

Article

Perfect Roman Domination: Aspects of Enumeration and Parameterization [†]

Kevin Mann ^{*}  and Henning Fernau ^{*} 

Theoretical Computer Science, Fachbereich IV, Trier University, 54296 Trier, Germany

^{*} Correspondence: mann@uni-trier.de (K.M.); fernau@uni-trier.de (H.F.)[†] This paper is an extended version of our paper published in Perfect Roman Domination: Aspects of Enumeration and Parameterization. In: Rescigno, A.A., Vaccaro, U. (eds) Combinatorial Algorithms. IWOCA 2024. Lecture Notes in Computer Science, vol 14764. Springer, Cham.

Abstract: Perfect Roman Dominating Functions and Unique Response Roman Dominating Functions are two ways to translate *perfect code* into the framework of Roman Dominating Functions. We also consider the enumeration of minimal Perfect Roman Dominating Functions and show a tight relation to minimal Roman Dominating Functions. Furthermore, we consider the complexity of the underlying decision problems PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION on special graph classes. For instance, split graphs are the first graph class for which UNIQUE RESPONSE ROMAN DOMINATION is polynomial-time solvable, while PERFECT ROMAN DOMINATION is NP-complete. Beyond this, we give polynomial-time algorithms for PERFECT ROMAN DOMINATION on interval graphs and for both decision problems on cobipartite graphs. However, both problems are NP-complete on chordal bipartite graphs. We show that both problems are $W[1]$ -complete if parameterized by solution size and FPT if parameterized by the dual parameter or by clique width.

Keywords: enumeration; (parameterized) complexity; domination in graphs; graph classes; perfect Roman domination; unique response Roman domination; extension problems



Citation: Mann, K.; Fernau, H. Perfect Roman Domination: Aspects of Enumeration and Parameterization. *Algorithms* **2024**, *17*, 576.
<https://doi.org/10.3390/a17120576>

Academic Editor: Adele Anna Rescigno

Received: 8 November 2024

Revised: 5 December 2024

Accepted: 9 December 2024

Published: 14 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

There are many ways to get along with NP-complete problems. For example, you could consider exponential-time algorithms or approximation algorithms. Here, we want to deal with PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION by looking into enumeration, into the complexity of related decision problems in special graph classes, and into parameterized complexity.

We only consider simple undirected graphs $G = (V, E)$ throughout this paper. We recall some notions next. $N_G(v) := \{u \in V \mid uv \in E\}$ describes the *open neighborhood* of $v \in V$ with respect to G . The *closed neighborhood* of $v \in V$ with respect to G is defined by $N_G[v] := N_G(v) \cup \{v\}$. $D \subseteq V$ is a *dominating set* (or a *perfect code*, respectively) of G if each vertex v finds a (unique) vertex u in $D \cap N_G[v]$. A function $f : V \rightarrow \{0, 1, 2\}$ is called a *Roman dominating function* (Rdf for short) if each $v \in V$ with $f(v) = 0$ is adjacent to a $u \in V$ with $f(u) = 2$. The weight of a function $f : V \rightarrow \{0, 1, 2\}$ is defined by $\omega(f) = \sum_{v \in V} f(v)$. The decision problem ROMAN DOMINATION has as input a graph G and a positive number k , and the question is whether there exists a Roman dominating function such that the weight of the function is at most k . ROMAN DOMINATION is a well-studied variation of the DOMINATING SET. It is motivated by a defense strategy of the Roman Empire. The idea is to position the armies in regions in such a way that either (1) one army is in this region or (2) two armies are in one neighbored region. We model the distribution of the armies on the region (viewed as vertices of the graph) by a function that maps each region to the number of armies placed on this region.

In recent years, this problem received notable attention [1–7]. As for the dominating set, there are also many variants of Roman Dominating Functions considered in the literature.

Examples are Roman- $\{2\}$ -domination (also known as Italian domination) [8] or double Roman Domination [9–11]. Clearly, as testified by books written on this topic [12–14], there are even more variations of the basic concept of domination in graphs.

Here, we will consider two further variations of Roman Domination, perfect and Unique Response Roman Domination. A *perfect Roman dominating function* (pRdf for short) (introduced by Henning et al. [15]) is a Roman Dominating Function where each vertex with value 0 has *exactly* one neighbor with value 2. If, moreover, all vertices with at least 1 as a value have no neighbor with value 2, then we have a *unique response Roman dominating function* (urRdf for short) (as introduced in [16]). Both variations can be seen as a way to translate the idea of perfect code into the realm of Roman dominating function. A further motivation can also be the idea of positioning armies in regions: if the armies are placed according to a perfect Roman dominating function and a region without any army is attacked, then it is clear from which region an army moves to secure the attacked region, so no time is wasted to first agree on who is the one to take action and move to the endangered region. This article is a revised and expanded version of a paper entitled “Perfect Roman Domination: Aspects of Enumeration and Parameterization”, which was presented at the 35th International Workshop Combinatorial Algorithms, IWOCA 2024, which took place on the island of Ischia, Italy, 1–3 July 2024. An extended abstract of this paper has appeared in the proceedings of IWOCA 2024; see [17]. More formally, the decision problems sketched so far are given as follows:

Problem name: UNIQUE RESPONSE ROMAN DOMINATION (URRD)
Given: A graph $G = (V, E)$ and $k \in \mathbb{N}$
Question: Is there a unique response Roman dominating function f on G with $\omega(f) \leq k$?

