (V, E)$ with $|V| = k+1$, adhering to a one-vertex extension ordering $(v_1, v_2, \ldots, v_k, v_{k+1})$. By the definition of a one vertex-extension ordering, G is constructed from G_k by appending v_{k+1} as either a pendant or a twin (true or false) to some vertex v_j, where $j \le k$. We inspect two potential scenarios.

Case 1: v_{k+1} **is a twin of** v_j (refer to Fig. 3).

Note that $Cut(G) = Cut(G_k) \setminus \{v_j\}$ and $V(G) \setminus Cut(G) = (V(G_k) \setminus Cut(G_k)) \cup \{v_j, v_{k+1}\}$. We show that $V(G) \setminus Cut(G)$ is contained in every MEG set of G. Since v_j and v_{k+1} are twins, they are contained in every MEG set of G, by Observation 2.1. Hence, $v_j, v_{k+1} \in Man(G)$.

By the induction hypothesis, the set $V(G_k) \setminus Cut(G_k)$ is contained in every MEG set of G_k. By Theorem 2.1, this implies that for any $v \in V(G_k) \setminus Cut(G_k)$, there exists $u_v \in N_{G_k}(v)$ such that any induced 2-path $u_v v x$ is part of a 4-cycle in G_k. Next, we show that $v \in Man(G)$. For the sake of contradiction, assume $v \notin Man(G)$; by Theorem 2.1, this implies that there exists an induced 2-path $u_v v x$ that is part of no 4-cycle in G. But this implies that this 2-path was not present in G_k (the path highlighted in blue in Fig. 3). Hence, $x = v_{k+1}$. This implies that $v \in N(v_{k+1})$ and $u_v \in N(v) \setminus N(v_{k+1})$; hence $v \in N(v_{k+1}) \setminus \{v_j\}$. But note that v was mandatory in G_k, which implies $u_v v v_j$ is part of a 4-cycle $u_v v v_j x'$ in G_k (refer to Fig. 3). This implies $u_v v v_{k+1}$ is also part of the 4-cycle $u_v v v_{k+1} x'$ in G, leading to a contradiction. Hence, $v \in Man(G)$.

As a result, every vertex of $V(G) \setminus Cut(G)$ is contained in $Man(G)$. The proof of the case when v_{k+1} is a false twin of v_j, is analogous, hence omitted.

Case 2: v_{k+1} **is a pendant vertex adjacent to** v_j.

Note that $Cut(G) = Cut(G_k) \cup \{v_j\}$. Since v_{k+1} is a pendant vertex, it is contained in $Man(G)$, by Theorem 2.2. By the induction hypothesis, $V(G_k) \setminus Cut(G_k)$ is contained in every MEG set of G_k. This implies that for every $v \in V(G_k) \setminus (Cut(G_k) \cup \{v_j\})$, there exists $u_v \in N(v)$ such that any induced 2-path $u_v v x$ is part of a 4-cycle. Hence, after the addition of the pendant vertex v_{k+1} adjacent to v_j, the vertices in $V(G_k) \setminus (Cut(G_k) \cup \{v_j\})$, which were mandatory in G_k, remain mandatory in G. This implies that $V(G) \setminus Cut(G)$ is contained in every MEG set of G.

Hence, by induction, we have shown that $V \setminus Cut(G) \subseteq Man(G)$. As a result, $V \setminus Cut(G) = Man(G)$ and $V \setminus Cut(G)$ is a minimum MEG set of G. □

4. MEG for bipartite permutation graphs

A *strong ordering* $(<_X, <_Y)$ of a bipartite graph $G = (X \cup Y, E)$ entails an ordering $<_X$ for the set X and an ordering $<_Y$ for the set Y. This ordering adheres to the condition that for any edges ab and a_0b_0, where $a, a_0 \in X$ and $b, b_0 \in Y$, if a precedes a_0 in $<_X$ and b_0 precedes b in $<_Y$, then both ab_0 and a_0b are edges in G. An ordering $<_X$ for X exhibits the *adjacency property* if, for each vertex in Y, its neighbors in X appear consecutively in $<_X$. Moreover, the ordering $<_X$ is said to have the *enclosure property* if, for any pair of vertices y and y_0 in Y where $N(y)$ is contained in $N(y_0)$, the vertices of $N(y_0) \setminus N(y)$ are consecutive in $<_X$. Strong ordering, adjacency property, and enclosure property, as delineated above, yield the subsequent result as discussed in previous literature.

Theorem 4.1 ([30]). *The following statements are equivalent for a bipartite graph $G = (X \cup Y, E)$:*

1. *$G = (X \cup Y, E)$ is a bipartite permutation graph.*
2. *G has a strong ordering.*
3. *There exists an ordering of X, which has the adjacency property and the enclosure property.*

Hence, from Theorem 4.1, we can conclude that if G is a bipartite permutation graph, then G has a strong ordering. We state another result from the previous literature.

Theorem 4.2 ([18]). *Let $(<_X, <_Y)$ be a strong ordering of a connected bipartite permutation graph $G = (X \cup Y, E)$. Then both $<_X$ and $<_Y$ have the adjacency property and the enclosure property.*

Let $G = (X \cup Y, E)$ be a bipartite permutation graph, which admits a strong ordering $(<_X, <_Y)$. Based on this ordering, we define the following notations:

- $f(v) = v'$, where $v' \in N(v)$ is the smallest neighbor of v with respect to the ordering ($v' < u$ for every $u \in N(v)$)
- $l(v) = v'$, where $v' \in N(v)$ is the largest neighbor of v with respect to the ordering ($u < v'$ for every $u \in N(v)$).

In the above definition, the notation $<$ is used for both $<_X$ and $<_Y$, depending on the partition to which the vertex belongs. Now we state a result from the previous literature which will be used in the upcoming proof.

Lemma 4.1 ([21]). *Given a connected bipartite permutation graph $G = (X \cup Y, E)$ with its strong ordering $(<_X, <_Y)$, G satisfies the following properties:*

1. *Given any vertex of G, its neighbor set consists of some consecutive vertices in $<_X$ or $<_Y$.*
2. *For a pair of vertices u, v from X or Y, if $u < v$ then $f(u) \leq f(v)$ and $l(u) \leq l(v)$.*

Throughout this section, we assume that $G = (X \cup Y, E)$ is a bipartite permutation graph. Given a strong ordering $(<_X, <_Y)$ of G, let the vertices of X and Y be ordered as $x_1 <_X x_2 <_X \dots x_p$ and $y_1 <_Y y_2 <_Y \dots y_q$, where $p = |X|$ and $q = |Y|$. When the context is clear, we write $<$ instead of $<_X$ or $<_Y$. In this section we call x_1, x_p, y_1, y_q as *extreme* vertices and all the other vertices as *interim* vertices.

