Some Rainbow Problems in Graphs Have Complexity Equivalent to Satisfiability Problems

Olivier Hudry, Antoine Lobstein

Some Rainbow Problems in Graphs have Complexity Equivalent to Satisfiability Problems

Olivier Hudry
LTCI, Télécom Paris, Institut polytechnique de Paris
19, place Marguerite Perey, 91120 Palaiseau, France
& Antoine Lobstein
Centre National de la Recherche Scientifique
Laboratoire de Recherche en Informatique, UMR 8623,
Université Paris-Sud
Bâtiment 650 Ada Lovelace, 91405 Orsay Cedex - France

olivier.hudry@telecom-paris.fr, antoine.lobstein@lri.fr

June 29, 2020

Abstract

In a vertex-coloured graph, a set of vertices S is said to be a rainbow set if every colour in the graph appears exactly once in S. We investigate the complexities of various problems dealing with domination in vertex-coloured graphs (existence of rainbow dominating sets, of rainbow locating-dominating sets, of rainbow identifying sets), including when we ask for a unique solution: we show equivalence between these complexities and those of the well-studied Boolean satisfiability problems.

Key Words: Graph theory, Complexity theory, Uniqueness of solution, Rainbow sets, Dominating codes, Locating-dominating codes, Identifying codes, Twin-free graphs

1 Introduction

We intend to locate in the classes of complexity several problems linked to the existence of rainbow dominating sets, of rainbow locating-dominating sets, and of rainbow identifying sets in a vertex-coloured graph, including when we ask for a unique solution. To this effect, we shall prove equivalence (up to polynomials) between the complexities of our problems and those of satisfiability problems.

This work is motivated by [1], [2], in particular their Theorem 2.1 (see Proposition 10 below) on the complexity of finding rainbow dominating sets in a vertex-coloured graph. Since locating-dominating sets and identifying sets are particular classes of dominating sets, and since locating-domination and identification are popular nowadays (see the ongoing bibliography at [24]), it seems natural to try to extend this theorem to these two classes, including when considering domination at distance r and uniqueness of solution.

1.1 Vertex-Coloured Graphs

Let $G = (V, E)$ be a finite, simple, undirected graph with vertex set V and edge set E, where an edge between $v_1 \in V$ and $v_2 \in V$ is indifferently denoted by v_1v_2 or v_2v_1. The *order* of G is its number of vertices.

If $\Phi = \{1, \dots, c\}$ is a set of colours, then $G_{col} = (V_{col}, E)$ denotes a vertex-coloured graph (or simply coloured graph) obtained from G by giving one colour taken in Φ to every vertex $v \in V$, with each colour given to at least one vertex; here it is not necessary that two neighbour vertices receive different colours. When useful, we denote by $\phi(v)$ the colour given to v. A subset of vertices $V^* \subseteq V$ is said to be *tropical* if every colour appears at least once in V^*. It is said to be *rainbow* if every colour appears exactly once in V^*; therefore, any rainbow set has c elements. When it is clear that we are dealing with a coloured graph, we shall often drop the subscript *col*.

Remark 1 *Here, we stick to the definition of a rainbow subset which is used in [1] and [2], because our article is intended to prolong and widen the complexity result therein. However, a different terminology can be found in the literature: a rainbow subset can also designate a vertex subset where each colour appears* at most *once, whereas in a* colourful *subset, they appear* exactly *once; see, e.g., [3] or [9].*

Since we shall study domination at distance r and, to this purpose, use the term rainbow r-domination, *we have to mention that* k-rainbow domination *is used with a different meaning in, e.g., [7].*

1.2 More Definitions and Notation in Graphs

All the following definitions apply to graphs and coloured graphs.

For any integer $r \geq 2$, the *r-th power* of G is the graph $G^r = (V, E^r)$, with $E^r = \{v_1v_2 : v_1, v_2 \in V, 0 < d_G(v_1, v_2) \leq r\}$.

For any integer $r \geq 1$, and for every vertex $v \in V$, we denote by $B_{G,r}(v)$ (and $B_r(v)$ when there is no ambiguity) the *ball of radius r centered at v*, i.e., the set of vertices at distance at most r from v:

$$B_r(v) = \{w \in V : 0 \leq d_G(v, w) \leq r\}.$$

Whenever $v \in B_r(w)$ (which is equivalent to $w \in B_r(v)$), we say that v and w *r-dominate* each other. When three vertices v, w, z are such that $z \in B_r(v)$ and $z \notin B_r(w)$, we say that z *r-separates* v and w in G (note that $z = v$ is possible). A set of vertices is said to r-separate v and w if at least one of its elements does.

A subset of vertices V^* will be indifferently called a set or a *code*, and its elements *codewords*. We denote by $I_{G,V^*,r}(v)$ (and $I_r(v)$ when there is no ambiguity) the set of codewords that r-dominate v: $I_{G,V^*,r}(v) = B_{G,r}(v) \cap V^*$.

A code V^* is said to be an *r-dominating set* or an *r-dominating code* (r-D code for short) if for all $v \in V$, we have $I_r(v) \neq \emptyset$. One can also find the terminology *dominating set at distance r*, or *distance r dominating set.*

A code V^* is said to be *r-locating-dominating* (r-LD for short) if for all $v \in V$, we have $I_r(v) \neq \emptyset$, and for any two distinct non-codewords $v_1, v_2 \in V \setminus V^*$, we have $I_r(v_1) \neq I_r(v_2)$.

A code V^* is said to be *r-identifying* (r-ID for short) if for all $v \in V$, we have $I_r(v) \neq \emptyset$, and for any two distinct vertices $v_1, v_2 \in V$, we have $I_r(v_1) \neq I_r(v_2)$.

In other words: every vertex must be r-dominated by at least one codeword for the three definitions; in addition, every pair of distinct non-codewords (respectively, vertices) must be r-separated by an r-LD (respectively, r-ID) code.

Two vertices $v_1, v_2 \in V$, $v_1 \neq v_2$, are said to be *r-twins* if $B_r(v_1) = B_r(v_2)$. Dominating and locating-dominating codes exist for all graphs; on the other hand, it is easy to see that a graph G admits an r-identifying code if and only if

$$\forall v_1 \in V,\ \forall v_2 \in V,\ v_1 \neq v_2 :\ B_r(v_1) \neq B_r(v_2). \tag{1}$$

A graph satisfying (1) is called *r-identifiable* or *r-twin-free.* The following useful remarks are quite trivial and need no proofs.

Remark 2 *Let $r \geq 2$ be any integer and $G = (V, E)$ be a graph.*
(a) A code V^* is 1-dominating in G^r, the r-th power of G, if and only if it is r-dominating in G.
(b) A code V^* is 1-locating-dominating in G^r if and only if it is r-locating-dominating in G.
(c) A code V^* is 1-identifying in G^r if and only if it is r-identifying in G.

Remark 3 *A code V^* is r-ID (respectively, r-LD) if and only if (a) for every vertex $v \in V$, $I_r(v) \neq \emptyset$, and (b) for every pair of distinct vertices $v_1, v_2 \in V$ (respectively, $v_1, v_2 \in V \setminus V^*$), we have*

$$[B_r(v_1)\Delta B_r(v_2)] \cap V^* \neq \emptyset, \tag{2}$$

where Δ stands for the symmetric difference.

For the vast topic of 1-domination in graphs, see [18]. For locating-dominating and identifying codes, see the large bibliography at [24].

1.3 Satisfiability Problems

We consider a set $\mathcal{X}$ of n *Boolean variables* x_i and a set $\mathcal{C}$ of m *clauses*; each clause contains *literals*, a literal being a variable x_i or its complement (or negated variable) $\overline{x}_i$. A *truth assignment* for $\mathcal{X}$ sets the variable x_i to TRUE, also denoted by T, and its complement to FALSE (or F), or *vice-versa.* A truth assignment is said to *satisfy* a clause if this clause contains at least one true literal, and to satisfy the set of clauses $\mathcal{C}$ if every clause contains at least one true literal. The following decision problems, for which the size of the instance is polynomially linked to $n + m$, are classical problems in complexity.

Problem SAT (Satisfiability):
Instance: A set $\mathcal{X}$ of variables, a collection $\mathcal{C}$ of clauses over $\mathcal{X}$, each clause containing at least two different literals.
Question: Is there a truth assignment for $\mathcal{X}$ that satisfies $\mathcal{C}$?

Problem 3-SAT (3-Satisfiability):
Instance: A set $\mathcal{X}$ of variables, a collection $\mathcal{C}$ of clauses over $\mathcal{X}$, each clause containing exactly three different literals.
Question: Is there a truth assignment for $\mathcal{X}$ that satisfies $\mathcal{C}$?

1.4 A Short Background on Complexity

See, e.g., [4], [16], [23] or [26] for more on this topic; we assume that the reader is already familiar with the classes *P*, *NP* and co-*NP*, with *NP*-complete problems, and with the notion of polynomial transformation between problems.

For problems which are not necessarily decision problems, a *Turing reduction* from a problem π_1 to a problem π_2 is an algorithm $\mathcal{A}$ that solves π_1 using a (hypothetical) subprogram $\mathcal{S}$ solving π_2 such that, if $\mathcal{S}$ were a polynomial algorithm for π_2, then $\mathcal{A}$ would be a polynomial algorithm for π_1. Thus, in this sense, π_2 is "at least as hard" as π_1. A problem π is *NP-hard* (respectively, *co-NP-hard*) if there is a Turing reduction from some *NP*-complete (respectively, co-*NP*-complete) problem to π [16, p. 113].

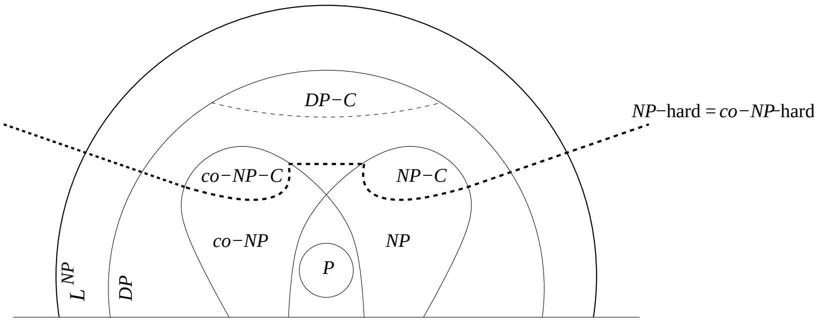

Figure 1: Some classes of complexity.

Remark 4 *Note that these two definitions, NP-hard and co-NP-hard, coincide [16, p. 114].*

The notions of completeness and hardness can be extended to classes other than *NP* or co-*NP*.

We shall also use the class L^{NP} [23] (also denoted by $P^{NP[O(\log n)]}$ or Θ_2), which contains the decision problems which can be solved by applying, with a number of calls which is logarithmic with respect to the size of the instance, a subprogram able to solve an appropriate problem in *NP* (usually, an *NP*-complete problem); and the class *DP* [27] (or DIF^P [5] or BH_2 [23], [28]) as the class of languages (or problems) $\mathcal{L}$ such that there are two languages $\mathcal{L}_1 \in NP$ and $\mathcal{L}_2 \in$ co-NP satisfying $\mathcal{L} = \mathcal{L}_1 \cap \mathcal{L}_2$. This class is not to be confused with $NP \cap$ co-NP (see the warning in, e.g., [26, p. 412]); actually, *DP* contains $NP \cup$ co-NP and is contained in L^{NP}. See Figure 1.

Membership to *P*, *NP*, co-*NP*, *DP* or L^{NP} gives an upper bound on the complexity of a problem (this problem is not more difficult than ...), whereas a hardness result gives a lower bound (this problem is at least as difficult as ...). Still, such results are conditional in some sense; if for example $P = NP$, they would lose their interest. But we do not know whether or where the classes of complexity collapse.