Problem name: PERFECT ROMAN DOMINATION (PRD)
Given: A graph $G = (V, E)$ and $k \in \mathbb{N}$
Question: Is there a perfect Roman dominating function f on G with $\omega(f) \leq k$?

Let $u_R(G)$ denote the smallest weight of any unique response Roman dominating function f on G . Furthermore, $\gamma_R^p(G)$ denotes the smallest weight of any perfect Roman dominating function on G . The following is known about the complexity of these decision problems. UNIQUE RESPONSE ROMAN DOMINATION is NP-complete on regular bipartite graphs [18]. Banerjee et al. [19] showed that it is NP-complete on chordal graphs, and polynomial-time solvable on distance-hereditary and interval graphs. They also proved that there is no UNIQUE RESPONSE ROMAN DOMINATION polynomial-time approximation algorithm within a factor of $n^{1-\epsilon}$ for any constant $\epsilon > 0$ and any n -vertex input graph, if $\text{NP} \neq \text{P}$. PERFECT ROMAN DOMINATION is NP-complete on chordal graphs, planar graphs, and bipartite graphs, and polynomial-time solvable on block graphs, cographs, series-parallel graphs, and proper interval graphs as shown in [20]. These results are complemented by our complexity results in this paper as surveyed in Table 1 that presents the main classical complexity results of our paper. Notice that, in particular, our results concerning split and chordal bipartite graphs solve questions posed as open in [19] (Table 1). We complement the NP-hardness results by providing FPT-algorithms for the parameters treewidth and solution size, as well as for the dual parameter, both for PERFECT ROMAN DOMINATION and for UNIQUE RESPONSE ROMAN DOMINATION. Also, if the clique decomposition is explicitly given, the problems can be solved in FPT-time when parameterized by the clique width of this decomposition.

Table 1. Survey of basic complexity results of PERFECT ROMAN DOMINATION (PRD) and of UNIQUE RESPONSE ROMAN DOMINATION (URRD) on graph various classes.

Graph Class	PRD	URRD
chordal	NP-complete [20]	NP-complete [19]
interval	P Theorem 6	P [19]
split	NP-complete Theorem 8	P Theorem 7
cobipartite	P Theorem 10	P Lemma 2
bipartite	NP-complete [20]	NP-complete [18]
chordal bipartite	NP-complete Theorem 13	NP-complete Theorem 13
distance-hereditary	P Theorem 14	P [19]
cographs	P [20]	P [19]
bounded treewidth	P Theorem 15	P Theorem 17

Enumeration is a wide area of research as testified by specialized workshops like [21]. For some examples, we refer to the survey [22]. From a practical point of view, enumeration can be interesting if not all aspects of the problem have been satisfactorily modeled. This also makes sense for the motivation of ROMAN DOMINATION (variants). It could be the case that between neighbored regions, there is a conflict unknown to the strategic planner of the army deployment. Such a conflict could be a reason why the regions do not want to defend each other. Enumerating minimal Roman dominating functions was considered by Abu-Khazam et al. [23]. There, it was shown that all pointwise (i.e., for $f, g : V \rightarrow \{0, 1, 2\}$, $f \leq g$ if and only if $f(v) \leq g(v)$ for all $v \in V$) minimal Roman dominating functions of a graph of order n can be enumerated in time $\mathcal{O}(1.9332^n)$ with *polynomial delay*. An enumeration algorithm has *polynomial delay* if the time between two subsequent outputs of the algorithm can be bounded by some polynomial, as well as the time from the start to the first output and the time between the last output and the termination of the algorithm. This property is interesting if a user (or another program) processes the results in a pipeline fashion. Also, it is not known if there is such an algorithm for enumerating (inclusion-wise) minimal dominating sets. In this context, we remark the following.

Observation 1. *Unless $P = NP$, there exists no polynomial-delay enumeration algorithm for (inclusion-wise minimal) perfect codes.*

Namely, such a polynomial-delay enumeration algorithm could also be used to decide (in polynomial time) if a perfect code exists in a given graph, a problem known to be NP-hard; see [24]. This contrasts with the polynomial-delay algorithms that we develop in this paper for enumerating minimal unique response and Perfect Roman Dominating Functions. We also obtain input-sensitive enumeration results in this way.

