

New Results on Vertices that Belong to Every Minimum Locating-Dominating Code

Ville Junnila

Tero Laihonen

Havu Miikonen

Department of Mathematics and Statistics, University of Turku, Turku, Finland

revisions 3rd Sep. 2025, 17th Mar. 2026; accepted 19th Mar. 2026.

Locating-dominating codes have been studied widely since their introduction in the 1980s by Slater and Rall. In this paper, we concentrate on vertices that must belong to all minimum locating-dominating codes in a graph. We call them *min-forced vertices*. We show that the number of min-forced vertices in a connected nontrivial graph of order n is bounded above by $\frac{2}{3}(n - \gamma^{LD}(G))$, where $\gamma^{LD}(G)$ denotes the cardinality of a minimum locating-dominating code. This implies that the maximum ratio between the number of min-forced vertices and the order of a connected nontrivial graph is at most $\frac{2}{5}$. Moreover, both of these bounds can be attained. In particular, the ratio $\frac{2}{5}$ is obtained by paths of order $5m$ having a unique minimum locating-dominating code of size $2m$. Furthermore, as a natural extension, we determine the number of different minimum locating-dominating codes in paths of all orders. In addition, we show that deciding whether a vertex is min-forced is co-NP-hard.

Keywords: locating-dominating code, forced vertex, algorithmic complexity, number of different codes, characterization

1 Introduction

Let $G = (V(G), E(G))$ be a simple, finite and undirected graph. We often denote the order of a graph $|V(G)|$ by n and the set of vertices $V(G) = \{v_1, v_2, \dots, v_n\}$. A graph is *nontrivial* if $n \geq 2$. A nonempty subset $S \subseteq V(G)$ is called a *code*. Elements of a code are called *codewords*. An edge between $u, v \in V(G)$ is denoted by $\{u, v\} = uv$. We denote by $N(v)$ the *open neighbourhood* of a vertex v which is defined as $N(v) = \{u \in V \mid vu \in E(G)\}$. The *closed neighbourhood* of a vertex $v \in V$ is $N[v] = N(v) \cup \{v\}$. If we wish to emphasize the underlying graph, we denote $N_G(v)$ and $N_G[v]$. Distinct vertices u and v are called *closed twins* if $N[u] = N[v]$, and *open twins* if $N(u) = N(v)$. Furthermore, vertices are *twins* if they are either closed or open twins. By the notation $G - S = G[V(G) \setminus S]$ (respectively, $G - v$ for a single vertex v) we mean the graph induced by the vertices of G not including S (resp. v). We use the following shorthand notation: $S[u \leftarrow v] = (S \setminus \{u\}) \cup \{v\}$ for the code where the codeword u is swapped for v . For a subset $E' \subseteq E(G)$, we also use the notation $G[E']$ to denote the graph induced by the edge set E' , that is, the graph with the edges of E' and their endpoints.

Definition 1.1. Let G be a graph and let $S \subseteq V(G)$ be a code. The *identifying set*, or *I-set*, of a vertex v is the set of codewords in the closed neighbourhood of v . Denote

$$I_G(S; v) = S \cap N_G[v].$$

We may omit the graph or the code in the notation if they are clear from the context, that is, $I_G(S; v) = I_G(v) = I(S; v) = I(v)$.

Definition 1.2. A code $S \subseteq V(G)$ is a *locating-dominating code*, or *LD-code*, if for all distinct non-codewords $x, y \in V(G) \setminus S$ their I -sets are nonempty and $I(x) \neq I(y)$. The cardinality of the minimum locating-dominating codes in a graph G is denoted by $\gamma^{LD}(G)$ and it is called the *locating-dominating number* of G .

Slater and Rall originally introduced locating-dominating codes in the 1980s, see for example, [10, 11, 12]. A lot of research has been done (see the numerous articles in [9]) considering the minimum possible cardinality of a locating-dominating code in graphs. In this paper, we continue this work. In particular, we are interested in certain critical vertices of a graph, which belong to all or no minimum locating-dominating codes.

Definition 1.3. A *minimum-forced vertex*, or *min-forced vertex*, is a vertex $v \in V(G)$ such that it is in every minimum locating-dominating code in G . In other words, $v \in \bigcap_{S \text{ is a minimum LD-code}} S$.

Similarly to the min-forced vertices, we define the following related concept.

Definition 1.4. A *minimum-void vertex*, or *min-void vertex*, is a vertex $v \in V(G)$ such that it is in no minimum locating-dominating code in G . In other words, $v \notin \bigcup_{S \text{ is a minimum LD-code}} S$.

These vertices have been considered in the case of *trees*, for example, in Blidia and Lounes [1]. Similar concepts for more general graphs have been considered in other contexts as well, see, for example, [2] for stable sets and [6] for metric bases. Notice also the resemblance of the min-forced vertices to the core vertices of a dominating set in [3].

In this paper, we first show a characterization of min-forced vertices that reduces the problem to LD-codes in smaller subgraphs. In Section 3, we provide, for example, the maximum ratio between the number of min-forced vertices and the order of a graph. We show that the maximum ratio is $\frac{2}{5}$. The ratio $\frac{2}{5}$ is attained by paths of order $5k$ having a unique locating-dominating code of size $2k$. In Section 4, we further determine the exact number of different minimum locating-dominating codes in paths of all orders. In the final section, we also show that it is computationally challenging to decide whether or not a vertex is min-forced in a graph. Notice that algorithmic complexity in the case where all the codewords are min-forced, *i.e.*, the LD-code is unique, is discussed in [8].

2 A characterization of min-forced vertices

In the following theorem, we give a characterization of a min-forced vertex.

Theorem 2.1. Let G be a graph. A vertex $v \in V(G)$ is min-forced if and only if either

- (i) v is isolated,
- (ii) $\gamma^{LD}(G - v) > \gamma^{LD}(G)$, or
- (iii) $\gamma^{LD}(G - v) = \gamma^{LD}(G)$, and for the graph $G - v$ there exists no minimum LD-code S such that $I_G(S; v) \neq \emptyset$ and $I_G(S; v) \neq I_G(S; w)$ for all $w \in V(G) \setminus (S \cup \{v\})$.

Proof: First, we assume that one of the conditions (i)–(iii) holds and show that then v is min-forced.

- (i) If v is isolated, then it is min-forced.
- (ii) If the graph $G - v$ cannot be locating-dominated using $\gamma^{LD}(G)$ vertices, then clearly the graph G cannot be locating-dominated using $\gamma^{LD}(G)$ vertices without using the vertex v , therefore v is min-forced in G .
- (iii) Assume that for all minimum LD-codes S in the graph $G - v$ we have either $I_G(S; v) = \emptyset$ or $I_G(S; v) = I_G(S; w)$ for some $w \in V(G) \setminus (S \cup \{v\})$. If there is an LD-code S' in G of order $\gamma^{LD}(G)$ such that $v \notin S'$, then S' is also a minimum LD-code in $G - v$. This is a contradiction: we assumed that no minimum LD-code in $G - v$ both distinguishes and dominates v . Therefore, the graph G cannot have minimum LD-codes S' that do not contain v , in other words, v is min-forced in G .

Next, we assume that v is min-forced and we show that one of the conditions (i)–(iii) follows. When the vertex v is removed from G , the locating-dominating number of the graph may increase, decrease or remain the same. We consider each case separately.

- (a) Suppose first that $\gamma^{LD}(G - v) < \gamma^{LD}(G)$. We will show that the vertex v is isolated, meeting the condition (i).

Let S be an LD-code in $G - v$ such that $|S| = \gamma^{LD}(G) - 1$. Notice that such a code always exists. If the vertex v has no neighbours in S (in other words, $I_G(S; v) = \emptyset$), then there are two possible cases:

- (1) If $N_G(v) = \emptyset$, then v is isolated.
- (2) There exists a vertex $u \in N_G(v)$. We know that $u \notin S$, because the I -set of v under S is empty. The set $S \cup \{u\}$ is an LD-code in G , because the I -sets of all vertices not contained in $S \cup \{u\}$ are distinct and nonempty: no vertex $w \in V(G) \setminus \{v\}$ has $I(S \cup \{u\}; w) = \{u\}$, because then $I(S; w) = \emptyset$, a contradiction. The I -sets that are distinct under the code S are distinct under $S \cup \{u\}$. This contradicts the fact that v is min-forced.

If the vertex v has neighbours in S , then $I_G(S; v) \neq \emptyset$. Furthermore, $I_G(S; v) = I_G(S; w)$ for at least one vertex $w \in V(G) \setminus (S \cup \{v\})$, since otherwise S would be an LD-code in G , leading to a contradiction as $|S| < \gamma^{LD}(G)$. If there were two or more such vertices with the same I -set, then S would not be locating-dominating in $G - v$ (a contradiction). Hence, we may assume that there is exactly one such vertex w . Now, $S \cup \{w\}$ is a minimum LD-code in G . Therefore, v is not min-forced, which is a contradiction.

- (b) If $\gamma^{LD}(G - v) > \gamma^{LD}(G)$, the condition (ii) holds.

- (c) Assume finally that $\gamma^{LD}(G - v) = \gamma^{LD}(G)$.

If there exists a minimum LD-code S for $G - v$ such that $I_G(S; v) \neq \emptyset$ and $I_G(S; v) \neq I_G(S; w)$ for all $w \in V(G) \setminus (S \cup \{v\})$, then S is also an LD-code in G . The vertex v does not belong to all minimum LD-codes in G , in particular, v does not belong to S . This contradicts the assumption that v is min-forced. \square

In the next example, we demonstrate how Theorem 2.1 can be used to show that a vertex is min-forced.

