The complexity of the identifying code problem in restricted graph classes*

Florent Foucaud

Universitat Politècnica de Catalunya, Building C3, C/ Jordi Girona 1-3, 08034 Barcelona, Spain.
florent.foucaud@gmail.com

Abstract. An identifying code is a subset of vertices of a graph such that each vertex is uniquely determined by its nonempty neighbourhood within the identifying code. We study the associated computational problem Minimum Identifying Code, which is known to be NP-hard, even when the input graph belongs to a number of specific graph classes such as planar bipartite graphs. Though the problem is approximable within a logarithmic factor, it is known to be hard to approximate within any sub-logarithmic factor. We extend the latter result to the case where the input graph is bipartite, split or co-bipartite. Among other results, we also show that for bipartite graphs of bounded maximum degree (at least 3), it is hard to approximate within some constant factor. We summarize known results in the area, and we compare them to the ones for the related problem Minimum Dominating Set. In particular, our work exhibits important graph classes for which Minimum Dominating Set is efficiently solvable, but Minimum Identifying Code is hard (whereas in all previously studied classes, their complexity is the same). We also introduce a graph class for which the converse holds.

1 Introduction

We study the computational complexity of the *identifying code problem*, where we want to find a set of vertices in a graph that uniquely identifies each vertex using its neighbourhood within the set. In particular, we study this complexity according to the graph class of the input. Identifying codes, introduced in 1998 [25], are a special case of the notion of a *test cover* in hypergraphs, first mentioned in Garey and Johnson's book [20]. Test covers have found applications in the areas of testing individuals (such as patients or computers) for diseases or faults, see [7,12]. In particular, as graphs model computer networks or buildings, identifying codes have been applied to the location of threats in facilities [34] and error detection in computer networks [25].

To avoid confusion, we will usually call hypergraphs and their vertex and edge sets $H = (I, A)$ and graphs $G = (V, E)$. Given a hypergraph H, a *set cover*

* An extended version of this paper, containing the full proofs and further results, is available on the author's website [16].

of H is a subset $\mathcal{S}$ of its edges such that each vertex v belongs to at least one set S of $\mathcal{S}$. We say that S *dominates* v. A *test cover* of H is a subset $\mathcal{T}$ of edges such that for each pair u, v of distinct vertices of H, there is at least one set T of $\mathcal{T}$ that contains exactly one of u and v [20]. We say that T (and also $\mathcal{T}$) *separates* u from v. A set of edges that is both a set cover and a test cover is called a *discriminating code* of H [7]. It has to be mentioned that some hypergraphs may not admit any set cover (if some vertex is not part of any edge) or test cover (if two vertices belong to the same set of edges).

Given a graph G and a vertex v of G, we denote by $N[v]$ the closed neighbourhood of v. An *identifying code* of G is a subset $\mathcal{C} \subseteq V(G)$ such that $\mathcal{C}$ is a *dominating set*, i.e. for each $v \in V(G)$, $N[v] \cap \mathcal{C} \neq \emptyset$ and $\mathcal{C}$ is a *separating code*, i.e. for each pair $u, v \in V(G)$, if $u \neq v$, then $N[u] \cap \mathcal{C} \neq N[v] \cap \mathcal{C}$. The minimum size of an identifying code of a given graph G will be denoted $\gamma^{\text{ID}}(G)$. Identifying codes were introduced in [25]. Note that a graph may not admit a separating code if it contains *twin* vertices, i.e. vertices having the same closed neighbourhood. In a graph containing no twins, the whole vertex set is an identifying code; we call such graphs *twin-free*.

Identifying codes and further related notions have been studied extensively in the literature. We refer to Lobstein's on-line bibliography [27] on these topics. In particular, see [1,2,5,9,15,17,18,21,26,29,31,32] for studies of the computational complexity of these problems.

For definitions of computational complexity, we refer to the books of Ausiello et al. and of Garey and Johnson [3,20]. Let us formally define the minimization problem associated with identifying codes (other problems used herein are defined analogously; we skip their definitions).

MIN ID CODE
INSTANCE: A graph G.
TASK: Find a minimum-size identifying code of G.

We will study MIN ID CODE from an approximation point of view, but also from a decision point of view; in that case MIN ID CODE is said to be NP-hard if the associated decision problem (consisting in deciding whether a given graph has an identifying code of a given size) is NP-hard.

We recall that the class APX is the class of all optimization problems that are c-approximable for some constant c. We also refer to the class log-APX as the class of all optimization problems that are $f(n)$-approximable, where n is the size of the instance and f is a poly-logarithmic function.

In this paper, we will study specific graph classes, of which many are standard, such as bipartite graphs, planar graphs or graphs of given maximum degree. Bipartite graphs which do not have any induced cycle of length more than 4 are called *chordal bipartite* (note that they are in general not chordal). Complements of bipartite graphs are called *co-bipartite* graphs. *Split graphs* are those whose vertex set can be partitioned into a clique and an independent set.

1.1 Related work

It is well-known that MIN DOMINATING SET is log-APX-hard (whereas a logarithmic factor approximation exists) [11, 22]. The same properties hold for MIN TEST COVER [12] (a result that is easily seen to be transferrable to MIN DISCRIMINATING CODE[1]) and MIN ID CODE (see [5, 26, 32], for different proofs).