Lemma 4.2. *Given a connected bipartite permutation graph $G = (X \cup Y, E)$ with strong ordering $(<_X, <_Y)$ and an interim vertex $u \in V(G)$, u is not a cut-vertex if and only if there exists $u_0 \in N(u)$ such that $f(u_0) < u$ and $l(u_0) > u$.*

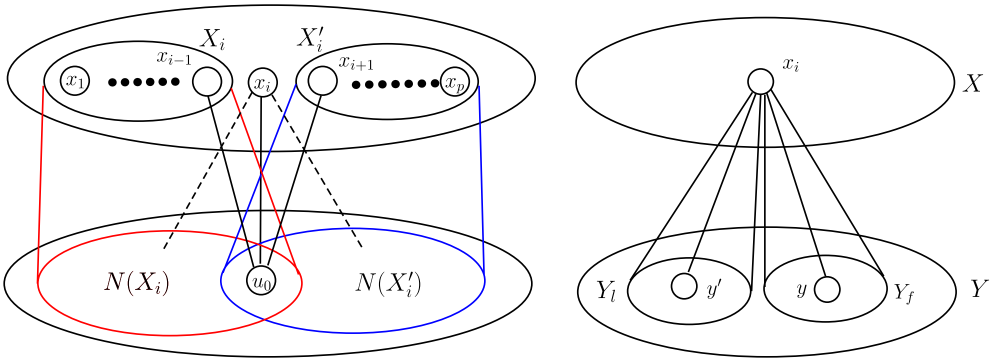
Proof. Without loss of generality, let $u \in X$ and $u = x_i$. Let there exist $u_0 \in Y$ such that $f(u_0) < u$ and $l(u_0) > u$. Suppose that u is a cut-vertex. Let $X_i = \{x_1, \dots, x_{i-1}\}$ and $X'_i = \{x_{i+1}, \dots, x_p\}$. Next we prove the following claim:

Claim 4.1. *$G[X_i \cup N(X_i)]$ and $G[X'_i \cup N(X'_i)]$ are connected.*

Proof. We prove that $G[X_i \cup N(X_i)]$ is connected, by showing that $N(x_j) \cap N(x_{j+1}) \neq \emptyset$, for all $x_j \in X$. Suppose that, there exists $x_j \in X_i$ such that $N(x_j) \cap N(x_{j+1}) = \emptyset$. So, $l(x_j) < f(x_{j+1})$ (by Lemma 4.1), hence $l(x_j) < f(x_k)$ for all $k \in \{j+1, \dots, p\}$. As a result, $N(x_j) \cap N(x_k) = \emptyset$, for all $k \in \{j+1, \dots, p\}$. This implies that there does not exist any path from x_j to x_{j+1} in G, which contradicts the fact that G is connected. Hence, $N(x_j) \cap N(x_{j+1}) \neq \emptyset$, for all $j \in \{1, \dots, i-1\}$. This implies that $G[X_i \cup N(X_i)]$ is connected. Analogously, it can be shown that $G[X'_i \cup N(X'_i)]$ is also connected. □

Since $u_0 \in N(X_i) \cap N(X'_i)$, $G[X_i \cup N(X_i) \cup X'_i \cup N(X'_i)]$ is also connected (refer to Fig. 4(a) for better understanding). As a result $G \setminus \{u\} = G[(X_i \cup N(X_i) \cup X'_i \cup N(X'_i))]$ is a connected graph, which implies that u is not a cut-vertex. If $u \in Y$, the proof follows from similar arguments.

Conversely, assume that u is not a cut-vertex. For the sake of contradiction, assume that for every $y_j \in N(x_i)$, either $f(y_j) = x_i$ or $l(y_j) = x_i$. Hence, $N(x_i)$ can be partitioned into two sets $Y_f = \{y \in N(x_i) \mid f(y) = x_i\}$ and $Y_l = \{y \in N(x_i) \mid l(y) = x_i\}$. Note that none of the sets Y_f and Y_l can be empty as x_i is an interim vertex and G is a connected graph. Consider any two vertices, $y \in Y_f$ and $y' \in Y_l$. Note that every y-y' path contains x_i (refer to Fig. 4(b)). As a result, $G \setminus \{x_i\}$ is disconnected, which leads to a contradiction. Hence there exists a vertex $u_0 \in N(u)$ such that $f(u_0) < u$ and $l(u_0) > u$. □



(a) $u = x_i$ is not a cut-vertex. (b) $u = x_i$ is not a cut-vertex.

Fig. 4. Figures used in the proof of Lemma 4.2.

Theorem 4.3. *Given a connected bipartite permutation graph $G = (X \cup Y, E)$, we have $Man(G) = V(G) \setminus Cut(G)$.*

Proof. We show that every non-cut-vertex is a mandatory vertex. Note that if an extreme vertex $u \in V(G)$ is not a cut-vertex, then it is either a pendant vertex or there exists u' such that $N(u) \subseteq N(u')$. In both cases, u is a mandatory vertex.

Now, we consider an interim vertex $x_i \in X$ (without loss of generality), which is not a cut-vertex. By Lemma 4.2, there exists $y_j \in N(x_i)$ such that $f(y_j) < x_i$ and $l(y_j) > x_i$. Note that for any induced 2-path $y_j x_i y_k$, two cases can occur.

Case 1: $y_k < y_j$. In this case $x_{i-1} \in N(y_k) \cap N(y_j)$. This implies that $y_j x_i y_k$ is part of a 4-cycle $y_j x_i y_k x_{i-1} y_j$.

Case 2: $y_k > y_j$. In this case $x_{i+1} \in N(y_k) \cap N(y_j)$. This implies that $y_j x_i y_k$ is part of a 4-cycle $y_j x_i y_k x_{i+1} y_j$.

Hence x_i is a mandatory vertex, by Theorem 2.1. The same can be shown for an interim vertex $y_i \in Y$ which is not a cut-vertex. As a result $Man(G) = V(G) \setminus Cut(G)$. □

5. MEG for P_4-sparse graphs

In this section, we propose an efficient algorithm that solves the MIN-MEG problem for P_4-sparse graphs. The class of P_4-sparse graphs is an extension of the class of cographs. Below, we state a characterization theorem for P_4-sparse graphs from the previous literature.

Theorem 5.1 ([20]). *A graph G is said to be P_4-sparse if and only if one of the following conditions hold*

- *G is a single vertex graph.*
- *$G = G_1 \cup G_2$, where G_1 and G_2 are P_4-sparse graphs.*
- *$G = G_1 \oplus G_2$, where G_1 and G_2 are P_4-sparse graphs.*
- *G is a spider which admits a spider partition (S, C, R) where either $G[R]$ is a P_4-sparse graph or $R = \emptyset$.*

Hence, by Theorem 5.1, a graph that is P_4-sparse and contains at least two vertices can be classified as either a join or union of two P_4-sparse graphs, or a particular type of spider (thick or thin). Consequently, in this section, we compute $meg(G)$ for each of the cases. First, we consider the case of G being a spider.

