ARTICLE IN PRESS

Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Domination problems with no conflicts

Alexis Cornet, Christian Laforest *

LIMOS, 1 rue de la Chebarde TSA 60125 CS 60026 63178 AUBIERE CEDEX, France

ARTICLE INFO

Article history:
Received 15 February 2017
Received in revised form 2 February 2018
Accepted 7 March 2018
Available online xxxx

Keywords:
Graph theory
Independent dominating set
Dominating set
Conflicts
Forbidden pairs
NP-completeness

ABSTRACT

Domination problems have been studied in graph theory for decades. In most of them, it is NP-complete to find an *optimal* solution, while it is easy (and even trivial in some cases) to find a solution in polynomial time, regardless of its size.

In recent works, authors added *conflicts* to classical discrete optimization problems. In this paper, a conflict is a pair of vertices that cannot be both in a solution. Set of conflicts can be viewed as edges of a so called *conflict graph*. An instance is then a *support graph* and a conflict graph. With these new constraints, the existence of a solution (*dominating set* or *independent dominating set*) with no conflicts is no more guaranteed. We explore this subject and we prove that it is NP-complete to decide the existence of a solution even in very restricted classes of graphs and conflicts (sparse or dense). We also propose polynomial algorithms for some sub-cases, using deterministic finite automata.

1. Introduction

Domination problems are central in the field of graph theory. Many variants have been introduced in the literature, see for example [5]. For Dominating Set and Independent Dominating Set, finding an optimal solution is NP-complete, and it is known that they are not constant-approximable unless P = NP [1]. However, it is always easy to construct a solution in polynomial time, regardless of its size.

In the real world, there can be structural incompatibilities between elements of a solution, for example non common interface, security reasons, and so on. To model this, we say that two vertices u and v are *in conflict* if u and v cannot be both in a solution. $G = (V, E)$ is called the *Support Graph*, and conflicts are interpreted as a graph on the same vertices, called the *Conflict Graph* $C = (V, F)$. The existence of a solution without conflict is no more guaranteed.

Problems with conflicts is an active area of research. In [6–8,11,12], authors study the complexity of finding structures without conflicts between edges, like path, Hamiltonian paths or cycle, and spanning tree. In [4,9,10,15], authors try to find paths without conflicts between vertices in various graph classes. Most of the results are NP-completeness proofs of deciding the existence of a solution.

In [3], authors prove that deciding the existence of solution with conflicts is NP-complete for Dominating Set, Independent Dominating Set, Connected Vertex Cover and Steiner Tree. However, the reductions use graphs and conflict graphs of non bounded degree and the conflict graphs do not have interesting structural properties. In [2], authors refine results for Connected Vertex Cover and Steiner Tree with conflicts and extend them to Total Domination with conflict.

In this paper, we precise the complexity of deciding the existence of a solution without conflicts, thus we will prove NP-completeness results in very restricted graph classes and point out some polynomial cases. Let us define the problems formally.

* Corresponding author.
E-mail addresses: alexis.cornet@uca.fr (A. Cornet), christian.laforest@isima.fr (C. Laforest).

https://doi.org/10.1016/j.dam.2018.03.018

Given (G, C) where $G = (V, E)$ and $C = (V, F)$, an independent dominating set with no conflict (IDSwnC) is a subset of vertices $S \subseteq V$ such that:

- for each $x \in V$, $x \in S$ or $\exists y \in S$ with $xy \in E$ (S is a dominating set of G)
- for each $xy \in E$, $x \notin S$ or $y \notin S$ (S is an independent set of G)
- for each $xy \in F$, $x \notin S$ or $y \notin S$ (S induces no conflicts in C).

Given (G, C) where $G = (V, E)$ and $C = (V, F)$, a dominating set with no conflicts (DSwnC) is a subset of vertices $S \subseteq V$ such that:

- for each $x \in V$, $x \in S$ or $\exists y \in S$ with $xy \in E$ (S is a dominating set of G)
- for each $xy \in F$, $x \notin S$ or $y \notin S$ (S induces no conflicts in C).

In Section 2, we will prove the NP-completeness of deciding the existence of IDSwnC and DSwnC in classes of sparse graphs and point some polynomial subclasses. In Section 3, we prove the NP-completeness of both problems in dense graphs, and Section 4 presents a polynomial algorithm for a subcase of IDSwnC when *path-cutwidth* is bounded (this parameter is defined in Section 4). For ease of reading, the problem of deciding the existence of an IDSwnC (resp. DSwnC) will also be called IDSwnC (resp. DSwnC). An instance of IDSwnC or DSwnC is of the form (G, C) where G is the support graph and C is the conflict graph, and will be called a *graph with conflicts*. Some of the following proofs will use reduction from a special case of 3 SAT, which we define below.

