Exact Algorithms for Weak Roman Domination*

Mathieu Chapelle[1], Manfred Cochefert[2], Jean-François Couturier[2], Dieter Kratsch[2], Mathieu Liedloff[3], and Anthony Perez[3]

[1] LIGM, Université Paris-Est Marne-La-Vallée
77454 Marne-La-Vallée Cedex 2, France
mathieu.chapelle@univ-mlv.fr
[2] LITA, Université de Lorraine
57045 Metz Cedex 01, France
{couturier,cochefert,kratsch}@univ-metz.fr
[3] LIFO, Université d'Orléans
45067 Orléans Cedex 2, France
{mathieu.liedloff,anthony.perez}@univ-orleans.fr

Abstract. We consider the Weak Roman Domination problem. Given an undirected graph $G = (V, E)$, the aim is to find a *weak roman domination* function (wrd-function for short) of minimum cost, *i.e.* a function $f : V \to \{0, 1, 2\}$ such that every vertex $v \in V$ is *defended* (*i.e.* there exists a neighbor u of v, possibly $u = v$, such that $f(u) \geqslant 1$) and for every vertex $v \in V$ with $f(v) = 0$ there exists a neighbor u of v such that $f(u) \geqslant 1$ and the function $f_{u\to v}$ defined by:

$$f_{u\to v}(x) = \begin{cases} 1 & \text{if } x = v \\ f(u) - 1 & \text{if } x = u \\ f(x) & \text{if } x \notin \{u, v\} \end{cases}$$

does not contain any undefended vertex. The *cost* of a wrd-function f is defined by $cost(f) = \sum_{v\in V} f(v)$. The trivial enumeration algorithm runs in time $\mathcal{O}^*(3^n)$ and polynomial space and is the best one known for the problem so far. We are breaking the trivial enumeration barrier by providing two faster algorithms: we first prove that the problem can be solved in $\mathcal{O}^*(2^n)$ time needing *exponential space*, and then describe an $\mathcal{O}^*(2.2279^n)$ algorithm using *polynomial space*. Our results rely on structural properties of a wrd-function, as well as on the best polynomial space algorithm for the Red-Blue Dominating Set problem.

Keywords: exact algorithm, graph algorithm, roman domination.

1 Introduction

In this paper we investigate a domination-like problem from the exact exponential algorithms viewpoint. In the classical Dominating Set problem, one is given an undirected graph $G = (V, E)$, and asked to find a dominating set S, i.e. every vertex $v \in V$ either belongs to S or has a neighbor in S, of minimum

* This work was supported by the French AGAPE project (ANR-09-BLAN-0159).

T. Lecroq and L. Mouchard (Eds.): IWOCA 2013, LNCS 8288, pp. 81–93, 2013.

size. The DOMINATING SET problem ranges among one of the most famous NP-complete covering problems [8], and has received a lot of attention during the last decades. In particular, the trivial enumeration algorithm of runtime $\mathcal{O}^*(2^n)$ [1] has been improved by a sequence of papers [7,14,23]. The currently best known algorithms for the problem run in time $\mathcal{O}^*(1.4864^n)$ using polynomial space, and in time $\mathcal{O}^*(1.4689^n)$ needing exponential space [14].

Many variants of the DOMINATING SET problem have been introduced and studied extensively both from structural and algorithmic viewpoints. The number of papers on domination in graphs and its variants is in the thousands, and several well-known surveys and books are dedicated to the topic (see, *e.g.*, [12]). One of those variants called ROMAN DOMINATION was introduced in [5] and motivated by the articles "Defend the Roman Empire!" of I. Stewart [21] and "Defendens Imperium Romanum: a classical problem in military strategy" of C.S. ReVelle and K.E. Rosing [20]. In general, the aim is to protect a set of locations (vertices of a graph) by using a smallest possible amount of legions (to be placed on those vertices). Motivated by a decree of the Emperor Constantine the Great in the fourth century A.D., ROMAN DOMINATION uses the following rules for protecting a graph: a vertex can *protect* itself if it has one legion, and protect all its neighbors if it owns two legions, since Constantine decreed that two legions must be placed at a location before one may move to a nearby location (adjacent vertex) to defend it. The ROMAN DOMINATION problem asks to minimize the number of legions used to defend all vertices.

Since then, numerous articles have been published around this problem, which has been studied from different viewpoints (see, *e.g.*, [1,2,4,6,17,18,24]). In particular, this NP-complete problem has been tackled using exact exponential algorithms. The first non-trivial one achieved had running time $\mathcal{O}^*(1.6183^n)$ and used polynomial space [15]. This result has recently been improved to $\mathcal{O}^*(1.5673^n)$ [22], which can be lowered to $\mathcal{O}^*(1.5014^n)$ at the cost of exponential space [22]. Moreover, the ROMAN DOMINATION problem can be related to several other variants of *defense-like* domination, such as *secure domination* (see, *e.g.*, [3,4,11]), or *eternal domination* (see, *e.g.*, [9,10]).