We now collect more notation that we will use in the paper. Let \mathbb{N} denote the set of all non-negative integers (including 0). For $n \in \mathbb{N}$, we will use the notation $[n] := \{1, \dots, n\}$. Let A, B be two sets. B^A is the set of all total functions $f : A \rightarrow B$. Likewise, $\binom{A}{2}$ is the set of all 2-element subsets of A . For $A \subseteq B$, $\chi_A : B \rightarrow \mathbb{N}$ denotes the characteristic function ($\chi_A(a) = 1$ if and only if $a \in A$; $\chi_A(a) = 0$, otherwise). Let $G = (V, E)$ be a graph. For a set $A \subseteq V$, the open neighborhood is defined as $N_G(A) := (\bigcup_{v \in A} N_G(v))$. The closed neighborhood of A is given by $N_G[A] := N_G(A) \cup A$. The *private neighborhood* of a vertex $v \in A$ with respect to G and A is denoted by $P_{G,A}(v) := N_G[v] \setminus N_G[A \setminus \{v\}]$. Let $G = (V, E)$ be a graph and $f : V \rightarrow \{0, 1, 2\}$. We define $V_i(f) := \{v \in V \mid f(v) = i\}$ for $i \in \{0, 1, 2\}$. If $v \in V$ obeys $f(v) = 2$, then $u \in N_G(v)$ with $f(u) = 0$ is called a *private neighbor* of v (with respect to f) if $|N_G(u) \cap V_2(f)| = 1$. In the previous notation, we dropped the subscript G if it was clear from the context.

Observation 2. *Let $G = (V, E)$ be a graph and $f : V \rightarrow \{0, 1, 2\}$. (1) If f is a perfect Roman dominating function, then for all $v \in V_2(f)$ and $u \in N(v) \cap V_0(f)$, u is a private neighbor of v .*

(2) If f is a Roman dominating function such that for all $v \in V_2(f)$ and $u \in N(v) \cap V_0(f)$, u is a private neighbor of v , then f is a perfect Roman dominating function.

Observation 3. Each unique response Roman dominating function on a graph without isolated vertices is a minimal Roman dominating function.

A final note on parameterized complexity: If P is some problem and κ is some parameter, we write κ - P to denote the problem P parameterized by κ .

2. Enumerating Minimal Perfect Roman Domination

The main goal of this section is to find a polynomial-delay algorithm to enumerate all minimal perfect Roman dominating functions even when solving the extension problem is hard, which blocks the standard approach to obtaining such enumeration algorithms.

Remark 1. It should be mentioned that there are graphs $G = (V, E)$ such that the set of perfect Roman dominating functions on G is not closed under \leq . This means that there exists a perfect Roman dominating function $f \in \{0, 1, 2\}^V$ and a $g \in \{0, 1, 2\}^V$ with $f \leq g$ such that g is no perfect Roman dominating function. One example of this is the 4-cycle

$$C_4 = (\{v_1, v_2, v_3, v_4\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_1\}\})$$

with the two functions

$$\begin{aligned} f(v_1) = 0, & \quad f(v_2) = 2, & \quad f(v_3) = 0, & \quad f(v_4) = 1, \\ g(v_1) = 0, & \quad g(v_2) = 2, & \quad g(v_3) = 0, & \quad g(v_4) = 2. \end{aligned}$$

This contradicts the idea of extension problems as they are defined in [25] because this violates the so-called upward closedness condition, but this condition is a matter of debate even in [25].

We often employ the following characterization result that is of independent interest.

Theorem 1. Let $G = (V, E)$ be a graph. A function $f : V \rightarrow \{0, 1, 2\}$ is a minimal perfect Roman dominating function if and only if the following conditions are true:

1. $\forall v \in V_0(f) : |N_G(v) \cap V_2(f)| = 1.$
2. $\forall v \in V_1(f) : |N_G(v) \cap V_2(f)| \neq 1.$
3. $\forall v \in V_2(f) : |N_G(v) \cap V_0(f)| \neq 0.$

Proof. Let f be a minimal perfect Roman dominating function. This implies the first condition. If there exists a $v \in V_1(f)$ with $|N_G(v) \cap V_2(f)| = 1$, then $f - \chi_{\{v\}}$ is also a perfect Roman dominating function, cf. Observation 2. For each $v \in V_2(f)$, $|N_G(v) \cap V_0(f)| \neq 0$, as, otherwise, $f - \chi_{\{v\}}$ is also a perfect Roman dominating function.

