

# On the minimum size of an identifying code over all orientations of a graph

Nathann Cohen

CNRS, LRI, Univ. Paris Sud  
Orsay, France

nathann.cohen@gmail.com

Frédéric Havet \*

CNRS, Université Côte d'Azur, I3S, INRIA  
Projet COATI

Sophia Antipolis, France

Frederic.Havet@cnrs.fr

Submitted: Jun 21, 2017; Accepted: Feb 17, 2018; Published: Mar 2, 2018

Mathematics Subject Classifications: 05C88, 05C89

## Abstract

If  $G$  be a graph or a digraph, let  $\text{id}(G)$  be the minimum size of an identifying code of  $G$  if one exists, and  $\text{id}(G) = +\infty$  otherwise. For a graph  $G$ , let  $\text{idor}(G)$  be the minimum of  $\text{id}(D)$  overall orientations  $D$  of  $G$ . We give some lower and upper bounds on  $\text{idor}(G)$ . In particular, we show that  $\text{idor}(G) \leq \frac{3}{2} \text{id}(G)$  for every graph  $G$ . We also show that computing  $\text{idor}(G)$  is NP-hard, while deciding whether  $\text{idor}(G) \leq |V(G)| - k$  is polynomial-time solvable for every fixed integer  $k$ .

**Keywords:** identifying code; orientations; NP-complete

## 1 Introduction

Let  $G$  be a graph. The *closed neighbourhood* of  $v$ , denoted  $N[v]$ , is the set of vertices that are either  $v$  or adjacent to  $v$  in  $D$ . A set  $C \subseteq V(G)$  is an *identifying code* in  $G$  if for every vertex  $v \in V(G)$ ,  $N[v] \cap C \neq \emptyset$ , and for any two distinct vertices  $u, v \in V(D)$ ,  $N[u] \cap C \neq N[v] \cap C$ . The set  $N[v] \cap C$  is called the *identifier* of  $v$  (with respect to  $C$  in  $G$ ) and is denoted by  $I_{C,G}(v)$ , or simply  $I_C(v)$  (resp.  $I(v)$ ) when  $G$  (resp.  $G$  and  $C$ ) is clear from the context. Two vertices  $u$  and  $v$  are *twins* in  $G$  if  $N[u] = N[v]$ . Clearly, if a graph has two twins  $u$  and  $v$ , then it has no identifying codes because  $I_C(u) = I_C(v)$  for any subset  $C \subseteq V(G)$ . Conversely, if a graph  $G$  is *twin-free* (i.e. has no twins), then  $V(G)$  is an identifying code. Hence we have the following.

**Proposition 1.** *A graph has an identifying code if and only if it is twin-free.*

---

\*Supported by ANR under contract STINT ANR-13-BS02-0007.

For a graph  $G$ , we define  $\text{id}(G)$  as the minimum size of an identifying code of  $G$  if one exists, and  $\text{id}(G) = +\infty$  otherwise. Hence Proposition 1 asserts that  $\text{id}(G)$  is finite if and only if  $G$  is twin-free.

Identifying codes have been widely studied since the introduction of the concept in [19] and have been applied to problems such as fault-diagnosis in multiprocessor systems [19], compact routing in networks [20, 21], emergency sensor networks in facilities [22] or the analysis of secondary RNA structures [17]. In particular, it has been proved [5] determining the value  $\text{id}(G)$  of a graph  $G$  is NP-hard.

Identifying codes have first been studied in undirected graphs, but the concept has naturally been extended to directed and oriented graphs [4, 3]. Let  $D$  be a digraph. The *closed out-neighbourhood* of  $v$ , denoted  $N^+[v]$ , is the set of vertices that are either  $v$  or dominated by  $v$  in  $D$ . A set  $C \subseteq V(D)$  is an *identifying code* in  $D$  if for every vertex  $v \in V(D)$ ,  $N^+[v] \cap C \neq \emptyset$ , and for any two distinct vertices  $u, v \in V(D)$ ,  $N^+[u] \cap C \neq N^+[v] \cap C$ . The set  $N^+[v] \cap C$  is called the *identifier* of  $v$  (with respect to  $C$  in  $D$ ) and is denoted by  $I_{C,D}(v)$ , or simply  $I_C(v)$  (resp.  $I(v)$ ) when  $D$  (resp.  $D$  and  $C$ ) is clear from the context.

Two vertices  $u$  and  $v$  are *twins* in  $D$  if  $N^+[u] = N^+[v]$ . Clearly, if a digraph has two twins  $u$  and  $v$ , then it has no identifying codes because  $I(u) = I(v)$  for any subset  $C \subseteq V(D)$ . Conversely, if a digraph is *twin-free* (i.e. has no twins), then  $V(D)$  is an identifying code. Hence we have the following.

**Proposition 2.** *A digraph has an identifying code if and only if it is twin-free. In particular, every oriented graph has an identifying code.*

For a digraph  $D$ , we define  $\text{id}(D)$  as the minimum size of an identifying code of  $D$  if one exists, and  $\text{id}(D) = +\infty$  otherwise. Hence Proposition 2 asserts that  $\text{id}(D)$  is finite if and only if  $D$  is twin-free.

Let  $G$  be a graph. The *symmetric digraph associated to  $G$* , denoted by  $D_G$ , is the digraph obtained from  $G$  by replacing each edge by two arcs in opposite directions. Clearly, a set  $C$  is an identifying code of  $G$  if and only if it is an identifying code of  $D_G$ . Hence all results on identifying codes in graphs may be seen as results on identifying codes in symmetric digraphs. In particular, it NP-hard to determine  $\text{id}(D)$  for a given (symmetric) digraph. Charon, Hudry, and Lobstein [5] proved that it is NP-hard to compute  $\text{id}(D)$  even if  $D$  is an acyclic bipartite digraph, and in particular if  $D$  is an oriented graph.

Let  $D$  be a twin-free digraph of order  $n$ . If  $C$  is an identifying code of  $D$ , then the sets of identifiers are distinct non-empty subsets of  $C$ , so  $n \leq 2^{|C|} - 1$ . Moreover,  $|C| \leq n$  since  $C$  is a subset of  $V(D)$ . Hence

$$\log_2(n + 1) \leq \text{id}(D) \leq n. \quad (1)$$

In this paper, we are interested by the following problem. We are given an undirected graph  $G$  of order  $n$ , and we want to find an orientation  $D$  of  $G$  such that  $\text{id}(D)$  is as small as possible. Let  $\text{idor}(G)$  be the minimum of  $\text{id}(D)$  over all possible orientations  $D$  of  $G$ .

Equation (1) yields directly the following.

$$\log_2(n+1) \leq \text{idor}(G) \leq n. \quad (2)$$

The bounds in this equation are tight. Indeed  $E_n$ , the empty graph of order  $n$ , satisfies  $\text{idor}(E_n) = n$ , and  $K_n$ , the complete graph of order  $n$ , satisfies  $\text{idor}(K_n) = \lceil \log_2(n+1) \rceil$  (see Proposition 5). The fact that the lower and upper bound are attained by the complete graphs and the empty graphs, respectively, is explained by the fact that  $\text{idor}$  decreases with the addition of edges (Proposition 10).

In Section 3, we prove additional upper bounds using several graph parameters. We also consider the case of trees: we show that for a tree  $T$  of order at least 5,  $\frac{|V(T)|+1}{2} \leq \text{idor}(T) \leq \frac{4}{3}\alpha(T)$ , where  $\alpha(T)$  denotes the maximum size of a stable set in  $T$ .

In Section 4, we investigate the relation between the parameters  $\text{id}$  and  $\text{idor}$ . Similarly, to Equation 1, we have  $\log_2(n+1) \leq \text{id}(G) \leq n$  for every twin-free graph  $G$ . Furthermore, Gravier and Moncel [16] proved that every non-empty twin-free graph  $G$  of order  $n$  satisfies  $\text{id}(G) \leq n-1$ . Hence, together with Equation (2), we obtain that  $\text{idor}(G) \leq 2^{\text{id}(G)} - 1$  and  $\text{id}(G) \leq 2^{\text{idor}(G)} - 2$  for every twin-free graph  $G$ . It is then natural to ask whether this bound are tight or not. We prove in Theorem 27 that  $\text{idor}(G) \leq \frac{3}{2} \text{id}(G)$ , and in Proposition 29, we show that this bound is tight. Foucaud [10] observed that there are twin-free graphs  $G$  for which  $\text{id}(G) \leq 2^{\text{idor}(G)} - 2$ ; see Proposition 31.

In Section 5, we study the complexity of determining  $\text{idor}(G)$  and the corresponding decision problem.

**IDOR**

Input: A graph  $G$  and an integer  $k$ .

Question:  $\text{idor}(G) \leq k$  ?

We prove that this problem is NP-complete (Theorem 32). On the the other hand, when  $k$  is fixed, deciding whether  $\text{idor}(G) \leq k$  can be done in polynomial time. Indeed, by Equation (2), a graph satisfying  $\text{idor}(G) \leq k$  has order at most  $2^k - 1$ . We can thus return ‘No’ if  $|V(G)| \geq 2^k$  and try all possibilities if  $|V(G)| \leq 2^k - 1$  (there are at most  $2^{|E(G)|} \leq 2^{|V(G)|^2}$  possible orientations and for each of them there is at most  $\binom{|V(G)|}{2} \leq \binom{2^k-1}{2}$  sets of  $k$  vertices). This procedure also shows that it can be done in FPT time when parameterized by  $k$ . An interesting question is to investigate whether  $\text{idor}(G)$  is close to the lower bound  $\log_2(|V(G)| + 1)$ .

**Problem 3.** What is the complexity of the following problem when parameterized by  $k$ ?

**SMALL-IDOR**

Input: A graph  $G$  and an non-negative integer  $k$ .

Question:  $\text{idor}(G) \leq \log_2(|V(G)| + 1) + k$  ?

Dually, one might wonder whether  $\text{idor}(G)$  is close to the upper bound  $|V(G)|$  and consider the following problem

## LARGE-IDOR

Input: A graph  $G$  and an non-negative integer  $k$ .

Question:  $\text{idor}(G) \geq |V(G)| - k$  ?

We prove in Theorem 36 that this problem is in  $XP$  when parameterized by  $k$ , that is that LARGE-IDOR is polynomial-time solvable for every fixed  $k$ . A natural question is then the following.

**Problem 4.** Is LARGE-IDOR fixed-parameter tractable (FPT) when parameterized by  $k$ ?

## 2 Preliminaries

For every positive integer  $n$ , we denote by  $[n]$  the set  $\{1, \dots, n\}$ .

Let  $G$  be a graph or a digraph. For every  $S \subseteq V(G)$ , we denote by  $G\langle S \rangle$  the subgraph or subdigraph induced by  $S$ .

**Proposition 5.**  $\text{idor}(K_n) = \lceil \log_2(n+1) \rceil$ .

*Proof.* Let us denote by  $[i]$  the set  $\{1, \dots, i\}$ .