We focus our attention on yet another variant of the ROMAN DOMINATION problem. In 2003, Henning et al. [13] considered the following idea: location t can also be protected if one of its neighbors possesses one legion that can be moved to t in such a way that the whole collection of locations (set of vertices) remains protected. This variation adds some kind of dynamics to the problem and gives rise to the WEAK ROMAN DOMINATION problem. Formally, it can be defined as follows:

WEAK ROMAN DOMINATION:
Input: An undirected graph $G = (V, E)$.
Output: A weak roman domination function f of G of minimum cost.

A weak roman domination (wrd-function) is a function $f : V \to \{0, 1, 2\}$ such that every vertex $v \in V$ is *defended* (*i.e.* there exists a neighbor u of v, possibly

[1] The notation $\mathcal{O}^*(f(n))$ suppresses polynomial factors.

$u = v$, such that $f(u) \geqslant 1$) and for every vertex $v \in V$ with $f(v) = 0$ there exists a neighbor u of v such that $f(u) \geqslant 1$ and the function $f_{u \to v}$ defined by $f_{u \to v}(x) = 1$ if $x = v$, $f_{u \to v}(x) = f(x) - 1$ if $x = u$ and $f_{u \to v}(x) = f(x)$ otherwise does not contain any undefended vertex. The *cost* of a wrd-function f is defined by $cost(f) = \sum_{v \in V} f(v)$.

Our Contribution. While several structural results on Weak Roman Domination are known, see, *e.g.*, [3,4,13,19], its algorithmic aspects have not been considered so far. In this paper, we give the first algorithms tackling this problem faster than by the $\mathcal{O}^*(3^n)$ bruteforce algorithm obtained by enumerating all legion functions. Both our algorithms rely on structural properties for weak roman domination functions, described in Section 3. In Section 4, we first give an $\mathcal{O}^*(2^n)$ time and exponential space algorithm. We then show how the exponential space can be avoided by using an exponential algorithm for the Red-Blue Dominating Set problem [22], which leads to an $\mathcal{O}^*(2.2279^n)$ algorithm.

2 Preliminaries and Notations

We consider simple undirected graphs $G = (V, E)$ and assume that $n = |V|$. Given a vertex $v \in V$, we denote by $N(v)$ its *open neighborhood*, by $N[v]$ its *closed neighborhood* (*i.e.* $N[v] = N(v) \cup \{v\}$). For $X \subseteq V$, let $N[X] = \cup_{v \in X} N[v]$ and $N(X) = N[X] \setminus X$. Similarly, given $S \subseteq V$, we use $N_S(v)$ to denote the set $N(v) \cap S$. A subset of vertices $S \subseteq V$ is a *dominating set* of G if for every vertex $v \in V$ either $v \in S$ or $N_S(v) \neq \emptyset$. Furthermore, $Y \subseteq V$ dominates $X \subseteq V$ in $G = (V, E)$ if $X \subseteq N[Y]$. A subset of vertices $S' \subseteq V$ is an *independent set* in G if there is no edge in G between any pair of vertices in S'. Finally, a graph $G = (V, E)$ is *bipartite* whenever its vertex set can be partitioned into two independent sets V_1 and V_2.

Legion and wrd-Functions. A function $f : V \to \{0, 1, 2\}$ is called a *legion function.* With respect to f, a vertex $v \in V$ is said to be *secured* if $f(v) \geqslant 1$, and *unsecured* otherwise. Similarly, a vertex $v \in V$ is said to be *defended* if there exists $u \in N[v]$ such that $f(u) \geqslant 1$. Otherwise, v is said to be *undefended.* The function f is a *weak roman domination function* (wrd-function for short) if there is no undefended vertex with respect to f, and for every vertex $v \in V$ with $f(v) = 0$ there exists a secured vertex $u \in N(v)$ such that the function $f' : V \to \{0, 1, 2\}$ defined by:

$$f'(x) = \begin{cases} 1 & \text{if } x = v \\ f(u) - 1 & \text{if } x = u \\ f(x) & \text{if } x \notin \{u, v\} \end{cases}$$

has no undefended vertex (see Figure 1 (a)). In the following, given any legion function f and two vertices v and $u \in N(v)$ such that $f(v) = 0$ and $f(u) \geqslant 1$, we use $f_{u\to v}$ to denote the function f' as defined above. In other words, $f_{u\to v}$ denotes the legion function obtained by *moving* one legion from u to v.