Now assume f is a function that fulfills these conditions. By Condition 1, we know f is a perfect Roman dominating function. Let g be a minimal perfect Roman dominating function with $g \leq f$. Thus, $V_0(f) \subseteq V_0(g)$ and $V_2(g) \subseteq V_2(f)$ hold. Assume there exists a $v \in V$ with $g(v) < f(v) = 2$. By Condition 3, there exists some $u \in N_G(v) \cap V_0(f)$. Because of condition 1, v is the only neighbor of u with value 2. Hence, $N_G(u) \cap V_2(g) \subseteq N_G(u) \cap (V_2(f) \setminus \{v\}) = \emptyset$. This contradicts the construction of g as a perfect Roman dominating function. Therefore, $V_2(g) = V_2(f)$. If there were a $v \in V$ with $g(v) = 0 < 1 = f(v)$, then this would contradict that g is a perfect Roman dominating function since $|N_G(v) \cap V_2(g)| = |N_G(v) \cap V_2(f)| \neq 1$. \square

Remark 2. This result implies that the constant function $\chi_V : V \rightarrow \{0, 1, 2\}, v \mapsto 1$ is the maximum minimal perfect Roman dominating function with respect to ω for each graph $G = (V, E)$, similar to maximum minimal Roman dominating functions as discussed in [26]. The reason is simple: if there exists a vertex $v \in V$ of value 2, then this vertex needs a neighbor u with value 0.

Since u has exactly one neighbor in $V_2(f)$, $\omega(f) \leq |V|$ for each perfect Roman dominating function f .

Problem name: EXTENSION PERFECT ROMAN DOMINATION (EXT PRD)
Given: A graph $G = (V, E)$ and $f \in \{0, 1, 2\}^V$.
Question: Is there a minimal perfect Roman dominating function g on G with $f \leq g$?

For the NP-hardness proof, we use MONOTONE 3-SAT-(2,2). This is a variant of 3-SAT where each clause includes only positive or negative literals and each variables appears twice positive and twice negative in clauses. It is known by [27] that also this restriction of 3-SAT is NP-complete.

Theorem 2. EXTENSION PERFECT ROMAN DOMINATION is NP-complete even on bipartite graphs of maximum degree 4.

Proof. For the NP-hardness result, we employ MONOTONE 3-SAT-(2,2). So, let $X := \{x_1, \dots, x_n\}$ be the set of variables and $C := \{c_1, \dots, c_m\}$ the set of clauses. To simplify the notation, denote $X' = \{x_i, \bar{x}_i \mid i \in [n]\}$. For each $j \in [m]$, let $c_j = l_{j,1} \vee l_{j,2} \vee l_{j,3} \in C$ with $l_{j,1}, l_{j,2}, l_{j,3} \in X'$.

Define $G = (V, E)$ with

$$\begin{aligned}
 V &:= \{v_i, \bar{v}_i, w_i, y_i, z_i \mid i \in [n]\} \cup \{p_j, q_j, r_j \mid j \in [m]\} \\
 E &:= \{\{v_i, w_i\}, \{\bar{v}_i, w_i\}, \{v_i, y_i\}, \{\bar{v}_i, y_i\}, \{z_i, y_i\} \mid i \in [n]\} \cup \\
 &\quad \{\{v_i, p_j\} \mid j \in [m], k \in [3], i \in [n], x_i = l_{j,k}\} \cup \\
 &\quad \{\{\bar{v}_i, p_j\} \mid j \in [m], k \in [3], i \in [n], \bar{x}_i = l_{j,k}\} \cup \\
 &\quad \{\{q_j, p_j\}, \{q_j, r_j\} \mid j \in [m]\}.
 \end{aligned}$$

The gadgets are also depicted in Figure 1. Furthermore, let

$$f := 2 \cdot \chi_{\{q_j \mid j \in [m]\} \cup \{y_i \mid i \in [n]\}} + \chi_{\{p_j \mid j \in [m]\}} \in \{0, 1, 2\}^V.$$

Clearly, G can be constructed in polynomial time. Furthermore, the sets $A := \{v_i, \bar{v}_i, z_i \mid i \in [n]\} \cup \{q_j \mid j \in [m]\}$ and $B := \{w_i, y_i \mid i \in [n]\} \cup \{p_j, r_j \mid j \in [m]\}$ are partition classes. This is easy to observe, as in each of the edges, we mention first the element of A and then the one of B . Since we consider MONOTONE 3-SAT-(2,2), for all $i \in [n], j \in [m]$

$$\begin{aligned}
 \deg(r_j) = \deg(z_i) = 1 < \deg(w_i) = \deg(q_j) = 2 < \deg(y_i) = 3 \\
 < \deg(v_i) = \deg(\bar{v}_i) = \deg(p_j) = 4.
 \end{aligned}$$



(a) Variable gadget: $i \in [n]$ (b) Clause gadget: $j \in [m]$

Figure 1. Construction for Theorem 2.

Claim 1. There is an assignment ϕ satisfying C if and only if there is a minimal perfect Roman dominating function $g \in \{0, 1, 2\}^V$ with $f \leq g$.