Regarding restrictions of the instances to specific graph classes, much is known for MIN DOMINATING SET: NP-hardness of MIN DOMINATING SET holds for many classes such as (chordal) bipartite graphs, split graphs, line graphs or planar graphs, but not for strongly chordal graphs, directed path graphs (which include the well-known interval graphs), or graphs having a dominating shortest path (see e.g. [13] for an on-line database, and [22] for a survey). The log-APX-completeness of MIN DOMINATING SET is known to hold even for bipartite graphs and split graphs [11], however it does not hold for planar graphs or unit disk graphs (in which MIN DOMINATING SET admits PTAS algorithms [4, 23]) or in (bipartite) graphs of bounded maximum degree (at least 3), where it is APX-complete [11].

In comparison, much less is known about MIN ID CODE; extending this knowledge is the main goal of this paper. It was known that, in general, MIN ID CODE is NP-hard, even for bipartite graphs [9], planar graphs of maximum degree 3 [1, 2], planar bipartite unit disk graphs [29], line graphs [17], split graphs [15, 18], and, interestingly, interval graphs [15, 18]. Regarding the approximation hardness, log-APX-completeness of MIN ID CODE is known only for general graphs [5, 26, 32], and APX-completeness, for graphs of bounded maximum degree at least 8 [21].

1.2 Our contribution and structure of the paper

We extend the knowledge about the computational complexity of MIN ID CODE when restricted to specific classes of graphs. We compare these results to the corresponding ones for MIN DOMINATING SET; see Table 1 for a summary of many known complexity results for these problems for selected graph classes.

We show in Section 2 that MIN ID CODE is log-APX-complete for bipartite, split and co-bipartite graphs. Prior, three different papers [5, 26, 32] showed that MIN ID CODE is log-APX-complete, but only in general graphs; intuitively speaking, the proximity between MIN DISCRIMINATING CODE and MIN ID CODE is used to design simpler reductions. Note that on co-bipartite graphs, MIN DOMINATING SET is trivially solvable in polynomial time; in contrast, our result shows that MIN ID CODE is computationally very hard on this class.

In Section 3, we show that MIN ID CODE is APX-complete for bipartite graphs of maximum degree 3, improving on a result from Gravier et al. [21]. We also show that it is NP-hard for the same class with the additional restriction of planarity, as well as for chordal bipartite graphs.

[1] We can reduce MIN TEST COVER to MIN DISCRIMINATING CODE: given a hypergraph H, construct H' by adding to H a single vertex x and the set $V(H) \cup \{x\}$. Now H has a test cover of size k if and only if H' has a discriminating code of size $k + 1$.

Finally, in Section 4, we exhibit a class of graphs, which we call SC-graphs, where Min Dominating Set is NP-hard, but Min Id Code is solvable in polynomial time. Until now, all known results for given graph classes were showing that Min Id Code was at least as hard as Min Dominating Set.

graph class	Min Id Code LB	Min Id Code UB	Min Dominating Set LB	Min Dominating Set UB
in general	log-APX-h [5,26,32]	$O(\ln n)$ [12]	log-APX-h [11]	$O(\ln n)$ [24]
bipartite	log-APX-h (Co. 4)	$O(\ln n)$ [12]	log-APX-h [11]	$O(\ln n)$ [24]
chordal bipartite (∗)	NP-h (Th. 15)	$O(\ln n)$ [12]	NP-h [28]	$O(\ln n)$ [24]
split, chordal	log-APX-h (Th. 7)	$O(\ln n)$ [12]	log-APX-h [11]	$O(\ln n)$ [24]
planar (+ bipartite max. degree 3) (∗)	NP-h (Th. 11)	7 [31]	NP-h [33]	PTAS [4]
line (∗)	APX-h [15]	4 [17]	APX-h [10]	2 [10]
$K_{1,\ell}$-free ($\ell \geq 3$)	log-APX-h (Th. 7)	$O(\ln n)$ [12]	APX-h [10]	$\ell - 1$ [11]
max. degree Δ	APX-h $\Delta \geq 8$: [21]	$O(\ln \Delta)$ [12]	APX-h $\Delta \geq 3$: [30]	$O(\ln \Delta)$ [24]
max. degree $\Delta \geq 3$ and bipartite	APX-h (Th. 11)	$O(\ln \Delta)$ [12]	APX-h [11]	$O(\ln \Delta)$ [24]
unit disk (∗)	NP-h [29]	$O(\ln n)$ [12]	NP-h [8]	PTAS [23]
co-bipartite	log-APX-h (Th. 7)	$O(\ln n)$ [12]	P (trivial)	
interval, (∗)	NP-h [18]	$O(\ln n)$ [12]	P [6]	
permutation (∗)	OPEN	$O(\ln n)$ [12]	P [14]	
(planar) SC-graphs (∗)	P (Th. 18)		NP-h (Th. 19)	$O(\ln n)$ [24]

Table 1. Comparison of complexity lower bounds,"LB", and upper bounds, "UB", on approximation ratios (as functions of the order n of the input graph) of Min Id Code and Min Dominating Set for selected graph classes. Underlined entries are new results proved in this paper. Graph classes for which the precise complexity class of Min Id Code is not fully determined are marked with (∗). SC-graphs will be defined in Section 4. Definitions for classes that are not defined here can be found in [13].

2 Bipartite, co-bipartite and split graphs

In this section, we provide three reductions from Min Discriminating Code to Min Id Code for bipartite, split and co-bipartite graphs. We begin with preliminary considerations.