Given a legion function f, we let V_f^1, V_f^2 denote the sets $\{v \in V \ : \ f(v) = 1\}$ and $\{v \in V \ : \ f(v) = 2\}$, respectively, and define its *underlying set* as $V_f = V_f^1 \cup V_f^2$. The *cost* of f is then defined by $cost(f) = \sum_{v\in V} f(v) = |V_f^1| + 2|V_f^2|$. Notice that when f is a wrd-function, the set V_f is a (not necessarily minimal) dominating set of G.

Safely-Defended Vertices. We now distinguish two types of *defended* vertices. Let $v \in V$ be any vertex and f be a legion function. We say that v is *safely defended by* f if one of the following holds:

- v is secured (i.e. $f(v) \geqslant 1$).
- there exists a neighbor u of v such that $f(u) = 2$.
- there exists a neighbor u of v such that $f(u) = 1$ and the vertices undefended by $f_{u\to v}$ are the same as the ones undefended by f, *i.e.*, $f_{u\to v}$ creates no new undefended vertex.

Otherwise, we say that v is *non-safely defended.* Notice that a legion function f is a wrd-function if and only if every vertex $v \in V$ is safely-defended by f.

Observe that for any non-safely defended vertex v, we have $f(v) = 0$, $f(u) = 1$ for every secured neighbor u of v and the legion function $f_{u\to v}$ previously defined contains (among possibly others) an undefended vertex $w \in N(u)$ for any such neighbor u. In the following, we will refer to w as *weakly defended by* u, *weakly defended due to* v, or simply *weakly defended* when the context is clear. Observe that a weakly defended vertex has exactly one secured neighbor. These notions are illustrated in Figure 1 (b).

3 Structure of a Weak Roman Domination Function

In this section, we prove several key structural properties of a wrd-function that will be used in our algorithms.

Given a graph $G = (V, E)$ and a subset of vertices $V' \subseteq V$, we define the legion function $\chi^{V'}$ as the indicator function of the subset V':

$$\chi^{V'}(x) = \begin{cases} 1 \text{ if } x \in V' \\ 0 \text{ otherwise.} \end{cases}$$

Lemma 1. *Let $G = (V, E)$ be a graph, f be a wrd-function of G of minimum cost, and V_f its underlying set. Then V_f^2 is a* minimum dominating set *of the vertices non-safely defended by χ^{V_f}.*

Proof. Let $u \in V \setminus V_f$ be a vertex non-safely defended by χ^{V_f}. Recall that u is non-safely defended by χ^{V_f} if for every $u' \in N_{V_f}(u)$ the legion function $\chi^{V_f}_{u'\to u}$ contains an undefended vertex. Hence, for every vertex $u' \in N_{V_f}(u)$, there exists a vertex u'' weakly defended due to u. In particular, this means that $u'u'' \in E$ and $uu'' \notin E$. We prove Lemma 1 through the following claims.

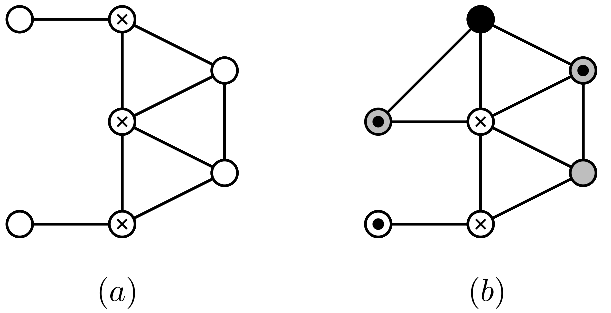

Fig. 1. (a) A graph $G = (V, E)$, and a wrd-function where each legion is represented by a cross. Any vertex is safely defended. (b) The black vertex is safely defended (one can safely move a legion on it without creating any undefended vertex), the gray vertices are non-safely defended (any move creates an undefended vertex) and the disked vertices are weakly defended.

Claim 1. *V_f^2 is a dominating set of the vertices non-safely defended by χ^{V_f}.*

Proof. Assume for a contradiction that there exists a vertex $u \in V \setminus V_f$ non-safely defended by χ^{V_f} such that $N_{V_f^2}(u) = \emptyset$. Let u'' be any vertex weakly defended due to u, and let u' be the common neighbor of u and u'' in V_f. Recall that $N(u'') \cap V_f = \{u'\}$, since otherwise u'' would be defended by $\chi^{V_f}_{u' \to u}$. Moreover, we know by assumption that $f(u') = 1$. Hence, the vertex u'' is undefended by $\chi^{V_f}_{u' \to u}$, which contradicts the fact that f is a wrd-function. ⋄

Claim 2. *V_f^2 is a minimal dominating set of the vertices non-safely defended by χ^{V_f}.*