Proof. Let ϕ be an assignment satisfying the clauses. Then, define

$$g := f + 2 \cdot \chi_{\{v_i | \phi(x_i)=1\} \cup \{\bar{v}_i | \phi(x_i)=0\}}.$$

Trivially, $f \leq g$. For $j \in [m]$, $r_j \in V_0(g)$ is only dominated by $q_j \in V_2(g)$. Let $i \in [n]$. Observe that $z_i \in V_0(g)$ and that $N(z_i) \cap V_2(g) = \{y_i\}$. The last type of vertex to discuss in $V_0(g)$ is w_i . If $\phi(x_i) = 1$, w_i is only dominated by v_i . For $\phi(x_i) = 0$, $N(w_i) \cap V_2(g) = \{\bar{v}_i\}$. Hence, the first condition of Theorem 1 is fulfilled.

In order to see the validity of the second condition, let $j \in [m]$. Assume c_j includes only positive literals (otherwise, the proof is analogous). Since ϕ is a solution to the SAT instance, there exists $x_i \in c_j$ with $\phi(x_i) = 1$, which implies $|N(p_j) \cap V_2(g)| \geq 2$. As $V_1(g) = \{p_j \mid j \in [m]\}$, the second condition is also satisfied.

This leaves us to show the last condition. For each $j \in [m]$, $N(r_j) \cap V_2(g) = \{q_j\}$. Let $i \in [n]$. The vertex z_i is a private neighbor of $y_i \in V_2(g)$. Since ϕ is an assignment, w_i is the private neighbor of the vertex in $V_2(g) \cap \{v_i, \bar{v}_i\}$. Hence, g is a minimal perfect Roman dominating function.

For the reverse direction, assume there is a minimal perfect Roman dominating function $g \in \{0, 1, 2\}^V$ with $f \leq g$. Observe that $V \setminus N(V_2(f)) = \{w_i \mid i \in [n]\}$. Because each vertex in $V_2(g)$ needs to have a private neighbor, $V_2(g) \setminus V_2(f) \subseteq \{v_i, \bar{v}_i \mid i \in [n]\}$. For any $i \in [n]$, $\{v_i, \bar{v}_i\} \not\subseteq V_2(g)$. Otherwise, v_i would not have a private neighbor.

Define $\phi := \chi_{\{x_i | i \in [n], g(v_i)=2\}}$. Assume there is a clause c_j which is not satisfied by ϕ . Then, q_j is the only neighbor of p_j in $V_2(g)$. Since $g(p_j) = 1$, this contradicts Condition 2 of Theorem 1 and g is not a minimal perfect Roman dominating function. Hence, ϕ is a solution to the SAT instance. \square

This proves that we have described a valid polynomial-time computable reduction and hence we have shown NP-hardness. Membership in NP is easy by a guess-and-check approach. This concludes the proof of Theorem 2. \square

We use Theorem 1 also to obtain the announced polynomial-delay enumeration algorithm. This algorithm uses the idea that there is a bijection between all minimal perfect Roman dominating functions and all minimal Roman dominating functions of a graph. To show this bijection, we need the following characterization theorem from [23].

Theorem 3. Let $G = (V, E)$ be a graph, $f : V \rightarrow \{0, 1, 2\}$ be a function and let $G' := G[V_0(f) \cup V_2(f)]$. Then, f is a minimal Roman dominating function if and only if the following holds:

1. $N_G[V_2(f)] \cap V_1(f) = \emptyset$.
2. $\forall v \in V_2(f) : P_{G', V_2(f)}(v) \not\subseteq \{v\}$.
3. $V_2(f)$ is a minimal dominating set on G' .

Let $G = (V, E)$ be a graph. Define the following sets of functions:
 $\mu\text{-RDF}(G) = \{f : V \rightarrow \{0, 1, 2\} \mid f \text{ is a minimal Roman dominating function}\}$ and
 $\mu\text{-PRDF}(G) = \{f : V \rightarrow \{0, 1, 2\} \mid f \text{ is a minimal perfect Roman dominating function}\}.$

Theorem 4. Let $G = (V, E)$ be a graph. There is a bijection $B : \mu\text{-RDF}(G) \rightarrow \mu\text{-PRDF}(G)$. Furthermore, $B(f)$ and $B^{-1}(g)$ can be computed in polynomial time (with respect to G) for each $f \in \mu\text{-RDF}(G)$ and $g \in \mu\text{-PRDF}(G)$.

Proof. Let $f \in \mu\text{-RDF}(G)$ and $g \in \mu\text{-PRDF}(G)$. Define $B(f)$ by the three sets

$$\begin{aligned} V_0(B(f)) &= \{v \in V_0(f) \mid |N_G(v) \cap V_2(f)| = 1\}, \\ V_1(B(f)) &= \{v \in V_0(f) \mid |N_G(v) \cap V_2(f)| \geq 2\} \cup V_1(f), \quad V_2(B(f)) = V_2(f). \end{aligned}$$

By the definition of B , $|N_G(v) \cap V_2(f)| = 1$ holds for all $v \in V_0(B(f))$. Since each $v \in V$ with $f(v) = 2$ needs a private neighbor except itself by Theorem 3, $N_G(v) \cap V_0(B(f)) \neq \emptyset$

holds. As any $v \in V_1(f)$ has no neighbor in $V_2(f)$ and any $v \in V_1(B(f)) \setminus V_1(f)$ has at least two neighbors in $V_2(f)$, all conditions of Theorem 1 are met. Therefore, $B(f) \in \mu\text{-PRDF}(G)$.