2.1 Useful bounds and constructions

Theorem 1 ([7]). *Let $H = (I, A)$ be a hypergraph admitting a discriminating code $\mathcal{C}$. Then $|\mathcal{C}| \geq \log_2(|I| + 1)$. If $\mathcal{C}$ is inclusion-wise minimal, then $|\mathcal{C}| \leq |I|$.*

We now describe two constructions that ensure that the vertices of some vertex set A are correctly identified using the vertices of another set L.

Construction 2 (bipartite logarithmic identification of A over (A, L)). Given two sets of vertices A and L with $|A| \leq 2^{|L|} - 1$, the bipartite logarithmic identification of A over (A, L), denoted $\mathcal{LOG}(A, L)$, is the graph of vertex set $A \cup L$ and where each vertex of A has a distinct nonempty subset of L as its neighbourhood.

The next construction is similar, but makes sure that each vertex of A has at least two neighbours in L.

Construction 3 (non-singleton bipartite logarithmic identification of A over (A, L)). Given two sets of vertices A and L with $|A| \leq 2^{|L|} - |L| - 1$,[2] the non-single bipartite logarithmic identification of A over (A, L), denoted $\mathcal{LOG}^*(A, L)$, is the graph of vertex set $A \cup L$ and where each vertex of A has a distinct subset of L of size at least 2 as its neighbourhood.

2.2 Bipartite graphs

Theorem 4. MIN ID CODE *is log-APX-complete, even for bipartite graphs.*

Theorem 4 is proved using the following reduction.

Reduction 5. Given a hypergraph (I, A), we construct in polynomial time the bipartite graph $G(I, A)$ on $|I| + |A| + 9\lceil\log_2(|A| + 1)\rceil + 3$ vertices, with vertex set $V(G(I, A)) = I \cup A \cup \{x, y, z\} \cup \{a_j, b_j, c_j, d_j, e_j, f_j, g_j, h_j, i_j \mid 1 \leq j \leq \lceil\log_2(|A| + 1)\rceil\}$, and edge set:

$$\begin{aligned}
E(G(I, A)) = \{x, y\} \cup \{y, z\} \cup \{\{z, i\} \mid i \in I\} \cup\ & E(\mathcal{B}(I, A)) \\
\cup\ & E(\mathcal{LOG}(A, \{a_j \mid 1 \leq j \leq \lceil\log_2(|A| + 1)\rceil\})) \\
\cup\ & \{\{a_j, b_j\}, \{b_j, c_j\}, \{a_j, d_j\}, \{d_j, g_j\} \mid 1 \leq j \leq \lceil\log_2(|A| + 1)\rceil\} \\
\cup\ & \{\{d_j, e_j\}, \{e_j, f_j\}, \{g_j, h_j\}, \{h_j, i_j\} \mid 1 \leq j \leq \lceil\log_2(|A| + 1)\rceil\}.
\end{aligned}$$

where $\mathcal{B}(I, A)$ denotes the bipartite incidence graph of (I, A) and $E(\mathcal{LOG}(A, L))$ denotes the bipartite logarithmic identification of A over (A, L) (see Construction 2). The construction is illustrated in Figure 1.

Theorem 6. *A hypergraph (I, A) has a discriminating code of size at most k if and only if graph $G(I, A)$ has an identifying code of size at most $k + 6\lceil\log_2(|A| + 1)\rceil + 2$, and one can construct one using the other in polynomial time.*

[2] There are exactly $2^{|L|} - |L| - 1$ distinct subsets of L with size at least 2.

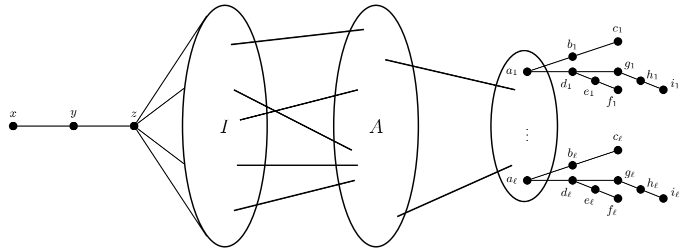


Fig. 1. Reduction from Min Discriminating Code to Min Id Code.

Proof. Let $\mathcal{D} \subseteq A$ be a discriminating code of (I, A), $|\mathcal{D}| = k$. We define $\mathcal{C}(\mathcal{D})$ as follows: $\mathcal{C}(\mathcal{D}) = \mathcal{D} \cup \{x, z\} \cup \{a_j, c_j, d_j, f_j, g_j, i_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1) \rceil\}$. One can easily check that $\mathcal{C}(\mathcal{D})$ has size $k + 6\lceil \log_2(|A| + 1)\rceil + 2$. Code $\mathcal{C}(\mathcal{D})$ has size $k+6\lceil \log_2(|A|+1)\rceil+2$ is clearly a dominating set. To see that it is an identifying code of $G(I, A)$, observe that vertex z separates all vertices of I from all vertices which are not in $I \cup \{z\}$. Vertex z itself is the only vertex dominated only by z (each vertex of $\mathcal{I}$ being dominating by some vertex of $\mathcal{D}$); y is dominated by both x, y and x, only by itself. Since $\mathcal{D}$ a discriminating code of (I, A), all vertices of I are dominated by a distinct subset of $\mathcal{D}$. Furthermore, due to the bipartite logarithmic identification of A over $(A, \{a_j \mid 1 \leq j \leq \lceil \log_2(|A|+1)\rceil\})$ (and since each vertex a_j belongs to the code), all vertices of A are dominated by a unique subset of $\{a_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\}$. Finally, it is easy to check that all vertices of type $a_j, b_j, c_j, d_j, e_j, f_j, g_j, h_h, i_j$ are correctly separated.