Proof. Assume for a contradiction that there exists $u \in V_f^2$ such that $V_f^2 \setminus \{u\}$ is a dominating set of the vertices non-safely defended by χ^{V_f}. We claim that the legion function f_u defined as:

$$f_u(x) = \begin{cases} 1 \text{ if } x = u \\ f(v) \text{ otherwise} \end{cases}$$

is a wrd-function. To see this, observe that since $V_f^2 \setminus \{u\}$ is a dominating set of the vertices non-safely defended by χ^{V_f}, any vertex of $N_{V \setminus V_f}(u)$ is safely defended by f_u. It follows that f_u is a wrd-function with $cost(f_u) < cost(f)$, a contradiction. ⋄

Now, since f is a wrd-function of *minimum cost*, it follows from Claims 1 and 2 that V_f^2 is a *minimum* dominating set of the vertices non-safely defended by χ^{V_f}. This completes the proof of Lemma 1. □

We conclude this section by showing that, given a dominating set V' of a graph $G = (V, E)$, a wrd-function can be obtained by computing a dominating set of the set $\overline{D}$ of all vertices non-safely defended by $\chi^{V'}$.

Lemma 2. *Let $V' \subseteq V$ be a dominating set of a graph $G = (V, E)$, and let S be a dominating set of all vertices $\overline{D}$ non-safely defended by $\chi^{V'}$. Then the function $f : V \to \{0, 1, 2\}$ defined by*

$$f(x) = \begin{cases} 2 \text{ if } x \in (V' \cap S) \\ 1 \text{ if } x \in (V' \cup S) \setminus (V' \cap S) \\ 0 \text{ otherwise} \end{cases}$$

is a wrd-function.

Proof. Let S be a dominating set of $\overline{D}$ in G. Observe first that since $V_f = V' \cup S$, and since V' is a dominating set, then so is V_f. We now show that the set $\overline{D}'$ of vertices non-safely defended by f is empty. Observe that since $V' \subseteq V_f$, we have $\overline{D}' \subseteq \overline{D} \setminus S$. Assume for a contradiction that $\overline{D}' \neq \emptyset$, and let $x \in \overline{D}'$. We distinguish two cases:

(i) If $N(x) \cap (V' \cap S) \neq \emptyset$ then x has a neighbor of f-value 2, and thus x is safely-defended, contradicting the choice of x.
(ii) Otherwise, by definition of V' and S, x has a neighbor y in S which does not belong to V'. We claim that the legion function $f_{y \to x}$ cannot contain any undefended vertex. Indeed, since y does not belong to the original dominating set V', all vertices are defended by V' in $f_{y \to x}$ (recall that any vertex v of V' satisfies $f(v) \geqslant 1$).

These two cases imply that $\overline{D}'$ is empty, and thus f is a wrd-function. □

4 Exact Algorithms for Weak Roman Domination

We now describe our exact algorithms solving the Weak Roman Domination problem. Observe that this problem can trivially be solved in $\mathcal{O}^*(3^n)$ time by enumerating all three-partitions of the set of vertices, which constitutes the best known bound for the problem so far. We first present an $\mathcal{O}^*(2^n)$ time and space algorithm, then an $\mathcal{O}^*(2.2279^n)$ time algorithm that only uses polynomial space.

4.1 Using Exponential Space

We first show that a wrd-function of minimum cost can be computed in $\mathcal{O}^*(2^n)$ time and space. Thanks to Lemma 1, a wrd-function f of minimum cost can be obtained by first guessing its underlying set V_f and then computing a minimum dominating set $V_f^2 \subseteq V_f$ of the vertices non-safely defended by χ^{V_f}. Finding such a set V_f^2 is done by a preprocessing step which involves a dynamic programming

Algorithm 1. The preprocessing step algorithm.

```
for k = 0 to n do
    DS[∅, k] = ∅;
foreach X ⊆ V s.t. |X| ≥ 1 do
    DS[X, 0] = {∞};
    // The set {∞} is a sentinel used to denote the non existence of
       a set Y_k which dominates a nonempty set X; its cardinality is
       set to ∞.
foreach X ⊆ V by increasing order of cardinality do
    for k = 1 to n do
        DS[X, k] = { a set of minimum cardinality chosen amongst
                     DS[X, k − 1] and {v_k} ∪ DS[X \ N[v_k], k − 1]. }
```

inspired by the one given in [16]. This preprocessing step results in an exponential space complexity, which will be reduced to polynomial space in Section 4.2. However, instead of guaranteeing that indeed $V_f^2 \subseteq V_f$, the preprocessing step computes a minimum dominating set V_f^2 of the vertices non-safely defended by χ^{V_f} without constraint, *i.e.* $V_f^2 \subseteq V$ is allowed. We show in Lemma 3 the correctness of this approach. Let us first describe the preprocessing step; its correctness is shown after the description of the main algorithm.