Define $B^{-1}(g)$ by the three sets

$$V_0(B^{-1}(g)) = V_0(g) \cup \{v \in V_1(g) \mid |N_G(v) \cap V_2(g)| \geq 2\},$$

$$V_1(B^{-1}(g)) = \{v \in V_1(g) \mid |N_G(v) \cap V_2(g)| = 0\}, \quad V_2(B^{-1}(g)) = V_2(f).$$

By the definition of B^{-1} , $N_G(v) \cap V_2(B^{-1}(g)) = \emptyset$ for each $v \in V_1(B^{-1}(g))$. By Theorem 1, each $v \in V_0(g)$ has exactly one neighbor in $V_2(g)$. Therefore, each vertex in $V_0(B^{-1}(g))$ has a neighbor in $V_2(B^{-1}(g))$, and $V_2(B^{-1}(g))$ is a dominating set on $G[V_0(B^{-1}(g)) \cup V_2(B^{-1}(g))]$. Theorem 1 implies that each $v \in V_2(g)$ has a $u \in N_G(v) \cap V_0(g)$ with $\{v\} = N_G(u) \cap V_2(g)$. Hence, u is a private neighbor of v and $B^{-1}(g) \in \mu\text{-RDF}(G)$. This leaves to show that $B^{-1}(B(f)) = f$ and $B(B^{-1}(g)) = g$. Trivially,

$$V_2(B^{-1}(B(f))) = V_2(f) \text{ and } V_2(B(B^{-1}(g))) = V_2(g)$$

holds. Let $v \in V_1(f)$. By Theorem 3, we know $N_G(v) \cap V_2(f) = \emptyset$. This implies $v \in V_1(B(f)) \cap V_1(B^{-1}(B(f)))$. Let $v \in V_0(f)$ with $|N_G(v) \cap V_2(f)| = 1$. Thus, $v \in V_0(B(f)) \cap V_0(B^{-1}(B(f)))$. Let $v \in V_0(f)$ with $|N_G(v) \cap V_2(f)| \geq 2$. Therefore, $v \in V_1(B(f))$ holds. The definition of B^{-1} implies $v \in V_0(B^{-1}(B(f)))$. Hence, $B^{-1}(B(f)) = f$. Assume $v \in V_0(g)$. Theorem 1 implies $|N_G(v) \cap V_2(f)| = 1$. Therefore, $v \in V_0(B(B^{-1}(f))) \cap V_0(B^{-1}(f))$. For $v \in V_1(g)$ with $N_G(v) \cap V_2(g) = \emptyset$, the construction of the functions B, B^{-1} implies $v \in V_1(B(B^{-1}(f))) \cap V_1(B^{-1}(f))$. Let $v \in V_1(g)$ with $|N_G(v) \cap V_2(g)| \geq 2$. This leads to $v \in V_0(B^{-1}(g))$ and $v \in V_1(B(B^{-1}(g)))$. Hence, $B(B^{-1}(g)) = g$. Therefore, the theorem holds. \square

This bijection is a so-called *parsimonious reduction*. This is a class of reductions designed for enumeration problems. For more information, we refer to [28]. Even without this paper, Theorem 4 implies some further results thanks to [23].

Corollary 1. *There are graphs of order n that have at least $\sqrt[5]{16}^n \in \Omega(1.7441^n)$ many minimal perfect rdf.*

Theorem 5. *There is a polynomial-space, polynomial-delay algorithm that enumerates all minimal perfect Roman dominating functions of a given graph of order n in time $\mathcal{O}^*(1.9332^n)$.*

Remark 3. *It should be mentioned that the function B defined in the proof of Theorem 4 inherits inclusion-wise minimality, but this does not mean that, if f is a minimum Roman Dominating Function, then $B(f)$ is a minimum perfect Roman dominating function. To see this, we consider the graph $G = (V, E)$ with $V := \{v_1, \dots, v_8\}$ and*

$$E := \{\{v_1, v_3\}, \{v_2, v_3\}, \{v_4, v_3\}, \{v_5, v_3\}, \{v_4, v_6\}, \{v_5, v_6\}, \{v_7, v_6\}, \{v_8, v_6\}\}.$$