The decision problems SAT and 3-SAT are two of the basic and most well-known *NP*-complete problems [14], [16, p. 39, p. 46 and p. 259]. If we consider their variants U-SAT and U-3-SAT, where the question now is "Is there a *unique* assignment ...", the following result was proved in [22].

Proposition 5 *[22, Th. 10] The decision problems U-SAT and U-3-SAT have equivalent complexities, up to polynomials.*

Using results from [5] and [26, p. 415], it is then rather simple to obtain the following result.

Corollary 6 *(a) The decision problems U-SAT and U-3-SAT are NP-hard.*

(b) The decision problems U-SAT and U-3-SAT belong to the class DP.

Remark 7 *It is not known whether these problems are DP-complete. In [26, p. 415], it is said that "U-SAT is not believed to be DP-complete". It is shown in [5] that there exists one oracle under which U-SAT is not DP-complete; and one oracle under which it is, if NP$\neq$co-NP.*

Note that uniqueness of solutions, which may be seen as part of the wider and rather unexplored issue of the number of solutions of a problem, had been studied earlier in a few papers (see, e.g., [5], [6], [8], [15], [17], [25]).

Let us now turn to the decision problems arising from the definitions given in Section 1.2; they are all stated for a fixed integer $r \geq 1$:

Problem DC$_r$ / LDC$_r$ / IDC$_r$ ({r-Dominating / r-Locating-Dominating / r-IDentifying} Code with bounded size):
Instance: A graph G and an integer k.
Question: Does G admit an {r-dominating / r-locating-dominating / r-identifying} code of size at most k?

Proposition 8 *Let $r \geq 1$ be any integer.*
(a) [16, p. 75 and p. 190, for $r = 1$], [19, Prop. 9] The decision problem DC$_r$ is NP-complete.
(b) [13, for $r = 1$], [10] The decision problem LDC$_r$ is NP-complete.
(c) [12, for $r = 1$], [10] The decision problem IDC$_r$ is NP-complete.

We have also results on the complexities of these problems when the question is about the uniqueness of the existence of a suitable set:

Proposition 9 *Let $r \geq 1$ be any integer.*
(a) [21, Th. 25] The decision problems U-SAT and U-DC$_r$ have equivalent complexities, up to polynomials.
(b) [20, Th. 20] The decision problems U-SAT and U-LDC$_r$ have equivalent complexities, up to polynomials.
(c) [20, Th. 35] The decision problems U-SAT and U-IDC$_r$ have equivalent complexities, up to polynomials.

As a consequence, by Corollary 6, the problems U-DC$_r$, U-LDC$_r$ and U-IDC$_r$ are *NP*-hard and belong to *DP*.

What about the same problems (with or without uniqueness of solution) when we consider coloured graphs? Note that it would not be interesting to ask whether there is, e.g., a tropical r-dominating code of size at most k in a coloured graph G_{col}: in this case, we can simply observe that with a graph coloured with only one colour, we are brought back to the basic problem DC$_r$. Much more interesting is to consider the existence of rainbow sets, that is, to try to locate the following problems in the classes of complexity, and this is what we shall do in the sequel, with the additional requirement that all graphs be connected (see Remark 23):

Problem [U-]{RDC_r / RLDC_r / RIDC_r} ([Unique] Rainbow {r-Dominating / r-Locating-Dominating / r-IDentifying} Code):
Instance: A connected, coloured graph G_{col}.
Question: Does G_{col} admit a [unique] rainbow {r-dominating / r-locating-dominating / r-identifying} code?

1.5 Outline of the Paper

In Sections 2.1–2.3, we give results on the *NP*-completeness of the problems RDC_1, RLDC_1 and RIDC_1.

In Sections 2.4–2.6, we prove that the three problems U-RDC_1, U-RLDC_1 and U-RIDC_1 have a complexity which is equivalent to that of U-SAT or U-3-SAT.

In some cases, we have results which hold even for paths or trees, or for graphs with a small number of occurrences for the colours appearing in the graph (typically, 2 or 3).

Then, in Sections 2.7–2.9, we shall extend our results to any r.

For the three types of codes, the general approach is the same, but the results slightly vary, and each type requires proofs which are different in their technical details, and cannot be merged. The starting point is the following result.

Proposition 10 *[1, Th. 2.1], [2, Th. 2.1] The problem RDC_1 is NP-complete, even when restricted to coloured paths.*

We give the proof, because it will be used and transformed for subsequent proofs. When, during the construction of a coloured graph, we say that a vertex has or is given a unique colour, we mean that this colour, at the end of the construction, appears exactly once. By extension, a vertex with unique colour is said to be *unique*, so that any unique vertex necessarily belongs to any rainbow set.

Proof of Proposition 10. In view of the subsequent proofs, we change slightly the proof from [1], [2]. The problem is clearly inside *NP*. We give a polynomial transformation from 3-SAT to RDC_1. Let the set $\mathcal{C}$ of m clauses over n variables x_1, x_2, …, x_n be an instance of 3-SAT, for which we may assume that each variable x_i appears with its two forms, x_i and $\overline{x}_i$ (otherwise, if it appears only in, say, its negated form, it suffices to take $x_i = \text{F}$, and this variable and the clauses where it appears do not need to be considered anymore).

We write $\mathcal{C} = \{\{\ell_1, \ell_2, \ell_3\}, \{\ell_4, \ell_5, \ell_6\}, \ldots, \{\ell_{3m-2}, \ell_{3m-1}, \ell_{3m}\}\}$, where each literal ℓ_j is a variable x_i or its complement $\overline{x}_i$. From this instance, we define a coloured path $\mathcal{P}$ such that $\mathcal{P}$ admits a rainbow 1-dominating set if and only if there is an assignment of the variables that satisfies $\mathcal{C}$. Example 12 below is intended to help to understand the notation.

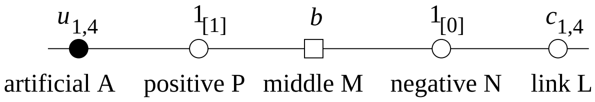

Figure 2: The gadget $W_{1,4}$ ($i = 1, i(1) = 4$). The artificial vertex is represented by a black circle because it belongs to any rainbow set, the middle vertex by a square because it does not belong to any rainbow 1-dominating set, the other three vertices are unspecified.

We first construct a path

$$\mathcal{P}_0 = z_1 z_2 S_1 v_1 v_2 v_3 S_2 v_4 v_5 v_6 S_3 \ldots S_m v_{3m-2} v_{3m-1} v_{3m} S_{m+1},$$

and we colour it in the following way. Vertices z_1 and z_2 are Blue (colour b). Each vertex S_s, $1 \le s \le m+1$, receives a unique colour. Each vertex v_i, $1 \le i \le 3m$, which corresponds to the literal ℓ_i and is called a *clausal* vertex, is coloured with the colour $\phi(v_i) = i_{[0]}$.

Next, we define a number of gadgets as follows. Whenever a pair of literals ℓ_p, ℓ_q satisfies $\ell_p = \overline{\ell}_q$, we say that they are *antithetic* to each other, and the same applies for the corresponding clausal vertices v_p and v_q. For each literal ℓ_i, $1 \le i \le 3m$, we consider the list of all the literals $\ell_{i(1)}, \ell_{i(2)}, \ldots, \ell_{i(k_i)}$, that are antithetic to ℓ_i (by assumption, there is at least one). To each literal $\ell_{i(f)}$, $1 \le f \le k_i$, is associated a *constraint gadget* $W_{i,i(f)}$ consisting of a path on five vertices, $A_{i,i(f)}$, $P_{i,i(f)}$, $M_{i,i(f)}$, $N_{i,i(f)}$ and $L_{i,i(f)}$; vertex $A_{i,i(f)}$ is called *artificial* and has a unique colour, $u_{i,i(f)}$; vertex $P_{i,i(f)}$ is the *positive* vertex of $W_{i,i(f)}$ and has colour $i_{[f]}$; vertex $M_{i,i(f)}$ is the *middle* vertex of $W_{i,i(f)}$, and $\phi(M_{i,i(f)}) = b$; vertex $N_{i,i(f)}$ is the *negative* vertex of $W_{i,i(f)}$ and has colour $i_{[f-1]}$; vertex $L_{i,i(f)}$ is the *link* vertex of $W_{i,i(f)}$ and has colour $c_{i,i(f)}$ if $i < i(f)$, colour $c_{i(f),i}$ otherwise.

Remark 11 *(a) In $\mathcal{P}$, the vertex z_1 will not be linked to any vertex other than z_2; therefore, z_1 or z_2 necessarily belongs to any 1-dominating set in $\mathcal{P}$, and no Blue vertex other than z_1 or z_2 can belong to any rainbow 1-dominating set in $\mathcal{P}$, i.e., no middle vertex.*

(b) Every vertex S_s, $1 \le s \le m+1$, and every artificial vertex $A_{i,i(f)}$, $1 \le i \le 3m$, $1 \le f \le k_i$, necessarily belong to any rainbow set.

(c) Every positive vertex $P_{i,i(k_i)}$ (which has colour $i_{[k_i]}$) is unique (see Remark 13 below) and belongs to any rainbow set.

Example 12 *Assume that $\ell_1 = \ell_7 = \ell_{13} = x_1$, $\ell_4 = \ell_{10} = \overline{x}_1$ and there is no other occurrence of x_1 nor $\overline{x}_1$ in $\mathcal{C}$. Figure 2 represents the gadget $W_{1,4}$.*

We have $k_1 = k_7 = k_{13} = 2$, $k_4 = k_{10} = 3$, and $\ell_{1(1)} = \ell_4$, $\ell_{1(2)} = \ell_{10}$, $\ell_{7(1)} = \ell_4$, …, $\ell_{10(3)} = \ell_{13}$.

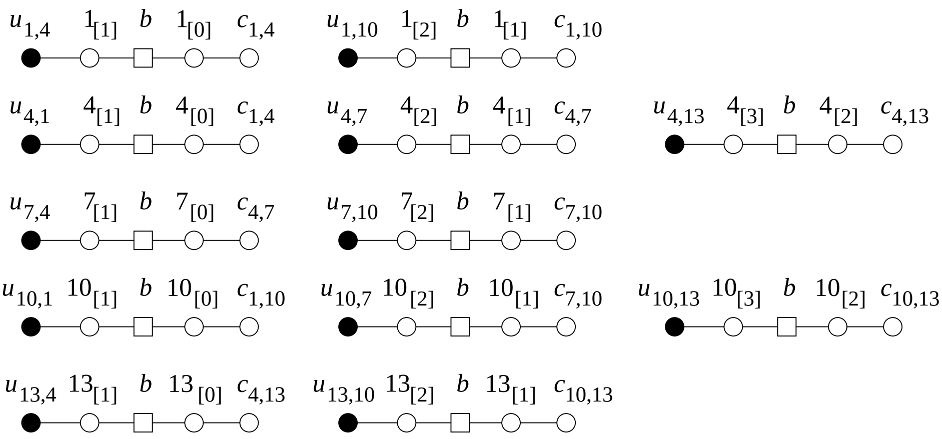

Figure 3: The 12 gadgets from Example 12.

Figure 3 represents the 12 gadgets $W_{1,4}$, $W_{1,10}$, $W_{4,1}$, $W_{4,7}$, $W_{4,13}$, $W_{7,4}$, $W_{7,10}$, $W_{10,1}$, $W_{10,7}$, $W_{10,13}$, $W_{13,4}$, $W_{13,10}$, *produced by* $\ell_1 = x_1$, $\ell_4 = \overline{x}_1$, $\ell_7 = x_1$, $\ell_{10} = \overline{x}_1$, $\ell_{13} = x_1$.