Set  $k = \lceil \log_2(n+1) \rceil$ . Let  $X = \{x_1, \dots, x_k\}$  be a set of  $k$  vertices of  $K_n$  and let  $Y = V(K_n) \setminus X$ . Orient the edges of  $K_n\langle X \rangle$  in a transitive way, i.e.  $x_i x_j$  is an arc if and only if  $j > i$ . Let  $\mathcal{I} = \{[i] \mid 1 \leq i \leq k\}$ . Let us label the vertices of  $Y$  with distinct non-empty subsets of  $[k]$  not in  $\mathcal{I}$ . This is possible because  $n \geq 2^k - 1$ . We orient  $Y$  arbitrarily. Finally, we orient an edge  $yx_i$  from  $y$  to  $x_i$  if and only if  $i$  is in the label of  $y$ . Let us denote by  $D$  the resulting orientation of  $K_n$ .

We claim that  $X$  is an identifying code of  $D$ . Indeed, with respect to  $X$ , the identifier of  $x_i$  is  $\{x_j \mid j \in [i]\}$  and for every  $y \in Y$  the identifier of  $y$  is  $\{x_j \mid j \text{ in the label of } y\}$ . Hence all the identifiers are distinct and non-empty. Therefore  $\text{idor}(K_n) \leq k = \lceil \log_2(n+1) \rceil$ . By Equation (2),  $\text{idor}(K_n) \geq \lceil \log_2(n+1) \rceil$ .  $\square$

This proposition shows that the lower bound  $\log_2(n+1) \leq \text{idor}(G)$  of Equation 2 is tight. The empty graphs show that the upper bound  $\text{idor}(G) \leq n$  of Equation (2) is tight. We shall now completely characterize the graphs  $G$  of order  $n$  satisfying  $\text{idor}(G) = n$ .

We need the following useful lemma.

**Lemma 6.** Let  $G$  be a graph,  $(V_1, V_2)$  a partition of  $G$ , and set  $G_i = G\langle V_i \rangle$ .  $\text{idor}(G) \leq \text{idor}(G_1) + \text{idor}(G_2)$ .

*Proof.* For  $i = 1, 2$ , let  $D_i$  be an orientation of  $G_i$  having an identifying code  $C_i$  of order  $\text{idor}(G_i)$ . Let  $D$  be the orientation of  $G$  obtained from  $D_1$  and  $D_2$  by orienting all edges between  $V_1$  and  $V_2$  from  $V_1$  to  $V_2$ . Let us prove that  $C_1 \cup C_2$  is an identifying code of  $D$ . Obviously  $I_C(v) \neq \emptyset$  for every vertex  $v$  because  $I_{C_1}(v)$  or  $I_{C_2}(v)$  was not empty. Let  $u$  and  $v$  be two vertices. If  $u$  and  $v$  are both in  $V_2$ , then  $I_C(u) = I_{C_2}(u) \neq I_{C_2}(v) = I_C(v)$ . If  $u$  and  $v$  are both in  $C_1$  then  $I_C(u) \cap V_1 \neq I_{C_1}(u) \neq I_{C_1}(v) = I_C(v) \cap V_1$ , so  $I_C(u) \neq I_C(v)$ . If  $u \in V_1$  and  $v \in V_2$ , then  $I_C(u) \cap V_1 \neq \emptyset$  and  $I_C(v) \cap V_1 = \emptyset$ , so  $I_C(u) \neq I_C(v)$ . Therefore, all the identifiers (according to  $C$ ) are distinct.  $\square$

**Corollary 7.** *Let  $G$  be a graph and let  $k$  be an integer. If  $G$  has an induced subgraph  $H$  such that  $\text{idor}(H) \leq |V(H)| - k$ , then  $\text{idor}(G) \leq |V(G)| - k$ .*

Let  $G_1$  and  $G_2$  be two graphs. The disjoint union of  $G_1$  and  $G_2$  is denoted by  $G_1 + G_2$  and their join is denoted by  $G_1 \oplus G_2$ .

**Corollary 8.** *Let  $G_1$  and  $G_2$  be two graphs.  $\text{idor}(G_1 + G_2) = \text{idor}(G_1) + \text{idor}(G_2)$ .*

*Proof.* By Lemma 6,  $\text{idor}(G_1 + G_2) \leq \text{idor}(G_1) + \text{idor}(G_2)$ .

Consider now an identifying code  $C$  of an orientation  $D$  of  $G_1 + G_2$ . For  $i = 1, 2$ , let  $C_i = C \cap V(G_i)$  and  $D_i = D \langle C_i \rangle$ . One easily checks that  $C_i$  is an identifying code of  $D_i$ , so  $\text{idor}(G_1 + G_2) = |C_1| + |C_2| \geq \text{idor}(G_1) + \text{idor}(G_2)$ .  $\square$

**Proposition 9.** *Let  $G$  be a graph of order  $n$ .  $\text{idor}(G) = n$  if and only if  $G$  is the disjoint union of copies of  $K_1$  and copies of  $K_2$ .*

*Proof.* It is clear that if  $G$  is the disjoint union of copies of  $K_1$  and copies of  $K_2$ , then  $\text{idor}(G) = n$ .

Assume now that  $G$  is not the disjoint union of copies of  $K_1$  and copies of  $K_2$ . Necessarily,  $G$  has an induced subgraph isomorphic to either  $P_3$  the path of order 3, or  $K_3$  the complete graph of order 3.  $K_3$  can be oriented into the directed cycle  $\vec{C}_3$  and  $P_3$  can be oriented into the directed path  $\vec{P}_3$ . These two digraphs have an identifying code of size 2, hence  $\text{idor}(K_3), \text{idor}(P_3) \leq 2$ . Thus, by Corollary 7,  $\text{idor}(G) \leq n - 1$ .  $\square$

This proposition implies that one can decide in polynomial time whether a graph  $G$  of order  $n$  satisfies  $\text{idor}(G) \leq n - 1$ . In Subsection 5.2, we extend this by showing that, for every fixed  $k$ , it is polynomial-time solvable to decide whether a graph  $G$  of order  $n$  satisfies  $\text{idor}(G) \leq n - k$ .

**Proposition 10.** *Let  $G$  be a graph and  $H$  a spanning subgraph of  $G$ . Then  $\text{idor}(G) \leq \text{idor}(H)$ .*

*Proof.* It suffices to prove the result for  $H = G \setminus e$  for some edge  $e = xy$ . The result then follows by an immediate induction.

Let  $D$  be an orientation of  $H$  having an identifying code  $C$  of size  $\text{idor}(H)$ . Set  $U = V(G) \setminus C$ . If one endvertex of  $e$ , say  $y$ , is in  $U$ , then consider the orientation  $D'$  of  $G$  obtained from  $D$  by adding the arc  $xy$ . Clearly,  $I_{C,D'}(v) = I_{C,D}(v)$  for all  $v \in V(G)$ . Hence  $C$  is an identifying code of  $D'$ . Thus  $\text{idor}(G) \leq \text{idor}(H)$ .

Assume now that both  $x$  and  $y$  are in  $C$ . Let  $D'$  be the orientation obtained from  $D$  by adding the arc  $xy$ . Observe that  $I_{C,D'}(v) = I_{C,D}(v)$  for all  $v \in V(G) \setminus \{x\}$ , and  $I_{C,D'}(x) = I_{C,D}(x) \cup \{y\}$ . If no vertex  $v$  distinct from  $x$  has identifier  $I_{C,D'}(x)$ , then  $C$  is an identifying code of  $D'$ . Otherwise, there is a vertex  $u$  such that  $N^+[u] \cap C = I_{C,D'}(u) = I_{C,D'}(x) = N_{D'}^+[x] \cap C$ . Necessarily,  $u$  must be in  $U$ , and  $u$  dominates  $y$ . Let  $D''$  be the orientation of  $G$  obtained from  $D'$  by reversing the arc  $uy$  into  $yu$ . Then  $I_{C,D''}(v) = I_{C,D}(v)$  for all  $v \in V(G) \setminus \{x, u\}$ ,  $I_{C,D''}(x) = I_{C,D}(u)$ , and  $I_{C,D''}(u) = I_{C,D}(x)$ . Hence  $C$  is an identifying code of  $D''$ . In both cases,  $\text{idor}(G) \leq \text{idor}(H)$ .  $\square$

### 3 Bounds using various parameters

#### 3.1 Upper bounds

We denote by  $\omega(G)$  the clique number of the graph  $G$ . Proposition 5 and Corollary 7 imply directly the following.

**Proposition 11.**  $\text{idor}(G) \leq |V(G)| - \omega(G) + \lceil \log_2(\omega(G) + 1) \rceil$  for all graphs  $G$ .

Proposition 11 is tight for complete graphs and disjoint union of copies of  $K_1$ , copies of  $K_2$  and one complete graph. However, we might expect better upper bounds for other graphs, for example for connected triangle-free graphs.

The minimum degree of a graph  $G$  is denoted by  $\delta(G)$ . Proposition 9 shows that  $\text{idor}(G) \leq n - 1$  if and only  $\delta(G) \geq 2$ . We now extend this for graphs with larger minimum degree.

**Proposition 12.** For every graph  $G$ ,  $\text{idor}(G) \leq |V(G)| - \delta(G)/2 + 1$ .

*Proof.* We prove the result by induction on  $|E(G)|$ , the result holding when  $\delta(G) \leq 4$  by Proposition 9.

Let  $G$  be a graph with  $\delta(G) \geq 5$ . Necessarily  $G$  contains a cycle. Let  $C$  be a shortest cycle in  $G$ . Let  $G' = G - C$ . Since  $C$  is a shortest cycle, then every vertex in  $V(G')$  has at most three neighbours in  $C$ . Hence  $\delta(G') \geq \delta(G) - 3$ .

Assume first that  $\delta(G') \geq \delta(G) - 2$ . By the induction hypothesis,  $\text{idor}(G') \leq |V(G')| - \delta(G')/2 + 1 \leq |V(G')| - \delta(G)/2 + 2$ . Moreover by Proposition 33, we have  $\text{idor}(C) \leq |V(C)| - 1$ . Hence, by Lemma 6,  $\text{idor}(G) \leq \text{idor}(G') + \text{idor}(C) \leq |V(G)| - \delta(G)/2 + 1$ .

Assume now that  $\delta(G') = \delta(G) - 3$ . There is a vertex  $v$  of  $G'$  having three neighbours in  $C$ . Since  $C$  is a shortest cycle, it implies that  $|C| = 3$ . Hence  $V(C) \cup \{v\}$  is a clique  $K$  on four vertices, say  $K = \{t, u, v, w\}$ . Since  $\delta(G) \geq 5$ , there exists distinct vertices  $v'$  and  $w'$  in  $G - K$  such that  $v'$  is adjacent to  $v$  and  $w'$  is adjacent to  $w$ . Set  $H = G \langle K \cup \{v', w'\} \rangle$ . Let  $D$  be the orientation of  $H$  such that  $t$  is a source,  $(u, v, w, u)$  is a directed cycle,  $\{v'v, w'w\} \subset A(D)$  and all other edges between  $\{u, v, w\}$  and  $\{v', w'\}$  are oriented towards  $\{v', w'\}$ . The set  $\{u, v, w\}$  is an identifying code of  $D$  as  $I(t) = \{u, v, w\}$ ,  $I(u) = \{u, v\}$ ,  $I(v) = \{v, w\}$ ,  $I(w) = \{w, u\}$ ,  $I(v') = \{v\}$ , and  $I(w') = \{w\}$ . So  $\text{idor}(H) \leq 3$ . By the induction hypothesis,  $\text{idor}(G - H) \leq |V(G - H)| - \delta(G - H)/2 + 1 \leq |V(G - H)| - \delta(G)/2 + 5/2$ . Hence, by Lemma 6,  $\text{idor}(G) \leq \text{idor}(H) + \text{idor}(G - H) \leq |V(G)| - \delta(G)/2 + 1$ .  $\square$

The *average degree* of a graph  $G$ , denoted by  $\text{Ad}(G)$ , is  $\frac{2|E(G)|}{|V(G)|}$ . The *maximum average degree* of a graph  $G$ , denoted by  $\text{Mad}(G)$ , is  $\max\{\text{Ad}(H) \mid H \text{ subgraph of } G\}$ .