For the other direction, Let $\mathcal{C}$ be an identifying code of $G(I, A)$, $|\mathcal{C}| = k + 6\lceil \log_2(|A| + 1)\rceil + 2$. We first "normalize" $\mathcal{C}$ by constructing an identifying code $\mathcal{C}^*$ of $G(I, A)$, $|\mathcal{C}^*| \leq |\mathcal{C}|$, such that the two following properties hold:

$$|\mathcal{C}^* \cap \{V(G(I, A)) \setminus \{I \cup A\}\}| = 6\lceil \log_2(|A| + 1)\rceil + 2 \quad (1)$$

$$|\mathcal{C}^* \cap I| = \emptyset. \quad (2)$$

To get Condition (1), we replace $|\mathcal{C} \cap \{V(G(I, A)) \setminus \{I \cup A\}\}|$ by $\{x, z\} \cup \{a_j, c_j, d_j,\ f_j, g_j, i_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\}$ to get code $\mathcal{C}'$ (whose structure is similar to the one of the code constructed in the first part of the proof). We claim that $|\mathcal{C}'| \leq |\mathcal{C}|$. First of all, observe that we had $|\mathcal{C} \cap \{V(G(I, A)) \setminus \{I \cup A\}\}| \geq 6\lceil \log_2(|A| + 1)\rceil + 2$. To see this, note that vertex z is the only one separating $\{x, y\}$, and $|\mathcal{C} \cap \{x, y\}| \geq 1$ since $\mathcal{C}$ must dominate x. Similarly, for any $j \in \{1, \ldots, \log_2(|A|+1)\}$, vertices a_j, d_j, g_j are the only ones separating $\{b_j, c_j\}$, $\{e_j, f_j\}$ and $\{h_j, i_j\}$, respectively, and $|\mathcal{C} \cap \{b_j, c_j\}| \geq 1$, $|\mathcal{C} \cap \{e_j, f_j\}| \geq 1$ and $|\mathcal{C} \cap \{h_j, i_j\}| \geq 1$, since $\mathcal{C}$ must dominate c_j, f_j and i_j, respectively.

To fulfill Condition (2), we replace each vertex $i \in I \cap \mathcal{C}'$ by some vertex in A. If $\mathcal{C}' \setminus \{i\}$ is an identifying code, we may just remove i from the code. Otherwise, note that i is not needed for domination since all vertices of I are dominated by z and all vertices of A are dominated by some vertex in $\{a_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\}$. Hence, i separates i itself from some other vertex i' in I

(indeed, one can check that all other types of pairs which could be separated by i are actually already separated by some vertex of $\mathcal{C}' \cap (V(G(I,A)) \setminus I)$. But then, the pair $\{i, i'\}$ is unique (suppose i separates i itself from two distinct vertices i' and i'' of I, then i' and i'' would not be separated by $\mathcal{C}'$, a contradiction). Since (I, A) admits a discriminating code, there must be some vertex a of A separating i from some i'. Hence we replace i by a. Doing this for every $i \in \mathcal{C}' \cap I$, we get code $\mathcal{C}^*$, and $|\mathcal{C}^*| \leq |\mathcal{C}'| \leq |\mathcal{C}|$.

Using these observations and similar arguments as in the first part of the proof, one can check that the obtained code $\mathcal{C}^*$ is still an identifying code.

To complete the proof, we claim that $\mathcal{C}^* \cap A$ is a discriminating code of (I, A): indeed, all pairs $\{I, I'\}$ of I are separated by $\mathcal{C}^*$. By Condition (1), they must be separated by some vertex of A (note that z is adjacent to all vertices of I), and we are done. □

Theorem 6 proves that Min Id Code for bipartite graphs is NP-hard, and can be used to prove Theorem 4:

Proof (Proof of Theorem 4). We use Theorem 6 to show that any c-approximation algorithm $\mathscr{A}$ for Min Id Code for bipartite graphs can be turned into a $7c$-approximation algorithm for Min Discriminating Code. Min Discriminating Code being log-APX-complete [12] and Min Id Code being in log-APX, we get the claim.

Let (I, A) be a hypergraph with optimal value OPT, and let $G(I, A)$ be the bipartite graph constructed using Reduction 5. By Theorem 6, we have:

$$\gamma^{\text{ID}}(G(I,A)) \leq OPT + 6\lceil \log_2(|A|+1) \rceil + 2. \tag{3}$$

Let $\mathcal{C}$ be an identifying code of $G(I, A)$ computed by $\mathscr{A}$. We have:

$$|\mathcal{C}| \leq c\gamma^{\text{ID}}(G(I,A)). \tag{4}$$

By Theorem 6, we can compute in polynomial time a discriminating code $\mathcal{D}$ of (I, A). Using Inequalities 3 and 4 together with the fact that $\lceil \log_2(|A|) \rceil \leq OPT \leq |\mathcal{D}|$ (Theorem 1), we get:[3]

$$\begin{aligned}
|\mathcal{D}| &\leq |\mathcal{C}| - 6\lceil \log_2(|A|+1) \rceil - 2 \leq c\gamma^{\text{ID}}(G(I,A)) - 6\lceil \log_2(|A|+1) \rceil - 2 \\
&\leq c(OPT + 6\lceil \log_2(|A|+1) \rceil + 2) - 6\lceil \log_2(|A|+1) \rceil - 2 \\
&\leq cOPT + (c-1)(6\lceil \log_2(|A|) \rceil + 8) \leq cOPT + (c-1)(6OPT + 8) \\
&\leq 7cOPT.
\end{aligned}$$

□

2.3 Split graphs and co-bipartite graphs

Theorem 7. Min Id Code *is log-APX-complete for split graphs and for co-bipartite graphs.*

[3] For the last line inequality, we assume here that $OPT \geq 2$.