Let $G = (V, E)$ be a graph of the Weak Roman Domination problem, and let $V = \{v_1, v_2, \ldots, v_n\}$. For each subset $X \subseteq V$ we start by computing a minimum dominating set Y of X in G, i.e. a subset $Y \subseteq V$ such that $X \subseteq N[Y]$. This is done by dynamic programming: for each subset X and each integer k ($1 \leqslant k \leqslant n$), $\mathsf{DS}[X, k]$ denotes a minimum dominating set Y_k of X such that $Y_k \subseteq \{v_1, v_2, \ldots, v_k\}$, if one exists. Algorithm 1 computes a corresponding table DS by dynamic programming.

Main Algorithm. The main steps of our exact algorithm are depicted in Algorithm 2. For each subset $V' \subseteq V$, we first verify whether $\chi^{V'}$ is (already) a wrd-function, *i.e.*, whether the set $\overline{D}$ of vertices non-safely defended by $\chi^{V'}$ is empty. Otherwise, we need to compute the set V_f^2. The preprocessing step then ensures that $S = \mathsf{DS}[\overline{D}, n]$ is a minimum dominating set of $\overline{D}$. If S is a subset of V', then a wrd-function f can be computed by Lemma 2; otherwise Lemma 3 ensures that there exists some other underlying set V'', being better than V'.

Lemma 3. *Let $V_1' \subseteq V$ be a dominating set of a graph $G = (V, E)$ and let S_1 be a minimum dominating set of the set $\overline{D_1}$ of all vertices non-safely defended by $\chi^{V_1'}$. Suppose that $S_1 \nsubseteq V_1'$. Then there exists a superset $V_2' \supset V_1'$ such that for any minimum dominating set S_2 of the set $\overline{D_2}$ of all vertices non-safely*

Algorithm 2. An $\mathcal{O}^*(2^n)$ exponential space algorithm for Weak Roman Domination.

```
foreach dominating set V' ⊆ V do
    foreach v ∈ V do
        Let f(v) = 1 if v ∈ V', and f(v) = 0 otherwise;
    Compute the set D̄ of vertices non-safely defended by χ^{V'};
    if D̄ ≠ ∅ then
        S = DS[D̄, n];
        if S ⊆ V' then
            foreach v ∈ S do
                Let f(v) = f(v) + 1;
return the computed wrd-function f of minimum cost;
```

defended by $\chi^{V_2'}$*, it holds that* $cost(f_2) \leq cost(f_1)$*, where* f_i *(*$i \in \{1, 2\}$*) is the legion function defined as:*

$$f_i(x) = \begin{cases} 2 \textit{ if } x \in (V_i' \cap S_i) \\ 1 \textit{ if } x \in (V_i' \cup S_i) \setminus (V_i' \cap S_i) \\ 0 \textit{ otherwise} \end{cases}$$

Proof. Assume that there exist three sets V_1', S_1 and $\overline{D_1}$ as stated in the lemma and assume that $S_1 \nsubseteq V_1'$. Let $V_2' = V_1' \cup S_1$. Since $S_1 \nsubseteq V_1'$, it follows that $V_2' \supset V_1'$. Let $\overline{D_2}$ be the set of vertices non-safely defended by $\chi^{V_2'}$. Observe that $\overline{D_2} \subseteq \overline{D_1}$, since $V_1' \subset V_2'$. By Lemma 2, we know that the legion function f_1 is in fact a wrd-function. Hence, by Lemma 1, we also have that $(V_1' \cap S_1)$ is a dominating set of $\overline{D_1}$, and thus of $\overline{D_2}$.

Denote by S_2 a minimum dominating set of $\overline{D_2}$. Then $|S_2| \leqslant |V_1' \cap S_1|$. We now consider the legion function f_2 as defined in the lemma. By Lemma 2, we know that f_2 is a wrd-function. Finally, since $|V_2'| = |V_1'| + |S_1 \setminus V_1'|$ and $|S_2| \leq |V_1' \cap S_1|$, we conclude the proof by the relation $cost(f_1) = |V_1'| + |S_1| = |V_1'| + |S_1 \setminus V_1'| + |S_1 \cap V_1'| \geq |V_2'| + |S_2| = cost(f_2)$. □

Correctness. The correctness of the preprocessing step is based on arguments of [16]. If the set X is empty then the initialization $\mathsf{DS}[\emptyset, k] = \emptyset$, for any $0 \leq k \leq n$, is clearly correct. If the set X is non empty but no vertex can be used to dominate X (i.e. $k = 0$), then $\mathsf{DS}[X, 0]$ is set to $\{\infty\}$ as a sentinel, meaning that there is no set Y (with $Y = \emptyset$) that can dominate X. The cardinality of $\{\infty\}$ is set to ∞. Finally the computation of $\mathsf{DS}[X, k]$ is done via an induction formula: either $v_k \notin \mathsf{DS}[X, k]$ or $v_k \in \mathsf{DS}[X, k]$ and in that latter case, $N(v_k)$ is dominated by v_k. As the sets X are considered by increasing order as well as the values of k, we note that the values $\mathsf{DS}[X, k-1]$ and $\mathsf{DS}[X \setminus N[v_k], k-1]$ have already been computed when the computation of $\mathsf{DS}[X, k]$ is done.