Lemma 5.1. *Let $G = (V, E)$ be a thin spider and (S, C, R) be the spider partition. Then, $Man(G) = S \cup R$, which is a minimum MEG set of G.*

Proof. Note that by Observation 2.1, S is contained in every MEG set (as every vertex of S is a pendant vertex) and no vertex in C can be part of a minimum MEG set, as they are cut-vertices. Now if $R \neq \emptyset$, then for each $r \in R$ and for any $c \in C$, $N[r] \subset N[c]$, hence by Observation 2.1, r is contained in every MEG set. As a result $meg(G) = |Man(G)| = |S| + |R|$. □

Lemma 5.2. *Let $G = (V, E)$ be a thick spider and (S, C, R) be the spider partition. Then, $meg(G) = |S| + |R|$, if $|C| = 2$ and $meg(G) = |Man(G)| = |V|$, otherwise.*

Proof. Note that if $|C| = 2$, then G is a thin spider; hence, the result follows from the previous lemma. Now, let $|C| \geq 3$. Consider any two distinct vertices c_i, c_j from C. The only induced 2-path of type $c_j c_i x$ is $c_j c_i s_j$, which is part of a 4-cycle $c_j c_i s_j c_k$, where $k \notin \{i, j\}$; such k exists as $|C| \geq 3$. This implies that c_i is a mandatory vertex for every $c_i \in C$. Now if $R \neq \emptyset$, then for any $r \in R$ (or $s \in S$), $N[r] \subset N[c]$ (or $N[s] \subset N[c]$), for any $c \in C$. Hence, by Observation 2.1, $R \cup S$ is contained in every MEG set. As a result $meg(G) = |V| = |Man(G)|$. □

Now, the only case that remains to be handled is when G is a join of two P_4-sparse graphs G_1 and G_2. Note that if G_1 and G_2 both are nontrivial graphs (having more than one vertex), then $meg(G) = |V(G)|$. Now assume that, without loss of generality, $|V(G_1)| = 1$ and $V(G_1) = \{v\}$. Two cases can appear. In the first case, if G_2 is a disconnected graph, then v is a cut-vertex and $N[x] \subseteq N[v]$ for any $x \in V(G_2)$, hence $meg(G) = |V(G_2)|$. For the other case, let G_2 be a connected graph. In that case, we propose the following lemma.

Lemma 5.3. *Let $G = G_1 \oplus G_2$ and $V(G_1) = \{v\}$ and G_2 is a nontrivial connected P_4-sparse graph. Then,*

1. *$meg(G) = n$, if there exists a vertex u in G_2, which satisfies the property that $d_{G_2}(u, x) \leq 2$ for every $x \in V(G_2) \setminus \{u\}$.*
2. *$meg(G) = n - 1$, otherwise.*

Proof. Note that for every vertex $x \in V(G_2)$, $N[x] \subseteq N[v]$. As a result, x belongs to every MEG set of G, implying $V(G_2) \subseteq M$, where M is any minimum MEG set of G. Suppose that there exists a vertex $u \in V(G_2)$, with the property: $d_{G_2}(u, x) \leq 2$ for every $x \in X$. Now consider any induced 2-path uvy. Clearly, u, y are not adjacent. Hence $d_{G_2}(u, y) = 2$. So, uvy is part of a 4-cycle, as y is at distance 2 from u in the graph G_2. Hence, by Theorem 2.1, v is a mandatory vertex in G, for every $v \in V(G)$; which implies that $meg(G) = n$.

Now, let us assume that no such vertex u exists. Consider any edge vx, $x \in V(G_2)$. By assumption, there exists a vertex $x' \in V(G_2)$ such that $d_{G_2}(x, x') \geq 3$. Hence, the pair x, x' monitors the edge vx. Since x is arbitrary in G_2, then every edge of the form vx can be monitored by some pair of vertices in G_2. Hence, $meg(G) = n - 1$. □

Hence, from the above discussion and Lemmas 5.1–5.3, the following theorem can be concluded:

Theorem 5.2. *Given a connected P_4-sparse graph $G = (V, E)$, $meg(G) = |Man(G)|$.*

6. MEG for strongly chordal graphs

In this section, we consider that $G = (V, E)$ is a connected strongly chordal graph. We use the definition of strongly chordal graphs from [8]. We define a vertex $v \in V$ as *simple* if the vertices within its closed neighborhood can be arranged as follows: $N_G[v] = v_1, v_2, \dots, v_r$, where $v_1 = v$ and $N_G[v_i] \subseteq N_G[v_j]$ for $1 \leq i \leq j \leq r$. A graph G is strongly chordal if every induced subgraph of G contains a simple vertex. An ordering $\alpha = (v_1, v_2, \dots, v_n)$ of vertices in V is called a strong elimination ordering (SEO) if v_i is a simple vertex in $G[v_i, v_{i+1}, \dots, v_n]$. A graph is strongly chordal if it admits an SEO. We show that the set of mandatory vertices forms an MEG set for all strongly chordal graphs.

Theorem 6.1. *For any strongly chordal graph G, $Man(G)$ forms a minimum MEG set of G.*

Proof. We show this by induction on the order n the graph G, n. For $n = 3$, the strongly chordal graphs are P_3 and C_3; the statement is true for both of these graphs. As a result, the statement is true for base cases. Now, let the statement be true for $n = k$. For $n = k + 1$, let us consider a strongly chordal graph G_{k+1} with SEO $v_1 < v_2 < \cdots < v_{k+1}$. Now define $G_k := G_{k+1}[v_2, v_3, \dots, v_{k+1}]$. Let us consider the neighborhood of v_1 as $N(v_1) = \{v_{i_1}, v_{i_2}, \dots, v_{i_l}\}$, where $N[v_{i_1}] \subseteq N[v_{i_2}] \subseteq \ldots \subseteq N[v_{i_l}]$. First, we show the following claims.

Claim 6.1. *$Man(G_k) \setminus \{v_{i_l}\} \subseteq Man(G_{k+1}) \subseteq Man(G_k) \cup \{v_1, v_{i_l}\}$.*

Proof. Note that by Observation 2.1, $N[v_1] \setminus \{v_{i_l}\} \subseteq Man(G_{k+1})$ as for every $v \in N[v_1] \setminus \{v_{i_l}\}$, $N[v] \subseteq N[v_{i_l}]$. Now, $V(G_{k+1})$ can be partitioned as $V(G_{k+1}) = N[v_1] \cup V'$, where V' is the set of all non neighbors of v_1. Let us consider a vertex $v \in V'$. Two cases can appear.

Case 6.1.1: $v \in Man(G_k)$

This implies that a vertex $u \in N_{G_k}(v)$ exists such that every induced 2-path uvx is part of a 4-cycle. Now, after adding v_1 to the vertex set, the neighbor set of v has not expanded, as v is not a neighbor of v_1. This implies that every induced 2-path uvx is still part of a 4-cycle. This implies $v \in Man(G_{k+1})$.

Case 6.1.2: $v \notin Man(G_k)$

For the sake of contradiction, let $v \in Man(G_{k+1})$. This implies that there exists $u \in N_{G_{k+1}}(v)$ such that every induced 2-path uvx is part of a 4-cycle $uvxx'$. Note that $u \neq v_1$, as v is not adjacent to v_1. But since v was not a mandatory vertex in G_k, for this u, there must exist an induced 2-path uvx in G_k, which is not part of any 4-cycle in G_k. This implies that the 4-cycle $uvxx'$ in G_{k+1}, of which uvx is part of, must include v_1. This implies that $x' = v_1$ and $u, x \in N(v_1)$. As a result, there is an edge between u and x (refer to Fig. 5), as the graph induced on $N[v_1]$ is a clique. So, uvx is not an induced 2-path, which leads to a contradiction. Hence, $v \notin Man(G_{k+1})$.