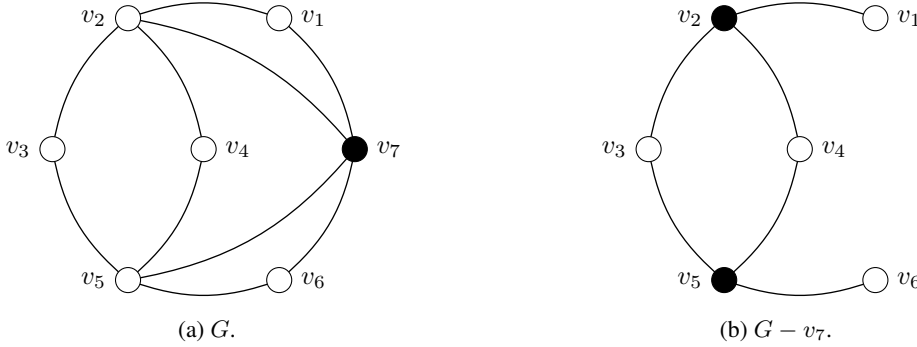


Figure 1: The black vertices are shown to be min-forced in their respective graphs.

Example 2.2. Consider the graph G illustrated in Figure 1(a). The location-domination number of G is 3; the set $\{v_2, v_3, v_7\}$ is an LD-code in G , and two codewords are not enough to distinguish the remaining 5 vertices, since only three nonempty I -sets can be formed using two codewords. We claim that the vertex v_7 is min-forced in G . Let us examine the graph $G - v_7$, illustrated in Figure 1(b), and its minimum LD-codes.

The vertex v_2 is min-forced in $G - v_7$ by Theorem 2.1(ii). Indeed, the graph $G - \{v_7, v_2\}$ is composed of an isolated vertex and the star graph $K_{1,3}$, thus $\gamma^{LD}(G - \{v_7, v_2\}) = 1 + 3 > 3$. By the same argument, the vertex v_5 is min-forced in $G - v_7$ as well. Therefore, all minimum LD-codes S in $G - v_7$ must contain both v_2 and v_5 .

The vertices v_3 and v_4 are twins, so to distinguish them, at least one of them must be in each LD-code in $G - v_7$. Together with the two min-forced vertices, this gives us $\gamma^{LD}(G - v_7) \geq 3$. It is easy to verify that the sets $\{v_2, v_3, v_5\}$ and $\{v_2, v_4, v_5\}$ are LD-codes of order 3 in $G - v_7$, and by the reasoning above, they are the only minimum LD-codes in $G - v_7$. Additionally, we see that $\gamma^{LD}(G - v_7) = \gamma^{LD}(G)$, and to show that v_7 is min-forced in G , we use Theorem 2.1(iii).

Now we consider the minimum LD-codes of $G - v_7$ in the graph G . For the code $S_1 = \{v_2, v_3, v_5\}$, we observe that $I_G(S_1; v_7) = \{v_2, v_5\} = I_G(S_1; v_4)$. Similarly, for $S_2 = \{v_2, v_4, v_5\}$, we see that $I_G(S_2; v_7) = \{v_2, v_5\} = I_G(S_2; v_3)$. We conclude that there are no minimum LD-codes in $G - v_7$ that distinguish the vertex v_7 in G ; the condition (iii) of Theorem 2.1 is met and therefore, v_7 is min-forced in G .

As we saw above, determining whether a vertex $v \in V(G)$ is min-forced, can involve the computation of the location-domination number of a graph of order $|V(G)| - 1$, which can be tedious in general. In Section 5, we prove that determining whether a vertex is min-forced is actually a co-NP-hard problem. Therefore, a simple method regarding a min-forced vertex *in general* can be hard to find. However, for *trees*, a characterization of min-forced vertices and an efficient algorithm for determining them is given in Lemma 3 and Theorem 3 of [1].

3 A tight bound for the maximum number of min-forced vertices

A natural question regarding the min-forced and min-void vertices is the following. What is the maximum ratio of such vertices in a graph with respect to the order n ? For min-void vertices this is easily answered.

Indeed, for any $h \geq 2$, take the graph G_h of order $n = 2^h - 1 + h$, where h vertices form an independent set and the rest of the vertices have the $2^h - 1$ different and nonempty subsets of the independent set as their neighbourhoods. It is well known (and also easy to check) that the independent set is a minimum LD-code giving $\gamma^{LD}(G_h) = h$. By [8, Lemma 7], the LD-code is unique, and hence, the graph has as many min-void vertices as there are non-codewords (that is, the maximum amount $n - \gamma^{LD}(G_h) = 2^h - 1$). Notice that here $\gamma^{LD}(G_h)$ is as small as possible with respect to n since for any graph it holds that $n \leq 2^{|S|} - 1 + |S|$ if S is a locating-dominating code. Thus, the maximum number k of min-void vertices satisfies

$$k = \frac{2^h - 1}{2^h - 1 + h} n,$$

where the ratio k/n approaches 1 as h grows.

However, the question for min-forced vertices is a harder problem. This question will be answered in Corollary 3.14. To that end, inspired by [7] (and to some extent also by [6] for resolving sets), we introduce the concept of the colour graph associated with a locating-dominating code. Our colour graph is different in essential ways from the so-called S -associated graph defined in [7] (and the colour graph of [6]), for example, it contains the codewords (as vertices) as well as their edges between non-codewords and also has a crucial auxiliary vertex, which is not a vertex in the original graph. The extension of the S -associated graph of [7] to our colour graph is needed in order to prove the optimal result in Theorem 3.9.

Definition 3.1. Let G be a graph and let $S \subseteq V(G)$ be a locating-dominating code in G . The vertices of G are denoted $V(G) = \{v_1, \dots, v_n\}$. We define the *colour graph* G_S as follows: $V(G_S) = V(G) \cup \{v_{-1}\}$ and

$$E(G_S) = \left\{ \{x, y\} \in \binom{V(G_S)}{2} \mid u \in S, x, y \notin (S \setminus \{u\}) \text{ and } I_G(S \setminus \{u\}; x) = I_G(S \setminus \{u\}; y) \right\},$$

where we interpret that $I_G(S'; v_{-1}) = \emptyset$ for all codes $S' \subseteq V(G)$. We call vertices in $V(G)$ *true vertices* and the vertex v_{-1} an *auxiliary vertex*. Associating a colour with each codeword $u \in S$, we assign the colour u to the edge $xy \in E(G_S)$ if $I_G(S \setminus \{u\}; x) = I_G(S \setminus \{u\}; y)$.

An example of a colour graph is given in Figure 2. Notice that G_S is a simple graph, not a multigraph, and that each edge has exactly one colour. The idea behind the colour graph is that if there is an edge $xy \in E(G_S)$ with colour u , then u is the only codeword that distinguishes x and y . If either x or y is the auxiliary vertex v_{-1} , say, $y = v_{-1}$, then u is the only codeword that dominates x . For edges $ux \in E(G_S)$, where $u \in S$, the interpretation is that if the vertex u was removed from S , then the I -sets of u and x under the code $S \setminus \{u\}$ would be equal. Finally, the edge $uv_{-1} \in E(G_S)$ means that u has no codeword neighbours in G . These observations immediately imply the following result.

Lemma 3.2. Let G be a graph and let S be a minimal locating-dominating code in G . Then for each codeword u there is an edge in G_S with the colour u .

Despite the differences between the colour graph given here and the graphs considered in [7] and [6], they share many similar properties like the ones stated in Proposition 5 of [7] and Lemma 25 of [6], as we shall see in the next lemma.

Lemma 3.3. Let G be a graph and let S be a locating-dominating code in G .

- (i) There are no edges $uv \in E(G_S)$ such that $u, v \in S$.

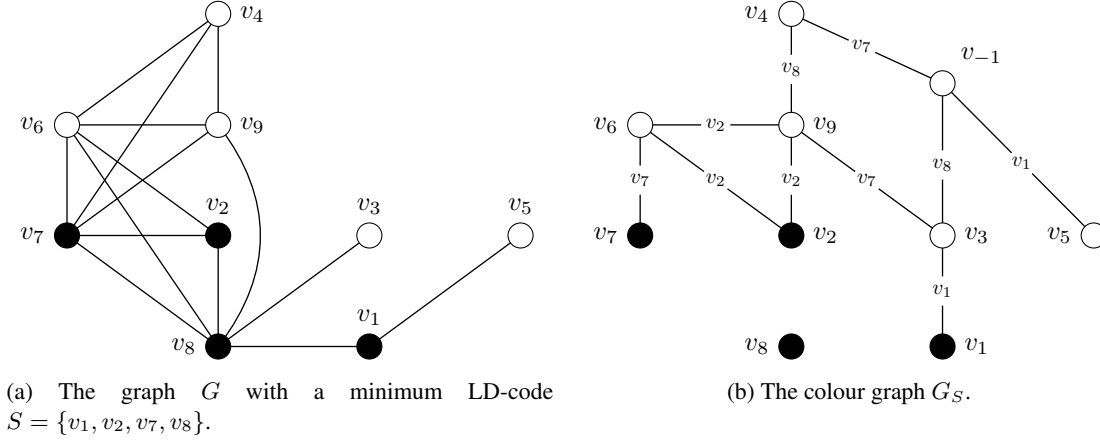


Figure 2: The vertex v_8 is min-forced as can be verified using Theorem 2.1(ii).