Now we show the correctness of Algorithm 2. It enumerates all possible sets V' as being possible candidates for the underlying set V_f. In particular, we discard any subset V' that does not induce a dominating set. By Lemma 1, it is sufficient to compute a dominating set $S \subseteq V'$ of the set of vertices $\overline{D}$ being non-safely defended by $\chi^{V'}$. Lemma 3 shows that if S is not included in V', then there exists a proper superset of V' which gives a wrd-function of cost being no more than the one obtained from V' and S, by Lemma 2. Let $V'_0 = V'$ and $S_0 = S$. As the graph is finite and the superset given by Lemma 3 is proper, there exists a finite $\ell \leq n$ and a sequence $V'_0 \subset V'_1 \subset ... \subset V'_\ell \subseteq V$ such that $S_i \nsubseteq V'_i$, for all $0 \leq i < \ell$, and $S_\ell \subseteq V'_\ell$. Since the algorithm enumerates all supersets of V', it follows that the set V'_ℓ will be considered at some iteration of the for-loop. This shows the correctness of Algorithm 2.

Complexity. The preprocessing step needs to consider each subset X of V and each value of k, $1 \leq k \leq n$. For each such couple (X, k), it retrieves the values of $\mathsf{DS}[X, k-1]$ and $\mathsf{DS}[X \setminus N[v_k], k-1]$ previously computed, and stores the new value in DS. Thus the preprocessing step requires $\mathcal{O}^*(2^n)$ time and space. The main part of the algorithm considers each (dominating set) $V' \subseteq V$, and computes in polynomial-time the set $\overline{D}$ of vertices non-safely defended by $\chi^{V'}$. A dominating set S of $\overline{D}$ is then retrieved in the already computed table DS in polynomial-time.

Theorem 3. Weak Roman Domination *can be solved in* $\mathcal{O}^*(2^n)$ *time and space.*

4.2 Using Polynomial Space

In order to obtain an exact exponential algorithm using only polynomial space, we need to avoid any exponential space consuming *preprocessing step* such as the one in the previous section. For this purpose, we use instead an exact exponential algorithm for Red-Blue Dominating Set using polynomial space to decide which vertices will be valued 2 to dominate the non-safely defended vertices.

Red-Blue Dominating Set:
Input: A bipartite graph $G = (R \cup B, E)$.
Output: A subset $S \subseteq R$ of minimum size dominating B.

Theorem 4 ([22]). *The* Red-Blue Dominating Set *problem can be solved in* $\mathcal{O}^*(1.2279^{|R|+|B|})$ *time and polynomial space.*

Algorithm. We consider the algorithm depicted in Algorithm 3, which might be seen as some modification of the previous Algorithm 2.

Observe that before computing a minimum red-blue dominating set on the bipartite graph $(\overline{C} \cup \overline{D}, E)$, we may modify the sets $\overline{C}$ and $\overline{D}$ as follows: for every vertex $v \in \overline{C}$, if v has at least two weakly non-safely defended neighbors, then we set $f(v) = 2$, and remove v from $\overline{C}$ and $N_{\overline{D}}(v)$ from $\overline{D}$.

Algorithm 3. An $\mathcal{O}^*(2.2279^n)$ poly-space algorithm for the Weak Roman Domination problem.

```
foreach dominating set V' ⊆ V do
    foreach v ∈ V do
        Let f(v) = 1 if v ∈ V', and f(v) = 0 otherwise;
    Compute the set D̄ of vertices non-safely defended by χ^{V'};
    if D̄ ≠ ∅ then
        Compute the set C̄ ⊆ V' of secured vertices which have at least one
        neighbor in D̄;
        /* Cleaning step */
        foreach v ∈ C̄ with at least two weakly non-safely defended neighbors in
        D̄ do
            Set f(v) = 2;
            Remove N_D̄(v) from D̄;
            Remove v from C̄;
        Let I = (C̄ ∪ D̄, E) be an instance of Red-Blue Dominating Set;
        if I admits a minimum red-blue dominating set S ⊆ C̄ then
            Set f(v) = 2 for every v ∈ S;
        else
            The current function f cannot yield a wrd-function;
return the computed wrd-function f of minimum cost;
```