This implies $Man(G_k) \cap V' = Man(G_{k+1}) \cap V'$ and $N(v_1) \setminus \{v_{i_l}\} \subseteq Man(G_k)$ and $N[v_1] \setminus \{v_{i_l}\} \subseteq Man(G_{k+1})$. As a result, the claim follows. □

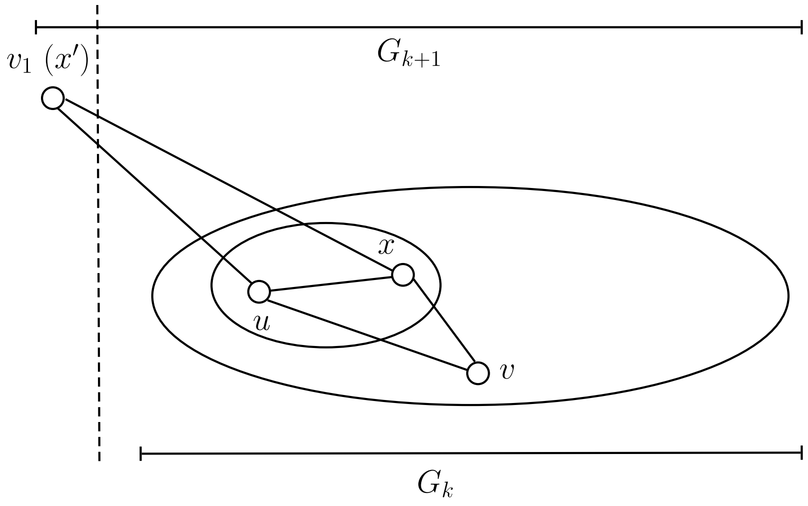
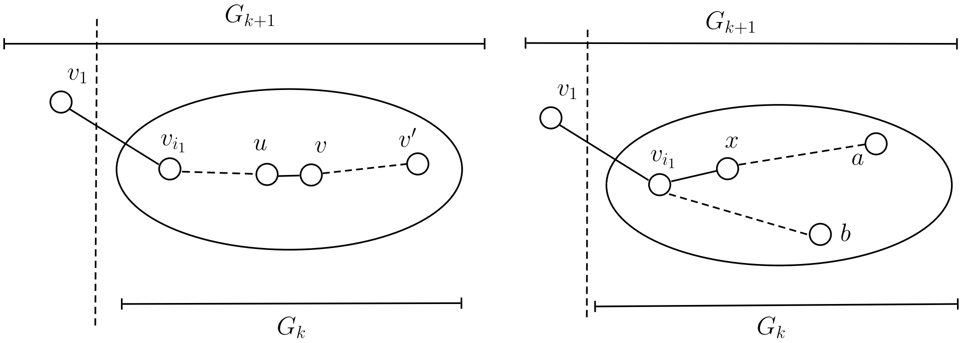


Fig. 5. v is a mandatory in G_{k+1} but not in G_k.



(a) The pair v_1, v' monitors both the edge $v_1v_{i_1}$ and uv.

(b) The pairs v_1, a and v_1, b both monitors the edge $v_{i_1}v_1$.

Fig. 6. v_1 is a pendant vertex and v_{i_1} is the stem.

Claim 6.2. *If v_1 is a pendant vertex, then $Man(G_{k+1})$ is a minimum MEG set of G.*

Proof. Note that if v_1 is a pendant vertex in G_{k+1}, then $N(v_1) = \{v_{i_1}\}$. This implies that $Man(G_{k+1}) = (Man(G_k)\setminus\{v_{i_1}\})\cup\{v_1\}$. By the induction hypothesis, $Man(G_k)$ is an MEG set of G_k. If $v_{i_1} \in Man(G_k)$, then there exists an edge $uv \in E(G_k)$ that is only being monitored by v_{i_1} and v' for some $v' \in Man(G_k)$; such that no other pair in $Man(G_k)$ monitors uv. Evidently, uv and $v_1v_{i_1}$ can both be monitored by v_1, v' in G_{k+1} (refer to Fig. 6(a)).

If $v_{i_1} \notin Man(G_k)$, then all the edges in G_{k+1} except $v_1v_{i_1}$ are being monitored by some pair of vertices in $Man(G_k)$. The only edge that remains to be monitored is $v_1v_{i_1}$. Consider any edge of the form $v_{i_1}x$ in G_k. It is monitored by some $a, b \in Man(G_k)$. Hence, it is not hard to see that either the pair a, v_1 or the pair b, v_1 monitors the edge $v_1v_{i_1}$ (refer to Fig. 6(b)). As a result in both of the cases, $meg(G_{k+1}) = |Man(G_{k+1})|$. □

So, from now on, let us assume that $|N_{G_{k+1}}(v_1)| \geq 2$, that is v_1 is not a pendant vertex in G_{k+1}.

Claim 6.3. *If $v_{i_l} \in Man(G_{k+1})$, then $v_{i_l} \in Man(G_k)$.*

Proof. Suppose that, $v_{i_l} \in Man(G_{k+1})$ but $v_{i_l} \notin Man(G_k)$. Then for every $u \in N_{G_k}(v_{i_l})$, there exists an induced 2-path $uv_{i_l}x$ in G_k, which is not part of any 4-cycle in G_k. But since, $v_{i_l} \in Man(G_{k+1})$, there exists $u_i \in N_{G_{k+1}}(v_{i_l})$ such that every induced 2-path $u_iv_{i_l}x$ is part of a 4-cycle $u_iv_{i_l}xx'$ in G_{k+1}. This 4-cycle was clearly not present in G_k, which implies that either $u_i = v_1$ or $x' = v_1$. If $u_i = v_1$, then for every neighbor x of v_{i_l}, x is also adjacent to a vertex x' in $N(v_1) \setminus \{v_{i_l}\}$ (refer to Fig. 7). This implies $x \in N[v_{i_{l-1}}]$, hence $N[v_{i_{l-1}}] = N[v_{i_l}]$, which implies that $v_{i_l} \in Man(G_k)$, by Observation 2.1, leading to a contradiction. If $x' = v_1$, then $u_i, x \in N(v_1)$. This implies that an edge exists from u_i to x, as the graph induced on $N[v_1]$ is a clique, which contradicts the fact that u_ivx is an induced 2-path. As a result, $v_{i_l} \in Man(G_k)$. □

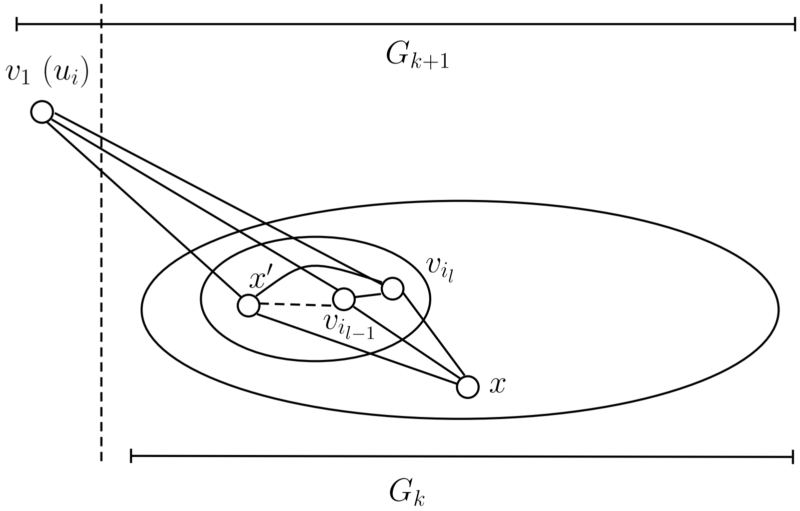


Fig. 7. $u_i = v_1$ implies that $N[v_{i_{l-1}}] = N[v_{i_l}]$.

