Theoretical Computer Science 626 (2016) 1–12

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

More results on the complexity of identifying problems in graphs

Olivier Hudry [a,*], Antoine Lobstein [b]

[a] Institut Télécom, Télécom ParisTech & CNRS, LTCI UMR 5141, 46, rue Barrault, 75634 Paris Cedex 13, France
[b] CNRS, LTCI UMR 5141 & Institut Télécom, Télécom ParisTech, 46, rue Barrault, 75634 Paris Cedex 13, France

ARTICLE INFO

Article history:
Received 3 June 2015
Received in revised form 16 November 2015
Accepted 16 January 2016
Available online 27 January 2016
Communicated by C. Kaklamanis

Keywords:
Graph theory
Complexity
Complexity classes
Polynomial hierarchy
NP-completeness
Hardness
Identifying codes
Twin-free graphs

ABSTRACT

We investigate the complexity of several problems linked with identification in graphs; for instance, given an integer $r \geq 1$ and a graph $G = (V, E)$, the existence of, or search for, optimal r-identifying codes in G, or optimal r-identifying codes in G containing a subset of vertices $X \subset V$. We locate these problems in the complexity classes of the polynomial hierarchy.

1. Introduction and preliminary results

Following [17], which investigates the complexity of Slater's problems in tournaments, our goal in this paper is to study the algorithmic complexity of different variants of the identifying problem in graphs.

In [18], we do the same work for domination problems.

1.1. Outline of the paper

In Subsection 1.2, we present the necessary notation and definitions about identifying codes; Subsection 1.3 gives preliminary results on identifying codes. In Section 2, we present seven problems, decision, optimization or search problems, related to identification, we give some known results, before we motivate our research and give our own development. We shall provide the necessary notions of complexity as we go along. The conclusion recapitulates our results.

1.2. Definitions and notation

We first give the necessary definitions and notation for identification in graphs; see the seminal paper [20], and also [21] for a large bibliography.

* Corresponding author.
E-mail addresses: hudry@telecom-paristech.fr (O. Hudry), lobstein@telecom-paristech.fr (A. Lobstein).

http://dx.doi.org/10.1016/j.tcs.2016.01.021
0304-3975/© 2016 Published by Elsevier B.V.

We shall denote by $G = (V, E)$ a finite, simple, undirected graph with vertex set V and edge set E, where an *edge* between $x \in V$ and $y \in V$ is indifferently denoted by xy or yx. The *order* of the graph is its number of vertices, $|V|$. A *path* $P_k = x_1x_2 \dots x_k$ is a sequence of k distinct vertices x_i, $1 \le i \le k$, such that x_ix_{i+1} is an edge for $i \in \{1, 2, \dots, k-1\}$. The *length* of P_k is its number of edges, $k-1$.

A graph G is called *connected* if for any two vertices x and y, there is a path between them; it is called *disconnected* otherwise. In a connected graph G, we can define the *distance* between any two vertices x and y, denoted by $d_G(x, y)$, as the length of any shortest path between x and y, since at least one such path exists. This definition can be extended to disconnected graphs, using the convention that $d_G(x, y) = +\infty$ if no path exists between x and y. The subscript G can be dropped when there is no ambiguity.

For an integer $k \ge 2$, the *k-th transitive closure*, or *k-th power* of $G = (V, E)$ is the graph $G^k = (V, E^k)$ defined by $E^k = \{uv : u \in V, v \in V, d_G(u, v) \le k\}$.