Remark 13 *One can see that the colours* $c_{i,i(f)}$ *appear exactly twice, on the link vertices of the gadgets* $W_{i,i(f)}$ *and* $W_{i(f),i}$*, and the same is true for the colours* $i_{[0]}$ *(on one clausal vertex and in one gadget),* $i_{[1]}$ *(in two gadgets), ..., except for the "last" colour* $i_{[k_i]}$*, which appears only once, on the positive vertex* $P_{i,i(k_i)}$ *(in Figure 3, these colours are* $1_{[2]}$, $4_{[3]}$, $7_{[2]}$, $10_{[3]}$ *and* $13_{[2]}$*). Only the Blue colour appears more than twice.*

Finally, the path $\mathcal{P}$ is obtained by concatenating $\mathcal{P}_0$ and all the different gadgets, and creating a unique vertex J, which is linked to the last vertex of the last gadget. The gadgets are ordered lexicographically; thus, in our example, we obtain $\mathcal{P} = \mathcal{P}_0 W_{1,4} W_{1,10} W_{4,1} \dots W_{13,10} J$.

Clearly, the construction is polynomial in the size of the instance of 3-SAT.

(a) We assume that there is a solution to 3-SAT, i.e., an assignment of the variables that satisfies $\mathcal{C}$. We construct a rainbow 1-dominating set V^* by putting the following vertices in V^*: (i) the vertex z_1 and all the unique vertices; (ii) for every true literal ℓ_i, the clausal vertices v_i (with colour $i_{[0]}$) and, for every $f \in \{1, \dots, k_i\}$, the positive vertices $P_{i,i(f)} \in W_{i,i(f)}$ and the link vertices $L_{i,i(f)} \in W_{i,i(f)}$; (iii) for every false literal ℓ_i and for every $f \in \{1, \dots, k_i\}$, the negative vertices $N_{i,i(f)} \in W_{i,i(f)}$.

See Figure 4 for the gadgets of Example 12 with $x_1 =$ FALSE. Note that some vertices may be codewords for two reasons, (i) and (ii). In Figure 4, this is the case for the two positive vertices with colours $4_{[3]}$ and $10_{[3]}$ (the two large black vertices).

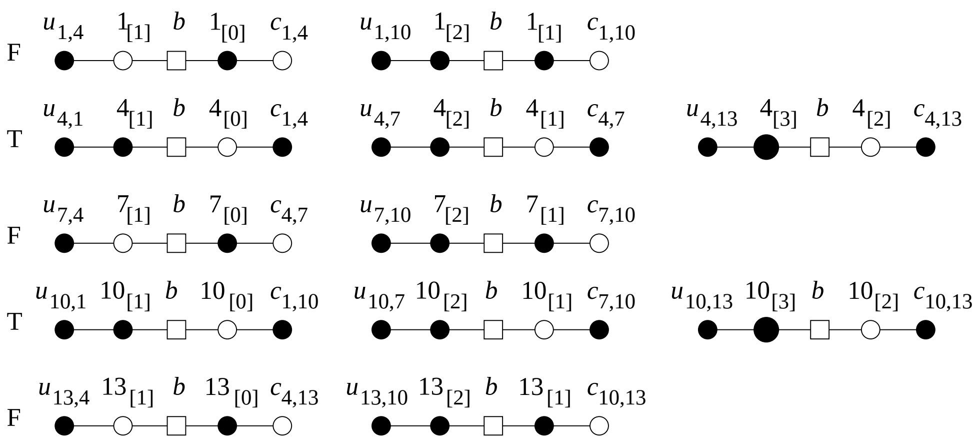

Figure 4: In black, the codewords in the 12 gadgets from Example 12, with $x_1 =$F, i.e., $\ell_1 =$F, $\ell_4 =$T, $\ell_7 =$F, $\ell_{10} =$T, $\ell_{13} =$F.

We claim that V^* is both rainbow and 1-dominating.

It is straightforward to check that the Blue colour, the unique colours and every colour $i_{[f]}$, $1 \leq i \leq 3m$, $0 \leq f \leq k_i$, appear exactly once in V^*. All that remains to be checked are the colours $c_{i,i(f)}$ of the link vertices in the gadgets $W_{i,i(f)}$ and $W_{i(f),i}$; since these two gadgets correspond to two literals that are antithetic, exactly one of them has been set TRUE by the assignment and exactly one of these two link vertices has been taken in the code. So V^* is rainbow.

For every $s \in \{1, \ldots, m\}$, the clausal vertices v_{3s-2} and v_{3s} are 1-dominated by $S_s \in V^*$ and $S_{s+1} \in V^*$, respectively; since by assumption there is at least one true literal in each clause, the clausal vertex v_{3s-1} is 1-dominated by at least one clausal vertex which is a codeword. Inside a gadget, either the negative vertex, or the positive and link vertices, are codewords, and in both cases, every vertex is 1-dominated by V^*. Hence V^* is a 1-dominating set.

(b) We assume that there is a solution to RDC$_1$, i.e., a rainbow 1-dominating set V^*. By Remark 11(a), no middle vertex can belong to V^*.

For every clausal vertex v_i (with colour $i_{[0]}$) that is a codeword, we assign the value to the corresponding variable in such a way that ℓ_i is TRUE. We claim that this (partial) assignment is consistent. Assume on the contrary that there is a pair of antithetic literals ℓ_p and ℓ_q receiving the same value by the assignment just defined, i.e., that the two antithetic clausal vertices v_p and v_q are codewords. Assume without loss of generality that $p < q$. There is an $f \geq 1$ such that $\ell_q = \ell_{p(f)}$, and two gadgets, $W_{p,p(f)}$ and $W_{p(f),p}$, both containing a link vertex with the common colour $c_{p,p(f)}$.

Consider $W_{p,p(f)}$ and its link vertex $L_{p,p(f)}$, and assume that $L_{p,p(f)} \notin$

V^*. Assume first that $f > 1$. Because $M_{p,p(f)} \notin V^*$, we have $N_{p,p(f)} \in V^*$, then $P_{p,p(f-1)} \notin V^*$ (because it has the same colour as $N_{p,p(f)}$), then $N_{p,p(f-1)} \in V^*$ (to have one codeword that 1-dominates $M_{p,p(f-1)}$), ..., $N_{p,p(1)} \in V^*$; but $N_{p,p(1)}$ has colour $p_{[0]}$, like v_p which is also a codeword, and this contradicts the fact that V^* is rainbow. If $f = 1$, we get immediately the same conclusion. So $L_{p,p(f)}$ must belong to V^*. But the same argument can be applied to $W_{p(f),p}$ and $L_{p(f),p}$, so the two link vertices, which have the same colour, are codewords, which is impossible. Therefore, the assignment defined by V^* is valid. If necessary, we complete the assignment by giving the value TRUE to the remaining unassigned variables.

Finally, this assignment satisfies 3-SAT because, for domination reasons, in each clause at least one clausal vertex belongs to V^*, i.e., at least one literal is true.

So the answer to the initial instance of 3-SAT is YES if and only if the answer to the constructed instance of RDC$_1$ is YES. $\triangle$

2 New Results

The first three Subsections of this Section are devoted to the problems RDC$_1$, RLDC$_1$ and RIDC$_1$: we study one variant for RDC$_1$, then we prove that RLDC$_1$ and RIDC$_1$ are *NP*-complete, i.e., their complexity is equivalent, up to polynomials, to that of, e.g., SAT.

The following three Subsections show that the same three problems with unique solution have complexity equivalent, up to polynomials, to that of U-SAT.

Then, in Subsections 2.7–2.9, we shall extend our results to RDC$_r$, RLDC$_r$ and RIDC$_r$, for any $r > 1$.

2.1 Rainbow 1-Dominating Codes

We can go further and ask for a fixed number of occurrences for the colours appearing in the graph. With this number restricted to 2, we have a result of *NP*-completeness for coloured trees.

Proposition 14 *The problem RDC$_1$ is NP-complete, even when restricted to trees where each colour appears at most twice.*

Proof. We consider the construction in the proof of Proposition 10, where every middle vertex $M_{i,i(f)}$ in the gadget $W_{i,i(f)}$ had been given the colour b. As noticed in Remark 13, the Blue colour is the only colour appearing more than twice in the graph. To get rid of these multiple occurrences, we proceed as follows: (i) we delete the vertices z_1, z_2 and the edges z_1z_2 and z_2S_1; (ii) for every middle vertex, we set $\phi(M_{i,i(f)}) = b_{i,i(f)}$ and we create the graph $H_{i,i(f)} = (V_{i,i(f)}, E_{i,i(f)})$ with

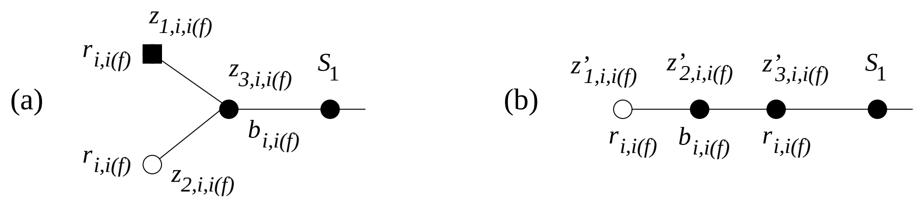

Figure 5: (a) The graph $H_{i,i(f)}$ of Propositions 14 and 17, with its link to S_1; (b) The graph $H'_{i,i(f)}$ of Propositions 19 and 28, with its link to S_1. In black, the codewords, in white the non-codewords; black circles are forced codewords.

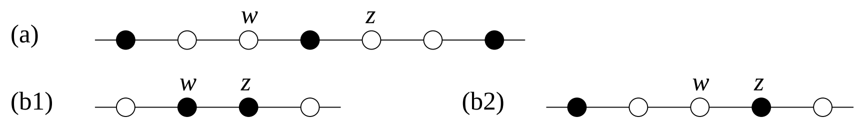

Figure 6: In black, codewords, and in white, non-codewords. The vertices w and z are not 1-separated by any codeword.

$$V_{i,i(f)} = \{z_{1,i,i(f)}, z_{2,i,i(f)}, z_{3,i,i(f)}\}, E_{i,i(f)} = \{z_{1,i,i(f)}z_{3,i,i(f)}, z_{2,i,i(f)}z_{3,i,i(f)}\},$$

together with the edge $z_{3,i,i(f)}S_1$. We set $\phi(z_{3,i,i(f)}) = b_{i,i(f)}$ and give a new colour, say $r_{i,i(f)}$ (for Red), to both $z_{1,i,i(f)}$ and $z_{2,i,i(f)}$: see Figure 5(a). Now the new graph is a tree and has no colour appearing more than twice. Since the Red vertices $z_{1,i,i(f)}$ and $z_{2,i,i(f)}$ must be 1-dominated by some codeword and cannot both belong to a rainbow set, we see that $z_{3,i,i(f)}$ necessarily belongs to any rainbow 1-dominating code (together with exactly one of $z_{1,i,i(f)}$ and $z_{2,i,i(f)}$); the consequence is that again, no middle vertex can belong to any rainbow 1-dominating set. The proof then goes exactly as for Proposition 10. $\triangle$

2.2 Rainbow 1-Locating-Dominating Codes

We now turn to 1-locating-dominating codes. We obtain the same results as for 1-dominating codes, one for paths without limitation on the occurrences of colours, one for trees where every colour appears at most twice.