Claim 6.3 implies that if the vertex v_{i_l} is mandatory in G_{k+1}, then it was also mandatory in G_k. Two cases may appear for v_{i_l}. The rest of the proof proceeds as follows:

- In Case 1, we consider $v_{i_l} \in Man(G_{k+1})$ and show that $Man(G_{k+1})$ forms an MEG set.
- In Case 2, we consider $v_{i_l} \notin Man(G_{k+1})$. Here again, two subcases may appear, $v_{i_l} \notin Man(G_k)$ (Case 2.1) and $v_{i_l} \in Man(G_k)$ (Case 2.2). In both of the subcases, we show that $Man(G_{k+1})$ forms an MEG set.

Case 1: $v_{i_l} \in Man(G_{k+1})$

By Claim 6.3, $v_{i_l} \in Man(G_k)$. In this case, $Man(G_{k+1}) = Man(G_k) \cup \{v_1\}$. Now, by induction hypothesis, $Man(G_k)$ is an MEG set of G_k. Next we prove the following claim:

Claim 6.4. *v_1 does not belong to any shortest x-y path, where $x, y \in V(G_k)$.*

Proof. Clearly, none of the x, y is v_1 as they both belong to $V(G_k)$. Suppose that, $P = ((x =)x_1x_2 \dots x_k(= y))$ is a shortest x-y path which contains v_1. Let $x_i = v_1$ for some $1 < i < k$. Then the vertices $x_{i-1},\ x_{i+1}$ are adjacent in G_k (and in G_{k+1} as well), as $N[v_1]$ induces a clique. This implies that $x_1x_2 \dots x_{i-1}x_{i+1} \dots x_k$ is a path between x and y which has smaller length than P, which contradicts the fact that P is a shortest x-y path. As a result v_1 does not belong to any shortest x-y path. □

Hence, all the edges in G_{k+1} that do not have v_1 as an endpoint are being monitored by some pair of vertices in $Man(G_k)$. Now since $N[v_1] \subseteq Man(G_{k+1})$, then all the edges of the form v_1x are being monitored by v_1 and x, as $x \in N[v_1] \subseteq Man(G_{k+1})$. Hence, $Man(G_{k+1})$ is an MEG set of G_{k+1}.

Case 2: $v_{i_l} \notin Man(G_{k+1})$

In this case, either $v_{i_l} \in Man(G_k)$ or $v_{i_l} \notin Man(G_k)$.

Case 2.1: $v_{i_l} \notin Man(G_k)$

If $v_{i_l} \notin Man(G_k)$, then all the edges in $E(G_{k+1})$ which do not contain v_1 as an endpoint are monitored by some pair in $Man(G_k)$. All the edges of the form $v_1v_{i_j}$ are monitored by v_1, v_{i_j}, for every $1 \le j \le l-1$. Now, the edge that is yet to be monitored is $v_1v_{i_l}$. Note that the edge $v_{i_{l-1}}v_{i_l}$ is being monitored by $v_{i_{l-1}}$ and v' for some $v' \in Man(G_k)$. This implies that every shortest path between $v_{i_{l-1}}$ and v' contains the edge $v_{i_{l-1}}v_{i_l}$. As a result, every shortest v_1-v' path also contains the edge $v_1v_{i_l}$, if not then there exists a shortest $v_{i_{l-1}}$-v' path that bypasses the edge $v_{i_{l-1}}v_{i_l}$, which leads to a contradiction (refer to Fig. 8). As a result, $Man(G_{k+1})$ forms an MEG set of G_{k+1}.

Case 2.2: $v_{i_l} \in Man(G_k)$

Let $v_{i_l} \in Man(G_k)$, which implies that $Man(G_{k+1}) = (Man(G_k) \setminus \{v_{i_l}\}) \cup \{v_1\}$. Now define E_l as the set of edges in G_k which were only being monitored by v_{i_l}, v', for some $v' \in Man(G_k)$ and no other pairs in $Man(G_k)$. Note that every edge in $E(G_{k+1}) \setminus E_l$ is being monitored by some pair in $Man(G_{k+1})$. Hence, it is sufficient to prove that every edge in E_l can be monitored by some pair in $Man(G_{k+1})$. Consider an edge $e \in E_l$.

Case 2.2.1: v_{i_l} is not an endpoint of e.

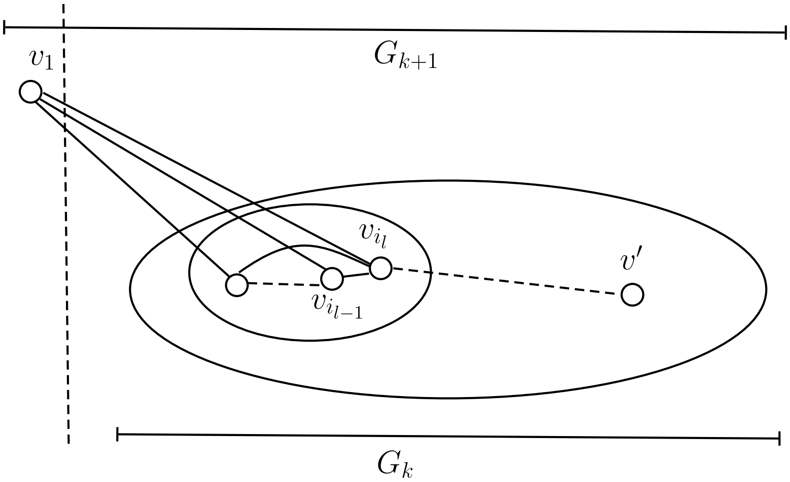


Fig. 8. The pair v_1, v' monitors the edge $v_1v_{i_1}$.

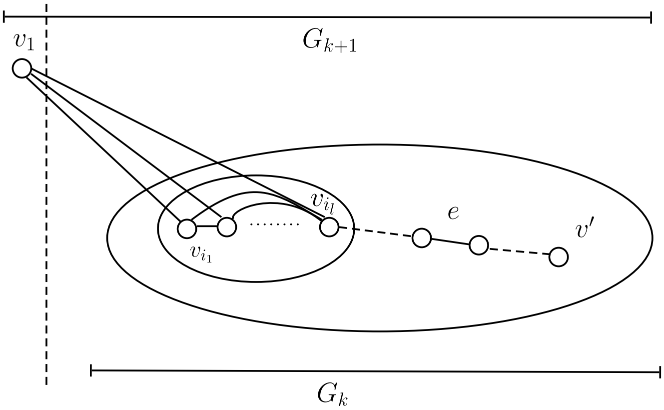


Fig. 9. e is not incident to v_{i_l}.