**Proposition 13** (Folklore). Every graph contains an (induced) subgraph with minimum degree at least  $\text{Ad}(G)/2$ .

Propositions 12, 13, and 9 imply the following.

**Corollary 14.** For every graph  $G$ ,  $\text{idor}(G) \leq |V(G)| - \text{Mad}(G)/4 + 1 \leq |V(G)| - \text{Ad}(G)/4 + 1$ .

This result suggests the more general problems:

**Problem 15.**

- What is the maximum value  $g(k)$  such that  $\text{idor}(G) \leq |V(G)| - g(k)$  for every graph  $G$  with  $\delta(G) \geq k$  ?
- What is the maximum value  $g_1(\alpha)$  such that  $\text{idor}(G) \leq |V(G)| - g_1(\alpha)$  for every graph  $G$  with  $\text{Ad}(G) \geq \alpha$  ?
- What is the maximum value  $g_2(\alpha)$  such that  $\text{idor}(G) \leq |V(G)| - g_2(\alpha)$  for every graph  $G$  such that  $\text{Mad}(G) \geq \alpha$  ?

Observe that  $g_1 = g_2$  : As  $\text{Ad} \geq \text{Mad}$ , we trivially have that  $g_2(\alpha) \geq g_1(\alpha)$ . Conversely, let  $G$  be a graph with  $\text{Mad}(G) = \alpha$ . It contains a subgraph  $H$  such that  $\text{Mad}(H) = \text{Ad}(H) = \alpha$ .  $\text{idor}(H) \leq |V(H)| - g_1(\alpha)$ , so by Lemma 6,  $\text{idor}(G) \leq |V(G)| - g_1(\alpha)$ . Hence  $g_2(\alpha) \geq \frac{k}{4} - 1$ .

Proposition 12 and Corollary 14 yields the lower bounds  $g(k) \geq \frac{k}{2} - 1$  and  $g_1(\alpha) \geq \frac{\alpha}{4} - 1$ . We now show some upper bounds.

We denote by  $K_{a,b}$  the complete bipartite graph with  $a$  vertices in one part and  $b$  vertices in the other.

**Proposition 16.**  $\text{idor}(K_{k,n-k}) \geq n - 2^k + 1$ .

*Proof.* Let  $(A, B)$  be a bipartition of  $K_{k,n-k}$  with  $|A| = k$  and  $|B| = n - k$ . Let  $C$  be an identifying code in an orientation  $D$  of  $K_{k,n-k}$ . Set  $P = A \cap C$ ,  $Q = B \setminus C$ ,  $p = |P|$ , and  $q = |Q|$ . The identifiers of vertices in  $Q$  are distinct non-empty subsets of  $P$ . Hence  $q \leq 2^p - 1$ . Therefore  $|C| = n - k - q + p \geq n - k - 2^p + p + 1$ . Now  $p \in [k]$ , so  $|C| \geq n - 2^k + 1$ .  $\square$

Observe that if  $n \geq 2k$ ,  $\delta(K_{k,n-k}) = k$ ,  $\text{Ad}(K_{k,n-k}) = \text{Mad}(K_{k,n-k}) = 2k - \frac{2k}{n}$ . Thus

$$\begin{aligned} g(k) &\leq 2^k - 1 ; \\ g_1(\alpha) &\leq 2^{\lfloor \alpha/2 \rfloor} - 1. \end{aligned}$$

### 3.2 Lower bounds

**Theorem 17.** Let  $G$  be a graph with maximum degree  $\Delta$ . Then  $\text{idor}(G) \geq \frac{2}{\Delta+2}|V(G)|$ .

*Proof.* Let  $C$  be an identifying code of an orientation  $D$  of  $G$ . We shall use the Discharging Method to prove that  $|C| \geq \frac{2}{\Delta+2}|V(G)|$ . This method was often use to get lower bounds on the size of identifying codes, in particular for infinite grids, see e.g. [2, 6, 7, 8, 9].

We give an initial charge of 1 to every vertex  $v$ . Hence the total charge is  $|V(G)|$ . Then every vertex distribute its charge uniformly to the vertices in its identifier. In other word, we apply the following rule: every vertex sends  $\frac{1}{|I(v)|}$  to every vertex of  $I(v)$ .

Consider now the final charge  $w(v)$  of a vertex. If  $v \notin C$ , then it gives all its charge and receives nothing, so  $w(v) = 0$ . If  $v \in C$ , then it gives all its charge, but receives

from all the vertices whose identifier contains  $v$  (including itself). Observe that there are at most  $\Delta + 1$  such vertices, and that at most one of this vertex has  $\{v\}$  for identifier. So  $v$  receives 1 from at most one vertex, and at most  $1/2$  from all the others. Hence  $w(v) \leq 1 + \frac{\Delta}{2} = \frac{\Delta+2}{2}$ .

The total charge is unchanged, so

$$|V(G)| = \sum_{v \in V(G)} w(v) = \sum_{v \in C} w(v) \leq |C| \frac{\Delta+2}{2}.$$

So,  $|C| \geq \frac{2}{\Delta+2} |V(G)|$ . □

The bound  $\frac{2}{\Delta+2} |V(G)|$  of Theorem 17 is tight as shown by the following proposition. The *incidence graph* of a graph  $H$  is the bipartite graph  $G$  with bipartition  $(V(H), E(H))$  where there is an edge between  $v \in V(H)$  and  $e \in E(H)$  if and only if  $v$  is an end vertex of  $e$ .

**Proposition 18.** *If  $G$  is the incidence graph of a  $\Delta$ -regular graph, then  $\text{idor}(G) = \frac{2}{\Delta+2} |V(G)|$ .*

*Proof.* Let  $H$  be the  $\Delta$ -regular graph with incidence graph  $G$ . Set  $p = |V(H)|$ . We have  $|E(H)| = \frac{\Delta p}{2}$ , and so  $|V(G)| = \frac{(\Delta+2)p}{2}$ . Let  $D$  be the orientation of  $G$  in which all edges are oriented from  $E(H)$  towards  $V(H)$ . The set  $V(H)$  is an identifying code of  $G$ . Indeed the identifier of every  $v \in V(H)$  is  $\{v\}$ , and the identifier of every  $e = uv \in E(H)$  is  $\{u, v\}$ . Hence  $\text{idor}(G) \leq p = \frac{2}{\Delta+2} |V(G)|$ . By Theorem 17, we get  $\text{idor}(G) = \frac{2}{\Delta+2} |V(G)|$ . □

### 3.3 Trees

Of course better lower and upper bounds might be obtained when  $G$  belongs to some graph classes. In this subsection, we give tight lower and upper bounds for trees. Identifying codes of trees have been studied in [1].

We shall need the following lemma.

**Lemma 19.** *Let  $G$  be a graph having three vertices  $u, v, w$  such that  $N(u) = \{v\}$  and  $N(v) = \{u, w\}$ . Any identifying code of any orientation of  $G$  contains at least two vertices in  $\{u, v, w\}$ . Moreover, there is a set of size  $\text{idor}(G)$  containing  $w$  that is an identifying code of an orientation  $D$  of  $G$ .*

*Proof.* Because  $u$  and  $v$  have non-empty distinct identifiers, any identifying code of any orientation of  $G$  contains at least two vertices in  $\{u, v, w\}$ .

Suppose now that  $C$  is an identifying code of an orientation  $D$  of  $G$ . If  $C$  does not contain  $w$ , then  $\{u, v\} \subseteq C$ . Let  $C' = (C \setminus \{v\}) \cup \{w\}$  and let  $D'$  be the orientation of  $G$  obtained from  $D$  by reorienting (if necessary) all edges incident to  $v$  away from this vertex. It is simple matter to check that  $C'$  is an identifying code of  $D'$ . □

**Theorem 20.** *If  $T$  is a tree of order  $n \geq 2$ , then  $\text{idor}(T) \geq \lceil (n+1)/2 \rceil$ .*

*Proof.* Let us prove the result by induction on  $n$ , the result holding trivially when  $n \leq 3$ . Let  $T$  be a tree of order  $n \geq 3$ . Let  $P = (v_1, \dots, v_k)$  be a diameter of  $T$ , and let  $L$  be the set of all neighbours of  $v_2$  distinct from  $v_3$ . All vertices in  $L$  are leaves of  $T$  because  $P$  is a diameter,

Let  $C$  be an identifying code of an orientation  $D$  of  $T$ . Assume that  $v_2 \notin C$ . Then  $L \subseteq C$ . Moreover,  $C \setminus L$  is an identifying code of  $D - (L \cup \{v_2\})$ . Thus, by the induction hypothesis,  $|C \setminus L| \geq \lceil (n - |L|)/2 \rceil$ , so  $|C| \geq \lceil (n + 1)/2 \rceil$ . Henceforth we may assume that  $v_2 \in C$ . Moreover we may assume that  $v_2$  dominates all vertices in  $L \cap C$  in  $D$ .

Because at most one vertex has  $\{v_2\}$  as an identifier, all vertices of  $L$  except possibly one are in  $C$ .

We distinguish two cases according to whether  $|L| \geq 2$  or not.

Case 1: Assume  $|L| \geq 2$ . If  $L \subseteq C$ , then  $C \setminus \{v_1\}$  is an identifying code of  $D - v_1$ , as  $v_2$  has a vertex of  $L$  in its identifier in  $D - v_1$ . Thus, by the induction hypothesis,  $|C \setminus \{v_1\}| \geq \lceil n/2 \rceil$ , so  $|C| \geq \lceil (n + 1)/2 \rceil$ . Henceforth, we may assume that one vertex in  $L$ , say  $v_1$ , is not in  $C$ . Observe that  $I(v_1) = \{v_2\}$ . Henceforth,  $C \setminus L$  is an identifying code of  $D - L$  because either  $v_3$  dominates  $v_2$  and its identifier in  $D - L$  is not  $\{v_2\}$  because  $C$  was a code in  $D$ , or  $v_2$  dominates  $v_3$  and  $I(v_3)$  does not contain  $v_2$ . Thus, by the induction hypothesis,  $|C \setminus L| \geq \lceil (n - |L| + 1)/2 \rceil$ , so  $|C| = |C \setminus L| + |L| - 1 \geq \lceil (n + 1)/2 \rceil$ , because  $|L| \geq 2$ .

