Journal of Discrete Algorithms 31 (2015) 48–68

Contents lists available at ScienceDirect

Journal of Discrete Algorithms

www.elsevier.com/locate/jda

Decision and approximation complexity for identifying codes and locating-dominating sets in restricted graph classes [☆]

Florent Foucaud [a,b,c,*]

[a] Department of Mathematics, University of Johannesburg, Johannesburg, South Africa
[b] LAMSADE – CNRS UMR 7243, PSL, Université Paris-Dauphine, Paris, France
[c] Departament de Matemàtica Aplicada III, Universitat Politècnica de Catalunya, Barcelona, Spain

ARTICLE INFO

Article history:
Available online 6 September 2014

Keywords:
Test cover
Separating system
Identifying code
Locating-dominating set
NP-completeness
Approximation

ABSTRACT

An identifying code is a subset of vertices of a graph with the property that each vertex is uniquely determined (identified) by its nonempty neighbourhood within the identifying code. When only vertices out of the code are asked to be identified, we get the related concept of a locating-dominating set. These notions are closely related to a number of similar and well-studied concepts such as the one of a test cover. In this paper, we study the decision problems Identifying Code and Locating-Dominating Set (which consist in deciding whether a given graph admits an identifying code or a locating-dominating set, respectively, with a given size) and their minimization variants Minimum Identifying Code and Minimum Locating-Dominating Set. These problems are known to be NP-hard, even when the input graph belongs to a number of specific graph classes such as planar bipartite graphs. Moreover, it is known that they are approximable within a logarithmic factor, but 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: both problems remain hard in these cases. Among other results, we also show that for bipartite graphs of bounded maximum degree (at least 3), the two problems are hard to approximate within some constant factor, a question which was open. We summarize all known results in the area, and we compare them to the ones for the related problem Dominating Set. In particular, our work exhibits important graph classes for which Dominating Set is efficiently solvable, but Identifying Code and Locating-Dominating Set are hard (whereas in all previous works, their complexity was the same). We also introduce graph classes for which the converse holds, and for which the complexities of Identifying Code and Locating-Dominating Set differ.

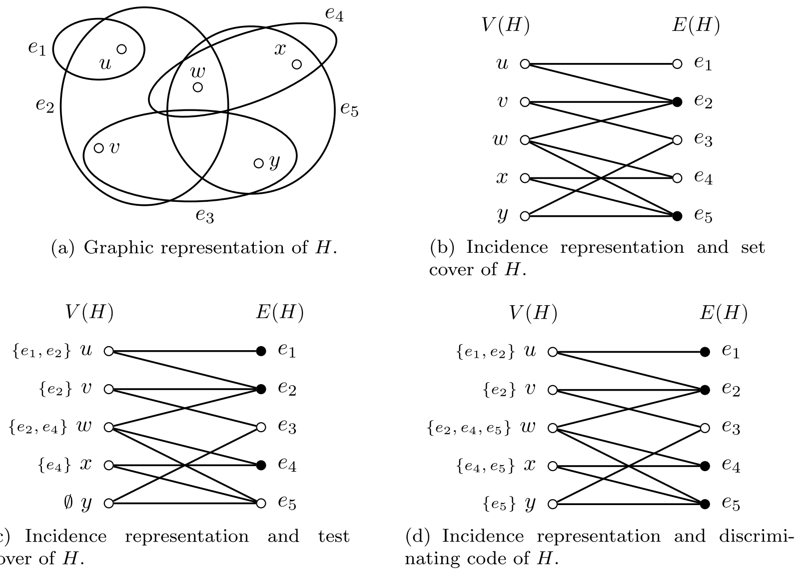
1. Introduction

This paper studies the computational complexity of problems where one wants to find a set of vertices in a graph that uniquely identifies each vertex. In particular, we study this complexity according to the graph class of the input. We mainly

☆ Most results of this paper are from the author's PhD thesis [28], and were found while he was a PhD student at the LaBRI, University Bordeaux 1, 351 Cours de la Libération, 33405 Talence Cedex, France. An extended abstract of this paper containing most results about identifying codes appeared in the proceedings of IWOCA 2013 [29].

* Correspondence to: Department of Mathematics, University of Johannesburg, Auckland Park 2006, South Africa.

http://dx.doi.org/10.1016/j.jda.2014.08.004
1570-8667/



(a) Graphic representation of H.

(b) Incidence representation and set cover of H.

(c) Incidence representation and test cover of H.

(d) Incidence representation and discriminating code of H.

Fig. 1. A hypergraph H with a set cover, a test cover and a discriminating code (black edges). In the two last figures, the sets next to the vertices represent the sets of edges of the solution containing that vertex. The test cover is not a set cover and hence it is not a discriminating code.

focus on identifying codes, which are subsets of vertices that identify all vertices using the intersection of their closed neighbourhood with the code. The slightly less restrictive concept of locating-dominating sets will also be studied.

1.1. Definitions and problems

The graph-theoretic problems we will consider are special cases of more general ones, defined on hypergraphs, that we shall describe first.

In this paper, 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* 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. 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. 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 exactly the same set of edges). See Fig. 1 for examples of these concepts.

While set covers are a standard and widespread notion in combinatorics and theoretical computer science, test covers are less known. They were introduced under the name of *separating systems* by Rényi [53] (see also Bollobás and Scott [8] for more recent work in the same line of research). They were independently studied in Garey and Johnson's book [33] under the name of *test collections* and further studied, see [24,47]. Discriminating codes were more recently (and independently) introduced and studied in [12,13]. As we will see later, the notions of test covers and discriminating codes are very similar in nature. Due to their properties enabling the unique identification of elements (vertices) of a system (hypergraph) using the set of their attributes (edges), they have had a number of interesting applications in the areas of testing individuals (such as patients or computers) for diseases or faults, see [12,24,47].

In this paper, we are mainly interested in special cases that can be defined over *graphs* rather than hypergraphs. The notion that we will mainly study, introduced in 1998 in [40], is the one of identifying codes. Given a graph G and a vertex v of G, we denote by $N(v)$ and by $N[v]$ the open and the closed neighbourhood of v, respectively. 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 $\gamma^{\mathrm{ID}}(G)$. The notion of an identifying code is a generalization of the well-studied *locating-dominating sets*, introduced three decades ago [54,55]. Given a graph G, a *locating-dominating set* of G is a subset $\mathcal{C} \subseteq V(G)$ which is both a dominating set and which separates all vertices that are not in the code, i.e. for each pair $u, v \in V(G) \setminus \mathcal{C}$, if $u \neq v$, then $N(u) \cap \mathcal{C} \neq N(v) \cap \mathcal{C}$. The minimum size of a locating-dominating set of a given graph G will be denoted $\gamma^{\mathrm{LD}}(G)$. It is easily seen that any identifying code is a locating-dominating set. See Fig. 2 for examples of these concepts.

Whereas it is clear that any graph admits a (locating-)dominating set (its whole vertex set), 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 a separating code (and therefore an identifying code); we call such graphs *twin-free*.