Remark 15 *On a path, the only configuration for a set which is 1-dominating and not 1-LD is given by Figure 6(a). One can see that this will never occur in the gadgets described in the proof of Proposition 10.*

Proposition 16 *The problem $RLDC_1$ is NP-complete, even when restricted to coloured paths.*

Proof. We consider again the construction in the proof of Proposition 10. We duplicate the vertices S_s, $1 \leq s \leq m$, so the path $\mathcal{P}_0$ now reads

$$\mathcal{P}_0 = z_1 z_2 S_1 S_1' v_1 v_2 v_3 S_2 S_2' v_4 v_5 v_6 S_3 S_3' \dots S_m S_m' v_{3m-2} v_{3m-1} v_{3m} S_{m+1},$$

where each S_s' is unique. The proof now goes exactly as previously:

(a) We construct a code V^* by following the same rules (i)–(iii) as in the Case (a) of the proof of Proposition 10. It is straightforward to see that all non-codewords are 1-dominated and 1-separated by the code in the gadgets (cf. Remark 15) as well as in $\mathcal{P}_0$, since the duplications avoid the configurations of the type:
$v_{3s-2} \in V^*$, $v_{3s-1} \notin V^*$, $v_{3s} \notin V^*$, $S_{s+1} \in V^*$, $v_{3s+1} \notin V^*$, $v_{3s+2} \notin V^*$, $v_{3s+3} \in V^*$
(cf. Figure 6(a)), that could exist in the previous construction. So V^* is a rainbow 1-LD set.

(b) Assume that there exists a rainbow 1-LD set V^*. Then we can apply *mutatis mutandis* the Case (b) of the proof of Proposition 10, and construct a valid assignment satisfying 3-SAT. △

Proposition 17 *The problem RLDC$_1$ is NP-complete, even when restricted to trees where each colour appears at most twice.*

Proof. Compared to the proof of Proposition 10, we duplicate the vertices S_s, $1 \leq s \leq m$, like we did for Proposition 16, and, like for Proposition 14, (i) we delete the vertices z_1, z_2 and the edges $z_1 z_2$ and $z_2 S_1$; (ii) for every middle vertex, we set $\phi(M_{i,i(f)}) = b_{i,i(f)}$ and we create the same graph $H_{i,i(f)}$ and the edge $z_{3,i,i(f)} S_1$, cf. Figure 5(a). Again, the new graph is a tree and has no colour appearing more than twice, and $z_{3,i,i(f)}$ necessarily belongs to any rainbow 1-LD code. The proof then goes like the previous ones. △

2.3 Rainbow 1-Identifying Codes

We now consider 1-identifying codes. We do not obtain a result which would be valid for paths, as was the case for the problems RDC$_1$ and RLDC$_1$, but we do have a result on trees with occurrences of colours at most 2.

Remark 18 *On a path, the only three configurations for a set which is dominating and not ID are given by Figure 6(a)–(b1)–(b2).*

Proposition 19 *The problem RIDC$_1$ is NP-complete, even when restricted to trees where each colour appears at most twice.*

Proof. We consider again the construction for the proof of Proposition 10.

(i) We triplicate the vertices S_s, $1 \leq s \leq m+1$, so the path $\mathcal{P}_0$ now reads

$$z_1 z_2 S_1 S_1' S_1'' v_1 v_2 v_3 S_2 S_2' S_2'' v_4 \dots S_m S_m' S_m'' v_{3m-2} v_{3m-1} v_{3m} S_{m+1} S_{m+1}' S_{m+1}'',$$

where S'_s and S''_s are unique. We also triplicate all the artificial vertices, and the gadget $W_{i,i(f)}$ now reads $A_{i,i(f)}A'_{i,i(f)}A''_{i,i(f)}P_{i,i(f)}M_{i,i(f)}N_{i,i(f)}L_{i,i(f)}$, where $A'_{i,i(f)}$ and $A''_{i,i(f)}$ are unique. Moreover, we triplicate the final vertex J by creating the unique vertices J' and J'', together with the edges JJ' and $J'J''$.

(ii) For each middle vertex $M_{i,i(f)} \in W_{i,i(f)}$, we create the unique vertex $Y_{i,i(f)}$, which is linked to $M_{i,i(f)}$: this will avoid the configuration given by Figure 6(b2), which appears for instance in the first gadget of Figure 4. The graph thus constructed is already not a path anymore.

(iii) We get rid of the multiple Blue colours of the middle vertices by deleting the vertices z_1, z_2 and the edges z_1z_2 and z_2S_1, and creating, for every middle vertex $M_{i,i(f)}$, the graph $H'_{i,i(f)} = (V'_{i,i(f)}, E'_{i,i(f)})$, with

$$V'_{i,i(f)} = \{z'_{1,i,i(f)}, z'_{2,i,i(f)}, z'_{3,i,i(f)}\}, E'_{i,i(f)} = \{z'_{1,i,i(f)}z'_{2,i,i(f)}, z'_{2,i,i(f)}z'_{3,i,i(f)}\},$$

together with the edge $z'_{3,i,i(f)}S_1$. The colours are: $\phi(M_{i,i(f)}) = \phi(z'_{2,i,i(f)}) = b_{i,i(f)}$ and $\phi(z'_{1,i,i(f)}) = \phi(z'_{3,i,i(f)}) = r_{i,i(f)}$, see Figure 5(b). The only vertex 1-separating $z'_{1,i,i(f)}$ and $z'_{2,i,i(f)}$ is $z'_{3,i,i(f)}$, so $z'_{3,i,i(f)}$ belongs to every 1-identifying code. Then $z'_{1,i,i(f)}$ cannot be a codeword, and $z'_{2,i,i(f)}$ is a codeword: observe already that here, we have no choice for these three vertices, unlike in the case of the graphs $H_{i,i(f)}$ for rainbow 1-D and 1-LD codes.

The graph G just constructed is a tree, and no colour appears more than twice.

(a) We construct a code V^* in G by following the same rules (i)–(iii) as in the Case (a) of the proof of Proposition 10. In particular, every artificial vertex and every $Y_{i,i(f)}$ are codewords, because they are unique, and inside a gadget, either the negative vertex, or the positive and link vertices, are codewords. It is easy to see why the three forbidden configurations of Figure 6 cannot appear in $\mathcal{P}_0$. Also thanks to the triplications, inside each gadget, the artificial, positive and link vertices are all 1-dominated and 1-separated from all vertices by the artificial vertices.

If we have taken $N_{i,i(f)}$ in the code, then $I_{G,V^*,1}(N_{i,i(f)}) = \{N_{i,i(f)}\}$ and $I_{G,V^*,1}(M_{i,i(f)}) = \{N_{i,i(f)}, Y_{i,i(f)}\}$.

If we have taken $P_{i,i(f)}$ and $L_{i,i(f)}$ in the code, then $I_{G,V^*,1}(N_{i,i(f)}) = \{L_{i,i(f)}\}$ and $I_{G,V^*,1}(M_{i,i(f)}) = \{P_{i,i(f)}, Y_{i,i(f)}\}$.

In both cases, $I_{G,V^*,1}(Y_{i,i(f)}) = \{Y_{i,i(f)}\}$, and all vertices are 1-dominated and 1-separated by V^*: we can conclude that V^* is a rainbow 1-identifying set.

(b) Assume that there exists a rainbow 1-ID set V^*. Then we can still use the same argument as in the Cases (b) of the proofs of Propositions 10 and 16, in particular because every middle vertex $M_{i,i(f)}$ still needs to be 1-dominated by a codeword belonging to $W_{i,i(f)} \setminus \{M_{i,i(f)}\}$, in order to be 1-separated from $Y_{i,i(f)}$ by V^*. $\triangle$

2.4 Unique Rainbow 1-Dominating Codes

We are going to prove that U-SAT and U-RDC$_1$ have equivalent complexities, up to polynomials; to this effect, we give a polynomial transformation from U-RDC$_1$ to U-SAT (Proposition 20) and a polynomial transformation from U-3-SAT to U-RDC$_1$ (Proposition 21). One originality of this work is that we need to go both ways; in particular, we need the first transformation because U-SAT, or U-3-SAT, is not sufficiently well located inside DP.

Proposition 20 *There exists a polynomial transformation from U-RDC$_1$ to U-SAT.*

Proof. We start from an instance of U-RDC$_1$, a coloured graph $G = (V, E)$ of order n, where $V = \{v_1, v_2, \dots, v_n\}$; the set $\{c_1, c_2, \dots, c_\gamma\}$ is the set of γ colours used on V, and the number of occurrences of the colour c_i is λ_i; we set $\Lambda_i = \Sigma_{j=1}^{i} \lambda_j$ for $1 \leq i \leq \gamma$. Without loss of generality, we can assume that $V_1 = \{v_1, \dots, v_{\Lambda_1}\}$ is the set of vertices with colour c_1, $V_i = \{v_{\Lambda_{i-1}+1}, \dots, v_{\Lambda_i}\}$ the set of vertices with colour c_i $(2 \leq i \leq \gamma)$, so that the sets V_i, $1 \leq i \leq \gamma$, partition V and the vertices are ranked by increasing index of colour. For each vertex v_i, we denote by $v_i^1, \dots, v_i^{s(i)}$ the $s(i)$ neighbours of v_i.

For each vertex v_i, we create the variable x_i. The set of clauses for U-SAT is constructed in the following way:

(a) for every $i \in \{1, \dots, n\}$, we create the clause $\{x_i, x_i^1, \dots, x_i^{s(i)}\}$;

(b1) for every $i \in \{1, \dots, \gamma\}$, we create the clause $\{x_j : v_j \in V_i\}$;

(b2) for every $i \in \{1, \dots, \gamma\}$ and for every pair of vertices $\{v_p, v_q\} \subseteq V_i$, $p < q$, we create the clause $\{\overline{x}_p, \overline{x}_q\}$.

Note that the number of variables and clauses is polynomial with respect to n, the order of G.

Now assume that we have a unique rainbow 1-dominating set V^* in G. Define the assignment $\mathcal{A}_1$ on the variables x_i by $\mathcal{A}_1(x_i) = \mathrm{T}$ if and only if $v_i \in V^*$. It is quite easy to see that the clauses described above are all satisfied: the clauses in (a) because V^* is 1-dominating, the clauses in (b1) because every colour appears at least once in V^*, and the clauses in (b2) because every colour appears at most once in V^*.

Is $\mathcal{A}_1$ unique? Assume on the contrary that another assignment, $\mathcal{A}_2$, also satisfies the constructed instance of U-SAT, and define the vertex set V^+ by the rule $v_i \in V^+$ if and only if $\mathcal{A}_2(x_i) = \mathrm{T}$. Since $\mathcal{A}_1 \neq \mathcal{A}_2$, we have $V^+ \neq V^*$.

Because at least one literal is set TRUE by $\mathcal{A}_2$ in each clause from (a), every vertex v_i is in V^+ or has a neighbour in V^+, so the set V^+ is 1-dominating; the clauses in (b1), when satisfied by $\mathcal{A}_2$, show that every colour appears at least once in V^+ and the clauses in (b2), that every colour appears at most once in V^+. Therefore, V^+ is a rainbow 1-dominating set, but this contradicts the uniqueness of V^*.

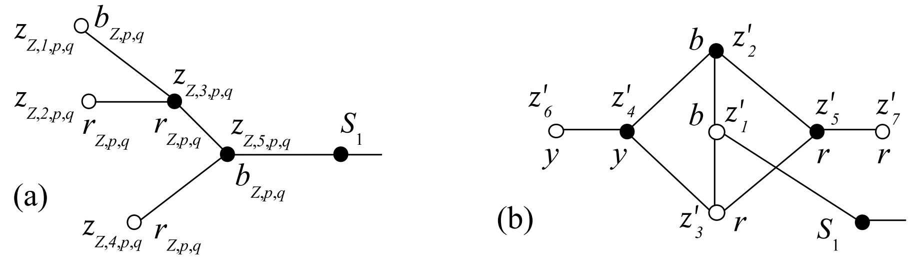

Figure 7: (a) The graph $H_{Z,p,q}$ for Proposition 21; (b) The graph $H'_{Z,p,q}$ for Proposition 25, with lightened notation. Codewords are in black; all codewords are forced.