Case 2: Assume  $|L| = 1$ . By Lemma 19, we may also assume that  $v_3 \in C$ . If  $v_1 \notin C$ , then  $I(v_1) = \{v_2\}$ . Necessarily,  $I(v_2) = \{v_2, v_3\}$ . In particular,  $v_2$  dominates  $v_3$  in  $D$ . Consequently,  $C \setminus \{v_2\}$  is an identifying code of  $D - \{v_1, v_2\}$ . Thus, by the induction hypothesis,  $|C \setminus \{v_2\}| \geq \lceil (n - 1)/2 \rceil$ , so  $|C| \geq \lceil (n + 1)/2 \rceil$ . Henceforth, we may assume that  $v_1 \in C$ . Without loss of generality, we may assume that  $v_2$  dominates  $v_1$ . Hence  $C \setminus \{v_1\}$  is an identifying code of  $D - v_1$ . Thus, by the induction hypothesis,  $|C \setminus \{v_1\}| \geq \lceil n/2 \rceil$ , so  $|C| \geq \lceil (n + 1)/2 \rceil$ .  $\square$

The bound of Theorem 20 is tight as shown by paths:

**Proposition 21.** *Let  $P_n$  be the path of order  $n$ .  $\text{idor}(P_n) = \lceil (n + 1)/2 \rceil$ .*

*Proof.* Set  $P_n = (v_1, \dots, v_n)$ .

Let  $D_n$  be the orientation of  $P_n$  where  $v_i \rightarrow v_{i+1}$  if and only if  $i$  is even.

If  $n$  is odd, then  $\{v_i \mid i \text{ odd}\}$  is an identifying code of  $P_n$ : if  $i$  is odd then  $I(v_i) = \{v_i\}$ , and if  $i$  is even then  $I(v_i) = \{v_{i-1}, v_{i+1}\}$ .

If  $n$  is even, then  $\{v_i \mid i \text{ odd}\} \cup \{v_n\}$  is an identifying code of  $P_n$ : if  $i$  is odd then  $I(v_i) = \{v_i\}$ , if  $i$  is even and  $i < n$  then  $I(v_i) = \{v_{i-1}, v_{i+1}\}$ , and  $I(v_n) = \{v_{n-1}, v_n\}$ . Hence  $\text{idor}(P_n) \leq \lceil (n + 1)/2 \rceil$ .

By Theorem 20,  $\text{idor}(P_n) = \lceil (n + 1)/2 \rceil$ .  $\square$

$K_1$  and  $K_2$  are trees  $T$  such that  $\text{idor}(T) = |V(T)|$ . However, Proposition 9 asserts that they are the only ones. By Proposition 21, if  $T = P_4$ , then  $\text{idor}(T) = |V(T)| - 1$ . The stars also satisfy this equality. Recall that the *star* of order  $n$  is the graph  $S_n$  in which a vertex called the *centre* is adjacent to all other vertices, called *leaves*.

**Proposition 22.**  $\text{idor}(S_n) = n - 1$ .

*Proof.* Let  $c$  be the centre of the star and  $u_1, \dots, u_{n-1}$  be its leaves. One can check that  $V(S_n) \setminus u_1$  is an identifying code of the orientation of  $S_n$  with arc set  $\{(u_1, c)\} \cup \{(c, u_i) \mid 2 \leq i \leq n-1\}$ .

Assume now that  $C$  is an identifying code of an orientation  $D$  of  $S_n$ . If all leaves of  $S_n$  are in  $C$ , then  $|C| \geq n - 1$ . Henceforth, we assume that a leaf, say  $u_1$  is not in  $C$ . Since  $I(u_1) \neq \emptyset$ , necessarily,  $(u_1, c) \in A(D)$ ,  $c \in C$  and  $I(u_1) = c$ . Moreover, since  $I(u_i) \neq I(u_1)$ , we necessarily have  $u_i \in C$  for  $2 \leq i \leq n - 1$ . Thus  $|C| \geq n - 1$ .  $\square$

However, we can improve on the upper bound  $\text{idor}(T) = |V(T)| - 1$  via some tree parameters. Let us denote by  $\text{leav}(T)$  the number of leaves in a tree. A *leaf* in a tree is a vertex of degree at most 1. When  $|V(T)| > 2$ , we have  $\text{leav}(T) \leq |V(T)| - 1$ . A vertex of a tree that is not a leaf is called a *node*.

**Theorem 23.** *If  $T$  is a tree, then  $\text{idor}(T) \leq \left\lfloor \frac{|V(T)| + \text{leav}(T)}{2} \right\rfloor$ .*

*Proof.* Let  $T$  be a tree, and let  $L$  be its set of leaves.  $T - L$  is a tree. Therefore it has a stable set  $S$  of size at least  $\left\lceil \frac{|V(T-L)|}{2} \right\rceil$ . Let  $D$  be an orientation of  $T$  such that all edges between  $S$  and  $V(T) \setminus S$  are oriented away from  $S$ .

We claim that  $C = V(T) \setminus S$  is an identifying code of  $D$ . Indeed every vertex of  $S$  has at least two neighbours in  $T$  which are its out-neighbours in  $D$ . Hence the identifier of every vertex is non-empty, since the identifier of each vertex  $c \in C$  contains  $c$ . Furthermore, consider two distinct vertices  $u, v$  of  $V(T)$ . If one of them, say  $u$ , is in  $S$ , then  $N_T(u) = I(u)$ . But  $u$  is the only vertex adjacent to the (at least two) vertices of  $N_T(u)$  in  $T$ , because  $T$  is acyclic. Hence  $I(u) \neq I(v)$ . If both  $u$  and  $v$  are not in  $S$ , then they are in  $C$ . Without loss of generality,  $u$  does not dominate  $v$ , and so  $v \in I(v) \setminus I(u)$ , so  $I(u) \neq I(v)$ . This proves that  $C$  is an identifying code of  $D$ .

Hence  $\text{idor}(T) \leq |C| = |V(T)| - |V(S)| \leq |V(T)| - \left\lceil \frac{|V(T-L)|}{2} \right\rceil \leq \left\lfloor \frac{|V(T)| + \text{leav}(T)}{2} \right\rfloor$ .  $\square$

Observe that the bound of Theorem 23 is attained for stars and paths.

Note moreover that  $\text{leav}(T) \leq \alpha(T)$  and  $|V(T)|/2 \leq \alpha(T)$ . This directly implies the following corollary.

**Corollary 24.** *If  $T$  is a tree, then  $\text{idor}(T) \leq \frac{3\alpha(T)}{2}$ .*

This upper bound is attained for the paths  $P_2$  and  $P_4$  of order 2 and 4 respectively. However, they are the only ones.

**Theorem 25.** *Let  $T$  be a tree different from  $P_2$  and  $P_4$ . Then  $\text{idor}(T) \leq \frac{4}{3}\alpha(T)$ .*

*Proof.* We prove the result by induction. One can easily check that the results holds for  $|V(T)| \leq 5$ . In particular, for every tree  $T$  of order 5, we have  $\text{idor}(T) \leq \frac{4}{3}\alpha(T)$ . Suppose now that  $|V(T)| \geq 6$ .

Assume that some vertex  $v \in T$  is incident to at least two leaves  $v_1, v_2$ . Clearly,  $\alpha(T) = \alpha(T - v_1) + 1$  and, by Lemma 6,  $\text{idor}(T) \leq \text{idor}(T - v_1) + 1$ . By the induction hypothesis,  $\text{idor}(T - v_1) \leq \frac{4}{3}\alpha(T - v_1)$ . Consequently,  $\text{idor}(T) \leq \frac{4}{3}\alpha(T)$ .

Henceforth, we may assume that every vertex of  $T$  is adjacent to at most one leaf. We distinguish three cases, depending on the value of  $\text{leav}(T)$ :

- If  $\text{leav}(T) \leq n/3$ , we apply Theorem 23 and obtain  $\text{idor}(T) \leq \frac{2}{3}n = \frac{4}{3} \cdot \frac{n}{2} \leq \frac{4}{3}\alpha(T)$ .
- If  $n/3 < \text{leav}(T) < n/2$ , then there is a node  $v \in T$  not adjacent to any leaf. Orient  $T$  as an in-arborescence of root  $v$ , while setting  $C$  to be the set of nodes. We observe that all leaves and  $v$  have an identifier of cardinality 1, while all nodes distinct from  $v$  have identifiers of cardinality 2 (which are thus different from each other). This ensures that  $\text{idor}(T) \leq n - \text{leav}(T) \leq \frac{2}{3}n \leq \frac{4}{3}\alpha(T)$ .
- If  $\text{leav}(T) = n/2$ , then we define an orientation as previously using any leaf vertex  $v$  as the root, and set the code  $C$  to contain all nodes of  $T$  and  $v$ . This time, all nodes have an identifier of cardinality 2 while all leaves have an identifier of cardinality 1. This ensures that  $\text{idor}(T) \leq n/2 + 1 \leq \frac{2}{3}n \leq \frac{4}{3}\alpha(T)$  as  $n > 5$ .  $\square$

Theorem 25 is best possible. Indeed consider  $A_{6p}$  the tree which is the union of  $2p$  disjoint paths  $Q_i = (a_i, b_i, c_i)$ ,  $1 \leq i \leq 2p$  and the path  $P = (a_1, a_2, \dots, a_{2p})$ . One can easily see that a maximum matching of  $A_{6p}$  is  $\{c_i \mid 1 \leq i \leq 2p\} \cup \{a_{2j} \mid 1 \leq j \leq 2p\}$ , so  $\alpha(A_{6p}) = 3p$ . On the other hand, for any code  $C$  of any orientation  $D$  of  $A_{6p}$ , there must be at least two vertices of  $C$  in each  $\{a_i, b_i, c_i\}$ , by Lemma 19. Hence  $\text{idor}(A_{6p}) \geq 4p$ . Thus  $\text{idor}(A_{6p}) \geq \frac{4}{3}\alpha(A_{6p})$ .

## 4 Relations between id and idor

We denote by  $E(A, B)$  the set of edges with an endvertex in  $A$  and the other in  $B$ .

**Theorem 26.**  $\text{idor}(G) \leq 2 \text{id}(G)$  for all graph  $G$ .

*Proof.* Let  $G$  be a graph. If  $G$  has some twins, then  $\text{id}(G) = +\infty$  and the results holds trivially. Assume now that  $G$  is twin-free, that is  $\text{id}(G) < +\infty$ . Let  $C$  be a code of  $G$  with cardinality  $\text{id}(G)$  and set  $U = V(G) \setminus C$ . Let  $D$  be an orientation of  $G$  such that all edges of  $E(U, C)$  are oriented towards  $C$ . Observe that for every  $u \in U$ ,  $I_{C,D}(u) \cap C = I_{C,G}(u) \neq \emptyset$  and for every vertex  $w \in C$ ,  $w \in I_{C,D}(w)$ . Therefore,  $I_{C,D}(v) \neq \emptyset$  for all  $v \in V(G)$ .