If v_{i_l} is not an endpoint of e, then in G_k, e was monitored by $v_{i_l}, v' \in Man(G_k)$. Note that $v' \notin N[v_1]$. Every shortest v_{i_l}-v' path contains the edge e. Next we show that every shortest v_1-v' path also contains e (refer to Fig. 9).

Claim 6.5. *Every shortest v_1-v' path also contains e.*

Proof. Let there exist a shortest v_1-v' path $P = ((v_1 =)x_1x_2 \dots x_k(= v'))$ that does not contain e. Note that v_{i_l} does not belong to the path (as it will contradict the fact that every shortest v_{i_l}-v' path contains the edge e). Without loss of generality, let us assume that $x_2 = v_{i_{l-1}}$. Note that $d(v_1, v') = 1 + d(v_{i_l}, v')$, or $v_{i_l}x_2 \dots x_k(= v')$ is a shortest v_{i_l}-v' path that does not contain e, leading to a contradiction. This implies that $d(v_{i_l}, v') = d(v_{i_{l-1}}, v')$.

So, $v_{i_{l-1}}x_3 \dots x_k$ is a shortest $v_{i_{l-1}}$-v' path that bypasses e. Since $N[v_{i_{l-1}}] \subseteq N[v_{i_l}]$, x_3 is also adjacent to v_{i_l}, hence $v_{i_l}x_3 \dots x_k$ is a shortest v_{i_l}-v' path that avoids e, which is a contradiction to the fact that v_{i_l}, v' monitor e. Hence, every shortest v_1-v' path also contains e. □

Case 2.2.2: v_{i_l} is an endpoint of e.

If v_{i_l} is an endpoint of e, then let the other endpoint be v. Note that in G_k, the edge $v_{i_l}v$ was monitored by v_{i_l} and v' for some v'.

At first let us consider the case $v \notin N_{G_{k+1}}(v_{i_{l-1}})$; such v exists as $N[v_{i_{l-1}}] \subset N[v_{i_l}]$ (otherwise $v_{i_{l-1}}$, v_{i_l} are twins, hence v_{i_l} will be a mandatory vertex of G_{k+1}). Now note that $d_{G_{k+1}}(v, v_1) = 2$ and $vv_{i_l}v_1$ is the unique shortest path from v_1 to v. Hence, the edge $v_{i_l}v$ is contained in every shortest v_1-v' path (refer to Fig. 10). Note that the edges $v_1v_{i_l}$ and $v_{i_{l-1}}v_{i_l}$ are also monitored by the pairs v_1, v' and $v_{i_{l-1}}, v'$, respectively.

Let $v \in N_{G_{k+1}}(v_{i_{l-1}})$. Note that there exists $x \in N[v_{i_l}] \setminus N[v_{i_{l-1}}]$ such that the induced 2-path $vv_{i_l}x$ is part of no 4-cycle. If not, then by Theorem 2.1 v_{i_l} would be mandatory in G_{k+1} as well. Now the edges $v_{i_l}v$ and $v_{i_l}x$ are being monitored by

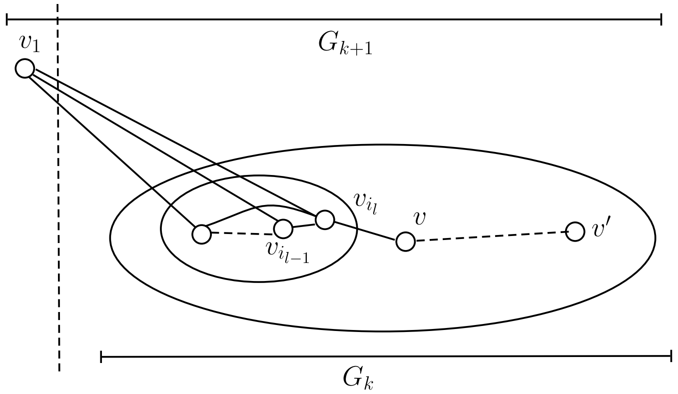


Fig. 10. $v_{i_l}v$ is being monitored by v_1 and v'.

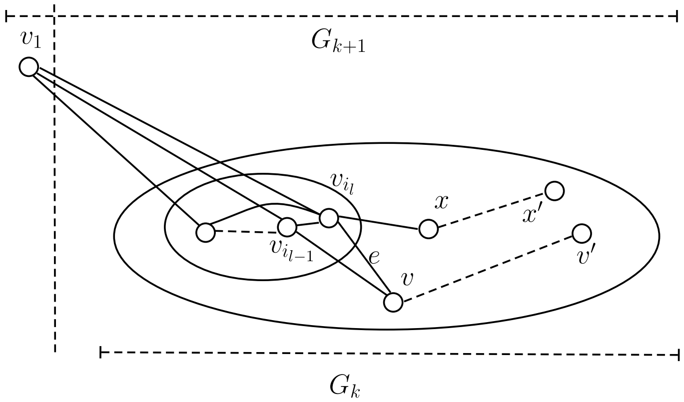


Fig. 11. e is incident to v_{l_1}.

v_{i_l}, v' and v_{i_l}, x', respectively in G_k (refer to Fig. 11). We also consider v' (and x') to be the minimum distance mandatory vertex from v_{i_l} such that v_{i_l} and v' (v_{i_l} and x') monitor the edge $v_{i_l}v$ ($v_{i_l}x$). Note that $v' \neq x'$; since if $v' = x'$, then every shortest v_{i_l}-v' path contains both of the edges $v_{i_l}v$ and $v_{i_l}x$, which is a contradiction.

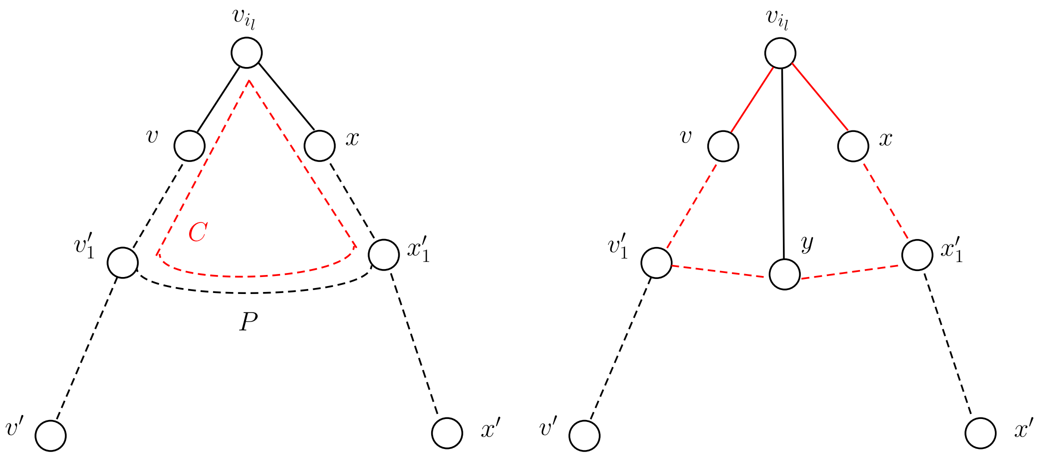
We show that v', x' monitor the edge $v_{i_l}v$ in G_{k+1}. Now, for the sake of contradiction, assume that there exists a shortest v'-x' path P that bypasses the edge $v_{i_l}v$ (refer to Fig. 12(a)).