So a YES answer for U-RDC$_1$ leads to a YES answer for U-SAT. Assume now that the answer to U-RDC$_1$ is negative. If it is negative because there are at least two rainbow 1-D codes, then we have at least two assignments satisfying the instance of U-SAT: we have seen above how to construct a suitable assignment from a rainbow 1-D code, and different rainbow 1-D codes obviously lead to different assignments. If there is no rainbow 1-D code, then there is no assignment satisfying U-SAT, because such an assignment would give a rainbow 1-D code, as we have seen above from $\mathcal{A}_2$. So in both cases, a NO answer to U-RDC$_1$ implies a NO answer to U-SAT. $\triangle$

Proposition 21 *There exists a polynomial transformation from U-3-SAT to U-RDC$_1$. Moreover, in the connected graph constructed for this transformation, each colour appears at most thrice.*

Proof. We consider again the construction for the proof of Proposition 10.

(i) We delete the vertices z_1, z_2 and the edges z_1z_2 and z_2S_1.

(ii) For each pair of antithetic literals ℓ_p, ℓ_q, we add one vertex $Z_{p,q}$ and the edges $v_pZ_{p,q}$, $Z_{p,q}v_q$, and set $\phi(Z_{p,q}) = b_{Z,p,q}$; since we assumed that each variable x_i appears under its two forms, x_i and $\overline{x}_i$, every clausal vertex is linked to at least one vertex of type Z. We then create the graph $H_{Z,p,q}$ with vertex set $V_{Z,p,q} = \{z_{Z,1,p,q}, z_{Z,2,p,q}, z_{Z,3,p,q}, z_{Z,4,p,q}, z_{Z,5,p,q}\}$ and edge set $E_{Z,p,q} = \{z_{Z,1,p,q}z_{Z,3,p,q}, z_{Z,2,p,q}z_{Z,3,p,q}, z_{Z,3,p,q}z_{Z,5,p,q}, z_{Z,4,p,q}z_{Z,5,p,q}\}$, together with the edge $z_{Z,5,p,q}S_1$. We set $\phi(z_{Z,1,p,q}) = \phi(z_{Z,5,p,q}) = b_{Z,p,q}$ and $\phi(z_{Z,2,p,q}) = \phi(z_{Z,3,p,q}) = \phi(z_{Z,4,p,q}) = r_{Z,p,q}$. See Figure 7(a).

(iii) We give the colour $b_{M,i,i(f)}$ to every middle vertex $M_{i,i(f)} \in W_{i,i(f)}$ and we create the graph $H_{M,i,i(f)}$ which is identical to the graph $H_{Z,p,q}$, except that we replace Z by M, p by i and q by $i(f)$, cf. Figure 7(a).

The number of vertices linked to S_1, apart from v_1, is equal to the number of pairs of antithetic clausal vertices plus the number of gadgets, i.e., 1.5

times the number of gadgets. One can already see that, in a rainbow 1-D code, in $H_{Z,p,q}$ and $H_{M,i,i(f)}$, the vertices $z_{Z,3,p,q}$ and $z_{Z,5,p,q}$ on the one hand, the vertices $z_{M,3,i,i(f)}$ and $z_{M,5,i,i(f)}$ on the other hand, are necessarily codewords, and the other vertices are not. As a consequence, no middle vertex and no vertex $Z_{p,q}$ can belong to any rainbow 1-D code.

In the constructed graph, each colour appears at most thrice.

As in Case (b) of the proof of Proposition 10 about the consistency of the assignment defined by a rainbow 1-D code, two antithetic causal vertices v_p and v_q cannot both be codewords, but, because $Z_{p,q}$ cannot belong to any rainbow 1-D code, one of them is a codeword; this extends to all the vertices antithetic to v_p, and so, all the clausal vertices corresponding to a variable x belong to a rainbow 1-D code and none corresponding to $\overline{x}$, or the other way round.

(a) Assume that there is a unique assignment satisfying the instance of 3-SAT. We build the set V^* in the following way: (i) for every true literal ℓ_i, the clausal vertex v_i, with colour $i_{[0]}$, belongs to V^*; (ii) all the vertices $z_{Z,3,p,q}$, $z_{Z,5,p,q}$, $z_{M,3,i,i(f)}$ and $z_{M,5,i,i(f)}$, and all the unique vertices belong to V^*; (iii) for every true literal ℓ_i and $f \in \{1, \dots, k_i\}$, the positive vertex $P_{i,i(f)} \in W_{i,i(f)}$ and the link vertex $L_{i,i(f)} \in W_{i,i(f)}$ belong to V^*; (iv) for every false literal ℓ_i and $f \in \{1, \dots, k_i\}$, the negative vertex $N_{i,i(f)} \in W_{i,i(f)}$ belongs to V^*.

This is by now a routine task to check that V^* is indeed a rainbow 1-dominating set. What is new is that, after the step (i) has been performed, i.e., once the decision has been made for all the clausal vertices, there is *no choice* left for the completion of a rainbow 1-D code. To prove this claim, all we have to check are the gadgets, since we have already seen that the step (ii) is forced. Consider the literal ℓ_1, its first antithetic literal $\ell_{1(1)}$, and the gadget $W_{1,1(1)}$, and assume first that $\ell_1 =$T. Then $v_1 \in V^*$; this implies that $N_{1,1(1)} \notin V^*$, which implies in turn that $L_{1,1(1)} \in V^*$, $P_{1,1(1)} \in V^*$ and $N_{1,1(2)} \notin V^*$. So we can go on in the same way for $W_{1,1(2)}$: $L_{1,1(2)} \in V^*$, $P_{1,1(2)} \in V^*$ and $N_{1,1(3)} \notin V^*$, and so on until we reach $W_{1,1(k_1)}$, the last gadget for ℓ_1. Here, we have $L_{1,1(k_1)} \in V^*$, $P_{1,1(k_1)} \in V^*$. We can repeat this argument for all the literals set TRUE by the assignment. Assume now that $\ell_1 =$F. Then all the antithetic vertices of v_1 are codewords, and for $1 \leq f \leq k_1$, all the link vertices $L_{1(f),1}$ are codewords, none of the link vertices $L_{1,1(f)}$ are codewords, all the negative vertices $N_{1,1(f)}$ are codewords, none of the positive vertices $P_{1,1(f-1)}$ (when $f = 1$, this is v_1) are codewords —and the vertex $P_{1,1(k_1)}$ is a codeword, because it is unique. This is true for all false literals, and so we had no choice for the positive, negative and link vertices (and we know from the beginning that the artificial vertices are codewords and the middle vertices are not).

Now assume that there exists another rainbow 1-dominating set, V^+. Then, denoting by $V_{\mathcal{C}}$ the set of clausal vertices, we have: $V_{\mathcal{C}} \cap V^* \neq V_{\mathcal{C}} \cap V^+$. Setting the variable x to TRUE if and only if one clausal vertex v_i

corresponding to one literal $\ell_i = x$ belongs to V^+ (and we have seen that then all the literals equal to x are in V^+, and those equal to $\overline{x}$ are not), we obtain a second valid assignment, which as before satisfies the instance of 3-SAT. This is a contradiction.

(b) Assume that the answer to U-3-SAT is NO, either because there is no assignment satisfying the instance, or because there are at least two such assignments. In the latter case, this would lead however to two different rainbow 1-dominating sets. On the other hand, if no assignment exists, then no rainbow 1-D code V^* exists, because V^* would lead, as before, to a valid assignment satisfying all the clauses —a contradiction.

Therefore, there is a YES answer to U-3-SAT if and only if there is a YES answer to U-RDC$_1$. $\triangle$

Remark 22 *We can be more specific than in the last sentence of the proof above: actually, the proof shows that there is zero, one, or more than one solution to the instance of 3-SAT if and only if there is zero, one, or more than one solution, respectively, in the constructed coloured graph; this proves that this construction could also have been used to provide a polynomial transformation from 3-SAT to RDC$_1$, i.e., a proof of NP-completeness for RDC$_1$. This would not however give a result for paths, like Proposition 10, nor for trees where each colour appears at most twice, like Proposition 14.*

Remark 23 *If we allow unconnected graphs, then it is easy to build a graph where each colour appears at most twice, by using isolated vertices with a colour that we do not wish elsewhere: instead of building the graph $H_{Z,p,q}$ so that the vertex $Z_{p,q}$ with the colour $b_{Z,p,q}$ cannot be a codeword, we can simply create one isolated vertex with this colour. The same is true for $H_{M,i,i(f)}$.*

2.5 Unique Rainbow 1-Locating-Dominating Codes

We have the same results for unique rainbow 1-LD codes as for unique rainbow 1-D codes.

Proposition 24 *There exists a polynomial transformation from U-RLDC$_1$ to U-SAT.*

Proof. The method is the same as in the proof of Proposition 20, and uses the characterization of LD codes in Remark 3, and in particular (2). Only the clauses in (a) will change, in order to fit the required location-domination property: compared to the previous construction, the clauses in (b1) and (b2) are unchanged, and, keeping the same notation, the new clauses read:

(a1) for every $i \in \{1, \dots, n\}$, we create the clause $\{x_i, x_i^1, \dots, x_i^{s(i)}\}$: here again, we translate the fact that we look for 1-dominating sets;

(a2) for each pair of vertices v_p and v_q, we consider the set $B_1(v_p)\Delta B_1(v_q) = \{v_{h_1}, v_{h_2}, \dots, v_{h_t}\}$ (where t depends on v_p and v_q) and we construct the clause $\{x_p, x_q, x_{h_1}, x_{h_2}, \ldots, x_{h_t}\}$; we shall say that $\{x_p, x_q\}$ is the first part of the clause, and $\{x_{h_1}, x_{h_2}, \ldots, x_{h_t}\}$ its second part, which exists only when $t > 0$ and may contain variables also appearing in the first part, a fact which is unimportant.

Assume that we have a unique rainbow 1-LD code, V^*; as previously, from V^* we define an assignment $\mathcal{A}_1$. Then the clauses in (a1) are satisfied by $\mathcal{A}_1$, because V^* is 1-dominating. And the clauses in (a2) also are satisfied: if at least one of v_p and v_q is in V^*, then the first part of the clause contains a true literal; if neither v_p nor v_q is a codeword, then at least one vertex in $B_1(v_p)\Delta\ B_1(v_q)$ must be, and the second part of the clause contains a true literal.

The end of the proof is similar to the proof of Proposition 20. △

Proposition 25 *There exists a polynomial transformation from U-3-SAT to U-RLDC$_1$. Moreover, in the connected graph constructed for this transformation, each colour appears at most thrice.*

Proof. Compared to the transformation from U-3-SAT to U-RDC$_1$ (Proposition 21), and after we have duplicated the vertices S_s, $1 \le s \le m$, like we did for going from Proposition 10 to Proposition 16, the differences are:

(i) for each pair of antithetic literals ℓ_p, ℓ_q, we create three vertices $Z_{p,q}$, $Z'_{p,q}$ and $Z''_{p,q}$, and the edges $v_p Z_{p,q}$, $Z_{p,q} Z''_{p,q}$, $Z''_{p,q} Z'_{p,q}$, and $Z'_{p,q} v_q$; the vertices $Z_{p,q}$ and $Z'_{p,q}$ are given the colours $b_{Z,p,q}$ and $b_{Z',p,q}$, respectively, while $Z''_{p,q}$ is unique; then, to deal with $Z_{p,q}$, we create the graph $H'_{Z,p,q}$ with vertex set $\{z'_{Z,i,p,q} : 1 \le i \le 7\}$ represented in Figure 7(b) where, for simplicity, we indicate only the second subscript of the vertices $z'_{Z,i,p,q}$. The vertex $z'_{Z,1,p,q}$ is linked to S_1 and the colours are: $\phi(z'_{Z,1,p,q}) = \phi(z'_{Z,2,p,q}) = b_{Z,p,q}$, $\phi(z'_{Z,3,p,q}) = \phi(z'_{Z,5,p,q}) = \phi(z'_{Z,7,p,q}) = r_{Z,p,q}$ and $\phi(z'_{Z,4,p,q}) = \phi(z'_{Z,6,p,q}) = y_{Z,p,q}$ (for Yellow). Again for simplicity, we drop the subscripts of the colours in Figure 7(b). Similarly, we create the graph $H'_{Z',p,q}$ for $Z'_{p,q}$.