- (ii) Edges uv of G_S with $u \in S$ have the colour u .
- (iii) Two edges of G_S between non-codewords that share an endpoint have distinct colours.
- (iv) If there is an edge $xy \in E(G_S)$ with the colour u , where $x, y \notin S$, then exactly one of x and y is a neighbour of u in G . This holds regardless of whether x and y are both true vertices or not.
- (v) Let u be a codeword and $x, y \in V(G_S) \setminus S$. If there exists two of the three edges ux , uy and xy in $E(G_S)$ with the colour u , then the remaining edge exists and has the colour u .
- (vi) There can be at most two edges incident with $u \in S$ in G_S .
- (vii) Every cycle in $G_S - S$ has an even number of edges with the colour u for each u appearing on the cycle. The graph $G_S - S$ is bipartite.
- (viii) Let $W = w_1w_2 \cdots w_k$ be a walk with no repeated edges in $G_S - S$. If W has an even number of edges with the colour u for each u appearing in the walk, then it is a closed walk.

Proof: (i) This claim trivially follows from the definition.

(ii) This follows from the definition, because for any other codeword $w \in S$, the condition $u, v \notin (S \setminus \{w\})$ does not hold.

(iii)–(iv) Assume that there is an edge $xy \in E(G_S)$ with the colour u , where $x, y \notin S$. Then the I -sets of x and y differ only by u . Without loss of generality, denote $I(x) = X \cup \{u\}$ and $I(y) = X$, where $u \notin X$ and $X \subseteq S$. By the definition of I -sets, $x \in N_G(u)$ and $y \notin N_G(u)$ if y is a true vertex. If $y = v_{-1}$, then $X = \emptyset$. In that case $x \in N_G(u)$. This proves the claim (iv).

Suppose there is a vertex $z \notin S$ with a u -coloured edge zx in G_S . Its I -set differs from that of x only by the vertex u , that is, $I(z) = X = I(y)$, a contradiction with S being a locating-dominating code, as y and z are not codewords. If $y = v_{-1}$, then $I(z) = \emptyset$, also a contradiction.

Similarly, if there is an edge $zy \in E(G_S)$ with the colour u , then $I(z) = I(y) \cup \{u\} = I(x)$, a contradiction. This proves the claim (iii).

(v) Any two of the u -coloured edges ux , uy and xy in $E(G_S)$ imply two of the following equalities:

$$\begin{aligned} I(S \setminus \{u\}; u) &= I(S \setminus \{u\}; x) \\ I(S \setminus \{u\}; u) &= I(S \setminus \{u\}; y) \\ I(S \setminus \{u\}; x) &= I(S \setminus \{u\}; y), \end{aligned}$$

where $I(S \setminus \{u\}; x) = \emptyset$ or $I(S \setminus \{u\}; y) = \emptyset$ if $x = v_{-1}$ or $y = v_{-1}$, respectively. The third equality follows from transitivity and in turn, implies the existence of the corresponding edge in G_S with the colour u .

(vi) Suppose there are three edges ux , uy and uz in G_S . Due to (i), we can assume that $x, y, z \notin S$. Moreover, by (ii), each edge has the colour u . By (v), there are edges xy and xz in $E(G_S)$ with the colour u , contradicting (iii).

(vii) Let $C = w_1w_2 \cdots w_kw_1$ be a cycle in $G_S - S$. Each edge w_iw_{i+1} in C corresponds to the removal (or addition) of a codeword u from (or to) $I(w_i)$ to get $I(w_{i+1})$. Each such removal (addition) has to be undone by adding (removing) the vertex u somewhere along the cycle to get back to $I(w_i)$. Thus, each colour u appears on the cycle C an even number of times, and by the well-known result by König, $G_S - S$ is bipartite.

(viii) Each appearance of a colour u along the walk $W = w_1w_2 \cdots w_k$ adds or removes the codeword u from the I -set of a vertex on the walk. Starting from $I(w_1)$, the additions and removals of codewords cancel each other out and we get $I(w_k) = I(w_1)$. If $I(w_1) = I(w_k)$, then $w_1 = w_k$ by S being a locating-dominating code. Thus, W is a closed walk. \square

In the following definition, we introduce a weaker version of min-forced vertices in order to show a stronger result in Lemma 3.5.

Definition 3.4. Let G be a graph and let S be a locating-dominating code in G . The vertex $v \in S$ is *non-swappable* with respect to S , if $S[v \leftarrow u]$ is not a locating-dominating code for any $u \neq v$.

Lemma 3.5. Let G be a connected nontrivial graph and let S be a minimal locating-dominating code in G . If $v \in S$ is non-swappable with respect to S , then there are at least 2 edges with the colour v in the graph $G_S - S$.

Proof: We shall show that a codeword u with fewer than two edges of colour u in $G_S - S$ can be swapped for another vertex while maintaining the locating-dominating properties of the code. We analyse cases based on how many u -coloured edges there are in the entire colour graph G_S .

- (0) If there are no edges in G_S with the colour u , then $S \setminus \{u\}$ is an LD-code in G , contradicting the minimality of S .
- (1) Suppose then that there exists exactly one edge with the colour u in G_S . If the only edge with the colour u in G_S is of type xy , where $x, y \notin S$ and $x \in N_G(u)$ (by Lemma 3.3(iv)), then $S[u \leftarrow x]$ is an LD-code. Indeed, the only pair of vertices that can violate the location-domination property in $S \setminus \{u\}$ is the pair x and y . Therefore, the code $S[u \leftarrow x]$ is locating-dominating since x is now a codeword. This holds regardless of whether y is the auxiliary vertex or a true vertex in G . It follows that u is *not* non-swappable.

If the only edge is of type ux , where $x \notin S$ and $x \neq v_{-1}$, then the set $S[u \leftarrow x]$ is an LD-code. If $x = v_{-1}$, then let w be any vertex in $N_G(u)$. (Indeed, such a vertex w exists because there are no

isolated vertices in G , and none of the neighbours of u are in S because v_{-1} is a neighbour of u in G_S .) The set $S[u \leftarrow w]$ is an LD-code and thus, the vertex u is not non-swappable.

- (2) Let there be exactly two edges with the colour u in G_S . If all four endpoints of the two edges are in $G_S - S$, we are done; the vertex u may or may not be non-swappable. Now, let at least one of the edges be incident with a vertex in S , and by Lemma 3.3(ii), the codeword in S must be u .

If there are two edges ux and uy (or ux and xy) with the colour u , then by Lemma 3.3(v) there is a third edge xy (respectively, uy) with the colour u , contradicting the assumption that there are exactly two such edges.

Next, we consider the case where the edges are of type xy and uz , where $x, y, z \notin S$ and $z \notin \{x, y\}$. Let one of x or y be the auxiliary vertex, say, $y = v_{-1}$. Now $I(S; x) = \{u\}$, and therefore $x \in N_G(u)$. If $x \notin N_G(z)$, then the code $S[u \leftarrow x]$ is still an LD-code, because z is dominated by codewords other than u (there is no edge $zv_{-1} = zy$ with the colour u in G_S) and x distinguishes z and u . On the other hand, if $x \in N_G(z)$, then $S[u \leftarrow z]$ is an LD-code. Now $I(S[u \leftarrow z]; x) = \{z\}$, which is a unique I -set with respect to $S[u \leftarrow z]$. The vertex u is dominated by some codeword (there is no edge uv_{-1} in G_S), and finally, u is distinguished (notice that z is now a codeword).

Now we may assume that x and y are true vertices. By Lemma 3.3(iv), we may assume without loss of generality that $x \in N_G(u)$ and $y \notin N_G(u)$. Let us first assume that $z \neq v_{-1}$. Since there are exactly two edges with the colour u in G_S , we know that all the pairs of vertices except $\{x, y\}$ and $\{u, z\}$ are distinguished and all vertices are dominated by codewords in $S \setminus \{u\}$. There are three different cases depending on whether the vertex z is adjacent to x or y in G :

- (a) If $|N_G(z) \cap \{x, y\}| = 1$, then $S[u \leftarrow z]$ is an LD-code. Indeed, now the codeword z distinguishes the pair $\{x, y\}$, and the pair $\{u, z\}$ does not need to be considered since z is a codeword in the LD-code $S[u \leftarrow z]$.
- (b) If $N_G(z) \cap \{x, y\} = \{x, y\}$, then $S[u \leftarrow y]$ is an LD-code. Now the codeword y distinguishes the pair $\{u, z\}$ and the pair $\{x, y\}$ does not need to be considered as $y \in S[u \leftarrow y]$.
- (c) If $N_G(z) \cap \{x, y\} = \emptyset$, then $S[u \leftarrow x]$ is an LD-code. Now the codeword x distinguishes the pair $\{u, z\}$ and the pair $\{x, y\}$ does not need to be considered as $x \in S[u \leftarrow x]$.

If $z = v_{-1}$, then $S[u \leftarrow x]$ is an LD-code.

- (3) Assume finally that there are at least three edges of colour u . By Lemma 3.3(ii), if a u -coloured edge is incident with a codeword, the codeword must be u . According to Lemma 3.3(vi), there cannot be three edges ux , uy and uz . Thus, there must be at least one u -coloured edge in $G_S - S$. If two u -coloured edges have both their endpoints in $G_S - S$, we are done. Hence, we may assume that there is exactly one such edge. There are at least two other edges with the colour u , so they must be ux and uy for some $x, y \notin S$. By Lemma 3.3(v), the edge xy is in $E(G_S)$ and it has the colour u . Therefore, the only u -coloured edge in $G_S - S$ is the edge xy . Lemma 3.3(iv) lets us assume that $x \in N_G(u)$. If $xy \in E(G)$, then $S[u \leftarrow y]$ is an LD-code. Otherwise $S[u \leftarrow x]$ is an LD-code, regardless of whether y is a true vertex or not. In any case, the vertex u is not non-swappable.