Observe that for every two distinct vertices  $u_1, u_2 \in U$ , we have  $I_{C,D}(u_1) \cap C = I_{C,G}(u_1) \neq I_{C,G}(u_2) = I_{C,D}(u_2) \cap C$ , so  $I_{C,D}(u_1) \neq I_{C,D}(u_2)$ . Recall moreover, that for two vertices  $w_1, w_2$  in  $C$ , we have  $I_{C,D}(w_1) \neq I_{C,D}(w_2)$ . Hence  $C$  is almost an identifying code of  $D$ . The only thing that prevents it to be a code are pairs  $(u, w) \in U \times C$  such that  $I_{C,D}(u) = I_{C,D}(w)$ . Call such pairs, *bad pairs*. Observe that for any bad pair  $(u, v)$ , we have  $(u, v) \in A(D)$ . Moreover, a vertex  $w \in C$  is in at most one bad pair, so there are at most  $|C|$  bad pairs. Let  $U'$  be the set of vertices of  $U$  which are the first vertex of a bad pair. One can easily check that  $C' = C \cup U'$  is an identifying code  $D$  since the identifier

of the two elements of a bad pair are now distinct. Moreover,  $|U'| \leq |C|$ , so  $|C'| \leq 2|C|$ . Hence  $\text{id}_o(G) \leq 2 \text{id}(G)$ .  $\square$

**Theorem 27.**  $\text{id}_o(G) \leq \frac{3}{2} \text{id}(G)$  for all graph  $G$ .

This theorem follows directly from the following lemma.

**Lemma 28.** Let  $G$  be a graph having an identifying code  $C$ . There exists an orientation  $D$  of  $G$  having an identifying code  $C'$  with  $C \subseteq C'$  and  $|C'| \leq \frac{3}{2}|C|$ .

*Proof.* Set  $U = V(G) \setminus C$ .

We first orient all the edges of  $E(U, C)$  towards  $C$ . We then partition our graph using the following process. Set  $U_1 := U$ ,  $C_1 := C$  and  $i = 1$ . As long as there is a vertex in  $U_i$  having at least two neighbours in  $C_i$ . Choose such a vertex  $u_i$ . Set  $B_i = N(u_i) \cap C$ ,  $S_i = N(B_i) \cap U_i \setminus \{u_i\}$ ,  $C_{i+1} = C_i \setminus B_i$ ,  $U_{i+1} = U_i \setminus (S_i \cup \{u_i\})$ , and  $i := i + 1$ . Reorient the edges of  $E(u_i, B_i)$  towards  $u_i$ , and reorient the edges of  $E(C_{i+1}, B_i)$  towards  $B_i$ . See Figure 1.

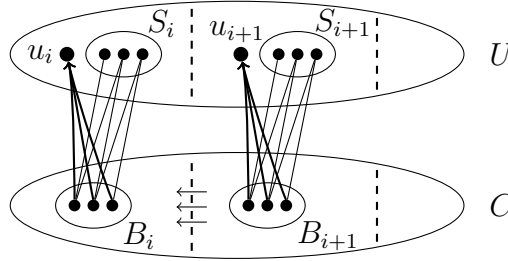


Figure 1: Partitioning the graph

Let  $p$  be the index  $i$  at which the process stops. Each vertex of  $U_p$  has no neighbour in  $C \setminus C_p$  for otherwise it would belong to some  $S_i$ . Hence it has degree 1 in  $C$  and its neighbour is in  $C_p$ . For every,  $u \in U_p$ , let  $c(u)$  be its neighbour. We have  $I_{C,G}(u) = \{c(u)\}$ , therefore  $c(u) \neq c(u')$  for all  $u, u' \in U_p$ . Let  $B_p$  be the set of vertices of  $C_p$  having a neighbour in  $U_p$ , i. e.  $B_p = \{c(u) \mid u \in U_p\}$ . Orient all edges of  $E(B_p, C_p \setminus B_p)$  towards  $C_p \setminus B_p$ . For each connected component  $H$  of  $G\langle B_p \rangle$  of order at least 3, choose a vertex  $r_H$  in  $H$ , orient the edges of  $H$  such that all vertices of  $H$  except possibly  $r_H$  have out-degree at least 1, and let  $w_H$  be the vertex of  $U_p$  such that  $c(w_H) = r_H$ . Let  $W$  be the set of all such  $w_H$ . Now consider the connected components  $H$  of  $G\langle B_p \rangle$  of order 2 with vertices  $c(u_1)$  and  $c(u_2)$ . Since  $I_{C,G}(u_1) = \{c(u_1)\}$  and  $I_{C,G}(u_2) = \{c(u_2)\}$ , there is a vertex  $b_H \in \{c(u_1), c(u_2)\}$  such that  $I_{C,G}(b_H)$  contains a vertex in  $C \setminus \{c(u_1), c(u_2)\}$ . Let  $a_H$  be the other vertex of  $\{c(u_1), c(u_2)\}$ . Orient the edge  $a_H b_H$  towards  $b_H$ .

Set  $U' = U \setminus (\{u_1, \dots, u_{p-1}\} \cup W)$ . Orient the edges of  $E(U \setminus U', U')$  towards  $U'$ . Finally orient all the unoriented edges arbitrarily to obtain an orientation  $D$  of  $G$ . Set  $C' = C \cup \{u_1, \dots, u_{p-1}\} \cup W$ . Observe that  $|\{u_1, \dots, u_p\} \cup W| \leq |C|/2$ , so  $|C'| \leq 3|C|/2$ . We shall now prove that  $C'$  is an identifying code of  $D$ .

Observe first that for every vertex  $u \in U'$ ,  $I_{C,G}(u) = I_{C',D}(u)$ . In particular,  $I_{C',D}(u)$  is not empty. Since the identifier of the vertices of  $C'$  are trivially non-empty, all the identifiers are non-empty. Let us now prove that they are all distinct.

Two vertices in  $C'$  have clearly distinct identifiers. If  $u'_1$  and  $u'_2$  are two vertices of  $U'$ , then  $I_{C',D}(u'_1) = I_{C,G}(u'_1) \neq I_{C,G}(u'_2) = I_{C',D}(u'_2)$ .

Consider now a vertex  $u \in U'$  and  $v \in C'$ .

If  $v \in \{u_1, \dots, u_{p-1}\} \cup W$ , then  $v \in I_{C',D}(v) \setminus I_{C',D}(u)$  because all the edges of  $E(U \setminus U', U')$  are oriented towards  $U'$ ; If  $v$  is in  $B_i$  for some  $1 \leq i \leq p-1$ , then  $u_i \in I_{C',D}(v) \setminus I_{C',D}(u)$ ; If  $uv$  is not an arc, then  $v \in I_{C',D}(v) \setminus I_{C',D}(u)$ .

Henceforth we may assume that  $u \in U_p$ ,  $v \in B_p$  and  $c(u) = v$ . Note that  $I_{C',D}(u) = \{v\}$ . Let  $H$  be the component of  $v$  in  $G\langle B_p \rangle$ . If  $H$  has order at least 3, then  $v \neq r_H$  because  $u \neq w_H$  as  $u \in U'$ . Hence  $v$  has an out-neighbour in  $H$ , and so  $I_{C',D}(v) \neq \{v\}$ . If  $H$  has order 2, then  $v$  is either  $a_H$  or  $b_H$ . If  $v = a_H$ , then  $v$  dominates  $b_H$ . If  $v = b_H$ , then  $v$  has in  $G$  a neighbour  $w$  in  $C \setminus \{a_H\}$ . Since  $H$  is a component of  $G\langle B_p \rangle$ , necessarily  $w \in \bigcup_{i=1}^{p-1} B_i$  and so  $v$  dominates  $w$  in  $D$  by construction. In both cases,  $I_{C',D}(v) \neq \{v\}$ . If  $H$  has order 1, then  $v$  is adjacent to a vertex  $w$  in  $C$ , because  $I_{C,G}(v) \neq I_{C,G}(u) = \{v\}$ . Since  $v$  is isolated in  $G\langle B_p \rangle$ ,  $w \in \bigcup_{i=1}^{p-1} B_i$  and so  $v$  dominates  $w$  in  $D$  by construction. Thus  $I_{C',D}(v) \neq \{v\}$ .  $\square$

The bound of Theorem 27 is tight, as shown by the graph  $R$  depicted in Figure 2 and the graph made of disjoint copies of it.

**Proposition 29.** *Let  $R$  be the graph depicted in Figure 2.  $\text{idor}(R) = 6$  and  $\text{id}(R) = 4$ .*

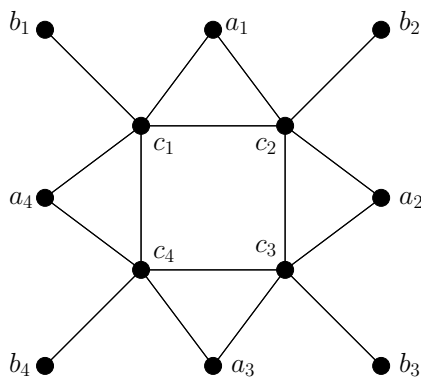


Figure 2: The graph  $R$

*Proof.* An identifying code of  $G$  must contain at least one vertex in each set  $\{b_i, c_i\}$ , so  $\text{id}(G) \geq 4$ . Now  $\{c_1, c_2, c_3, c_4\}$  is an identifying code of  $G$ . Indeed, for all  $i$ , we have  $I(a_i) = \{a_i, a_{i+1}\}$  (indices are modulo 4),  $I(b_i) = \{c_i\}$ , and  $I(c_i) = \{c_{i-1}, c_i, c_{i+1}\}$ . Hence  $\text{id}(R) = 4$ .

By Theorem 27,  $\text{idor}(R) \leq 6$ . We shall now prove that  $\text{idor}(R) \geq 6$ .

Assume for a contradiction that  $D$  contains an orientation of  $D$  having an identifying code  $C$  of order less than 6. Moreover, we take such a pair  $(D, C)$  such that  $C \cap \{c_1, c_2, c_3, c_4\}$  is maximum.

We claim that  $\{c_1, c_2, c_3, c_4\} \subseteq C$ . Indeed, if  $c_i \notin C$ , then the identifier of  $b_i$  is  $\{b_i\}$ , and in particular,  $b_i \in C$ . Let  $C' = (C \setminus \{b_i\}) \cup \{c_i\}$  and let  $D'$  be the digraph obtained from  $D$  by reorienting (if necessary) the edge  $b_i c_i$  towards  $c_i$ . One easily checks that  $C'$  is an identifying code of  $D'$  and so  $(D', C')$  contradicts the maximality of  $(D, C)$ . This proves our claim.

Now since  $|C| < 6$ , we have  $|C \cap \{a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4\}| \leq 1$ . Without loss of generality, we may assume that  $C \cap \{a_1, a_2, a_3, b_1, b_2, b_3\} = \emptyset$ . Therefore for  $i \in \{1, 2, 3\}$ , we have  $I(b_i) = \{c_i\}$ . Consequently,  $i \in \{1, 2\}$ , we have  $I(a_i) = \{c_i, c_{i+1}\}$ . Now  $I(c_2) = \{c_1, c_2, c_3\}$ , and so  $(c_2, c_3) \in A(D)$ . Thus  $c_2 \notin I(c_3)$  and so  $I(c_3) = \{c_3, c_4\}$  because  $\{c_3\} = I(b_3)$ . Now  $I(a_3) = \{c_4\}$ , and so  $b_4 \in I(b_4) \subseteq C$ . Thus  $c_1$  and  $a_4$  must have identifier  $\{c_1, c_4\}$ , a contradiction.  $\square$

**Problem 30.** What is the complexity of the following problems :

- Deciding whether  $\text{idor}(G) < \frac{3}{2} \text{id}(G)$ .
- Given a graph  $G$  with an identifying code  $C$ , deciding whether an orientation  $D$  of  $G$  and an identifying code  $C'$  of  $D$  such that  $C \subseteq C'$  and  $|C'| < \frac{3}{2}|C|$ .