(ii) We give the colour $b_{M,i,i(f)}$ to every middle vertex $M_{i,i(f)} \in W_{i,i(f)}$ and we create the graph $H'_{M,i,i(f)}$ which is identical to the graph $H'_{Z,p,q}$, except that we replace Z by M, p by i and q by $i(f)$, cf. Figure 7(b).

Remark 26 *Assume that V^* is a rainbow 1-LD code. Because $z'_{Z,2,p,q}$ and $z'_{Z,3,p,q}$ have the same neighbours, at least one of them must belong to V^*; if however $z'_{Z,3,p,q} \in V^*$, then $z'_{Z,7,p,q}$ cannot be 1-dominated by V^*. So $z'_{Z,2,p,q} \in V^*$, $z'_{Z,3,p,q} \notin V^*$, and $z'_{Z,1,p,q} \notin V^*$. It is then straightforward to check that necessarily the Yellow and Red codewords are $z'_{Z,4,p,q}$ and $z'_{Z,5,p,q}$, respectively (and $z'_{Z,6,p,q} \notin V^*$, $z'_{Z,7,p,q} \notin V^*$).*

As a consequence, no middle vertex and no vertex $Z_{p,q}$, $Z'_{p,q}$ can belong to any rainbow 1-LD code.

In the constructed graph, each colour appears at most thrice.

As in Case (b) of the proof of Proposition 21 about the consistency of the assignment defined by a rainbow 1-D code, two antithetic causal vertices

v_p and v_q cannot both be codewords; on the other hand, because neither $Z_{p,q}$ nor $Z'_{p,q}$ can belong to any rainbow 1-LD code, one of v_p and v_q is a codeword, so that $Z_{p,q}$ and $Z'_{p,q}$ are 1-separated by a codeword. This extends to all the vertices antithetic to v_p; so, all the clausal vertices corresponding to a variable x belong to a rainbow 1-LD code and none corresponding to $\overline{x}$, or the other way round.

The proof then follows the lines of the proof of Proposition 21. $\triangle$

2.6 Unique Rainbow 1-Identifying Codes

We obtain a better result for unique rainbow 1-ID codes, in the sense that, starting from U-3-SAT, we are able to construct a coloured graph where each colour appears at most twice. But first we go from U-RIDC$_1$ to U-SAT:

Proposition 27 *There exists a polynomial transformation from U-RIDC$_1$ to U-SAT.*

Proof. Same technique as for Propositions 20 and 24. We start from a twin-free graph and define the clauses of type (a) as follows (it relies on the characterization of Remark 3):

(a1) for every $i \in \{1, \dots, n\}$, we create the clause $\{x_i, x_i^1, \dots, x_i^{s(i)}\}$;

(a2) for each pair of vertices v_p and v_q, we consider the set $B_1(v_p)\Delta B_1(v_q) = \{v_{h_1}, v_{h_2}, \dots, v_{h_t}\}$; because G is twin-free, we have $t > 0$. Then we construct the clause $\{x_{h_1}, x_{h_2}, \dots, x_{h_t}\}$, which is simply the second part of the clause defined in the Case (a2) of the proof of Proposition 24.

The end of the proof is similar to the previous two proofs of this type. $\triangle$

Proposition 28 *There exists a polynomial transformation from U-3-SAT to U-RIDC$_1$. Moreover, in the connected graph constructed for this transformation, each colour appears at most twice.*

Proof. With respect to the original construction, we triplicate the vertices S_s, $1 \leq s \leq m+1$, all the artificial vertices, and the final vertex J. For each pair of antithetic literals ℓ_p, ℓ_q, we add two vertices $Z_{p,q}$, $Z'_{p,q}$ and the edges $v_pZ_{p,q}$, $Z_{p,q}v_q$ and $Z_{p,q}Z'_{p,q}$, and set $\phi(Z_{p,q}) = b_{Z,p,q}$; the vertex $Z'_{p,q}$ is unique. We give the colour $b_{M,i,i(f)}$ to every middle vertex $M_{i,i(f)} \in W_{i,i(f)}$. For each middle vertex $M_{i,i(f)}$, we create the unique vertex $Y_{i,i(f)}$, which is linked to $M_{i,i(f)}$.

For every pair of antithetic literals ℓ_p, ℓ_q, we create the graph $H'_{Z,p,q}$ given by Figure 5(b), with the edge $z'_{Z,1,p,q}S_1$, and for every middle vertex $M_{i,i(f)}$, the similar graph $H'_{M,i,i(f)}$ and the edge $z'_{M,1,i,i(f)}S_1$.

We have already observed that $z'_{Z,2,p,q}$ and $z'_{Z,3,p,q}$, not $z'_{Z,1,p,q}$, necessarily belong to any rainbow 1-ID code, and the same is true for $z'_{M,2,i,i(f)}$ and $z'_{M,3,i,i(f)}$, and $z'_{M,1,i,i(f)}$. Then one can see that exactly one of the two antithetic vertices, v_p or v_q, belongs to any rainbow 1-ID code.

The end of the proof is the same as that of Propositions 21 and 25. △

2.7 Generalization to $r > 1$: Dominating Sets

We start with an easy lemma, which is actually common to the three problems.

Lemma 29 *Let $r \geq 2$ be any integer. There is a polynomial transformation from U-RDC$_r$ to U-SAT, from U-RLDC$_r$ to U-SAT, and from U-RIDC$_r$ to U-SAT.*

Proof. Let $G = (V, E)$ be a coloured graph, assumed to be r-twin-free in the case of U-RIDC$_r$, and consider G^r, the r-th power of G. By Remark 2, there is a unique rainbow 1-D code (respectively, 1-LD code, 1-ID code) in G^r if and only if there is a unique rainbow r-D code (respectively, r-LD code, r-ID code) in G. Therefore, we have a first transformation, from U-RDC$_r$ to U-RDC$_1$, from U-RLDC$_r$ to U-RLDC$_1$, or from U-RIDC$_r$ to U-RIDC$_1$. Then we apply Propositions 20, 24 and 27, with their transformations from U-RDC$_1$ to U-SAT, from U-RLDC$_1$ to U-SAT, from U-RIDC$_1$ to U-SAT, and the transitivity of polynomial transformations. △

Then we turn to rainbow r-dominating sets.

Proposition 30 *Let $r \geq 2$ be any integer. There is a polynomial transformation from U-RDC$_1$ to U-RDC$_r$.*

Proof. Let $G = (V, E)$ be any instance of U-RDC$_1$, that is, any coloured graph, and let $e = v_1v_2$ be any edge in G. Let

$$V_e^\S = \{\alpha_{e,i} : 1 \leq i \leq r-1\} \cup \{\beta_{e,i,j} : 1 \leq i \leq r-1, 1 \leq j \leq 3r\},$$

$$\begin{aligned}&E_e^\S = \{v_1\alpha_{e,1}, \alpha_{e,1}\alpha_{e,2}, \ldots, \alpha_{e,r-2}\alpha_{e,r-1}, \alpha_{e,r-1}v_2\} \cup \\ &\{\alpha_{e,i}\beta_{e,i,1}, \beta_{e,i,1}\beta_{e,i,2}, \ldots, \beta_{e,i,r}\beta_{e,i,r+1}, \ldots, \beta_{e,i,2r-1}\beta_{e,i,2r} : 1 \leq i \leq r-1\} \\ &\cup \{\beta_{e,i,r}\beta_{e,i,2r+1}, \beta_{e,i,2r+1}\beta_{e,i,2r+2}, \ldots, \beta_{e,i,3r-1}\beta_{e,i,3r} : 1 \leq i \leq r-1\},\end{aligned}$$

see Figure 8. For $i \in \{1, \ldots, r-1\}$, the vertex $\alpha_{e,i}$ and all the vertices $\beta_{e,i,j}$, $1 \leq j \leq 3r$, (i.e., the branch linked to $\alpha_{e,i}$) receive the colour $b_{i,e}$, which will appear nowhere else. Then we set

$$V^\S = V \cup (\cup_{e \in E} V_e^\S),\ E^\S = \cup_{e \in E} E_e^\S,$$

and $G^\S = (V^\S, E^\S)$ is the instance of U-RDC$_r$ (the colours in V remain the same). Note in particular that any two vertices at distance 1 in G are at distance r in $G^\S$; this is why we shall say that the edge e is *dilated*.

We claim that an instance of U-RDC$_1$ is positive if and only if the corresponding constructed instance of U-RDC$_r$ is.

(a) First, we assume that there is a YES answer for U-RDC$_1$: there is a unique rainbow 1-dominating code V_1^* in G. Let W be the set consisting of

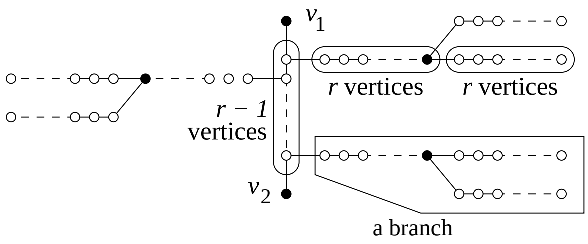

Figure 8: How the edge $e = v_1v_2 \in E$ gives $V_e^{\S}$ and $E_e^{\S}$. The black vertices on the branches are the vertices $\beta_{e,i,r}$.

the $(r-1)|E|$ vertices $\beta_{e,i,r}$, $e \in E$, $1 \leq i \leq r-1$ (they are represented as black vertices in Figure 8). Note that W r-dominates exactly $V^{\S} \setminus V$. Then $V_1^+ = V_1^* \cup W$ is a rainbow r-dominating set in $G^{\S}$. Is V_1^+ unique?

Assume on the contrary that V_2^+ is another rainbow r-D code in $G^{\S}$. Because $\beta_{e,i,r}$ is the only vertex r-dominating the two extremities of its branch, $\beta_{e,i,2r}$ and $\beta_{e,i,3r}$, and for rainbow reasons, we have

$$(V^{\S} \setminus V) \cap V_2^+ = W.$$

Let $V_2^* = V_2^+ \setminus W = V_2^+ \cap V$. Clearly, V_2^* is a rainbow 1-D code in G, different from V_1^*, a contradiction.

(b) Next, we assume that the answer to U-RDC$_1$ is NO: either there is no rainbow 1-dominating code in G, or there is more than one. In the latter case, we have more than one rainbow r-D code in $G^{\S}$: simply add the set W to the codes in G. So we assume that we are in the first case. If there is a rainbow r-D code V^+ in $G^{\S}$, then again $V^+ \setminus W$ would be a rainbow 1-D code in G, a contradiction. In all cases, the answer to U-RDC$_r$ is also NO. $\triangle$

Corollary 31 *Let $r \geq 1$ be any integer. The decision problem RDC$_r$ is NP-complete.*

Proof. We can use Remark 22 here: the above proof of Proposition 30 shows that there is zero, one, or more than one solution to the instance of U-RDC$_1$ if and only if there is zero, one, or more than one solution, respectively, in the constructed coloured graph $G^{\S}$; this proves that we have a polynomial transformation from RDC$_1$ to RDC$_r$, i.e., a proof of *NP*-completeness for RDC$_r$, since RDC$_1$ is *NP*-complete (Propositions 10 or 14) and RDC$_r$ obviously belongs to *NP*. $\triangle$

However, a better result can be obtained.