Any other choice of three or more edges with the colour u results in at least two edges with the colour u between non-codewords in G_S .

In conclusion, codewords with fewer than 2 edges in $G_S - S$ can be swapped for another vertex to get an LD-code, therefore, non-swappable vertices in a code S have at least 2 edges in their colour in $G_S - S$. \square

The converse of the claim in the previous lemma does not hold in general as will be seen next. There are graphs G with locating-dominating codes S such that there are “swappable vertices” that have at least two colour edges between non-codewords in G_S . For example, the colour graph in Figure 2(b) has two edges with the colour v_7 between non-codewords, but the codeword v_7 is not non-swappable. Indeed, the vertices v_6 and v_7 can be freely swapped in any LD-code containing one but not the other as they are twins, that is, they have the same closed neighbourhood in G .

Corollary 3.6. Let G be a connected nontrivial graph and let S be a minimum locating-dominating code in G . If $v \in S$ is min-forced, then there are at least 2 edges with the colour v in the graph $G_S - S$.

Proof: Minimum LD-codes are minimal. Min-forced vertices are non-swappable with respect to every minimum LD-code in G . Therefore, by Lemma 3.5, there are at least 2 edges with the colour v in the graph $G_S - S$ for each min-forced $v \in S$. \square

Notice that our new (extended) colour graph always has at least two edges of the same colour in $G_S - S$ for every min-forced vertex. This allows us to utilise a fruitful approach discussed in [7]. Although our graph G_S is different from the one in [7], we can use, due to the similar properties in $G_S - S$ of our colour graph, the same reasoning as in Lemma 1 of [7] to prove the analogous result below in Lemma 3.7. Indeed, it is straightforward to check that the arguments to prove Lemma 1 in [7] relies on the properties (see Proposition 5 in [7]) of their colour graph that hold in our colour graph as well (due to (iii), (vii) and (viii) stated in Lemma 3.3). For the following lemma, we define that a *cactus* is a graph such that no two cycles share an edge.

Lemma 3.7. Let S be a locating-dominating code in a nontrivial graph G and let $S' \subseteq S$. Consider a subgraph H of $G_S - S$ induced by a set of edges containing exactly two edges with colour u for each $u \in S'$. Then all the connected components of H are cacti.

To illustrate the previous lemma, see Figure 3(a) for an example of a graph with four min-forced vertices. The cactus graph in Figure 3(b) is the induced subgraph of the colour graph with two edges of each colour.

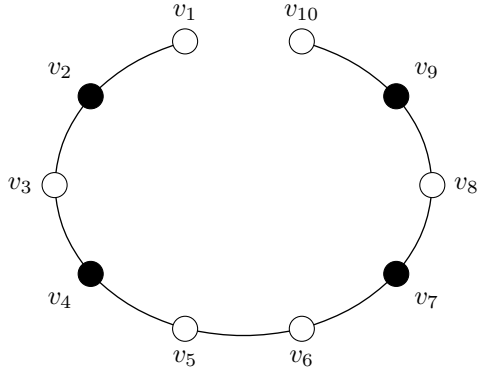
Lemma 3.8 (Lemma 2 in [7]). Let H be a bipartite graph such that $|V(H)| \geq 4$ and all its connected components are cacti. Then $|V(H)| \geq \frac{3}{4}|E(H)| + cc(H)$, where $cc(H)$ is the number of connected components in H .

The graph in Figure 3(b) reaches the bound of Lemma 3.8. The next theorem gives us upper bounds for the number of min-forced vertices. In particular, it shows that the maximum ratio between the number of min-forced vertices and the order of a nontrivial graph is $2/5$, which is also shown to be optimal in Example 3.13.

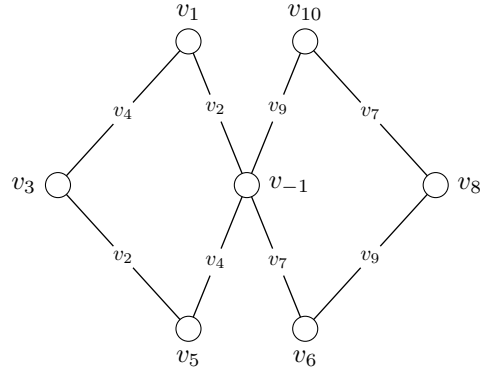
Theorem 3.9. If G is a connected nontrivial graph with $k \geq 1$ min-forced vertices, then

$$k \leq \frac{2}{3} (n - \gamma^{LD}(G))$$

and, in particular, $k \leq \frac{2}{5}n$. Furthermore, for all nontrivial graphs G with min-forced vertices, we have $\gamma^{LD}(G) \leq n - 3$.



(a) The graph P_{10} and its unique minimum LD-code $S = \{v_2, v_4, v_7, v_9\}$.



(b) $(P_{10})_S - S$.

Figure 3: The illustration using the path P_{10} .

Proof: Let $S \subseteq V(G)$ be a minimum locating-dominating code in a graph G with $k \geq 1$ min-forced vertices. Denote the non-codewords $U = V(G) \setminus S$. By Lemma 3.6, there are at least 2 edges with the colour v in $G_S[U \cup \{v_{-1}\}] = G_S - S$ for each min-forced vertex $v \in S$. Let $E' \subseteq E(G_S - S)$ be a set of edges consisting of exactly 2 edges of colour v for each min-forced v . Since $k \geq 1$ and the edges with the same colour do not share an endpoint by Lemma 3.3(iii), there are at least 4 vertices in $G_S[E']$. Having 4 vertices in $G_S[E']$ implies that $\gamma^{LD}(G) \leq (n+1) - 4 = n - 3$. By Lemmas 3.7 and 3.3(vii), the connected components of $G_S[E']$ are bipartite cacti. Therefore, by Lemma 3.8, we have $|U| + 1 \geq |V(G_S[E'])| \geq \frac{3}{4}|E(G_S[E'])| + cc(G_S[E']) \geq \frac{3}{2}k + 1$. Then, using $k \leq \gamma^{LD}(G)$ we get

$$\frac{3}{2}k \leq |U| = n - \gamma^{LD}(G) \leq n - k,$$

from which we get the bounds $k \leq \frac{2}{3}(n - \gamma^{LD}(G))$ and $k \leq \frac{2}{5}n$. \square

In what follows, we show that the upper bound $k \leq \frac{2}{3}(n - \gamma^{LD}(G))$ can be attained for all even positive integers k . For this purpose, we need some auxiliary results. We first recall the minimum cardinality of LD-codes in paths, which is known due to Slater [11].

Lemma 3.10 ([11]). We have $\gamma^{LD}(P_n) = \lceil \frac{2n}{5} \rceil$.

Furthermore, we define a *weakened* version of LD-codes in paths, which turns out to be useful also in Section 4.

Definition 3.11. A code $S \subseteq V(P_n)$ is an *LD*-code*, if for all distinct non-codewords $v_i, v_j \in V(P_n) \setminus (S \cup \{v_n\})$ their I -sets are nonempty and $I(v_i) \neq I(v_j)$. In other words, S is a code for which the locating-dominating property is satisfied for all but the last vertex v_n . The cardinality of a minimum LD*-code in P_n is denoted by $\gamma^{LD^*}(P_n)$.

The cardinality $\gamma^{LD^*}(P_n)$ is determined in the following theorem.

Theorem 3.12. We have $\gamma^{LD^*}(P_n) = \lceil \frac{2(n-1)}{5} \rceil$.

Proof: Since all LD-codes in P_{n-1} are LD*-codes in P_n , we have $\gamma^{LD^*}(P_n) \leq \gamma^{LD}(P_{n-1}) = \left\lceil \frac{2(n-1)}{5} \right\rceil$.

Let S^* be an LD*-code in P_n . If $v_n \notin S^*$, then S^* is an LD-code in P_{n-1} . If $v_n \in S^*$, then $S^*[v_n \leftarrow v_{n-1}]$ is an LD-code in P_{n-1} with the same or smaller cardinality as S^* . It follows that $\gamma^{LD^*}(P_n) \geq \gamma^{LD}(P_{n-1}) = \left\lceil \frac{2(n-1)}{5} \right\rceil$. \square

Now we are ready to present in the following example a family of graphs attaining the upper bound $k \leq \frac{2}{3}(n - \gamma^{LD}(G))$ for all even positive integers k . It should be noted that the presented graphs $G_{s,t}$ are trees and, hence, their min-forced vertices could be determined based on the characterization given by Blidia and Lounes [1]. However, as their characterization of min-forced vertices in trees is somewhat technical, we have decided to present a simple independent proof based on Theorem 2.1.

Example 3.13. Assume that s and t are positive integers. Let $G_{s,t}$ be a graph formed by a path P_s , the vertices of which are denoted by v_1, v_2, \dots, v_s , and pendant vertices u_1, u_2, \dots, u_t , which are attached to v_s ; occasionally, $G_{s,t}$ has been called the *broom graph* (due to an apparent visual resemblance). In what follows, we first give a lower bound on $\gamma^{LD}(G_{s,t})$. For this purpose, let $s \geq 4$ and S be an LD-code in $G_{s,t}$. Now we can make the following observations:

- If $v_s \notin S$, then the vertices u_1, u_2, \dots, u_t belong to S since otherwise $I(u_i) = \emptyset$ for some $i \in \{1, \dots, t\}$, and furthermore, $\{v_{s-1}, v_{s-2}\} \cap S \neq \emptyset$, since otherwise $I(v_{s-1}) = \emptyset$.
- If $v_s \in S$, then at least $t - 1$ of the (open) twins u_i belong to S , say $\{u_1, u_2, \dots, u_{t-1}\} \subseteq S$. Furthermore, if $u_t \notin S$, then $v_{s-1} \in S$ or $v_{s-2} \in S$ as otherwise $I(v_{s-1}) = I(u_t)$.