Theorem 7 is proved using the two following reductions from Min Discriminating Code to Min Id Code.

Reduction 8. Given a hypergraph (I, A), we construct in polynomial time the following split graph $Sp(I, A)$ on $|I| + |A| + 6\lceil \log_2(|A| + 1)\rceil + 1$ vertices, with vertex set $V(Sp(I, A)) = K \cup S$ (K is a clique and S, an independent set). More specifically, $K = I \cup \{u\} \cup \{k_j \mid 1 \leq j \leq 2\lceil \log_2(|A| + 1)\rceil\}$ and $S = A \cup \{v\} \cup \{s_j, t_j \mid 1 \leq j \leq 2\lceil \log_2(|A| + 1)\rceil\}$.

$Sp(I, A)$ has edge set:

$$\begin{aligned} E(Sp(I, A)) = \{u, v\} \cup\ & E(\mathcal{B}(I, A)) \\ & \cup\ E(\mathcal{LOG}^*(A, \{k_j \mid 1 \leq j \leq 2\lceil \log_2(|A| + 1)\rceil\})) \\ & \cup \{\{k_j, s_j\}, \{k_j, t_j\} \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\} \\ & \cup \{a, b \mid a, b \in K, a \neq b\}, \end{aligned}$$

where $\mathcal{B}(I, A)$ denotes the bipartite incidence graph of (I, A) and $E(\mathcal{LOG}^*(A, L))$ denotes the non-singleton bipartite logarithmic identification of A over (A, L) (see Construction 3). The construction is illustrated in Figure 2(a).

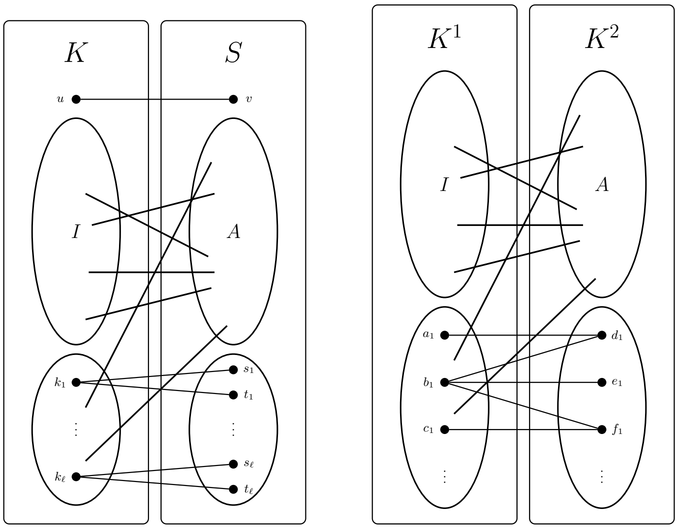
Reduction 9. Given a hypergraph (I, A), we construct in polynomial time the following co-bipartite graph $G(I, A)$ on $|I| + |A| + 6\lceil \log_2(|A| + 1)\rceil$ vertices, with vertex set $V(G(I, A)) = K^1 \cup K^2$, where K^1 and K^2 are two cliques over the two sets of vertices $K^1 = I \cup \{a_j, b_j, c_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\}$ and $K^2 = A \cup \{d_j, e_j, f_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\}$.

$G(I, A)$ has edge set:

$$\begin{aligned} E(G(I, A)) = E(\mathcal{B}(I, A)) \cup\ & E(\mathcal{LOG}(A, \{a_j \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\})) \\ & \cup \{\{a_j, d_j\}, \{b_j, d_j\}, \{b_j, e_j\}, \{b_j, f_j\}, \{c_j, f_j\} \mid 1 \leq j \leq \lceil \log_2(|A| + 1)\rceil\} \\ & \cup \{x, y \mid x, y \in K^1\} \cup \{x, y \mid x, y \in K^2\}. \end{aligned}$$

where $\mathcal{B}(I, A)$ denotes the bipartite incidence graph of (I, A) and $E(\mathcal{LOG}(A, L))$ denotes the bipartite logarithmic identification of A over (A, L) (see Construction 2). The construction is illustrated in Figure 2(b).

Proof (Sketch of proof of Theorem 7). Reductions 8 and 9 can be used to show that, given a hypergraph (I, A), (I, A) has a discriminating code of size at most k if and only if $Sp(I, A)$ has an identifying code of size at most $k + 4\lceil \log_2(|A| + 1)\rceil + 1$ and $G(I, A)$ has an identifying code of size at most $k + 5\lceil \log_2(|A| + 1)\rceil - 2$, respectively. Moreover these constructions can be performed in polynomial time. Using similar arguments as for bipartite graphs, we can show that any c-approximation algorithm for Min Id Code for split graphs or co-bipartite graphs can be turned into a $5c$- or $6c$-approximation algorithm for Min Discriminating Code, respectively.