Instance: (X, Cl) where X is a set of boolean variables and Cl a set of disjunctive 3-clauses over X.

Question: Is there an assignment on X satisfying Cl?

The 3 SAT problem is NP-complete, even if each variable is in at most 4 clauses [14] (in positive or negative form). This restriction is important for our reductions.

We also need a few additional notations. P_n denote the path on n vertices. If G_1 and G_2 are two paths, $G_1 + G_2$ denotes the concatenation of the paths, i.e. one arbitrary extremity of a path is linked to one extremity of the other. The complete bipartite graph with partitions of size x and y is denoted by $K_{x,y}$. Problem $\mathcal{P}$ denotes the generic problem of deciding the existence of a structure without conflicts (an IDSwnC or DSwnC in our case).

2. Sparse graphs

The purpose of this section is to prove NP-completeness of deciding the existence of IDSwnC or DSwnC in classes of very sparse graphs. The structure behind the reductions for these two problems is quite similar, although the results are not exactly the same.

Let us sketch the proof. We will reduce a (NP-complete) subcase of 3-SAT to our problems. For that, we define *clause gadgets* which simulate clauses of 3-SAT. Then, we add conflicts to ensure coherence (a variable and its negation not being both positive). This is the first part of our reduction. Then, in order to strengthen our results we use another kind of gadget to decompose the conflict graph into an even more sparse graph (namely, a graph of maximum degree 1). Finally, a last gadget allows us to connect the support graph and achieve our strongest result. To do this, we first define abstract gadgets and properties they must have and then we instanciate them for each problem. More formally:

Definition 1 (*Clause Gadget*)**.** A clause gadget for problem $\mathcal{P}$ for clause $c_l = (x_i \vee x_j \vee x_k)$ is a graph with conflicts (G_{c_l}, C_{c_l}) where G_{c_l} is a path and the conflict graph C_{c_l} is a disjoint union of P_1, P_2, P_3 and x_i, x_j, x_k are 3 distinguished vertices of G_{c_l} such that:

1. Any solution S of $\mathcal{P}$ must contain at least one vertex among x_i, x_j, x_k.
2. For any non empty subset X of x_i, x_j, x_k there must exist a solution S without conflict of $\mathcal{P}$ such that $S \cap \{x_i, x_j, x_k\} = X$.
3. No edge of C_{c_l} is incident to x_i, x_j, x_k. (x_i, x_j, x_k are not in conflict in C_{c_l})

We say that there exists a clause gadget for $\mathcal{P}$ if for any 3-clause one can construct in polynomial time a clause gadget for $\mathcal{P}$.

Lemma 1. *If $\mathcal{P}$ is a problem with conflicts such that there exists a clause gadget for $\mathcal{P}$, then $\mathcal{P}$ is NP-complete even when the support graph is a disjoint union of paths and the conflict graph is a union of complete bipartite graphs of at most 4 vertices.*

Proof. It is known from [14] that 3-SAT is NP-complete even in instances in which each variable appears in at most 4 clauses. Let (X, Cl) be a such 3-SAT instance. Construct an instance (G, C) of problem $\mathcal{P}$ as follows. For each clause $c_l = (x_i \vee x_j \vee x_k)$ of Cl, construct a clause gadget (G_{c_l}, C_{c_l}) and denote x_i, x_j, x_k as its three distinguished vertices. Set $G = \bigcup_{c_l}(G_{c_l})$ and $C' = \bigcup_{c_l}(C_{c_l})$. Create C by adding to C' for each $x_i, \bar{x}_i$ (a literal of X and its negation), edges to create a complete bipartite graph between vertices representing x_i and vertices representing $\bar{x}_i$. By definition, G is a disjoint union of paths and C is a disjoint union of P_1, P_2, P_3 (from clauses gadget) which are complete bipartites of less than 4 vertices, and $K_{1,1}, K_{1,2}, K_{1,3}, K_{2,2}$ (from conflicts between literals and their negation).

Suppose that there exists a solution A of the 3-SAT instance. A is an assignment. Construct a solution S of $\mathcal{P}$ in (G, C) as follows. For each literal x_α set to *true*, all vertices x_α are included in S. Since A is an assignment, a literal and its negation cannot be both *true*, and there are no conflicts between the selected vertices. Since each clause is satisfied, there is at least one distinguished vertex selected in each clause gadget. By properties 2 and 3 of the gadget definition, each gadget can have a solution without conflict for $\mathcal{P}$ and there exists a solution for $\mathcal{P}$ in (G, C).

Please cite this article in press as: A. Cornet, C. Laforest, Domination problems with no conflicts, Discrete Applied Mathematics (2018), https://doi.org/10.1016/j.dam.2018.03.018.