Thus, in conclusion, $|S \cap \{v_{s-2}, v_{s-1}, v_s, u_1, u_2, \dots, u_t\}| \geq t + 1$. Moreover, $S \cap \{v_1, v_2, \dots, v_{s-3}\}$ is an LD*-code in P_{s-3} . Hence, we have

$$\gamma^{LD}(G_{s,t}) \geq \gamma^{LD^*}(P_{s-3}) + t + 1. \quad (1)$$

Consider then the graph $G_{5m+4,t}$ for a nonnegative integer m . It is straightforward to verify that the set $\{v_{5\ell+2} \mid \ell = 0, 1, \dots, m\} \cup \{v_{5\ell+4} \mid \ell = 0, 1, \dots, m\} \cup \{u_1, u_2, \dots, u_{t-1}\}$ with $2m + t + 1$ vertices is locating-dominating in $G_{5m+4,t}$. Therefore, by (1) and Theorem 3.12, we have $\gamma^{LD}(G_{5m+4,t}) = 2m + t + 1$. In what follows, we show that the vertices of $\{v_{5\ell+2} \mid \ell = 0, 1, \dots, m\} \cup \{v_{5\ell+4} \mid \ell = 0, 1, \dots, m\}$ are min-forced in $G_{5m+4,t}$:

- For $\ell = 0, 1, \dots, m - 1$, the vertex $v_{5\ell+2}$ is min-forced by Theorem 2.1(ii) since

$$\begin{aligned} \gamma^{LD}(G_{5m+4,t} - v_{5\ell+2}) &= \gamma^{LD}(P_{5\ell+1}) + \gamma^{LD}(G_{5m+4-(5\ell+2),t}) \\ &\geq \left\lceil \frac{2(5\ell+1)}{5} \right\rceil + \gamma^{LD^*}(P_{5(m-\ell)-1}) + t + 1 \\ &= 2\ell + \left\lceil \frac{2(5(m-\ell)-2)}{5} \right\rceil + t + 2 \\ &= 2\ell + 2(m-\ell) + t + 2 \\ &> \gamma^{LD}(G_{5m+4,t}), \end{aligned}$$

where we apply Theorem 3.12 to compute $\gamma^{LD^*}(P_{5(m-\ell)-1})$.

- Similarly, for $\ell = 0, 1, \dots, m-1$, the vertex $v_{5\ell+4}$ can be shown to be min-forced by Theorem 2.1(ii).
- The vertex v_{5m+2} is min-forced by Theorem 2.1(ii) since

$$\begin{aligned}\gamma^{LD}(G_{5m+4,t} - v_{5m+2}) &= \gamma^{LD}(P_{5m+1}) + \gamma^{LD}(K_{1,t+1}) = \left\lceil \frac{2(5m+1)}{5} \right\rceil + t + 1 \\ &= 2m + t + 2 > \gamma^{LD}(G_{5m+4,t}).\end{aligned}$$

Indeed, it is well-known that for the star $K_{1,t+1}$, we have $\gamma^{LD}(K_{1,t+1}) = t + 1$.

- The vertex v_{5m+4} is min-forced by Theorem 2.1(ii) since $G_{5m+4,t} - v_{5m+4}$ consists of a path P_{5m+3} and t isolated vertices implying that

$$\gamma^{LD}(G_{5m+4,t} - v_{5m+4}) = \left\lceil \frac{2(5m+3)}{5} \right\rceil + t = 2m + t + 2.$$

Thus, in conclusion, we have $n = |V(G_{5m+4,t})| = 5m + t + 4$, $\gamma^{LD}(G_{5m+4,t}) = 2m + t + 1$ and there are $2m + 2$ min-forced vertices in $G_{5m+4,t}$. Hence, the graph $G_{5m+4,t}$ attains the upper bound of Theorem 3.9 as

$$\frac{2}{3}(n - \gamma^{LD}(G_{5m+4,t})) = \frac{2}{3}((5m + t + 4) - (2m + t + 1)) = 2m + 2.$$

Notice that if $t = 1$ in the previous example, then $G_{5m+4,1} = P_{5(m+1)}$ and the number of min-forced vertices is equal to $\gamma^{LD}(P_{5(m+1)}) = 2 \cdot 5(m+1)/5 = 2m + 2$. In other words, the minimum locating-dominating code in $P_{5(m+1)}$ is unique. The number of minimum LD-codes in paths is further discussed in the following section in Theorem 4.1, where another approach is given for the uniqueness of the code in $P_{5(m+1)}$. In conclusion, Theorem 3.9 and Example 3.13 are combined in the following corollary.

Corollary 3.14. If G is a connected nontrivial graph with $k \geq 1$ min-forced vertices, then

$$k \leq \frac{2}{3}(n - \gamma^{LD}(G)) \quad \text{and} \quad k \leq \frac{2}{5}n.$$

Moreover, there exist graphs for which the equalities hold. In particular, the maximum ratio between the number of min-forced vertices and the order of a connected nontrivial graph is $2/5$.

4 The number of different minimum locating-dominating codes in paths

In the previous section, we noticed that the LD-code formed by the min-forced vertices is unique in P_{5k} . A natural extension of this observation is the question of how many minimum locating-dominating codes there are in the path P_n in general. To address this question, we denote the number of minimum locating-dominating codes in P_n by $C(n)$ and the number of LD*-codes in P_n with cardinality k by $A(n, k)$. In the following theorem, we calculate the number of different minimum locating-dominating codes for all paths, that is, we determine $C(n)$ for lengths $n \geq 5$. For completeness, we mention that the following first values of $C(n)$ are straightforward to verify:

$$C(1) = 1, C(2) = 2, C(3) = 3, \text{ and } C(4) = 4.$$

Theorem 4.1. The number of minimum locating-dominating codes in the path P_{5m+r} for each $r \in \{0, \dots, 4\}$ and $m \geq 1$ are the following:

$$(M_0) \quad C(5m) = 1$$

$$(M_1) \quad C(5m + 1) = \frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 1$$

$$(M_2) \quad C(5m + 2) = m + 2$$

$$(M_3) \quad C(5m + 3) = \frac{1}{24}m^4 + \frac{11}{12}m^3 + \frac{131}{24}m^2 + \frac{103}{12}m + 3$$

$$(M_4) \quad C(5m + 4) = \frac{1}{2}m^2 + \frac{7}{2}m + 4.$$

Proof: Let S be an LD-code in P_n , where $n \geq 5$. Anticipating the recursion formula in Equation (2), the code S is classified into the following three categories based on the pattern of codewords in the final few vertices of the path:

- $v_n \in S$,
- $v_n \notin S$ and $v_{n-1}, v_{n-2} \in S$, or
- $v_n, v_{n-2} \notin S$ and $v_{n-1}, v_{n-3} \in S$.

These patterns are illustrated in Figure 4. In what follows, we discuss the minimum LD-codes in each category.

- Suppose first that $v_n \in S$. Clearly, the set $S \cap \{v_1, \dots, v_{n-1}\}$ is an LD*-code in P_{n-1} with cardinality $\gamma^{LD}(P_n) - 1$. On the other hand, if A is an LD*-code in P_{n-1} with cardinality $\gamma^{LD}(P_n) - 1$, then $A \cup \{v_n\}$ is a minimum LD-code in P_n . This is clear if $v_{n-1} \in A$, so assume that $v_{n-1} \notin A$. Now the vertex v_{n-1} is dominated and has a unique I -set due to v_n . Moreover, each A gives a different LD-code for P_n .
- If $v_n \notin S$, then necessarily $v_{n-1} \in S$. Now either $v_{n-2} \in S$ or $v_{n-2} \notin S$.
 - Let us first suppose that $v_{n-2} \in S$. Now the set $S \cap \{v_1, \dots, v_{n-3}\}$ is an LD*-code in P_{n-3} with cardinality $\gamma^{LD}(P_n) - 2$. Conversely, if A is an LD*-code in P_{n-3} with cardinality $\gamma^{LD}(P_n) - 2$, then $A \cup \{v_{n-2}, v_{n-1}\}$ is a minimum LD-code in P_n . Indeed if $v_{n-3} \in A$, then we are immediately done, and otherwise ($v_{n-3} \notin A$) the vertex v_{n-3} is dominated by v_{n-2} and its I -set contains v_{n-2} , which no other non-codeword does.
 - Suppose next that $v_{n-2} \notin S$. To distinguish the non-codewords v_n and v_{n-2} , we need to have $v_{n-3} \in S$. The set $S \cap \{v_1, \dots, v_{n-4}\}$ is an LD*-code in P_{n-4} with cardinality $\gamma^{LD}(P_n) - 2$. On the other hand, if A is an LD*-code in P_{n-4} with cardinality $\gamma^{LD}(P_n) - 2$, then $A \cup \{v_{n-3}, v_{n-1}\}$ is a minimum LD-code in P_n , since (if $v_{n-4} \notin A$) the vertex v_{n-4} is dominated by v_{n-3} and distinguished from other I -sets of v_i ($i \leq n - 5$) by v_{n-3} .

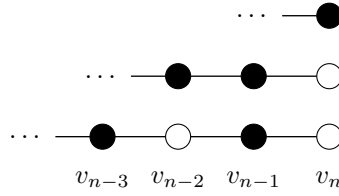


Figure 4: The different categories of LD-codes illustrated. The black and white vertices represent the codewords and non-codewords, respectively.