(a) Reduction for split graphs (b) Reduction for co-bipartite graphs

Fig. 2. Two reductions from Min Discriminating Code to Min Id Code.

3 Reductions for (planar) bipartite graphs of bounded maximum degree and chordal bipartite graphs

In this section, we improve results from the literature by showing that Min Id Code is NP-hard for planar bipartite graphs of maximum degree 3. We also improve and extend the APX-hardness results for Min Id Code for non-bipartite graphs of maximum degree at least 8 from [21] by showing that they are APX-hard even for bipartite graphs of maximum degree 3. Finally, we show that Min Id Code is NP-hard for chordal bipartite graphs.

We will use the standard concept of *L-reductions*, that is widely used to prove APX-hardness of optimization problems.

Definition 10 ([30]). *Let P and Q be two optimization problems. An* L-reduction *from P to Q is a four-tuple (f, g, α, β) where f and g are polynomial time computable functions and α, β are positive constants with the following properties:*

1. *Function f maps instances of P to instances of Q and for every instance I_P of P, $OPT_Q(f(I_P)) \leq \alpha \cdot OPT_P(I_P)$.*
2. *For every instance I_P of P and every solution $SOL_{f(I_P)}$ of $f(I_P)$, g maps the pair $(f(I_P), SOL_{f(I_P)})$ to a solution SOL_{I_P} of I_P such that $|OPT_P(I_P) - |SOL_{I_P}|| \leq \beta \cdot |OPT_Q(f(I_P)) - |SOL_{f(I_P)}||$.*

As discovered in [30], if there exists an L-reduction between two optimization problems P and Q with parameters α and β and it is NP-hard to approximate

P within ratio $r_P = 1 + \delta$, then it is NP-hard to approximate Q within ratio $r_Q = 1 + \frac{\delta}{\alpha\beta}$.

3.1 (Planar) bipartite graphs of maximum degree 3

Theorem 11. *Reduction 12 applied to graphs of maximum degree 3 is an L-reduction with parameters $\alpha = 4$ and $\beta = 1$. Therefore,* Min Id Code *is APX-complete, even for bipartite graphs of maximum degree 3. Moreover,* Min Id Code *is NP-hard, even for planar bipartite graphs of maximum degree 3.*

We prove Theorem 11 using the following reduction.

Reduction 12. Given a graph G, we construct the graph G' on vertex set

$$V(G') = V(G) \cup \{p_e, q_e \mid e \in E(G)\},$$

and edge set

$$E(G') = \{\{x, p_e\}, \{y, p_e\}, \{p_e, q_e\} \mid e = \{x, y\} \in E(G)\}.$$

The construction is illustrated in Figure 3 (where vertices of G are circled).

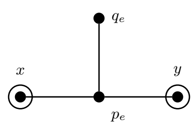


Fig. 3. Reduction 12 from Min Vertex Cover to Min Id Code.

For the following claims, let G be a graph and G', the graph obtained from G using Reduction 12.

Claim 13. *Let $\mathcal{N}$ be a vertex cover of G. Using $\mathcal{N}$, one can build an identifying code of G' of size at most $|\mathcal{N}| + |E(G)|$.*

Proof. First of all, we may assume that G is connected. Furthermore, it has no vertex of degree less than 2. Indeed, assuming we have a vertex cover containing a degree 2-vertex x, we can always replace it by its neigbour in the solution. Removing x and its neighbour from the graph, one gets a computationally equivalent instance.

Let $\mathcal{C} = \mathcal{N} \cup \{p_e \mid e \in E(G)\}$. Set $\mathcal{C}$ is an identifying code of G': any original vertex x of G is dominated by the unique set of vertices $\{p_e \mid x \in e, e \in E(G)\}$ (this set having at least two elements). For each edge $\{x, y\} = e \in E(G)$, vertex p_e is dominated by itself and at least one of x, y; q_e is dominated by p_e only. □

Claim 14. *Let $\mathcal{C}$ be an identifying code of G'. One can use $\mathcal{C}$ to build a vertex cover of G of size at most $|\mathcal{C}| - |E(G)|$.*

Proof. For each edge $e = \{x, y\}$ of G, one of p_e, q_e belongs to $\mathcal{C}$, since $\mathcal{C}$ has to dominate q_e. Moreover, one of x, y belongs to $\mathcal{C}$ since p_e, q_e need to be separated. Hence, $\mathcal{C} \cap V(G)$ is a vertex cover of G with size at most $|\mathcal{C}| - |E(G)|$. □

We are now ready to prove Theorem 11. In what follows, let $\tau(G)$ denote the minimum size of a vertex cover of G.

Proof (Proof of Theorem 11). Let G be a graph of maximum degree 3 and G' the graph constructed from G using Reduction 12. We have to prove Properties 1 and 2 from Definition 10.

By Claim 13, given an optimal vertex cover $\mathcal{N}^*$ of G, we can construct an identifying code $\mathcal{C}$ with $\gamma^{\text{ID}}(G') \leq |\mathcal{C}| \leq |\mathcal{N}^*| + |E(G)| = \tau(G) + |E(G)|$. By Claim 14, given an optimal identifying code $\mathcal{C}^*$ of G', we can construct a vertex cover $\mathcal{N}$ of G such that $\tau(G) \leq |\mathcal{N}| \leq |\mathcal{C}^*| - |E(G)| = \gamma^{\text{ID}}(G) - |E(G)|$. Hence:

$$\gamma^{\text{ID}}(G') = \tau(G) + |E(G)|. \tag{5}$$

Proof of Property 1. Since G has maximum degree 3, each vertex can cover at most three edges, hence we have $\tau(G) \geq \frac{|E(G)|}{3}$, so $|E(G)| \leq 3\tau(G)$. Using Equality (5), we get that $\gamma^{\text{ID}}(G') = \tau(G) + |E(G)| \leq 4\tau(G)$.