Proposition 32 *Let $r \geq 1$ be any integer. The decision problem RDC$_r$ is NP-complete, even when restricted to coloured trees.*

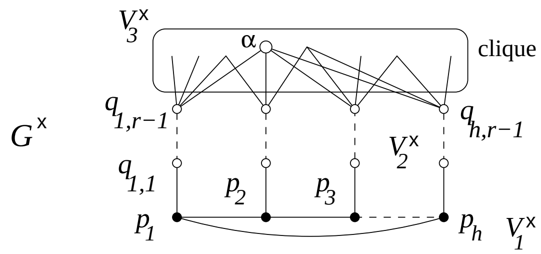

Figure 9: The graph $G^\times$. Black vertices are forced codewords. White vertices are forced non-codewords.

Proof. Apply the construction from the proof of Proposition 30, which yields a tree, to the path constructed in the proof of Proposition 10. $\triangle$

For $r > 1$, we do not have results with a fixed number of occurrences of the colours. We can now conclude for the problem U-RDC$_r$.

Proposition 33 *Let $r \geq 1$ be any integer. The decision problems U-SAT and U-RDC$_r$ have equivalent complexities, up to polynomials.*

Proof. There is a polynomial transformation from U-3-SAT to U-RDC$_1$ (Proposition 21), from U-RDC$_1$ to U-RDC$_r$ (Proposition 30), and from U-RDC$_r$ to U-SAT (Lemma 29). $\triangle$

2.8 Generalization to $r > 1$: Locating-Dominating Sets

In order to dilate the edges, we need, as with the branches and the set W in the case of dominating codes, a graph which has all its codewords (and non-codewords) forced.

Let $r \geq 2$ and $h = 2r + 1$. Let $G_1^\times = (V_1^\times, E_1^\times)$ be the cycle of length h, with $V_1^\times = \{p_i : 1 \leq i \leq h\}$. Then we construct $G_2^\times = (V_2^\times, E_2^\times)$, with $V_2^\times = \{q_{i,j} : 1 \leq i \leq h, 1 \leq j \leq r-1\}$ and $E_2^\times = \cup_{1 \leq i \leq h}\{q_{i,j}q_{i,j+1} : 1 \leq j \leq r-2\}$. The set of edges between $G_1^\times$ and $G_2^\times$ is $E_{1,2}^\times = \{p_i q_{i,1} : 1 \leq i \leq h\}$. Next, we construct $G_3^\times = (V_3^\times, E_3^\times)$ with $V_3^\times = \{s_i : 1 \leq i \leq 2^h - 1 - (r-1)h\}$ and $E_3^\times = \{s_{i_1}s_{i_2} : 1 \leq i_1 < i_2 \leq |V_3^\times|\}$, i.e., $G_3^\times$ is a clique.

We set $V^\times = V_1^\times \cup V_2^\times \cup V_3^\times$, see Figure 9.

In order to define the set $E_{2,3}^\times$ of edges between $\{q_{i,r-1} : 1 \leq i \leq h\}$ and $V_3^\times$, we introduce, for every vertex $v \in V_2^\times \cup V_3^\times$, the *signature* of v as the set $B_r(v) \cap V_1^\times$ of the elements of the cycle that r-dominate v, and we wish to have nonempty and distinct signatures. Since

(a) the h vertices in $V_1^\times$ can provide $2^h - 1$ such signatures;
(b) $|V_2^\times \cup V_3^\times| = |V_2^\times| + |V_3^\times| = 2^h - 1$;
(c) because h is sufficiently large, the vertices $q_{i,j}$ in $V_2^\times$ have nonempty and different signatures (of odd size $2r - 2j + 1 \geq 3$);

(d) a vertex in $V_3^\times$ which is linked (respectively, not linked) to $q_{i,r-1}$ is at distance equal to (respectively, greater than) r from p_i;

we can see that it is possible to construct $E_{2,3}^\times$ in such a way that the vertices in $V_3^\times$ have nonempty signatures which are different inside $V_3^\times$, and different from those for $V_2^\times$. In particular, in $V_3^\times$ there is a vertex which has signature equal to $V_1^\times$; we denote this vertex by α. Note also that we could not have more vertices with this signature property.

We set $E^\times = E_1^\times \cup E_{1,2}^\times \cup E_2^\times \cup E_{2,3}^\times \cup E_3^\times$ and $G^\times = (V^\times, E^\times)$. The order of $G^\times$ is $n^\times = 2^h - 1 + h$. Then we give colours to $G^\times$ in the following way: the vertices in $V_1^\times \setminus \{p_{r+1}\}$ are unique, the other vertices share the Blue colour.

Lemma 34 *The only rainbow r-LD code in $G^\times$ is the cycle $V_1^\times$.*

Proof. Every rainbow r-LD code must contain the $h-1$ unique vertices, plus one Blue vertex. Consider the four vertices w_1, w_2, w_3, w_4 in $V_3^\times$ with signatures $\{p_1\}$, $\{p_2\}$, $\{p_1, p_{r+1}\}$ and $\{p_2, p_{r+1}\}$, respectively. No vertex in $V_3^\times$, no vertex in $V_2^\times$, can r-separate w_1 and w_3, w_2 and w_4, only p_{r+1} can. On the other hand, $V_1^\times$ is a rainbow r-LD code, because now the signatures are simply the sets $I_{G^\times, V_1^\times, r}(v)$, for $v \in V_2^\times \cup V_3^\times = V^\times \setminus V_1^\times$: by construction, they are all nonempty and distinct. $\triangle$

Proposition 35 *Let $r \geq 2$ be any integer. There is a polynomial transformation from U-RLDC$_1$ to U-RLDC$_r$.*

Proof. We start from an instance of U-RLDC$_1$, i.e., a coloured graph $G = (V, E)$ of order n.

For each edge $e = v_1v_2 \in E$ that we want to dilate, we "paste" $r-1$ copies of the graph $G^\times$, by deleting the edge $e = v_1v_2$ and creating the edges $v_1\alpha_1$, $\alpha_1\alpha_2$, …, $\alpha_{r-1}v_2$, where the α_i's are copies of the vertex α in $G^\times$; see Figure 10. We denote by $G^\S = (V^\S, E^\S)$ the graph thus constructed. The colours in $G^\S$ are given as follows: the colours in V are unchanged, the vertices in each copy of $V_1^\times \setminus \{p_{r+1}\}$ are unique, and the other vertices in each copy of $G^\times$ share the colour b_i, i.e., one different colour for each copy.

The order of $G^\S$ is $|V| + |E|(r-1)(2^h + h - 1)$. Since r, hence $h = 2r+1$, is fixed, this does not affect the polynomiality of our construction with respect to n, the order of G.

We claim that there is a unique rainbow 1-LD code in G if and only if there is a unique rainbow r-LD code in $G^\S$.

(a) Assume first that there is a unique rainbow 1-LD code V_1^* in G. We construct the following code V_1^+ in $G^\S$: we add to V_1^* the set W of all the vertices in all the cycles $G_1^\times$ in all the copies of $G^\times$. Note that these vertices in W do not r-dominate any vertex in V. Obviously, V_1^+ is a rainbow r-LD code in $G^\S$. Is it unique?

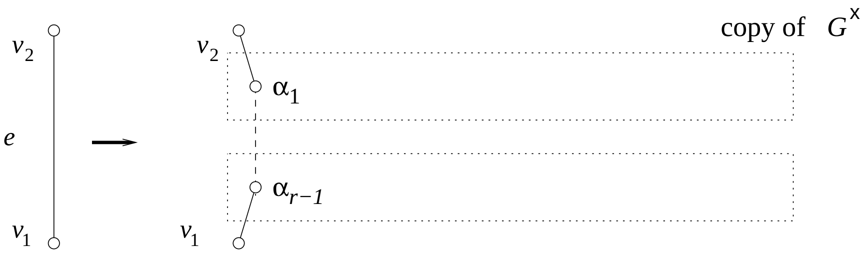

Figure 10: How the edge $e = v_1v_2 \in E$ is dilated for Proposition 35.

The argument is similar to the one for dominating codes: assume on the contrary that V_2^+ is another rainbow r-LD code in $G^\S$. The intersection of $V^\S \setminus V$ with V_2^+ is equal to W, because, since the vertices in V do not r-separate between any vertices in any copy of the clique $G_3^\times$, except possibly the vertex α, we can still apply the argument of Lemma 34, and it is still true that, in addition to the unique vertices, we must take as codewords all the copies p_{r+1}^i, $1 \leq i \leq |E|(r-1)$, of the vertex p_{r+1}. Let $V_2^* = V_2^+ \setminus W = V_2^+ \cap V$. Clearly, V_2^* is a rainbow 1-LD code in G, different from V_1^*, a contradiction.

(b) Next, we assume that the answer to U-RLDC$_1$ is NO: either there is no rainbow 1-LD code in G, or there is more than one. In the latter case, we have more than one rainbow r-LD code in $G^\S$: simply add the set W to the codes in G. So we assume that we are in the first case. But if there is a rainbow r-LD code V^+ in $G^\S$, then again $V^+ \setminus W$ would be a rainbow 1-LD code in G, a contradiction. In both cases, the answer to U-RLDC$_r$ is also NO. $\triangle$

The following consequences are immediate.

Corollary 36 *Let $r \geq 1$ be any integer. The decision problem RLDC$_r$ is NP-complete.*

Proposition 37 *Let $r \geq 1$ be any integer. The decision problems U-SAT and U-RLDC$_r$ have equivalent complexities, up to polynomials.*

2.9 Generalization to $r > 1$: Identifying Sets

Lemma 38 *Let $r \geq 1$ be any integer. Let $G = (V, E)$ be the (non coloured) path $\beta_1\beta_2 \ldots \beta_{3r+1}$. Then $V^* = \{\beta_{r+1}, \ldots, \beta_{2r}\}$ is included in any r-identifying code.*

Proof. Apply Lemma 4 in [11] with $n = 3r + 1$. Note that β_{2r+1} also belongs to any r-identifying code, but we do not need it for our purpose. $\triangle$

We colour the previous path in the following way: for $i \in \{1, \ldots, r\}$, $\phi(\beta_i) = \phi(\beta_{i+r}) = c_i$; the vertices β_i, $2r + 1 \leq i \leq 3r + 1$, are unique.

Lemma 39 *The coloured path defined above admits $V^+ = \{\beta_i : r+1 \le i \le 3r+1\}$ as its unique rainbow r-identifying code.*

Proof. A rainbow r-ID code must contain V^* and the unique vertices. The vertices β_i, $1 \le i \le r$, cannot be codewords for rainbow reasons. It is easy to check that V^+ is indeed a rainbow r-ID code, and is the only one. $\triangle$

Proposition 40 *Let $r \ge 2$ be any integer. There is a polynomial transformation from U-RIDC$_1$ to U-RIDC$_r$.*

Proof. Let $G = (V, E)$ be any instance of U-RIDC$_1$, that is, any coloured graph, and let $e = v_1v_2$ be any edge in G. Let

$$V_e^{\S} = \{\beta_{e,i,j} : 1 \le i \le r-1, 1 \le j \le 3r+1\},$$

$$E_e^{\S} = \{v_1\beta_{e,1,1}, \beta_{e,1,1}\beta_{e,2,1}, \dots, \beta_{e,r-2,1}\beta_{e,r-1,1}, \beta_{e,r-1,1}v_2\} \cup \{\beta_{e,i,1}\beta_{e,i,2}, \beta_{e,i,2}\beta_{e,i,3}, \dots, \beta_{e,i,3r}\beta_{e,i,3r+1} : 1 \le i \le r-1\},$$

i.e., in order to dilate all the edges, we have split every edge e with $r-1$ vertices $\beta_{e,i,1}$ which are the starting points of copies of the path defined for Lemma 38. Then we set

$$V^{\S} = V \cup (\cup_{e \in E} V_e^{\S}),\ E^{\S} = \cup_{e \in E} E_e^{\S},$$

and $G^{\S} = (V^{\S}, E^{\S})$. The set V keeps its colours unchanged. Each path is coloured as was done for Lemma 39, with its specific colours that will be nowhere else. Note that the set $\{\beta_{e,i,j} : r+1 \le j \le 3r+1\}$ r-dominates exactly the set $\{\beta_{e,i,j} : 1 \le j \le 3r+1\}$, for all e and i.