As seen above, the three categories described above partition the set of minimum LD-codes in P_n , and hence each LD-code is counted exactly once by counting the number of appropriate LD*-codes in shorter paths. Thus, recalling Lemma 3.10, we get the following formula for the number of minimum LD-codes in P_n (based on the patterns illustrated in Figure 4):

$$\begin{aligned}
 C(n) &= A(n-1, \lceil 2n/5 \rceil - 1) \\
 &\quad + A(n-3, \lceil 2n/5 \rceil - 2) \\
 &\quad + A(n-4, \lceil 2n/5 \rceil - 2).
 \end{aligned} \tag{2}$$

For the values of $A(n, k)$, we get the following recurrence relation:

$$\begin{aligned}
 A(n, k) &= A(n-1, k-1) \\
 &\quad + A(n-2, k-1) \\
 &\quad + A(n-4, k-2) \\
 &\quad + A(n-5, k-2).
 \end{aligned} \tag{3}$$

We now reason why the recurrence relation holds. Each LD*-code S in P_n (where $n \geq 6$) with $|S| = k$ is, as will be seen, in exactly one of the following four categories, illustrated in Figure 5:

- (i) $v_n \in S$,
- (ii) $v_n \notin S$ and $v_{n-1} \in S$,
- (iii) $v_n, v_{n-1} \notin S$ and $v_{n-2}, v_{n-3} \in S$, or
- (iv) $v_n, v_{n-1}, v_{n-3} \notin S$ and $v_{n-2}, v_{n-4} \in S$.

Next, we consider each type of LD*-code. First we notice that $S \cap \{v_1, \dots, v_{n-j}\}$ is an LD*-code of cardinality $k-1$ in P_{n-j} where $j = 1$ in the category (i) and $j = 2$ in the category (ii). Similarly, $S \cap \{v_1, \dots, v_{n-j}\}$ is an LD*-code of cardinality $k-2$ in P_{n-j} where $j = 4$ in the category (iii) and $j = 5$ in the category (iv). Next we consider different endings of LD*-codes in P_n and show that a suitable LD*-code in a shorter path provides an LD*-code in P_n .

- Let first $v_n \in S$. If A is an LD*-code with cardinality $k-1$ in P_{n-1} , then $A \cup \{v_n\}$ is an LD*-code of cardinality k in P_n as the I -set of v_{n-1} is nonempty and unique (the reasoning is needed only if $v_{n-1} \notin A$).

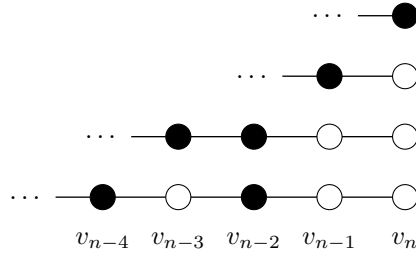


Figure 5: The four categories of LD*-codes illustrated.

- Suppose then that $v_n \notin S$ and $v_{n-1} \in S$. If A is an LD*-code with cardinality $k - 1$ in P_{n-2} , then $A \cup \{v_{n-1}\}$ is an LD*-code of cardinality k in P_n . Indeed, if $v_{n-2} \in A$, we are done, and if $v_{n-2} \notin A$, the set $I(v_{n-2})$ contains at least v_{n-1} , so it is nonempty and necessarily distinct from I -sets other than possibly $I(v_n)$, which is allowed in an LD*-code of P_n .
- If $v_n, v_{n-1} \notin S$, then we must have $v_{n-2} \in S$ in order for v_{n-1} to be dominated. (Recall that the final vertex need not be dominated in an LD*-code.) Then, in order to have distinct I -sets for the vertices $v_i \notin S$ where $i \in \{n - 3, n - 1\}$, either $v_{n-3} \in S$ or $v_{n-3} \notin S$ and $v_{n-4} \in S$.
 - Let first $v_{n-3} \in S$. If A is an LD*-code with cardinality $k - 2$ in P_{n-4} , then $A \cup \{v_{n-3}, v_{n-2}\}$ is an LD*-code of cardinality k in P_n , since (if $v_{n-4} \notin A$) the codeword v_{n-3} dominates and distinguishes v_{n-4} .
 - Finally, let $v_{n-3} \notin S$ and $v_{n-4} \in S$. If A is an LD*-code with cardinality $k - 2$ in P_{n-5} , then $A \cup \{v_{n-4}, v_{n-2}\}$ is an LD*-code of cardinality k in P_n , since (if $v_{n-5} \notin A$) the vertex v_{n-5} is dominated and distinguished by v_{n-4} .

As we see above, the four categories (i)–(iv) are the only possible patterns of codewords in an LD*-code at the end of the path P_n , and they are mutually exclusive, so each LD*-code in P_n with cardinality k is counted exactly once in the recursion formula (3).

Theorem 3.12 gives the boundary condition

$$A(n, k) = 0, \text{ when } k < \left\lceil \frac{2(n-1)}{5} \right\rceil. \quad (4)$$

We will first prove the following values of $A(n, k)$ in this order and refer back to them as needed:

$$A(5m + 1, 2m) = 1, \quad (5)$$

$$A(5m + 3, 2m + 1) = m + 2, \quad (6)$$

$$A(5m, 2m) = \frac{1}{2}m^2 + \frac{5}{2}m + 1, \quad (7)$$

$$A(5m + 2, 2m + 1) = \frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 2, \quad (8)$$

$$A(5m + 4, 2m + 2) = \frac{1}{24}m^4 + \frac{11}{12}m^3 + \frac{131}{24}m^2 + \frac{115}{12}m + 5. \quad (9)$$

In what follows, we prove Equations (5)–(9).

Equation (5): First, we calculate $A(5m + 1, 2m) = 1$. By (3),

$$\begin{aligned} A(5m + 1, 2m) &= A(5m + 1 - 1, 2m - 1) + A(5m + 1 - 2, 2m - 1) \\ &\quad + A(5m + 1 - 4, 2m - 2) + A(5m + 1 - 5, 2m - 2), \end{aligned}$$

where the first three terms are eliminated by the bound (4), because $2m - 1 < \left\lceil \frac{2(5m-1)}{5} \right\rceil$, $2m - 1 < \left\lceil \frac{2(5m-1-1)}{5} \right\rceil$, and $2m - 2 < \left\lceil \frac{2(5m-3-1)}{5} \right\rceil$. We are left with

$$A(5m + 1, 2m) = A(5(m - 1) + 1, 2(m - 1)),$$

where through iteration we get the base case $A(1, 0)$, that is, the number of LD*-codes in P_1 with no vertices. There is one such code, namely \emptyset . Therefore,

$$A(5m + 1, 2m) = A(1, 0) = 1.$$

Equation (6): Next, we calculate $A(5m + 3, 2m + 1) = m + 2$. By (3),

$$\begin{aligned} A(5m + 3, 2m + 1) &= A(5m + 3 - 1, 2m + 1 - 1) + A(5m + 3 - 2, 2m + 1 - 1) \\ &\quad + A(5m + 3 - 4, 2m + 1 - 2) + A(5m + 3 - 5, 2m + 1 - 2), \end{aligned}$$

where the first and third terms are eliminated by (4) and the value of the second term is known by Equation (5). We are left with the recurrence relation

$$A(5m + 3, 2m + 1) = 1 + A(5(m - 1) + 3, 2(m - 1) + 1),$$

where after m iterations we arrive at $A(3, 1)$, the number of LD*-codes in P_3 with one vertex. There are two such codes, namely $\{v_1\}$ and $\{v_2\}$. Thus, we obtain

$$A(5m + 3, 2m + 1) = m \cdot 1 + A(3, 1) = m + 2.$$

Equation (7): Let us prove that $A(5m, 2m) = \frac{1}{2}m^2 + \frac{5}{2}m + 1$. By (3),

$$\begin{aligned} A(5m, 2m) &= A(5m - 1, 2m - 1) + A(5m - 2, 2m - 1) \\ &\quad + A(5m - 4, 2m - 2) + A(5m - 5, 2m - 2), \end{aligned}$$

where the first term is eliminated by the bound (4) and the values of the second and third terms are known by Equations (6) and (5). We are left with the recurrence relation

$$A(5m, 2m) = (m + 1) + 1 + A(5(m - 1), 2(m - 1)),$$

where, after $m - 1$ iterations, we arrive at $A(5, 2)$, the number of LD*-codes in P_5 with two elements. There are 4 such codes: $\{v_1, v_3\}$, $\{v_1, v_4\}$, $\{v_2, v_3\}$ and $\{v_2, v_4\}$. Thus, we obtain

$$\begin{aligned} A(5m, 2m) &= (m + 1) + 1 + A(5(m - 1), 2(m - 1)) \\ &= \sum_{i=2}^m (i + 2) + A(5, 2) \\ &= \frac{1}{2}m^2 + \frac{5}{2}m + 1. \end{aligned}$$

Equation (8): Let us show that $A(5m + 2, 2m + 1) = \frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 2$. By (3),

$$\begin{aligned} A(5m + 2, 2m + 1) &= A(5m + 2 - 1, 2m + 1 - 1) + A(5m + 2 - 2, 2m + 1 - 1) \\ &\quad + A(5m + 2 - 4, 2m + 1 - 2) + A(5m + 2 - 5, 2m + 1 - 2), \end{aligned}$$

where the first three terms are known by Equations (5), (7) and (6). We get the recurrence relation

$$A(5m + 2, 2m + 1) = 1 + \left(\frac{1}{2}m^2 + \frac{5}{2}m + 1 \right) + (m + 1) + A(5(m - 1) + 2, 2(m - 1) + 1),$$

where after m iterations we get the base case $A(2, 1)$, the number of one-vertex LD*-codes in P_2 . There are two such codes, namely $\{v_1\}$ and $\{v_2\}$. Thus, we obtain

$$\begin{aligned} A(5m + 2, 2m + 1) &= \frac{1}{2}m^2 + \frac{7}{2}m + 3 + A(5(m - 1) + 2, 2(m - 1) + 1) \\ &= \sum_{i=1}^m \left(\frac{1}{2}i^2 + \frac{7}{2}i + 3 \right) + A(2, 1) \\ &= \frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 2. \end{aligned}$$