Proposition 1. *The cleaning step on $\overline{C}$ and $\overline{D}$ does not modify a solution for* Red-Blue Dominating Set *on instance $I = (\overline{C} \cup \overline{D}, E)$.*

Proof. Let $v \in \overline{C}$ be a secured vertex with at least two weakly non-safely defended neighbors, say w_1 and w_2. Since w_1 and w_2 are weakly defended, their only secured neighbor is $v \in V'$; since they are non-safely defended, they need to be dominated by V_f^2 in order for f to be a wrd-function (Lemma 1). Thus we must set $f(v) = 2$. It follows that any minimum red-blue dominating set on instance $I = (\overline{C} \cup \overline{D}, E)$ must put $v \in \overline{C}$ into the red-blue dominating set in order to dominate all weakly non-safely defended neighbors of v in $\overline{D}$.

Now, observe that since all the neighbors of v are safely defended (because dominated by V_f^2), they can safely be removed from $\overline{D}$. Since v has no non-safely defended neighbor left, it can be removed from $\overline{C}$. □

Correctness. The correctness of the algorithm follows from Lemma 1 and the proof of correctness of Algorithm 2. The main difference lies in the computation of the dominating set of the vertices non-safely defended by $\chi^{V'}$. Indeed, in that case, we use Theorem 4 to find the vertices of V' that must have value 2 in order to dominate the vertices non-safely defended by $\chi^{V'}$. The correctness of this step follows from Lemma 1 and Proposition 1.

Complexity. Let us now give the time and space complexities of Algorithm 3. It is easy to see that for every subset $V' \subseteq V$, the initialisation of $f(x)$ for every $x \in V$ as well as the computation of the set $\overline{D}$ can be done in polynomial time and space, and that the cleaning step is also polynomial.

Regarding the legion function f being constructed, for any $V' \subseteq V$, our algorithm computes and reduces the set $\overline{D}$ of vertices non-safely defended by $\chi^{V'}$, and the set $\overline{C}$ of secured vertices which have at least one neighbor in $\overline{D}$. Those two sets are considered as an instance of Red-Blue Dominating Set to be solved in $\mathcal{O}^*(1.2279^{|\overline{D}|+|\overline{C}|})$ time and polynomial space using an algorithm from van Rooij [22]. To conclude our analysis, we need the following result.

Proposition 2. *For any* $V' \subseteq V$, $|\overline{D}| + |\overline{C}| \leq |V| - |V'|$.

Proof. For every vertex $v \in V'$, one of the following statements holds:

(i) v has no neighbor in $\overline{D}$, that is no neighbor non-safely defended by $\chi^{V'}$;
(ii) there exists at least one vertex $w \in V \setminus V'$ which is weakly defended by v.

First notice that (i) and (ii) are the only two possible cases. Indeed, if there exists $v \in V'$ such that $N_{\overline{D}}(v) \neq \emptyset$ but no vertex in $V \setminus V'$ is weakly defended by v, then the vertices in $N_{\overline{D}}(v)$ are safely defended, which is a contradiction.

If the first statement holds, then v is not included in $\overline{C}$. If the second statement holds, then either w is safely defended, or w is non-safely defended. If w is safely defended (that is no other neighbor of v is weakly defended by v), then w is not included in $\overline{D}$. If w is non-safely defended, then v has at least two weakly non-safely defended neighbors. Indeed, since w is weakly defended by v (as the second statement holds), v is the only neighbor of w in V'. Hence, there exists a nonempty set $\overline{D}_{v,w} = N_{\overline{D}}(v) \setminus N(w)$ such that w is weakly defended due to each vertex in $\overline{D}_{v,w}$. Now, since w is non-safely defended by v, there must exist a vertex $w' \in \overline{D}_{v,w}$ which is also weakly defended due to w. Then the cleaning step on $\overline{D}$ and $\overline{C}$ applies, which implies that v is removed from $\overline{C}$ and all neighbors of v (including w) are removed from $\overline{D}$. Altogether, for every vertex $v \in V'$, at least one vertex from V is not included in $\overline{C} \cup \overline{D}$, and hence at least $|V'|$ vertices from V are not included in $\overline{D} \cup \overline{C}$. $\diamond$

The overall algorithm iteratively runs all the previously described computations for every subset $V' \subseteq V$, and stores the minimum wrd-function considered so far using polynomial space. We claim that its worst-case time complexity corresponds to the following:

$$\mathcal{O}^*\Big(\sum_{i=1}^{n} \binom{n}{i} \cdot T(n-i)\Big) = \mathcal{O}^*\big(\sum_{i=1}^{n} \binom{n}{i} \cdot 1.2279^{n-i}\big) = \mathcal{O}^*\left(2.2279^{n}\right)$$

where $T(p)$ stands for the time complexity needed to compute a minimum red-blue dominating set in a graph with p vertices (here we use the one of [22]). Indeed, for any subset $V' \subseteq V$ containing i vertices, we apply Theorem 4 on the bipartite graph induced by $\overline{C}$ and $\overline{D}$, which contain less than $|V| - |V'| = n - i$ vertices (Proposition 2).