Proof of Property 2. Let $\mathcal{C}$ be an identifying code of G'. Using Claim 14 applied to $\mathcal{C}$, we obtain a vertex cover $\mathcal{N}$ with $|\mathcal{N}| \leq |\mathcal{C}| - |E(G)|$. By Equality (5), we have $-\tau(G) = |E(G)| - \gamma^{\text{ID}}(G')$. So we obtain:

$$\begin{aligned} |\mathcal{N}| - \tau(G) \leq& |\mathcal{C}| - |E(G)| + |E(G)| - \gamma^{\text{ID}}(G') \\ |\tau(G) - |\mathcal{N}|| \leq& |\gamma^{\text{ID}}(G') - |\mathcal{C}||. \end{aligned}$$

For the second part of the statement, Min Vertex Cover is known to be APX-complete for graphs of maximum degree 3 [11]. It is easy to check that the constructed graphs have maximum degree 3 and are bipartite. For the final part of the statement, we apply Reduction 12 to Min Vertex Cover for *planar* graphs of maximum degree 3, which is known to be NP-hard [19]. Claims 13 and 14 applied to an optimal vertex cover and an optimal identifying code show that $\gamma^{\text{ID}}(G') = \tau(G) + |E(G)|$. □

3.2 Chordal bipartite graphs

Theorem 15. Min Id Code *is NP-hard, even for chordal bipartite graphs.*

Reduction 16. Given a graph G, we construct the graph G' on vertex set

$$V(G') = V(G) \cup \{a_x, b_x, c_x, d_x, e_x \mid x \in V(G)\},$$

and edge set

$$\begin{aligned} E(G') =& E(G) \cup \{\{x, a_x\}, \{x, e_x\}, \{a_x, b_x\}, \{a_x, c_x\}, \{a_x, d_x\}, \{e_x, b_x\}, \\ & \{e_x, c_x\}, \{e_x, d_x\} \mid x \in V(G)\}. \end{aligned}$$

The construction is illustrated in Figure 4.

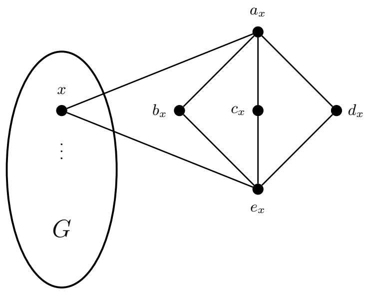


Fig. 4. Reduction from Min Dominating Set to Min Id Code.

To prove Theorem 15, we show that G has a dominating set of size at most k if and only if G' has an identifying code of size at most $k + 3|V(G)|$. The proof is omitted due to lack of space.

4 Further classes of graphs for which the complexities of Min Dominating Set, and Min Id Code differ

We saw that for co-bipartite graphs, Min Id Code is hard (whereas Min Dominating Set is trivially solvable in polynomial time). In this section, we define a class for which the converse holds: Min Dominating Set is NP-hard, but Min Id Code is solvable in polynomial time. We call these graphs *SC-graphs.*

Definition 17. *A graph G is an* SC-graph *if it can be built from a bipartite graph with parts S and T and an additional set S' with $|S'| = 2|S|$ such that:*

- *for each vertex x of S, there is a path x, u_x, v_x of length 2 starting at x with $u_x, v_x \in S'$, $deg_G(u_x) = 2$ and $deg_G(v_x) = 1$, and*
- *each vertex of T has a distinct neighbourhood within S, and this neighbourhood has at least two elements.*

An example of an SC-graph is pictured in Figure 5. We have the following theorems (proofs are omitted due to lack of space).

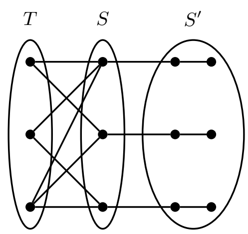


Fig. 5. Example of an SC-graph.

Theorem 18. *Let G be an SC-graph built from a bipartite graph with parts S and T, with S_1, the set of all degree 1-vertices of the pendant paths attached to the vertices of S. We have $\gamma^{ID}(G) = 2|S|$ and $S \cup S_1$ is an identifying code of G. Hence,* Min Id Code *can be solved in polynomial time in the class of SC-graphs.*

Theorem 19. Min Dominating Set *is NP-hard in planar (bipartite) SC-graphs of maximum degree 4.*

5 Open problems

The complexity for Min Id Code is open for several important input graph classes, as shown in Table 1. Regarding interval graphs, the approximation complexity of Min Id Code is still an open question. It is also of interest to determine the complexity of Min Id Code for permutation graphs (for which Min Dominating Set is polynomial-time solvable [14]). Finally, we remark that Min Dominating Set admits PTAS algorithms for planar graphs [4] and for unit disk graphs [23]. Does the same hold for Min Id Code?