Equation (9): Let us verify that $A(5m + 4, 2m + 2) = \frac{1}{24}m^4 + \frac{11}{12}m^3 + \frac{131}{24}m^2 + \frac{115}{12}m + 5$. By (3),

$$\begin{aligned} A(5m + 4, 2m + 2) &= A(5m + 4 - 1, 2m + 2 - 1) + A(5m + 4 - 2, 2m + 2 - 1) \\ &\quad + A(5m + 4 - 4, 2m + 2 - 2) + A(5m + 4 - 5, 2m + 2 - 2), \end{aligned}$$

where the first three terms are known by Equations (6), (8) and (7). We get the recurrence relation

$$\begin{aligned} A(5m + 4, 2m + 2) &= (m + 2) + \left(\frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 2 \right) + \left(\frac{1}{2}m^2 + \frac{5}{2}m + 1 \right) \\ &\quad + A(5(m - 1) + 4, 2(m - 1) + 2) \end{aligned}$$

where after m iterations we get $A(4, 2)$, the number of two-vertex LD*-codes in P_4 . There are 5 such codes, namely all two-vertex subsets of $V(P_4)$ except $\{v_3, v_4\}$. Thus, we obtain

$$\begin{aligned} A(5m + 4, 2m + 2) &= \frac{1}{6}m^3 + \frac{5}{2}m^2 + \frac{25}{3}m + 5 + A(5(m - 1) + 4, 2(m - 1) + 2) \\ &= \sum_{i=1}^m \left(\frac{1}{6}i^3 + \frac{5}{2}i^2 + \frac{25}{3}i + 5 \right) + A(4, 2) \\ &= \frac{1}{24}m^4 + \frac{11}{12}m^3 + \frac{131}{24}m^2 + \frac{115}{12}m + 5. \end{aligned}$$

Now we have solved all the necessary values of $A(n, k)$, so let us calculate $C(n)$, the number of different minimum locating-dominating codes in P_n for all $n \geq 5$. In what follows, Cases M_0 – M_4 of the claim are shown.

Case M_0 : To warm up, we calculate $C(5m)$ with our new method (although we already know from previous section that $C(5m) = 1$). Now Equation (2) gives us

$$C(5m) = A(5m - 1, 2m - 1) + A(5m - 3, 2m - 2) + A(5m - 4, 2m - 2).$$

The first two terms are eliminated by the bound (4). We apply Equation (5) and obtain that $C(5m) = A(5m - 4, 2m - 2) = 1$.

Case M_1 : Next, we calculate $C(5m + 1)$. Equation (2) gives us

$$C(5m + 1) = A(5m, 2m) + A(5m - 2, 2m - 1) + A(5m - 3, 2m - 1).$$

Now we respectively apply Equations (7), (6) and (8), and we obtain

$$\begin{aligned} C(5m + 1) &= \left(\frac{1}{2}m^2 + \frac{5}{2}m + 1 \right) + (m - 1 + 2) + \left(\frac{1}{6}(m - 1)^3 + 2(m - 1)^2 + \frac{29}{6}(m - 1) + 2 \right) \\ &= \frac{1}{6}m^3 + 2m^2 + \frac{29}{6}m + 1. \end{aligned}$$

Case M_2 : Let us next calculate $C(5m + 2)$. Equation (2) gives us

$$C(5m + 2) = A(5m + 1, 2m) + A(5m - 1, 2m - 1) + A(5m - 2, 2m - 1).$$

After eliminating the middle term by (4) and applying Equations (5) and (6) to the other two terms, we get

$$C(5m + 2) = 1 + (m - 1 + 2) = m + 2.$$

Case M_3 : Let us calculate $C(5m + 3)$. Equation (2) gives us

$$C(5m + 3) = A(5m + 2, 2m + 1) + A(5m, 2m) + A(5m - 1, 2m).$$

By applying Equations (8), (7) and (9), we obtain

$$C(5m + 3) = \frac{1}{24}m^4 + \frac{11}{12}m^3 + \frac{131}{24}m^2 + \frac{103}{12}m + 3.$$

Case M_4 : Finally, let us compute $C(5m + 4)$. Equation (2) gives us

$$C(5m + 4) = A(5m + 3, 2m + 1) + A(5m + 1, 2m) + A(5m, 2m).$$

We apply Equations (6), (5) and (7) to get

$$C(5m + 4) = \frac{1}{2}m^2 + \frac{7}{2}m + 4. \quad \square$$

5 Computational complexity of determining min-forced and min-void vertices

In this section, we show that the problems “given a graph G and a vertex $u \in V(G)$, is u min-forced?” and “given a graph G and a vertex $u \in V(G)$, is u min-void?” are co-NP-hard problems. Similar questions have been studied in the context of resolving sets and basis forced vertices in [6].

We use the following notation for the 3-satisfiability problem (3-SAT). Let $X = \{x_1, \dots, x_n\}$ be the set of variables and $U = \{x_1, \dots, x_n, \bar{x}_1, \dots, \bar{x}_n\}$ the set of literals, where \bar{x} is the negation of x . Let $F = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_m\}$ be a 3-SAT instance over X , where each clause $\mathcal{C}_j \subseteq U$ contains exactly three literals. An assignment of truth values is a set $A \subseteq U$ where either $x_i \in A$ or $\bar{x}_i \in A$ for each $x_i \in X$. The logical interpretation of these definitions is that F corresponds to the formula $\mathcal{C}_1 \wedge \mathcal{C}_2 \wedge \dots \wedge \mathcal{C}_m$ and the clause $\mathcal{C}_j = \{u_{j,1}, u_{j,2}, u_{j,3}\} \subseteq U$ corresponds to the formula $u_{j,1} \vee u_{j,2} \vee u_{j,3}$. In a truth assignment A , the variable x_i is set as *true* if $x_i \in A$ and *false* if $\bar{x}_i \in A$. The truth assignment A satisfies the instance F if each clause \mathcal{C}_j contain at least one true literal, in other words, $\mathcal{C}_j \cap A \neq \emptyset$.

The proof presented here is inspired by the proof of Lemma 3.1 in [4].

Theorem 5.1. Let G be a graph and $u \in V(G)$.

- (i) Deciding whether u is a min-forced vertex of G is a co-NP-hard problem.
- (ii) Deciding whether u is a min-void vertex of G is a co-NP-hard problem.

Proof:

We reduce the 3-SAT problem polynomially to each of the problems studied.

Let $X = \{x_1, \dots, x_n\}$ be a set of variables and $F = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_m\}$ be a 3-SAT instance over X . For each variable $x_i \in X$, we construct a graph G_{x_i} in the following way:

$$\begin{aligned} V(G_{x_i}) &= \{x_i, \bar{x}_i, a_i, b_i\}, \\ E(G_{x_i}) &= \{x_i a_i, a_i \bar{x}_i, \bar{x}_i b_i, b_i x_i\}. \end{aligned}$$

For each clause $\mathcal{C}_j \in F$, we construct a graph $G_{\mathcal{C}_j}$ in the following way:

$$\begin{aligned} V(G_{\mathcal{C}_j}) &= \{\alpha_j, \beta_j, \gamma_j\}, \\ E(G_{\mathcal{C}_j}) &= \{\alpha_j \beta_j, \beta_j \gamma_j\}. \end{aligned}$$

We combine the variable graphs and clause graphs into $G = (V, E)$ as follows:

$$\begin{aligned} V &= \bigcup_{i=1}^n V(G_{x_i}) \cup \bigcup_{j=1}^m V(G_{\mathcal{C}_j}) \cup \{w, v\}, \\ E &= \bigcup_{i=1}^n E(G_{x_i}) \cup \bigcup_{j=1}^m E(G_{\mathcal{C}_j}) \cup \{\alpha_j u \mid u \in \mathcal{C}_j, 1 \leq j \leq m\} \cup \{w \alpha_j \mid 1 \leq j \leq m\} \cup \{wv\}. \end{aligned}$$

The graph G is illustrated in Figure 6. We see that $|V(G)| = 4n + 3m + 2$ and $|E(G)| = 4n + 6m + 1$.

Regardless of whether F is satisfiable, there is an LD-code S in G with $2n + m + 1$ vertices:

$$S = \{x_i, a_i \mid 1 \leq i \leq n\} \cup \{\beta_j \mid 1 \leq j \leq m\} \cup \{w\}.$$

The I -sets of vertices not belonging to S are nonempty and distinct:

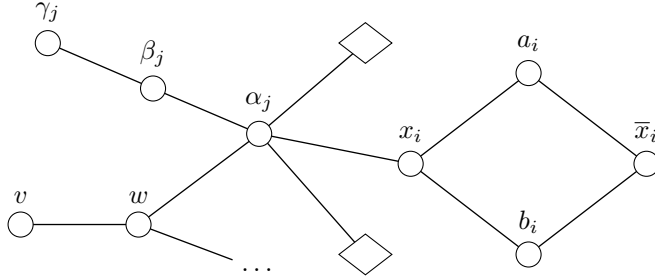


Figure 6: The vertex w is connected to all vertices α_j , represented by the ellipsis. The vertex α_j is connected to three variable gadgets, two of which are drawn as diamonds for the sake of simplicity.