Recall that  $\text{id}(G) \leq 2^{\text{idor}(G)} - 2$  for every twin-free graph  $G$ . The following proposition, observed by Florent Foucaud [10], shows that this bound is tight. Let  $J_{2p+1}$  be the graph obtained from a complete graph on  $2p+1$  vertices by removing a matching of size  $p$ .

**Proposition 31.** Let  $k \geq 1$  be an integer.  $\text{idor}(J_{2^k-1}) = k$  and  $\text{id}(J_{2^k-1}) = 2^k - 2$ .

*Proof.* Twin-free graphs such that  $\text{id}(G) = |V(G)| - 1$  have been characterized by Foucaud et al. [13]. The graphs  $J_{2p+1}$  are among them, so  $\text{id}(J_{2^k-1}) = 2^k - 2$ .

Now, let  $C$  be a set of  $k$  vertices of  $J_{2^k-1}$  such that  $C$  is complete to  $V(J_{2^k-1}) \setminus C$ . Such a set clearly exists by definition of  $J_{2^k-1}$ . Take an arbitrary orientation  $D_C$  of  $G \setminus C$ . Let  $\mathcal{F}$  be the set of closed out-neighbourhoods in  $D_C$ :  $\mathcal{F} = \{N_{D_C}^+(v) \mid v \in C\}$ . Let us label the vertices of  $V(J_{2^k-1}) \setminus C$  with distinct non-empty subsets of  $C$  not in  $\mathcal{F}$ . This is possible because  $J_{2^k-1}$  has  $2^k - 1$  vertices. Let  $D$  be an orientation of  $G$  obtained from  $D_C$  by orienting an edge  $uv$  with  $u \in V(J_{2^k-1}) \setminus C$  and  $v \in C$  from  $u$  to  $v$  if and only if  $v$  is in the label of  $u$ , and orienting the edges with both endvertices in  $V(J_{2^k-1}) \setminus C$  arbitrarily. One easily checks that  $C$  is an identifying code of  $D$ . Thus  $\text{idor}(J_{2^k-1}) = k$ .  $\square$

## 5 Complexity

### 5.1 Complexity of IDOR

Finding the minimum size of an identifying code in a graph or a digraph is an NP-hard problem even on some very restricted classes of graphs or digraphs, see e.g. [5, 12, 11, 4]. We now prove analogous results for  $\text{idor}$ .

**Theorem 32.** IDOR is NP-complete, even when restricted to bipartite cubic graphs or when restricted to bipartite planar graphs of maximum degree 3.

*Proof.* Reduction from VERTEX COVER which consists, given a graph  $G$  and a non-negative integer  $k$ , in deciding whether  $G$  has a vertex cover of size at most  $k$ . Recall that a *vertex cover* in a graph  $G$  is a set  $S$  of vertices such that every edge of  $G$  has an endvertex in  $S$ . The minimum size of a vertex cover of  $G$  is denoted by  $\text{vc}(G)$ . This problem is one of the 21 first problems proved to be NP-complete by Karp [18]. It is known to be NP-complete even when restricted to cubic graphs [15] and when restricted to planar graphs of maximum degree 3 [14]. It can easily be shown that it also remains NP-complete when restricted to such planar graphs with minimum degree at least 2 and maximum degree 3. (From an instance  $G$  of vertex cover, consider the graph  $H = G - (V_1 \cup N(V_1))$  where  $V_1$  is the set of vertices of degree at most 1 and  $N(V_1)$  the set of vertices adjacent to vertices in  $V_1$ .)

Let  $G$  be a graph. Let  $H$  be the graph obtained from  $G$  by replacing each edge by the edge gadget  $F_e$  depicted Figure 3.

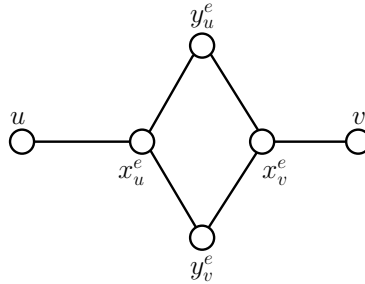


Figure 3: The edge-gadget  $F_e$  for an edge  $e = uv$ .

Observe that  $H$  is bipartite. Moreover if  $G$  is cubic then  $H$  is cubic, and if  $G$  is a planar graph with minimum degree at least 2 and maximum degree 3 so is  $H$ . We claim that  $\text{idor}(H) = \text{vc}(G) + 2|E(G)|$  which will imply the result.

We shall use the name of the vertices as written of this figure. For each edge  $e = uv$ , we set  $X_e = \{x_u^e, x_v^e\}$ .

Assume that  $G$  has a vertex cover  $S$ .

Let  $e = uv$  be an edge of  $G$ . We have  $|\{u, v\} \cap S| \geq 1$ . If  $|\{u, v\} \cap S| = 1$ , we orient the edges of  $F_e$  as shown on Figure 4 left, and if  $|\{u, v\} \cap S| = 2$ , we orient the edges of  $F_e$  as shown on Figure 4 right. This result in an orientation  $D$  of  $G$ . Set  $C = S \cup \bigcup_{e \in E(G)} X_e$ . We claim that the set  $C$  is an identifying code in  $D$ . Indeed, for all  $v \in S$ ,  $I(v) = \{v\}$ , for all  $w \in V(G) \setminus S$ ,  $I(w) = N_H(w)$ , for every edge  $e = uv$  with  $v \in S$ ,  $I(y_u^e) = \{x_u^e, x_v^e\}$ ,  $I(y_v^e) = \{x_v^e\}$ ,  $I(x_v^e) = \{x_v^e, v\}$  and  $I(x_u^e) = \{x_u^e, u\}$  if  $u \in S$  and  $I(x_u^e) = \{x_u^e\}$  otherwise. (Here the fact that  $G$  has minimum degree at least 2 implies that for all  $w \in V(G) \setminus S$ ,  $I(w)$  is non-empty and distinct from the other identifiers.) Hence  $\text{idor}(H) \leq |C| = \text{vc}(G) + 2|E(G)|$ .

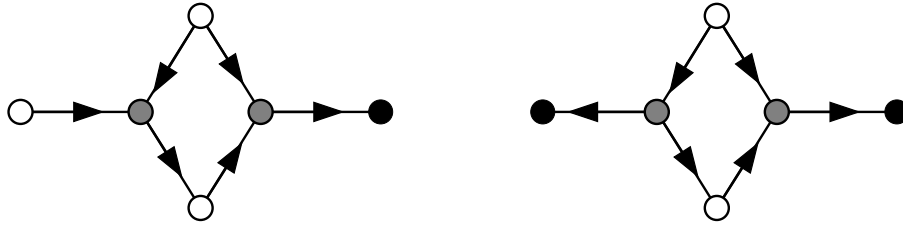


Figure 4: The orientation of the edge-gadget  $F_e$  when one endvertex of  $e$  is in  $S$  (left) and both endvertices of  $e$  are in  $s$  (right). Black vertices are those in  $S$  and black and grey vertices those in  $C$ .

Reciprocally, consider an orientation  $D$  of  $H$  having an identifying code  $C$  of size  $\text{idor}(H)$ . Moreover consider such an identifying code that maximizes the number of vertices in  $V(G)$ .

Let  $E'$  be the set of edges of  $G$  having no endvertex in  $C$ . For each  $e \in E'$  select an endvertex  $x_e$  of  $e$ . Let  $S = (C \cap V(G)) \cup \{x_e \mid e \in E'\}$ . By construction,  $S$  is a vertex cover of  $G$ . Moreover, for every edge  $e = uv$  of  $G$ ,  $C$  contains at least two vertices in  $\{x_u^e, x_v^e, y_u^e, y_v^e\}$  for otherwise either one of the identifiers of  $y_u^e$  and  $y_v^e$  is empty, or they are both the same. Furthermore, if  $e \in E'$ , there are at least three vertices in  $\{x_u^e, x_v^e, y_u^e, y_v^e\}$ : indeed, the identifiers of those four vertices are contained in  $\{x_u^e, x_v^e, y_u^e, y_v^e\} \cap C$  and this set must be of size at least 3 to have four non-empty distinct subsets. Consequently,  $|C| \geq |C \cap V(G)| + 2|E(G)| + |E'| = |S| + 2|E(G)|$ . Hence  $\text{idor}(H) \geq |C| = \text{vc}(G) + 2|E(G)|$ .  $\square$

Observe IDOR can be solved in polynomial time for graphs with maximum degree 2. Indeed such graphs are disjoint union of paths and cycles. Consequently  $\text{idor}$  can be computed in linear time (the time required to compute the connected components of the graphs) using Corollary 8, Proposition 21 and the following proposition.

**Proposition 33.** *Let  $C_n$  be the cycle of length  $n$ . If  $n \neq 4$  then  $\text{idor}(C_n) = \lceil n/2 \rceil$ , and  $\text{idor}(C_4) = 3$ .*

*Proof.* Assume first that  $n \neq 4$ . By Theorem 17,  $\text{idor}(C_n) \geq \lceil n/2 \rceil$ . Let  $D$  be an orientation of  $C_n$  with  $\lceil n/2 \rceil$  sources and  $\lfloor n/2 \rfloor$  sinks. One easily checks that the set of  $\lceil n/2 \rceil$  vertices of  $D$  which are not source is an identifying code of  $D$ . Hence  $\text{idor}(C_n) \leq \lceil n/2 \rceil$ .

Assume now that  $n = 4$ . Equation (2) yields  $\text{idor}(C_4) \geq 3$  and Proposition 9 implies  $\text{idor}(C_4) \leq 3$ .  $\square$

Observe that graphs with maximum degree 2 have treewidth at most 2. More generally, one can compute IDOR can be solved in polynomial time for graphs with bounded treewidth.

**Proposition 34.** *For every fixed integer  $t$ , IDOR is polynomial-time solvable when restricted to the class  $\mathcal{T}_t$  of graphs with treewidth at most  $t$ .*

*Proof.* Using standard dynamic programming. For each bag  $B$ , let  $H_B$  be the subgraph induced by the vertices contains in the bags of the subtree with root  $B$ . For each subset  $A$  and set  $P$  of unordered pairs of  $B$ , we keep the minimum size of a partial code  $C$  of an orientation  $D_B$  of  $H_B \setminus E(G\langle B \rangle)$  such that  $B \cap C = A$  and each pair  $\{u, v\}$  in  $P$  is distinguished (i.e.  $N_D^+(u) \neq N_D^+(v)$ ).  $\square$

**Problem 35.** What is the complexity of IDOR when restricted to cographs? when restricted to split graphs ? when restricted to interval graphs ? when restricted to chordal graphs ?