References

1. D. Auger. Minimal identifying codes in trees and planar graphs with large girth. *Eur. J. Combin.* 31(5):1372–1384, 2010.
2. D. Auger, I. Charon, O. Hudry and A. Lobstein. Complexity results for identifying codes in planar graphs. *Int. T. Oper. Res.* 17(6):691–710, 2010.
3. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi. *Complexity and approximation*, Springer, 1999.
4. B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. ACM* 41(1):153–180, 1994.
5. T. Y. Berger-Wolf, M. Laifenfeld and A. Trachtenberg. Identifying codes and the set cover problem. *Proc. 44th Allerton Conf. on Comm. Contr. and Comput.*, 2006.
6. K. S. Booth and H. J. Johnson. Dominating sets in chordal graphs. *SIAM J. Comput.* 11(1):191–199, 1982.
7. E. Charbit, I. Charon, G. Cohen, O. Hudry and A. Lobstein. Discriminating codes in bipartite graphs: bounds, extremal cardinalities, complexity. *Adv. Math. Commun.* 4(2):403–420, 2008.
8. B. N. Clark, C. J. Colbourn and D. S. Johnson. Unit disk graphs. *Discrete Math.* 86(1-3):165–177, 1990.
9. I. Charon, O. Hudry and A. Lobstein. Minimizing the size of an identifying or locating-dominating code in a graph is NP-hard. *Theor. Comput. Sci.* 290(3):2109–2120, 2003.
10. M. Chlebík and J. Chlebíková. Approximation hardness of edge dominating set problems. *J. Comb. Optim.* 11:279–290, 2006.
11. M. Chlebík and J. Chlebíková. Approximation hardness of dominating set problems in bounded degree graphs. *Inform. Comput.* 206:1264–1275, 2008.
12. K. M. J. De Bontridder, B. V. Halldórsson, M. M. Halldórsson, C. A. J. Hurkens, J. K. Lenstra, R. Ravi and L. Stougie. Approximation algorithms for the test cover problem. *Math. Programm. Ser. B* 98:477–491, 2003.

13. H. N. De Ridder and others. Information System on Graph Classes and their Inclusions (ISGCI). http://www.graphclasses.org
14. M. Farber and J. M. Keil. Domination in permutation graphs. *J. Algorithm.* 6:309–321, 1985.
15. F. Foucaud. *Combinatorial and algorithmic aspects of identifying codes in graphs.* PhD thesis, Université Bordeaux 1, France, 2012. http://tel.archives-ouvertes.fr/tel-00766138.
16. F. Foucaud. On the decision and approximation complexities for identifying codes and locating-dominating sets in restricted graph classes. Manuscript, 2013. http://www-ma4.upc.edu/~florent.foucaud/Research.
17. F. Foucaud, S. Gravier, R. Naserasr, A. Parreau and P. Valicov. Identifying codes in line graphs. To appear in *J. Graph Theor.*
18. F. Foucaud, G. Mertzios, R. Naserasr, A. Parreau and P. Valicov. Identifying codes in subclasses of perfect graphs. Manuscript, 2012.
19. M. R. Garey and D. S. Johnson. The rectilinear Steiner tree problem is NP-complete. *SIAM J. Appl. Math.* 32(4):826–834, 1977.
20. M. R. Garey and D. S. Johnson. *Computers and intractability: a guide to the theory of NP-completeness*, W. H. Freeman, 1979.
21. S. Gravier, R. Klasing and J. Moncel. Hardness results and approximation algorithms for identifying codes and locating-dominating codes in graphs. *Alg. Oper. Res.* 3(1):43–50, 2008.
22. T. W. Haynes, S. T. Hedetniemi and P. J. Slater, editors. *Domination in graphs: advanced topics*, Marcel Dekker, 1998.
23. H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, S. S. Ravi, D. J. Rosenkrantz and R. E. Stearns. NC-approximation schemes for NP-and PSPACE-hard problems for geometric graphs. *J. Algor.* 26:238–274, 1998.
24. D. S. Johnson, Approximation algorithms for combinatorial problems. *J. Comput. Sys. Sci.* 9:256–278, 1974.
25. M. G. Karpovsky, K. Chakrabarty and L. B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE T. Inform. Theory* 44:599–611, 1998.
26. M. Laifenfeld and A. Trachtenberg. Identifying codes and covering problems. *IEEE T. Inform. Theory* 54(9):3929–3950, 2008.
27. A. Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs: a bibliography. http://www.infres.enst.fr/~lobstein/debutBIBidetlocdom.pdf
28. H. Müller and A. Brandtädt. The NP-completeness of Steiner Tree and Dominating Set for chordal bipartite graphs. *Theor. Comput. Sci.* 53:257–265, 1987.
29. T. Müller and J.-S. Sereni. Identifying and locating-dominating codes in (random) geometric networks. *Comb. Probab. Comput.* 18(6):925–952, 2009.
30. C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *J. Comput. Sys. Sci.* 43(3):425–440, 1991.
31. P. J. Slater and D. F. Rall. On location-domination numbers for certain classes of graphs. *Congr. Numer.* 45:97–106, 1984.
32. J. Suomela. Approximability of identifying codes and locating-dominating codes. *Inform. Process. Lett.* 103(1):28–33, 2007.
33. J. Suomela. Answer to the question "Is the dominating set problem restricted to planar bipartite graphs of maximum degree 3 NP-complete?", http://cstheory.stackexchange.com/a/2508/1930.
34. R. Ungrangsi, A. Trachtenberg and D. Starobinski. An implementation of indoor location detection systems based on identifying codes. *Proc. Intelligence in Comm. Systems, INTELLCOMM 2004*, *LNCS* 3283:175–189, 2004.