The graph is visualized in Figure 2. Since v_1, v_2 and v_7, v_8 are pairs of false twins and v_1, v_7 have no common neighbors, for each Roman dominating function f on G , $\omega(f) \geq 4$. Therefore, $f \in \{0, 1, 2\}^V$ with $V_0(f) = \{v_1, v_2, v_4, v_5, v_7, v_8\}$, $V_1(f) = \emptyset$, and $V_2(f) = \{v_3, v_6\}$ is a minimum Roman dominating function. $B(f)$ is given by $V_0(B(f)) = \{v_1, v_2, v_7, v_8\}$, $V_1(B(f)) = \{v_4, v_5\}$, and $V_2(B(f)) = \{v_3, v_6\}$. The weight is $\omega(B(f)) = 6$. The minimal perfect Roman dominating function $g \in \{0, 1, 2\}^V$ with $V_0(g) = \{v_1, v_2, v_4, v_5\}$, $V_1(g) = \{v_6, v_7, v_8\}$ and $V_2(g) = \{v_3\}$ fulfills $\omega(g) = 5$. Hence, $B(f)$ is no minimum perfect Roman dominating function.

Let $h \in \{0, 1, 2\}^V$ be a minimal perfect Roman dominating function. For $h(v_3) = h(v_6) = 2$, $h = B(f)$, as $v_1, v_2, v_7, v_8 \in N(V_2(h))$ are pendant. Assume $h(v_3) = 2 \neq h(v_6)$. If $h(v_4) = 2$ (respectively, $h(v_5) = 2$), then $h(v_7) = h(v_8) = 1$ and $\omega(h) \geq 6$. For $h(v_7) = 2$ (respectively, $h(v_8) = 2$), $h(v_8) = 1$ (respectively, $h(v_7) = 1$) and $\omega(g) = 5 \geq \omega(h)$. As $v_1, v_2 \notin V_2(h)$

for $h(v_3) = 2$, the remaining possibility is $h = g$. Thus, $\omega(g) = 5 \leq \omega(h)$ holds for each minimal perfect Roman dominating function with $h(v_3) = 2 \neq h(v_6)$. Analogously, for perfect Roman dominating functions h with $h(v_6) \neq 2 = h(v_3)$, $\omega(g) = 5 \leq \omega(h)$. Let $h(v_3) \neq 2$ and $h(v_6) \neq 2$. Since $N(v_i) \subseteq \{v_3, v_6\}$ for $i \in \{1, 2, 4, 5, 7, 8\}$, $v_1, v_2, v_4, v_5, v_7, v_8 \notin V_0(h)$. Hence, $\omega(h) \geq 6$ and g is a minimum perfect Roman dominating function. Furthermore, $B^{-1}(g) = g$ is not a minimum Roman dominating function.

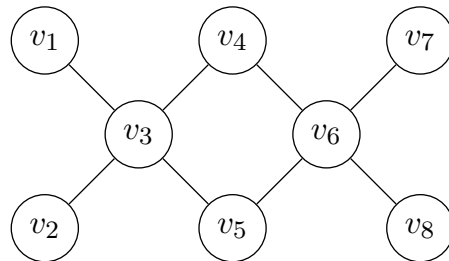


Figure 2. Counter-example of Remark 3.

According to [29], we obtain some further results for special graph classes: Let $G = (V, E)$, with $n := |V|$ being of the order of G . If G is a forest or interval graph, then there is a perfect Roman dominating function enumeration algorithm that runs in time $\mathcal{O}^*(\sqrt{3}^n)$ with polynomial delay. For both graph classes, a graph with many isolated edges is an example of a graph with $\sqrt{3}^n$ many minimal perfect Roman dominating function. This is also the worst-case example for chordal graphs which is known so far. We can enumerate all perfect Roman dominating functions of these graphs in time $\mathcal{O}(1.8940^n)$. If G is a split or cobipartite graph, then we can enumerate all perfect Roman dominating functions of G in $\mathcal{O}^*(\sqrt[3]{3}^n)$ with polynomial delay. Further, both classes include graphs of order n with $\Omega(\sqrt[3]{3}^n)$ many perfect Roman dominating functions. There is also a polynomial-time recursive algorithm to count all perfect Roman dominating functions of paths. The enumeration of unique response Roman dominating functions in split graphs is considered in Section 3.4.

3. Complexity and Enumeration in Special Graph Classes

In this section, we consider the complexity of our problems PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION on different graph classes: interval, split, cobipartite, and chordal bipartite graphs. PERFECT ROMAN DOMINATION is known to be NP-complete on chordal graphs [20].

3.1. Interval Graphs

We first start with a polynomial-time algorithm for PERFECT ROMAN DOMINATION for interval graphs. A graph $G = (V, E)$ is called an *interval graph* (see [30]) if, for $v \in V$, there is an interval $I_v = [l_v, r_v]$ such that $\{v, u\} \in E$ if and only if $I_v \cap I_u \neq \emptyset$. We can assume that the interval representation obeys $\forall u, v \in V : r_u = r_v \implies u = v$.

Theorem 6. PERFECT ROMAN DOMINATION is polynomial-time solvable on interval graphs.