Let v'_1 (and x'_1) be the nearest vertex form v_{i_l} that is present in $P \cap P_{v_{i_l}v'}$ (and $P \cap P_{v_{i_l}x'}$) (refer to Fig. 12(a)), where $P_{a,b}$ is defined to be any shortest a-b path. Since P is a shortest path from v' to x', then $d_P(v'_1, x'_1) \leq d(v'_1, v_{i_l}) + d(x'_1, v_{i_l})$. Let C be the cycle which is constructed with the boundaries $P(v'_1, x'_1)$, $P_{x'_1v_{i_l}}$ and $P_{v_{i_l}v'_1}$ (Refer to Fig. 12(a)). Note that there must be at least one chord attached to v_{i_l} in the cycle C. If not, since $vv_{i_l}x$ form an induced 2-path, then $vv_{i_l}x$ is part of a cycle of length at least 4, which does not have any chord, leading to a contradiction. Hence, v_{i_l} has at least one chord attached to it in C. Let $v_{i_l}y$ be one such chord; then we show that y lies in $P(v'_1, x'_1)$ in the next claim.

Claim 6.6. *y lies in $P(v'_1, x'_1)$.*

Proof. If y does not lie in $P(v'_1, x'_1)$, then y must lie in the path between v_{i_l} and x'_1 (or the path between v_{i_l} and v'_1). If it lies on the path between v_{i_l} and x'_1 (resp. between v_{i_l} and v'_1), then we have a path from v_{i_l} to x' (resp. between v_{i_l} and v') which has length smaller than $d(v_{i_l}, x')$ (resp. $d(v_{i_l}, v')$), which is a contradiction. Hence, y lies in $P(v'_1, x'_1)$ (refer to Fig. 12(b)). □

Note that $d(x'_1, y) \geq d(x'_1, v_{i_l})$ and $d(v'_1, y) \geq d(v'_1, v_{i_l})$. If not, then there exists a shorter or equal length path from v' to v_{i_l} (from x' to v_{i_l}) that bypasses the edge $v_{i_l}v$ (resp. $v_{i_l}x$), which is a contradiction to the fact that the pair v_{i_l}, v' (resp. v_{i_l}, x') monitors the edge $v_{i_l}v$ (resp. $v_{i_l}x$). Hence $d_P(x'_1, v'_1) = d(x'_1, y) + d(y, v'_1) \geq d(v'_1, v_{i_l}) + d(x'_1, v_{i_l})$, implying $d_P(x'_1, v'_1) = d(v'_1, v_{i_l}) + d(x'_1, v_{i_l})$. This implies that $d(x'_1, y) = d(x'_1, v_{i_l})$ and $d(v'_1, y) = d(v'_1, v_{i_l})$, which implies that the vertex y is unique in $P(v'_1, x'_1)$. This fact together with Claim 6.6 implies that in C, v_{i_l} is incident to exactly one chord.



(a) Another shortest path P from v' to x'. (b) v_{i_l} is incident to exactly one chord in C.

Fig. 12. The cycle C created by two distinct paths from v' to x'.

Now note that if v and y are not adjacent, then the induced 2-path $yv_{i_l}v$ is part of a chordless cycle of length at least 4, which is a contradiction. Hence $vy \in E(G_{k+1})$. Similarly, it can be shown that $xy \in E(G_{k+1})$. Hence $vv_{i_l}x$ is part of a 4-cycle $vv_{i_l}xyv$, which leads to a contradiction. Hence v' and x' monitor vv_{i_l}.

Hence, until now, we showed that all edges in E_l can be monitored by the vertices of $Man(G_k) \setminus \{v_{i_l}\}$. Hence, combining this fact with Claims 6.1 and 6.3, we can conclude that $Man(G_{k+1})$ forms a minimum MEG set of G_{k+1}. Hence, $meg(G) = |Man(G)|$ for every strongly chordal graph G. □

7. Conclusion and future aspects

In this paper, we solved the complexity status of the MIN-MEG problem for some well-known graph classes. Next, it will be interesting to address the following questions.

Question 1. *For any bipartite permutation graph G, we showed that $Man(G) = V \setminus Cut(G)$, and the graph mentioned in [11, Figure 4] (a path on six vertices with an added universal vertex, which is a permutation graph) shows that this is not always the case for permutation graphs. However, do we have $|Man(G)| = meg(G)$ when G is a permutation graph?*

Question 2. *For any strongly chordal graph G, we showed that $|Man(G)| = meg(G)$, and this was also shown to be true for well-partitioned chordal graphs in [10,11]. Does this property hold for chordal graphs as well?*

Question 3. *As can be seen in Fig. 1, the complexity status of the MIN-MEG problem is open for graph classes such as circle graphs, permutation graphs, chordal graphs, cocomparability graphs, etc. Is the MIN-MEG problem efficiently solvable for these graph classes?*

Question 4. *Does there exist a sufficient condition for the following property: $meg(G) = |Man(G)|$ or $meg(G) = |Man(G)| = |V \setminus Cut(G)|$, to hold for a graph G?*

While the idea of identifying a general sufficient condition under which $meg(G) = |Man(G)|$ or $meg(G) = |Man(G)| = |V \setminus Cut(G)|$ holds, is appealing; we believe that such a characterization may be extremely difficult to obtain, if it exists at all. Notably, any such condition would necessarily encompass all graphs satisfying $meg(G) = |V(G)|$, which represents a vast and structurally diverse collection of graphs. This includes, for example, complete bipartite graphs, hypercubes etc. The condition should also hold for all the graph classes which are considered in this paper. These families of graphs have very little in common from a structural standpoint.

Acknowledgments

Research of Florent Foucaud and Kaustav Paul was partially funded by the ANR project GRALMECO (ANR-21-CE48-0004), the French government IDEX-ISITE initiative 16-IDEX-0001 (CAP 20-25), the International Research Center “Innovation Transportation and Production Systems” of the I-SITE CAP 20-25, and the CNRS IRL ReLaX. Research in this article is partially supported by the FIST program of the Department of Science and Technology, Government of India, Reference No. SR/FST/MS-I/2018/22(C). Research of Arti Pandey is supported by CRG project, Grant Number-CRG/2022/008333, Science and Engineering Research Board (SERB), India

Data availability

No data was used for the research described in the article.

References

[1] H.J. Bandelt, H.M. Mulder, Distance-hereditary graphs, J. Combin. Theory Ser. B 41 (2) (1986) 182–208.

[2] L.W. Beineke, M.C. Golumbic, R.J. Wilson, Topics in algorithmic graph theory, vol. 178, Cambridge University Press, 2021.

[3] D. Bilò, G. Colli, L. Forlizzi, S. Leucci, On the inapproximability of finding minimum monitoring edge-geodetic sets (short paper), in: U. de'Liguoro, M. Palazzo, L. Roversi (Eds.), Proceedings of the 25th Italian Conference on Theoretical Computer Science, Torino, Italy, September 11-13, 2024, in: CEUR Workshop Proceedings, Vol. 3811, CEUR-WS.org, 2024, pp. 219–224.

[4] L. Bozyk, J. Derbisz, T. Krawczyk, J. Novotná, K. Okrasa, Vertex deletion into bipartite permutation graphs, Algorithmica 84 (8) (2022) 2271–2291, http://dx.doi.org/10.1007/S00453-021-00923-7.