For any vertex $v \in V$, the *open neighbourhood* $N(v)$ *of* v consists of the set of vertices adjacent to v, i.e., $N(v) = \{u \in V : uv \in E\}$; the *closed neighbourhood of* v is $B_1(v) = N(v) \cup \{v\}$. This notation can be generalized to any integer $r \ge 0$ by setting

$$B_r(v) = \{x \in V : d(x, v) \le r\}.$$

For $X \subseteq V$, we denote by $B_r(X)$ the set of vertices within distance r from X:

$$B_r(X) = \cup_{x \in X} B_r(x).$$

Two vertices x and y such that $B_r(x) = B_r(y)$, $x \ne y$, are called *r-twins*. If G has no r-twins, we say that G is *r-twin-free*. Whenever two vertices x and y are such that $x \in B_r(y)$ (which is equivalent to $y \in B_r(x)$), we say that x and y *r-cover* or *r-dominate* each other; note that every vertex r-covers itself. A set W is said to r-cover a set Z if every vertex in Z is r-covered by at least one vertex of W. When three vertices x, y, z are such that $z \in B_r(x)$ and $z \notin B_r(y)$, we say that z *r-separates* x and y in G (note that $z = x$ is possible). A set of vertices is said to r-separate x and y if it contains at least one vertex which does.

A *code* C is simply a subset of V, and its elements are called *codewords*. For each vertex $v \in V$, we denote the set of codewords r-covering v by $I_{G,C,r}(v)$, or, when there is no ambiguity on G, by $I_{C,r}(v)$:

$$I_{G,C,r}(v) = I_{C,r}(v) = B_r(v) \cap C.$$

We say that C is an *r-dominating* code in G if all the sets $I_{C,r}(v)$, $v \in V$, are nonempty; in other words, every vertex is r-dominated by C. We say that C is an *r-identifying* code if all the sets $I_{C,r}(v)$, $v \in V$, are nonempty and distinct: in other words, every vertex is r-covered by C, and every pair of vertices is r-separated by C. It is quite easy to observe that a graph G admits an r-identifying code if and only if G is r-twin-free; this is why r-twin-free graphs are also called *r-identifiable*. When G is r-twin-free, we denote by $i_r(G)$ the smallest cardinality of an r-identifying code in G, and call it the *r-identification number* of G. Any r-identifying code C such that $|C| = i_r(G)$ is said to be *optimal*.

1.3. Some useful facts on identification

In the sequel, we shall need the following results on identification.

Lemma 1. *Let $r \ge 1$ be an integer and G be a graph. If C is an r-identifying code in G, then any code $S \supseteq C$ also is.*

Proof. When we add the elements of $S \setminus C$ to the adequate sets $I_{C,r}(v)$, these new sets $I_{S,r}(v)$ are still nonempty and distinct, and distinct from the sets with no addition (those such that $I_{S,r}(v) = I_{C,r}(v)$). □

Lemma 2. *Let $r \ge 2$ be an integer and $G = (V, E)$ be a graph. A code C is 1-identifying in G^r, the r-th power of G, if and only if it is r-identifying in G.*

Proof. For every vertex $v \in V$, we have:

$$I_{G,C,r}(v) = \{c \in C : d_G(v, c) \le r\} = \{c \in C : d_{G^r}(v, c) \le 1\} = I_{G^r,C,1}(v). \qquad \square$$

Lemma 3. *Let $G = (V, E)$ be a 1-twin-free graph. For a given set of vertices $A = \{\alpha_1, \dots, \alpha_k\} \subseteq V$, we construct the following graph $G_A = (V_A, E_A)$, which depends on A (see Fig. 1):*

$$V_A = V \cup V_A^*, \text{ with } V_A^* = \cup_{1 \le j \le k} V_j^* \text{ and } V_j^* = \{\beta_{j,1}, \beta_{j,2}, \delta_j, \lambda_j\},$$

$$E_A = E \cup \{\alpha_j\beta_{j,1}, \beta_{j,1}\beta_{j,2}, \beta_{j,1}\delta_j, \beta_{j,1}\lambda_j, \beta_{j,2}\delta_j, \beta_{j,2}\lambda_j : 1 \le j \le k\},$$

where for $j \in \{1, \dots, k\}$, none of the vertices $\beta_{j,1}, \beta_{j,2}, \delta_j, \lambda_j$ belongs to V.

Then $A \subseteq V$ is included in at least one optimal 1-identifying code in G if and only if $i_1(G) = i_1(G_A) - 2|A|$.