## 5.2 LARGE-IDOR

The aim of this subsection is to prove that LARGE-IDOR is XP when parameterized by  $k$ . This is equivalent to the following statement.

**Theorem 36.** *For every fixed non-negative  $k$ , the following problem is polynomial-time solvable.*

$k$ -LARGE-IDOR

Input: A graph  $G$ .

Question:  $\text{idor}(G) \geq |V(G)| - k$  ?

Our proof of Theorem 36 is based on the notion of *atom*. A graph  $G$  is a  $k$ -atom if  $\text{idor}(G) = |V(G)| - k$  and  $\text{idor}(H) > |V(H)| - k$  for all proper induced subgraphs  $H$  of  $G$ . Let  $A_k$  be the set of  $k$ -atoms. Observe that Corollary 7 implies that every graph  $G$  such that  $\text{idor}(G) \leq |V(G)| - k$  contains a  $k$ -atom as an induced subgraph. In other words, the set of graphs such that  $\text{idor}(G) > |V(G)| - k$  is the set of  $A_k$ -free graphs.

We shall prove that for every  $k$  the set  $A_k$  is finite. Then a brute-force algorithm checking whether a given graph  $G$  contains a  $(k+1)$ -atom would solve  $k$ -LARGE-IDOR in polynomial time.

**Theorem 37.** *Let  $k$  be a positive integer. Every  $k$ -atom has order at most  $\binom{k}{2} + 2k + 1$ .*

*Proof.* Let  $G$  be a graph of order  $n$ . It has an orientation  $D$  which has an identifying code  $C$  of order  $n - k$ . Set  $U = V(G) \setminus C$ .

To prove the result, it is sufficient to prove that there is a set  $C'$  of size at most  $\binom{k}{2} + k + 1$  which is an identifying code of an orientation  $D'$  of  $G\langle U \cup C' \rangle$ . For such  $C'$  and  $D'$ , the following four properties will be satisfied.

(P1) for every vertex  $v \in V(D')$ ,  $I_{C'}(v) \neq \emptyset$ .

(P2) for every two distinct vertices  $u, u' \in U$ ,  $I_{C'}(u) \neq I_{C'}(u')$ .

(P3) for every two distinct vertices  $v, v' \in C'$ ,  $I_{C'}(v) \neq I_{C'}(v')$ .

(P4) for every  $u \in U$  and  $v \in C$ ,  $I_{C'}(u) \neq I_{C'}(v)$ .

For every two distinct vertices  $u, u' \in U$ , choose a vertex  $s(u, u')$  in  $I_C(u) \Delta I_C(u')$ . Set  $S = \{s(u, u') \mid \{u, u'\} \text{ 2-subset of } U\}$ . We have  $|S| \leq \binom{k}{2}$ . Moreover,  $s(u, u')$  is dominated by  $u$  or  $u'$  in  $D$  because it is in  $I_C(u) \Delta I_C(u')$ . Therefore at most one vertex of  $U$  has no out-neighbour in  $S$ . If such a vertex  $u_0$  exists, let  $t(u_0)$  be its out-neighbour in  $C$  and let  $C_0 = S \cup \{t(u_0)\}$ . Otherwise let  $C_0 = S$ . Let  $D_0 = D \langle C_0 \cup U \rangle$ . We shall now extend  $C_0$  into set  $C_i$  and  $D_0$  into an orientation  $D_i$  of  $G \langle C_i \cup U \rangle$  that satisfies the following properties.

Observe that  $C_0$  and  $D_0$  have been constructed so that P1 and P2 hold. Moreover, P3 holds for every oriented graph since for two elements in the code, one of them is not in the identifier of the other. However,  $C_0$  and  $D_0$  do not necessarily fulfill P4. Let  $B = \{(u_1, v_1), \dots, (u_p, v_p)\}$  be the pairs of  $U \times C_0$  that do not satisfy P4 (i.e.  $I_{C_0}(u_i) = I_{C_0}(v_i)$ ). Observe that, since P3 holds, all the  $u_i$  are distinct, and so  $p \leq k$ .

For  $i = 1, \dots, p$ , choose a vertex  $t(u_i, v_i)$  in  $I_C(u_i) \Delta I_C(v_i)$ . Let  $T = \{t(u_i, v_i) \mid 1 \leq i \leq p\}$ . Note that  $T \cap C_0 = \emptyset$ . Now orient all edges between  $C_0$  and  $T$  towards  $T$  and all edges between  $T$  and  $U$  towards  $U$ . Finally, if  $\{v_i, t(u_i, v_i)\}$  is not an edge, then we reorient  $\{u_i, t(v_i)\}$  in  $(u_i, t(v_i))$ . This results in an orientation  $D'$  of  $G \langle S \cup T \cup U \rangle$ .

We claim that  $C' = C_0 \cup T$  is an identifying code of  $D'$ . Clearly,  $C'$  and  $D'$  satisfy P1, P2 and P3. Let us now prove that P4 also holds. Let  $u$  and  $v$  be two vertices in  $U$  and  $C$  respectively.

Assume first that  $v \in C_0$ . If  $(u, v) \notin B$ , then  $I_{C_0}(u) \neq I_{C_0}(v)$ , so  $I_{C'}(u) \neq I_{C'}(v)$ . If  $(u, v) \in B$ , then either  $\{u, t(u, v)\}$  is an edge of  $G$  and  $t(u, v) \in N^+[v] \setminus N^+[u]$ , or  $\{u, t(u, v)\}$  is not an edge of  $G$  and  $t(u, v) \in N^+[u] \setminus N^+[v]$ . In both cases,  $I_{C'}(u) \neq I_{C'}(v)$ . Assume now that  $v \in T$ . There exists  $i$  such that  $v = t(u_i, v_i)$ . If  $u \neq u_i$ , then  $v \notin N^+[u]$  and so  $I_{C'}(u) \neq I_{C'}(v)$ . If  $u = u_i$ , then either  $v \notin N^+[u]$ , or  $\{v_i, v\}$  is not an edge of  $G$ . In both cases,  $I_{C'}(u) \neq I_{C'}(v)$ .  $\square$

In view of Theorem 37, one naturally asks the following questions.

**Problem 38.** What is precisely the set of  $k$ -atoms ? In particular, what is the maximum size of a  $k$ -atom ? How many  $k$ -atoms are there ?

Note that Proposition 9 implies that the set of 1-atoms is  $\{K_3, P_3\}$ .

### 5.3 ISCODE

**Problem 39.** What is the complexity of the following problem :

ISCODE

Input: A graph  $G$  and a set  $C \subseteq V(G)$ .

Question: Is there an orientation  $D$  of  $G$  for which  $C$  is an identifying code ?

**Theorem 40.** ISCODE is NP-complete.

*Proof.* ISCODE is clearly in NP. An orientation  $D$  of  $G$  in which  $C$  is an identifying code is clearly a certificate.

Let us now prove that IsCODE is NP-hard. We give a reduction from 3, 4-SAT which is a restriction of 3-SAT to instances in which each variable occurs at most 4 times. This problem was shown NP-complete by Tovey [23].

Consider a 3, 4-SAT formula. We first double every clause to obtain an equivalent 3, 4-SAT formula  $\mathcal{F}$  with  $2m$  clauses  $C_1, \dots, C_{2m}$  on  $n$  variables  $x_1, \dots, x_n$ . Let  $G$  be the graph and  $C$  its subset of vertices constructed as follows. For each variable  $x_i$ , we create a variable gadget  $VG_i$  with vertex set  $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2, y_i\}$ , and edge set  $\{x_i^1 y_i, x_i^2 y_i, \bar{x}_i^1 y_i, \bar{x}_i^2 y_i\}$ . For each clause  $C_j = \ell_j^1 \vee \ell_j^2 \vee \ell_j^3$ , we create a clause gadget  $CG_j$  with vertex set  $\{\ell_j^1, \ell_j^2, \ell_j^3, z_j\}$ , and edge set  $\{\ell_j^1 z_j, \ell_j^2 z_j, \ell_j^3 z_j\}$ . See Figure 5. Moreover, if  $\ell_j^k$ ,  $k \in [3]$ , is the negated literal  $\bar{x}_i$  then we connect the vertex  $\ell_j^k$  to  $x_i^1$  and  $x_i^2$ , and if  $\ell_j^k$  is the non-negated literal  $x_i$  then we connect the vertex  $\ell_j^k$  to  $\bar{x}_i^1$  and  $\bar{x}_i^2$ .

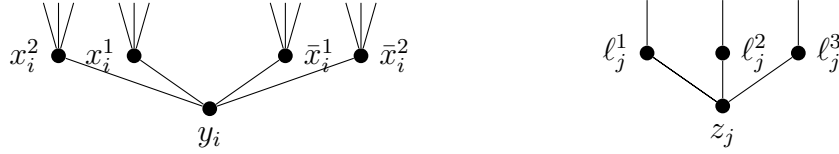


Figure 5: Variable gadget  $VG_i$  (left) and clause gadget  $CG_j$  (right).

We define  $C = \bigcup_{i=1}^n V(VG_i) \cup \bigcup_{j=1}^{2m} V(CG_j)$ .

The two gadgets are very similar in their principle: we want to ensure that, in any orientation of  $G$  in which  $C$  is an identifying codes, every variable/clause gadget has at least one vertex whose edges leaving the gadget are all directed away from it. To this aim, we add new vertices in  $G$  (but not in  $C$ ). For every vertex  $v$  in  $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ , let  $P[v] = N_G[v] \setminus \{y_i\}$ .

- For every variable  $x_i$ , we add a vertex whose unique neighbour is  $y_i$ ;
- For every clause  $C_j$ , we add a vertex whose unique neighbour is  $z_j$ ;
- For every set  $S$  which is a non-empty strict subset of  $P[v]$  for some vertex  $v \in \bigcup_{i=1}^n \{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ , we add a vertex whose neighbourhood is  $S$ .

Observe that in the last step, for every  $\ell_j^k$ , we create a vertex of  $V(G) \setminus C$  whose unique neighbour is  $\ell_j^k$ .

Observe that since every variable occurs in at most 8 clauses, there are at most  $2^9 - 2 = 510$  non-empty strict subsets of  $P[v]$  for all  $v \in \bigcup_{i=1}^n \{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ . Hence  $G$  has a size at most  $2046n + 2m$ .

Let us now show that  $\mathcal{F}$  is satisfiable if and only if  $G$  has an orientation in which  $C$  is an identifying code.

Suppose first that  $C$  is an identifying code of an orientation  $D$  of  $G$ . Observe that every vertex  $v \in V(G) \setminus C$  must satisfy  $I(v) = N_G(v)$ , and all edges of  $E(V(G) \setminus C, C)$  are oriented towards  $C$ .