We claim that the instance of U-RIDC$_1$ we started from, is positive if and only if the coloured graph $G^{\S}$ admits a unique rainbow r-ID code.

(a) First, we assume that there is a unique rainbow 1-ID code V_1^* in G. Let W be the set consisting of the $|E|(2r+1)(r-1)$ vertices $\beta_{e,i,j}$, $e \in E$, $1 \le i \le r-1$, $r+1 \le j \le 3r+1$. Note that W r-dominates exactly $V^{\S} \setminus V$. Then $V_1^+ = V_1^* \cup W$ is a rainbow r-identifying set in $G^{\S}$. Is V_1^+ unique?

Assume on the contrary that V_2^+ is another such code in $G^{\S}$. We have $(V^{\S} \setminus V) \cap V_2^+ = W$. Let $V_2^* = V_2^+ \setminus W = V_2^+ \cap V$. Clearly, V_2^* is a rainbow 1-ID code in G, different from V_1^*, a contradiction.

(b) Next, we assume that the answer to U-RIDC$_1$ is NO: either there is no rainbow 1-ID code in G, or there is more than one. In the latter case, we have more than one rainbow r-ID code in $G^{\S}$: simply add the set W to the codes in G. So we assume that we are in the first case. If there is a rainbow r-ID code V^+ in $G^{\S}$, then again $V^+ \setminus W$ would be a rainbow 1-ID code in G, a contradiction. In both cases, the answer to U-RIDC$_r$ is NO. $\triangle$

As previously, we have the following easy consequences.

Corollary 41 *Let $r \ge 1$ be any integer. The decision problem RIDC$_r$ is NP-complete.*

Proposition 42 *Let $r \geq 1$ be any integer. The decision problem RIDC$_r$ is NP-complete, even when restricted to coloured trees.*

Proposition 43 *Let $r \geq 1$ be any integer. The decision problems U-SAT and U-RIDC$_r$ have equivalent complexities, up to polynomials.*

3 Conclusion

We recapitulate all the above results and summarize them in Table 1 below.

(i) The decision problem RDC$_1$ was known to be *NP*-complete, even for paths [1], [2]. We proved that it is *NP*-complete, even when restricted to trees where each colour appears at most twice (Proposition 14). For all $r \geq 2$, we proved that the decision problem RDC$_r$ is *NP*-complete, even for trees (Proposition 32).

We proved that the decision problems U-RDC$_1$ and U-SAT have equivalent complexities, up to polynomials, and settled the case when each colour appears at most thrice (Propositions 20 and 21). For all $r \geq 2$, we proved that the decision problems U-RDC$_r$ and U-SAT have equivalent complexities, up to polynomials (Proposition 33). Using Corollary 6, we have the following result.

Proposition 44 *For all $r \geq 1$, the decision problem U-RDC$_r$ is NP-hard and belongs to the class DP.*

(ii) We proved that the decision problem RLDC$_1$ is *NP*-complete, even for paths (Proposition 16), and when restricted to trees where each colour appears at most twice (Proposition 17). For all $r \geq 2$, we proved that the decision problem RLDC$_r$ is *NP*-complete (Corollary 36).

We proved that the decision problems U-RLDC$_1$ and U-SAT have equivalent complexities, up to polynomials, and settled the case when each colour appears at most thrice (Propositions 24 and 25). For all $r \geq 2$, we proved that the decision problems U-RLDC$_r$ and U-SAT have equivalent complexities, up to polynomials (Proposition 37). Using Corollary 6, we have the following result.

Proposition 45 *For all $r \geq 1$, the decision problem U-RLDC$_r$ is NP-hard and belongs to the class DP.*

(iii) We proved that the decision problem RIDC$_1$ is *NP*-complete, even when restricted to trees where each colour appears at most twice (Proposition 19). For all $r \geq 2$, we proved that the decision problem RIDC$_r$ is *NP*-complete, even for trees (Proposition 42).

We proved that the decision problems U-RIDC$_1$ and U-SAT have equivalent complexities, up to polynomials, and settled the case when each colour appears at most twice (Propositions 27 and 28). For all $r \geq 2$, we proved that the decision problems U-RIDC$_r$ and U-SAT have equivalent complexities, up to polynomials (Proposition 43). Using Corollary 6, we have the following result.

Proposition 46 *For all $r \geq 1$, the decision problem U-RIDC$_r$ is NP-hard and belongs to the class DP.*

In Remark 7, it is said that "U-SAT is not believed to be *DP*-complete", so the same can be said for U-RDC$_r$, U-RLDC$_r$, and U-RIDC$_r$.

	r-D codes		r-LD codes		r-ID codes	
NP-complete, $r = 1$	path [1],[2] tree	? ≤ 2	path tree	? ≤ 2	tree	≤ 2
NP-complete, $r > 1$	tree	?	graph	?	tree	?
uniqueness: $\approx$ U-SAT, $r = 1$	graph	≤ 3	graph	≤ 3	graph	≤ 2
uniqueness: $\approx$ U-SAT, $r > 1$	graph	?	graph	?	graph	?

Table 1: The 2$^{\text{nd}}$, 4$^{\text{th}}$, and 6$^{\text{th}}$ columns give the structure of the graph, the 3$^{\text{rd}}$, 5$^{\text{th}}$, and 7$^{\text{th}}$ columns give the maximum number of colour occurrences.

References

[1] J.-A. ANGLÈS d'AURIAC, Cs. BUJTÁS, A. El MAFTOUHI, M. KARPINSKI, Y. MANOUSSAKIS, L. MONTERO, N. NARAYANAN, L. ROSAZ, J. THAPPER and ZS. TUZA: Tropical dominating sets in vertex-coloured graphs, *Proceedings of 10th International Workshop on Algorithms and Computation*, Kathmandu, Nepal, pp. 17–27, 2016.

[2] J.-A. ANGLÈS d'AURIAC, Cs. BUJTÁS, A. El MAFTOUHI, M. KARPINSKI, Y. MANOUSSAKIS, L. MONTERO, N. NARAYANAN, L. ROSAZ, J. THAPPER and ZS. TUZA: Tropical dominating sets in vertex-coloured graphs, *Journal of Discrete Algorithms*, Vol. 48, pp. 27–41, 2018.

[3] J.-A. ANGLÈS d'AURIAC, N. COHEN, A. El MAFTOUHI, A. HARUTYUNYAN, S. LEGAY, and Y. MANOUSSAKIS: Connected tropical subgraphs in vertex-colored graphs, *Discrete Mathematics & Theoretical Computer Science*, Vol. 17(3), P21, 2016.

[4] J.-P. BARTHÉLEMY, G. D. COHEN and A. C. LOBSTEIN: *Algorithmic Complexity and Communication Problems*, London: University College of London, 1996.

[5] A. BLASS and Y. GUREVICH: On the unique satisfiability problem, *Information and Control*, Vol. 55, pp. 80-88, 1982.

[6] M. BLIDIA, M. CHELLALI, R. LOUNES and F. MAFFRAY: Characterizations of trees with unique minimum locating-dominating sets, *Journal of Combinatorial Mathematics and Combinatorial Computing*, Vol. 76, pp. 225–232, 2011.

[7] B. BREŠAN, M. A. HENNING and D. F. RALL: Rainbow domination in graphs, *Taiwanese Journal of Mathematics*, Vol. 12(1), pp. 213–225, 2008.

[8] C. CALABRO, R. IMPAGLIAZZO, V. KABANETS and R. PATURI: The complexity of Unique k-SAT: an isolation lemma for k-CNFs, *Journal of Computer and System Sciences*, Vol. 74, pp. 386–393, 2008.

[9] V. T. CHAKARAVARTHY, M. KAPRALOV, P. MURALI, F. PETRINI, X. QUE, Y. SABHARWALI and B. SCHIEBER: Subgraph counting: Color coding beyond trees, arXiv:1602.04478 (2018).

[10] I. CHARON, O. HUDRY and A. LOBSTEIN: Minimizing the size of an identifying or locating-dominating code in a graph is *NP*-hard, *Theoretical Computer Science*, Vol. 290, pp. 2109–2120, 2003.

[11] C. CHEN, C. LU and Z. MIAO: Identifying codes and locating-dominating sets on paths and cycles, *Discrete Applied Mathematics*, Vol. 159(15), pp. 1540–1547, 2011.

[12] G. COHEN, I. HONKALA, A. LOBSTEIN and G. ZÉMOR: On identifying codes, *Proceedings of DIMACS Workshop on Codes and Association Schemes '99*, Piscataway, USA, Vol. 56, pp. 97–109, 2001.

[13] C. J. COLBOURN, P. J. SLATER and L. K. STEWART: Locating dominating sets in series parallel networks, *Congressus Numerantium*, Vol. 56, pp. 135–162, 1987.

[14] S. A. COOK: The complexity of theorem-proving procedures, *Proceedings of 3rd Annual ACM Symposium on Theory of Computing*, pp. 151–158, 1971.

[15] M. FISCHERMANN: Block graphs with unique minimum dominating sets, *Discrete Mathematics*, Vol. 240, pp. 247–251, 2001.

[16] M. R. GAREY and D. S. JOHNSON: *Computers and Intractability, a Guide to the Theory of NP-Completeness*, New York: Freeman, 1979.

[17] G. GUNTHER, B. HARTNELL, L. R. MARKUS and D. RALL: Graphs with unique minimum dominating sets, *Congressus Numerantium*, Vol. 101, pp. 55–63, 1994.

[18] T. W. HAYNES, S. T. HEDETNIEMI and P. J. SLATER: *Fundamentals of Domination in Graphs*, New York: Marcel Dekker, 1998.

[19] O. HUDRY and A. LOBSTEIN: More results on the complexity of domination problems in graphs, *International Journal of Information and Coding Theory*, Vol. 4, pp. 129–144, 2017.

[20] O. HUDRY and A. LOBSTEIN: Unique (optimal) solutions: complexity results for identifying and locating-dominating codes, *Theoretical Computer Science*, Vol. 767, pp. 83–102, 2019.

[21] O. HUDRY and A. LOBSTEIN: Complexity of unique (optimal) solutions in graphs: Vertex Cover and Domination, *Journal of Combinatorial Mathematics and Combinatorial Computing*, Vol. 110, pp. 217–240, 2019.

[22] O. HUDRY and A. LOBSTEIN: Some complexity considerations on the uniqueness of solutions for satisfiability and colouring problems, submitted.

[23] D. S. JOHNSON: A catalog of complexity classes, in: *Handbook of Theoretical Computer Science, Vol. A: Algorithms and Complexity*, van Leeuwen, Ed., Chapter 2, Elsevier, 1990.

[24] A. LOBSTEIN: A bibliography on watching systems, identifying, locating-dominating and discriminating codes in graphs, https://www.lri.fr/˜lobstein/debutBIBidetlocdom.pdf

[25] C. H. PAPADIMITRIOU: On the complexity of unique solutions, *Journal of the Association for Computing Machinery*, Vol. 31, pp. 392-400, 1984.

[26] C. H. PAPADIMITRIOU: *Computational Complexity*, Reading: Addison-Wesley, 1994.

[27] C. H. PAPADIMITRIOU and M. YANNAKAKIS: The complexity of facets (and some facets of complexity), *Journal of Computer and System Sciences*, Vol. 28, pp. 244–259, 1984.

[28] https://complexityzoo.uwaterloo.ca/Complexity_Zoo