- $I_G(\alpha_j) \supseteq \{\beta_j, w\}$,
- $I_G(\gamma_j) = \{\beta_j\}$,
- $I_G(\bar{x}_i) = \{a_i\}$,
- $I_G(b_i) = \{x_i\}$, and
- $I_G(v) = \{w\}$.

Therefore, $\gamma^{LD}(G) \leq 2n + m + 1$.

Next, we establish a lower bound for $\gamma^{LD}(G)$. Each clause gadget G_{e_j} necessarily contains at least one codeword to dominate γ_j : either β_j or γ_j , since $N_G[\gamma_j] = \{\gamma_j, \beta_j\}$. Each variable gadget G_{x_i} needs to contain at least two codewords: either a_i or b_i , as they are twins, and one more to dominate the other one. Either w or v is in any LD-code in G , again, since $N_G[v] = \{v, w\}$. Therefore, $\gamma^{LD}(G) = 2n + m + 1$.

Let us consider some observations.

Fact 1. No minimum LD-code in G contains the vertex α_j for any j . That is because an LD-code containing α_j contains two codewords in the clause gadget G_{e_j} , resulting in a code of at least $2n + m + 2$ codewords.

Fact 2. Exactly one of x_i and \bar{x}_i is in a minimum LD-code $S \subseteq V(G)$. It follows from Fact 1 that if $x_i, \bar{x}_i \notin S$, then $I_G(x_i), I_G(\bar{x}_i) \subseteq \{a_i, b_i\}$ in any minimum LD-code $S \subseteq V(G)$. Hence, $I_G(x_i) = I_G(\bar{x}_i)$ if $x_i, \bar{x}_i \notin S$. Including both x_i and \bar{x}_i in S would result in an LD-code of at least $2n + m + 2$ codewords. Therefore, to distinguish the vertices x_i and \bar{x}_i , exactly one of them is contained in every minimum LD-code.

Fact 3. Exactly one of w and v is in a minimum LD-code $S \subseteq V(G)$.

We claim that (i) the vertex w is min-forced if and only if the 3-SAT instance F is not satisfiable and (ii) the vertex v is min-void if and only if the 3-SAT instance F is not satisfiable. We prove these statements by contraposition.

First, we assume that F is satisfiable, and we show that w is not min-forced and v is not min-void. If F is satisfiable, then there exists a truth assignment $A \subseteq U$ such that $|\mathcal{C}_j \cap A| \geq 1$ for all clauses $\mathcal{C}_j \in F$. Now

$$S = A \cup \{a_i \mid 1 \leq i \leq n\} \cup \{\beta_j \mid 1 \leq j \leq m\} \cup \{v\}$$

is a minimum LD-code of G with $v \in S$ and $w \notin S$. The I -sets of vertices not belonging to S are nonempty and distinct:

- $I_G(\alpha_j) \supseteq \{\beta_j, u\}$, where u is (one of) the true literal(s) in \mathcal{C}_j ,
- $I_G(\gamma_j) = \{\beta_j\}$,
- if $x_i \in S$, then $I_G(\bar{x}_i) = \{a_i\}$, otherwise $\bar{x}_i \in S$ and $I_G(x_i) = \{a_i\}$,
- if $x_i \in S$, then $I_G(b_i) = \{x_i\}$, otherwise $\bar{x}_i \in S$ and $I_G(b_i) = \{\bar{x}_i\}$, and
- $I_G(w) = \{v\}$.

In this case, we see that w is not min-forced and v is not min-void.

Next, we separately assume that w is not min-forced and that v is not min-void, and in both cases it follows that F is satisfiable. Here the proofs of claims (i) and (ii) briefly diverge.

- (i) Assume that w is not min-forced, that is, there exists a minimum LD-code $S \subseteq V(G)$ such that $w \notin S$. Recall Fact 1: the vertices α_j are never in a minimum LD-code of G . Consider the I -set of the vertex α_j corresponding to the clause $\mathcal{C}_j = \{u_{j,1}, u_{j,2}, u_{j,3}\}$. If $\beta_j \in I(\alpha_j)$, then γ_j is not a codeword and $I(\gamma_j) = \{\beta_j\}$. For $I(\alpha_j)$ to be distinct, there must be $u_{j,k} \in I(\alpha_j)$ for some $k \in \{1, 2, 3\}$. If $\beta_j \notin I(\alpha_j)$, then for $I(\alpha_j)$ to be nonempty, there must be $u_{j,k} \in I(\alpha_j)$ for some $k \in \{1, 2, 3\}$.

By Fact 2, either $x_i \in S$ or $\bar{x}_i \in S$, but not both. When we set the variable x_i as *true* if $x_i \in S$ and as *false* if $\bar{x}_i \in S$, we get a valid truth assignment for the variables of X . Under this assignment, each clause \mathcal{C}_j in F contains at least one true literal, namely, the literal $u_{j,k}$ that is in the I -set of α_j . Therefore, F is satisfiable.

- (ii) Now we assume that v is not min-void, that is, there exists a minimum LD-code $S \subseteq V(G)$ such that $v \in S$. By Fact 3, $v \in S$ implies that $w \notin S$, in other words, w is not min-forced.

Just like in the proof of claim (i), it follows that F is satisfiable.

We have shown that the vertex w is min-forced (respectively, v is min-void) if and only if the 3-SAT instance F is not satisfiable. Thus, there exists a polynomial-time reduction of the complement of the 3-SAT problem to the problem of deciding whether a given vertex is min-forced (resp. min-void). Hence, the studied problems are co-NP-hard. \square

Note that the graph $G - \{v, w\}$ can be used to show that the decision problem “is there a locating-dominating code $S \subseteq V(G)$ of size at most k ?” is NP-complete (see Lemma 3.1 in [4]). The graph $G - \{v, w\}$ has fewer vertices than the graph used in the proof in [4], which in turn needs fewer vertices compared to the proof of Colbourn et al. [5].

Acknowledgements

The authors have been partially supported by Research Council of Finland grant number 338797. Havu Miiikonen has been partially supported by the Turku University Foundation. The authors would like to thank the anonymous referee for careful reading of the paper and insightful comments.

References

- [1] M. Blidia and R. Lounes. Vertices belonging to all or to no minimum locating dominating sets of trees. *Opuscula Math.*, 29(1):5–14, 2009. ISSN 1232-9274,2300-6919. doi: 10.7494/OpMath.2009.29.1.5. URL <https://doi.org/10.7494/OpMath.2009.29.1.5>.
- [2] E. Boros, M. C. Golumbic, and V. E. Levit. On the number of vertices belonging to all maximum stable sets of a graph. *Discrete Appl. Math.*, 124(1-3):17–25, 2002. ISSN 0166-218X,1872-6771. doi: 10.1016/S0166-218X(01)00327-4. URL [https://doi.org/10.1016/S0166-218X\(01\)00327-4](https://doi.org/10.1016/S0166-218X(01)00327-4). Workshop on Discrete Optimization (Piscataway, NJ, 1999).
- [3] V. Bouquet, F. Delbot, and C. Picouleau. On the vertices belonging to all, some, none minimum dominating set. *Discrete Appl. Math.*, 288:9–19, 2021. ISSN 0166-218X. doi: <https://doi.org/10.1016/j.dam.2020.08.020>. URL <https://www.sciencedirect.com/science/article/pii/S0166218X20303863>.
- [4] I. Charon, O. Hudry, and A. Lobstein. Minimizing the size of an identifying or locating-dominating code in a graph is NP-hard. *Theoret. Comput. Sci.*, 290(3):2109–2120, Jan. 2003. doi: 10.1016/s0304-3975(02)00536-4. URL [https://doi.org/10.1016/s0304-3975\(02\)00536-4](https://doi.org/10.1016/s0304-3975(02)00536-4).
- [5] C. J. Colbourn, P. J. Slater, and L. K. Stewart. Locating dominating sets in series parallel networks. *Congr. Numer.*, 56(1987):135–162, 1987.
- [6] A. Hakanen, V. Junnila, T. Laihonon, and I. G. Yero. On vertices contained in all or in no metric basis. *Discrete Appl. Math.*, 319:407–423, oct 2022. doi: 10.1016/j.dam.2021.12.004. URL <https://doi.org/10.1016/j.dam.2021.12.004>.
- [7] C. Hernando, M. Mora, and I. M. Pelayo. Locating domination in bipartite graphs and their complements. *Discrete Appl. Math.*, 263:195–203, 2019. ISSN 0166-218X,1872-6771. doi: 10.1016/j.dam.2018.09.034. URL <https://doi.org/10.1016/j.dam.2018.09.034>.
- [8] O. Hudry and A. Lobstein. Unique (optimal) solutions: Complexity results for identifying and locating–dominating codes. *Theoret. Comput. Sci.*, 767:83–102, May 2019. ISSN 0304-3975. doi: 10.1016/j.tcs.2018.09.034. URL <http://dx.doi.org/10.1016/j.tcs.2018.09.034>.
- [9] D. Jean and A. Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs. Published electronically, 2024. URL <https://dragazo.github.io/bibdom/main.pdf>. Accessed: June 5th 2024.
- [10] D. F. Rall and P. J. Slater. On location-domination numbers for certain classes of graphs. *Congr. Numer.*, 45:97–106, 1984.
- [11] P. J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987. doi: <https://doi.org/10.1002/net.3230170105>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/net.3230170105>.
- [12] P. J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci.*, 22(4):445–455, 1988.