In particular, for every  $v \in C$ , there is a vertex in  $V(G) \setminus C$  whose identifier is  $\{v\}$ . Consequently, all vertices of  $C$  must have at least one out-neighbour in  $C$ . In particular, each  $y_i$  has an out-neighbour which must be in  $C$  so in  $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ . Furthermore, we can assume that  $y_i$  has exactly one out-neighbour in  $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$  because reversing some arcs towards  $y_i$  leaves  $C$  an identifying code of  $D$  (provided that  $y_i$  keeps one out-neighbour). Let  $w_i$  be the out-neighbour of  $y_i$ . As mentioned above it is in  $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ . The identifier of  $w_i$  is contained in  $P[w_i]$ . Now by construction and the fact that all edges of  $E(V(G) \setminus C, C)$  are oriented towards  $C$ , every strict subset  $S$  of this set is the neighbourhood and the identifier of a vertex in  $V(G) \setminus C$ . Hence the identifier of  $w_i$  is  $P[w_i]$ . In particular, it dominates all its neighbours in clause gadgets.

Let  $\phi$  be the truth assignment defined by  $\phi(x_i) = \text{true}$  if  $w_i \in \{x_i^1, x_i^2\}$  and  $\phi(x_i) = \text{false}$  otherwise. Let us prove that it satisfies  $\phi$ .

Let  $C_j = \ell_j^1 \vee \ell_j^2 \vee \ell_j^3$  be a clause of  $\mathcal{F}$ . In  $D$ ,  $z_j$  has an out-neighbour, say  $\ell_j^k$ . Assume that the literal  $\ell_j^k$  corresponds to the negated variable  $\bar{x}_i$ . Then vertex  $N_G(\ell_j^k) = \{z_j, x_i^1, x_i^2\}$ . Hence its identifier is contained in  $\{\ell_j^k, x_i^1, x_i^2\}$ . But since  $x_i$  is in at least two clauses (because we doubled the clauses),  $\{\ell_j^k\}$ ,  $\{\ell_j^k, x_i^1\}$  and  $\{\ell_j^k, x_i^2\}$  are strict subsets of  $P[x_i^1]$  and  $P[x_i^2]$  respectively. Therefore, those three sets are identifiers of some vertices of  $V(G) \setminus C$ . Hence the identifier of  $\ell_j^k$  is  $\{\ell_j^k, x_i^1, x_i^2\}$ . In particular,  $\ell_j^k$  dominates  $x_i^1$  and  $x_i^2$  and so  $w_i$  is not in  $\{x_i^1, x_i^2\}$ . Hence by definition  $\phi(\ell_j^k) = \phi(\bar{x}_i) = \text{true}$ . So  $C_j$  contains a true literal.

Assume now that there is a truth assignment  $\phi$  satisfying  $\mathcal{F}$ . Let  $D$  be the orientation of  $G$  obtained as follows. Firstly we orient all edges of  $E(V(G) \setminus C, C)$  are oriented towards  $C$ . Secondly for every variable  $x_i$ , we orient the edges  $y_i x_i^1$  and  $y_i x_i^2$  away from  $y_i$  and the edges  $y_i \bar{x}_i^1$  and  $y_i \bar{x}_i^2$  towards  $y_i$  if  $\phi(x_i) = \text{true}$ , and we orient the edges in the other direction if  $\phi(x_i) = \text{false}$ . Finally, for every clause  $C_j = \ell_j^1 \vee \ell_j^2 \vee \ell_j^3$ , we orient  $z_j$  towards its literals which are *true*. (There is at least one of them). Moreover if  $\phi(\ell_j^k) = \text{true}$ , we orient all edges between  $\ell_j^k$  and vertices in variable gadgets away from  $\ell_j^k$ . Otherwise, we orient all these edges towards  $\ell_j^k$ . One can easily check that  $C$  is an identifying code of  $D$ .  $\square$

Despite Theorem 40, there are many cases when ISCODE can be solved in polynomial time. If  $C = V(G)$  the answer is trivially ‘yes’, and if  $2^{|C|} - 1 < V(G)$  then the answer is trivially ‘no’. If  $C$  is a stable set of  $G$ , or if  $G\langle C \rangle$  has a bounded number of edges, then it can be solved in polynomial time using matchings, as we shall now explain.

**Lemma 41.** *Given a graph  $G$ , a set  $C \subseteq V(G)$ , and an orientation  $D_C$  of  $G\langle C \rangle$ , one can check in polynomial time whether there exists an orientation  $D$  of  $G$  such that  $D\langle C \rangle = D_C$  and  $C$  is an identifying code of  $D$ .*

*Proof.* Observe first that for any orientation  $D$  such that  $D\langle C \rangle = D_C$ ,  $I_{C,D}(v) = N^+[v] \cap C$  for all  $v \in C$ .

For every vertex  $v \in V(G)$ , let  $P(v)$  be the set of potential identifiers of  $v$  and  $p(v)$  its cardinality: if  $v \in C$ , then  $P(v) = \{N^+[v] \cap C\}$  and  $p(v) = 1$ ; if  $v \in V(G) \setminus C$ , then

$P(v)$  consists in all non-empty subsets of  $N(v) \cap C$  and  $p(v) = 2^{|N(v) \cap C|} - 1$ . Note that  $p(v)$  can be computed in polynomial time.

We need to assign to each vertex  $v$  a set  $I(v)$  in  $P(v)$  such that the  $I(v)$  are pairwise distinct. Observe that if  $p(v) \geq |V(G)|$  whatever assignment is done for the other vertices, one can find a set  $I(v)$  in  $P(v)$  which is distinct from the  $I(u), u \neq v$ . Hence, we do the assignment of such vertices greedily at the end.

Let  $A$  be the set of vertices such that  $p(v) < |V(G)|$ . By the above observation, it suffices to assign identifiers to vertices in  $A$ . Let us construct an auxiliary bipartite graph  $H$ . The partition of its vertex set is  $(A, B)$  with  $B$  the set of subsets of  $C$  in  $\bigcup_{v \in A} P(v)$ . Note that  $B$  has size less than  $|V(G)|^2$  because  $p(v) < n$  for all  $v \in A$ . The edge set of  $H$  is  $E(H) = \{ab \mid a \in A, b \in P(a)\}$ . Finding an assignment of distinct identifiers for vertices in  $A$  is equivalent to find a matching saturating  $A$  in  $H$ . This can be done in polynomial time, by the celebrated Hungarian Method for example.  $\square$

Lemma 41 implies that IsCODE can be solved in polynomial time if  $C$  has few edges, that is at most  $\log_2(P(|V(G)|))$  for  $P$  a fixed polynomial.

**Problem 42.** What is the complexity of the following problem :

CLIQUECODE

Input: A graph  $G$  and a clique  $C \subseteq V(G)$ .

Question: Is there an orientation  $D$  of  $G$  for which  $C$  is an identifying code ?

## Acknowledgements

The authors would like to thank an anonymous referee for their careful reading and suggestions that helped to improve the presentation of the paper.

## References

- [1] Mostafa Blidia, Mustapha Chellali, Frédéric Maffray, Julien Moncel, and Ahmed Semri. Locating-domination and identifying codes in trees. *Australasian Journal of Combinatorics*, 39:219–232, 2007.
- [2] Marwane Bouznif, Frédéric Havet, and Myriam Preissmann. Minimum-density identifying codes in square grids. In *Algorithmic Aspects in Information and Management - 11th International Conference, AAIM 2016, Bergamo, Italy, July 18-20, 2016, Proceedings*, pages 77–88, 2016.
- [3] Irène Charon, Sylvain Gravier, Olivier Hudry, Antoine Lobstein, Michel Mollard, and Julien Moncel. A linear algorithm for minimum 1-identifying codes in oriented trees. *Discrete Applied Mathematics*, 154(8):1246–1253, 2006.
- [4] Irène Charon, Olivier Hudry, and Antoine Lobstein. Identifying and locating-dominating codes: NP-completeness results for directed graphs. *IEEE Transactions on Information Theory*, 48(8):2192–2200, Aug 2002.

- [5] Irène Charon, Olivier Hudry, and Antoine Lobstein. Minimizing the size of an identifying or locating-dominating code in a graph is NP-hard. *Theoretical Computer Science*, 290(3):2109 – 2120, 2003.
- [6] Daniel W. Cranston and Gexin Yu. A new lower bound on the density of vertex identifying codes for the infinite hexagonal grid. *Electron. J. Combin.*, 16(1):#R113, 2009.
- [7] Ari Cukierman and Gexin Yu. New bounds on the minimum density of an identifying code for the infinite hexagonal grid. *Discrete Appl. Math.*, 161(18):2910–2924, 2013.
- [8] Rennan Dantas, Frédéric Havet, and Rudini M. Sampaio. Identifying codes for infinite triangular grids with a finite number of rows. *Discrete Math.*, 340(7):1584–1597, 2017.
- [9] Rennan Dantas, Rudini M. Sampaio, and Frédéric Havet. Minimum density of identifying codes of king grids. *Electronic Notes in Discrete Mathematics*, 62:51–56, 2017.
- [10] Florent Foucaud. Personnel communication.
- [11] Florent Foucaud. Decision and approximation complexity for identifying codes and locating-dominating sets in restricted graph classes. *J. Discrete Algorithms*, 31:48–68, 2015.
- [12] Florent Foucaud, Sylvain Gravier, Reza Naserasr, Aline Parreau, and Petru Valicov. Identifying codes in line graphs. *Journal of Graph Theory*, 73(4):425–448, 2013.
- [13] Florent Foucaud, Eleonora Guerrini, Matjaz Kovse, Reza Naserasr, Aline Parreau, and Petru Valicov. Extremal graphs for the identifying code problem. *Eur. J. Comb.*, 32(4):628–638, 2011.
- [14] Michael R. Garey and David S. Johnson. *Computers and intractability*. W. H. Freeman and Co., San Francisco, Calif., 1979. A guide to the theory of NP-completeness, A Series of Books in the Mathematical Sciences.
- [15] Michael R. Garey, David S. Johnson, and Larry J. Stockmeyer. Some simplified np-complete graph problems. *Theoretical Computer Science*, 1(3):237 – 267, 1976.
- [16] Sylvain Gravier and Julien Moncel. On graphs having a set as an identifying code. *Discrete Mathematics*, 307(3-5):432 – 434, 2007.
- [17] Teresa Haynes, Debra Knisley, Edith Seier, and Yue Zou. A quantitative analysis of secondary rna structure using domination based parameters on trees. *BMC Bioinformatics*, 7(1):108, 2006.
- [18] Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972.
- [19] Mark G. Karpovsky, Krishnendu Chakrabarty, and Lev B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE Trans. Information Theory*, 44(2):599–611, 1998.
- [20] Moshe Laifenfeld, Ari Trachtenberg, Reuven Cohen, and David Starobinski. Joint monitoring and routing in wireless sensor networks using robust identifying codes. In *Fourth International Conference on Broadband Communications, Networks and Sys-*

- tems, (*BROADNETS 2007*), 10-14 September 2007, Raleigh, North-Carolina, USA, pages 197–206. IEEE, 2007.
- [21] Moshe Laifenfeld, Ari Trachtenberg, Reuven Cohen, and David Starobinski. Joint monitoring and routing in wireless sensor networks using robust identifying codes. *MONET*, 14(4):415–432, 2009.
  - [22] Saikat Ray, Rachanee Ungrangsi, Francesco De Pellegrini, Ari Trachtenberg, and David Starobinski. Robust location detection in emergency sensor networks. In *Proceedings IEEE INFOCOM 2003, The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, March 30 - April 3, 2003*. IEEE, 2003.
  - [23] Craig A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85 – 89, 1984.