[5] R.C. Brewster, G. MacGillivray, F. Yang, Broadcast domination and multipacking in strongly chordal graphs, Discrete Appl. Math. 261 (2019) 108–118, http://dx.doi.org/10.1016/J.DAM.2018.08.021.

[6] D. Chakraborty, F. Foucaud, A. Hakanen, Distance-based (and path-based) covering problems for graphs of given cyclomatic number, Discret. Math. 348 (11) (2025) 114595, http://dx.doi.org/10.1016/J.DISC.2025.114595.

[7] S.R. Dev, S. Dey, F. Foucaud, K. Narayanan, L.R. Sulochana, Monitoring edge-geodetic sets in graphs, Discrete Appl. Math. 377 (2025) 598–610, http://dx.doi.org/10.1016/J.DAM.2025.08.041.

[8] M. Farber, Domination, independent domination, and duality in strongly chordal graphs, Discrete Appl. Math. 7 (2) (1984) 115–130, http://dx.doi.org/10.1016/0166-218X(84)90061-1.

[9] F. Foucaud, S.-S. Kao, R. Klasing, M. Miller, J. Ryan, Monitoring the edges of a graph using distances, Discrete Appl. Math. 319 (2022) 425–438.

[10] F. Foucaud, P. Marcille, Z.M. Myint, R.B. Sandeep, S. Sen, S. Taruni, Monitoring edge-geodetic sets in graphs: Extremal graphs, bounds, complexity, in: S. Kalyanasundaram, A. Maheshwari (Eds.), Algorithms and Discrete Applied Mathematics - 10th International Conference, CALDAM 2024, Bhilai, India, February 15-17, 2024, Proceedings, in: Lecture Notes in Computer Science, Vol. 14508, Springer, 2024, pp. 29–43.

[11] F. Foucaud, C. Marcille, Z.M. Myint, R. Sandeep, S. Sen, S. Taruni, Bounds and extremal graphs for monitoring edge-geodetic sets in graphs, Discrete Appl. Math. 366 (2025) 106–119, http://dx.doi.org/10.1016/j.dam.2024.12.032.

[12] F. Foucaud, C. Marcille, R.B. Sandeep, S. Sen, S. Taruni, Algorithms and complexity for monitoring edge-geodetic sets in graphs, 2024, CoRR, abs/2409.19067.

[13] F. Foucaud, K. Narayanan, L.R. Sulochana, Monitoring edge-geodetic sets in graphs, in: A. Bagchi, R. Muthu (Eds.), Algorithms and Discrete Applied Mathematics - 9th International Conference, CALDAM 2023, Gandhinagar, India, February 9-11, 2023, Proceedings, in: Lecture Notes in Computer Science, Vol. 13947, Springer, 2023, pp. 245–256, http://dx.doi.org/10.1007/978-3-031-25211-2_19.

[14] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, vol. 57, Elsevier, 2004.

[15] P.L. Hammer, F. Maffray, Completely separable graphs, Discrete Appl. Math. 27 (1–2) (1990) 85–99.

[16] F. Harary, E. Loukakis, C. Tsouros, The geodetic number of a graph, Math. Comput. Modelling 17 (11) (1993) 89–95.

[17] J. Haslegrave, Monitoring edge-geodetic sets: Hardness and graph products, Discrete Appl. Math. 340 (2023) 79–84.

[18] P. Heggernes, P. van 't Hof, D. Lokshtanov, J. Nederlof, Computing the cutwidth of bipartite permutation graphs in linear time, SIAM J. Discret. Math. 26 (3) (2012) 1008–1021.

[19] ISGCI, Information system on graph classes and their inclusions, 2025, https://www.graphclasses.org/. (Accessed 08 June 2025).

[20] B. Jamison, S. Olariu, A tree representation for P_4-sparse graphs, Discrete Appl. Math. 35 (2) (1992) 115–129, http://dx.doi.org/10.1016/0166-218X(92)90036-A.

[21] T.-H. Lai, S.-S. Wei, Bipartite permutation graphs with application to the minimum buffer size problem, Discrete Appl. Math. 74 (1) (1997) 33–55.

[22] M. Lalou, H. Kheddouci, A polynomial-time algorithm for finding critical nodes in bipartite permutation graphs, Optim. Lett. 13 (6) (2019) 1345–1364, http://dx.doi.org/10.1007/S11590-018-1371-6.

[23] C. Lee, Exploring clique transversal variants on distance-hereditary graphs: Computational insights and algorithmic approaches, Algorithms 17 (8) (2024) 359, http://dx.doi.org/10.3390/A17080359.

[24] X. Li, W. Li, A. Tan, M. He, W. Chen, Monitoring edge-geodetic numbers of mycielskian graph classes, J. Interconnect. Networks 25 (2) (2025) 2450010:1–2450010:16, http://dx.doi.org/10.1142/S0219265924500105.

[25] R. Ma, Z. Ji, Y. Yao, Y. Lei, Monitoring-edge-geodetic numbers of radix triangular mesh and Sierpiński graphs, Int. J. Parallel Emergent Distrib. Syst. 39 (3) (2024) 353–361.

[26] T. Mu, C. Lin, Optimal algorithm for paired-domination in distance-hereditary graphs, 2024, CoRR, abs/2411.19476.

[27] S. Paul, D. Pradhan, S. Verma, Vertex-edge domination in interval and bipartite permutation graphs, Discuss. Math. Graph Theory 43 (4) (2023) 947–963, http://dx.doi.org/10.7151/DMGT.2411.

[28] K. Paul, A. Sharma, A. Pandey, Exploring algorithmic solutions for the independent roman domination problem in graphs, Discrete Appl. Math. 364 (2025) 143–152, http://dx.doi.org/10.1016/J.DAM.2024.12.017.

[29] G. Rong, W. Li, J. Wang, Y. Yang, Cycle extendability of Hamiltonian strongly chordal graphs, SIAM J. Discret. Math. 35 (3) (2021) 2115–2128, http://dx.doi.org/10.1137/20M1369920.

[30] J.P. Spinrad, A. Brandstädt, L. Stewart, Bipartite permutation graphs, Discrete Appl. Math. 18 (3) (1987) 279–292.

[31] A. Tan, W. Li, X. Wang, X. Li, Monitoring edge-geodetic numbers of convex polytopes and four networks, Int. J. Parallel Emergent Distrib. Syst. 38 (4) (2023) 301–312.

[32] V. Tripathi, A. Pandey, A. Maheshwari, A linear-time algorithm for semitotal domination in strongly chordal graphs, Discrete Appl. Math. 338 (2023) 77–88, http://dx.doi.org/10.1016/J.DAM.2023.05.030.

[33] X. Xu, C. Yang, G. Bao, A. Zhang, X. Shao, Monitoring-edge-geodetic sets in product networks, Int. J. Parallel Emergent Distrib. Syst. 39 (2) (2024) 264–277.

[34] Y. Zhang, F. Wang, C. Yang, Monitoring edge-geodetic numbers of some networks, J. Interconnect. Networks 25 (1) (2026) 2450001:1–2450001:14, http://dx.doi.org/10.1142/S0219265924500014.