Theorem 5. Weak Roman Domination *can be solved in* $\mathcal{O}^*(2.2279^n)$ *time and polynomial space.*

References

1. Chambers, E.W., Kinnersley, B., Prince, N., West, D.B.: Extremal problems for roman domination. SIAM J. Discret. Math. 23(3), 1575–1586 (2009)
2. Chellali, M., Rad, N.J., Volkmann, L.: Some results on roman domination edge critical graphs. AKCE Int. J. Graphs Comb. 9(2), 195–203 (2012)
3. Cockayne, E.J., Favaron, O., Mynhardt, C.M.: Secure domination, weak roman domination and forbidden subgraphs. Bull. Inst. Combin. Appl. 39, 87–100 (2003)
4. Cockayne, E.J., Grobler, P.J.P., Gründlingh, W.R., Munganga, J., van Vuuren, J.H.: Protection of a graph. Util. Math. 67, 19–32 (2005)
5. Cockayne, E.J., Dreyer Jr., P.A., Hedetniemi, S.M., Hedetniemi, S.T.: Roman domination in graphs. Discret. Math. 278(1-3), 11–22 (2004)
6. Favaron, O., Karami, K., Khoeilar, R., Sheikholeslami, S.M.: On the roman domination number of a graph. Discret. Math. 309, 3447–3451 (2009)
7. Fomin, F.V., Grandoni, F., Kratsch, D.: A measure & conquer approach for the analysis of exact algorithms. J. ACM 56(5) (2009)
8. Garey, M.R., Johnson, D.S.: Computers and intractability. Freeman (1979)
9. Goddard, W., Hedetniemi, S.M., Hedetniemi, S.T.: Eternal security in graphs. J. Combin. Math. Combin. Comput. 52, 160–180 (2005)
10. Goldwasser, J.L., Klostermeyer, W.F.: Tight bounds for eternal dominating sets in graphs. Discret. Math. 308, 2589–2593 (2008)
11. Grobler, P.J.P., Mynhardt, C.M.: Secure domination critical graphs. Discret. Math. 309, 5820–5827 (2009)
12. Haynes, T.W., Hedetniemi, S.T., Slater, P.J.: Domination in graphs: advanced topics. Pure and Applied Mathematics, vol. 209. Marcel Dekker Inc. (1998)
13. Henning, M.A., Hedetniemi, S.T.: Defending the Roman Empire: a new strategy. Discret. Math. 266(1-3), 239–251 (2003)
14. Iwata, Y.: A faster algorithm for dominating set analyzed by the potential method. In: Marx, D., Rossmanith, P. (eds.) IPEC 2011. LNCS, vol. 7112, pp. 41–54. Springer, Heidelberg (2012)
15. Liedloff, M.: Algorithmes exacts et exponentiels pour les problèmes NP-difficiles: domination, variantes et généralisations. Phd thesis, Laboratoire d'Informatique Théorique et Appliquée, Université Paul Verlaine, Metz (2007)
16. Liedloff, M.: Finding a dominating set on bipartite graphs. Inf. Proc. Lett. 107(5), 154–157 (2008)
17. Liedloff, M., Kloks, T., Liu, J., Peng, S.-L.: Efficient algorithms for roman domination on some classes of graphs. Discret. App. Math. 156, 3400–3415 (2008)
18. Liu, C.-H., Chang, G.J.: Roman domination on 2-connected graphs. SIAM J. Discret. Math. 26(1), 193–205 (2012)

19. Malini Mai, T.N.M., Roushini Leely Pushpam, P.: Weak roman domination in graphs. Discussiones Mathematicae Graph Theory 31(1), 161–170 (2011)
20. ReVelle, C.S., Rosing, K.E.: Defendens Imperium Romanum: a classical problem in military strategy. Math. Assoc. of America 107(7), 585–594 (2000)
21. Stewart, I.: Defend the Roman Empire!. Scientific American 281(6), 136–139 (1999)
22. van Rooij, J.M.M.: Exact exponential-time algorithms for domination problems in graphs. Phd thesis, Utrecht University, Netherlands (2011)
23. van Rooij, J.M.M., Bodlaender, H.L.: Exact algorithms for dominating set. Discret. App. Math. 159(17), 2147–2164 (2011)
24. Xing, H.-M., Chen, X., Chen, X.-G.: A note on roman domination in graphs. Discret. Math. 306, 3338–3340 (2006)