Proof. Let $G = (V, E)$ be an interval graph such that $I_v = [l_v, r_v]$ is the interval representation of $v \in V$. The dynamic programming works from “left to right” according to some linear extension of the partial order defined as $w \leq v$ if and only if $l_w \leq l_v \wedge r_w \leq r_v$. For $v \in V$, $L_v := \{w \in V \mid l_w \leq r_v\}$ denotes the set of vertices which are dominated by v or “to the left” of I_v . To simplify our notation, we define $P_v := \{w \in V \setminus \{v\} \mid w \leq v\}$ to easily refer to the already considered vertices in the dynamic program. $W_v := \{w \in V \mid r_v < r_w\}$ is the set of not yet considered vertices. For $v \in V$, $u \in P_v$ and $w \in P_u$, define $Z_{u,v} := (L_v \setminus (N[u] \cup N[v])) \cup (N(u) \cap N(v))$ and

$Y_{w,u,v} := ((N(u) \cap N(v)) \setminus N[w]) \cup (L_v \setminus (L_u \cup N[v]))$; see Figure 3. All these sets and their cardinalities can be computed before the actual dynamic programming starts. Also, we can compute the perfect Roman dominating functions that set at most one vertex to 2 and store their minimum weight in $\min_{\leq 1}$.

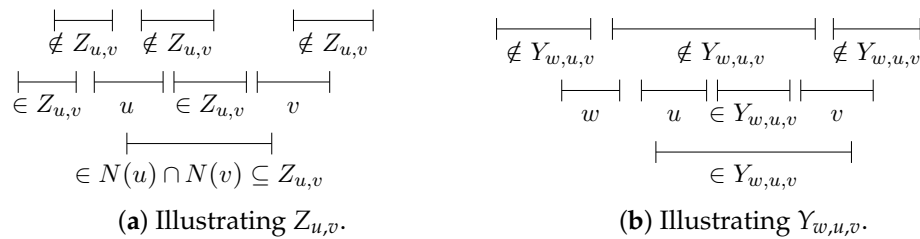


Figure 3. Examples illustrating our notions.

We need some more notation to formulate the dynamic programming algorithm and prove its correctness. For $v \in V$ and $u \in P_v$, define $\Gamma_{u,v} := \{f \in \{0, 1, 2\}^V \mid f|_{L_v} \in \mathcal{PRDF}(G[L_v]) \wedge f(u) = f(v) = 2 \wedge \forall w \in W_u \setminus \{v\} : f(w) \neq 2\}$ and $\gamma_{u,v} = \min_{f \in \Gamma_{u,v}} \omega(f)$. In the following, $f'_{u,v}$ will denote one function from $\Gamma_{u,v}$ that attains this minimum, i.e., $\omega(f'_{u,v}) = \gamma_{u,v}$. For the inductive computation, we will reuse the values $\gamma_{w,u}$ with $w \in P_u$. For the correctness of this approach, the following observations are helpful.

1. $Z_{u,v}$ is connected to the function $f_{u,v} \in \Gamma_{u,v}$ by setting $V_2(f_{u,v}) = \{u, v\}$ and $V_1(f_{u,v}) = Z_{u,v}$.
2. $Y_{w,u,v} = V_1(f_{w,u,v}) \setminus V_1(f'_{w,u})$, where $f_{w,u,v}$ is the auxiliary perfect Roman dominating function which we will construct from $f'_{w,u}$ in the induction step.
3. If $f \in \Gamma_{u,v}$, then $\tilde{f} := f + \chi_{V \setminus L_v}$ is a perfect Roman dominating function on G . If f is a minimum, then \tilde{f} is a minimum perfect Roman dominating function on G .

By the third property, after the inductive computations,

$$\min\{\min_{\leq 1}, \min\{\gamma_{u,v} + |V \setminus L_v| \mid v \in V, u \in P_v\}\}$$

gives the minimum weight of any perfect Roman dominating function for G .

Assume now that v is a vertex for which we want to compute all relevant $\gamma_{u,v}$ next, with $u \in P_v$. The following claim contains a formula that describes how to do this. It implicitly also covers the initial step: then, $\gamma_{u,v} = 4 + |Z_{u,v}|$ since $P_u = \emptyset$.

Claim 2. Let $v \in V$ and $u \in P_v$. Then,

$$\gamma_{u,v} = \min\{4 + |Z_{u,v}|, 2 - |\{v\} \cap N(u) \cap N(w)| + \gamma_{w,u} + |Y_{w,u,v}| \mid w \in P_u\}.$$

Proof. Let $f'_{w,u} \in \Gamma_{w,u}$ for $w \in P_u$ such that $\omega(f'_{w,u}) = \gamma_{w,u}$. Define

$$f_{w,u,v} := f'_{w,u} + (2 - \chi_{N(u) \cap N(w)}) \cdot \chi_{\{v\}} + \chi_{Y_{w,u,v}}.$$

Clearly, $\omega(f_{w,u,v}) = \gamma_{w,u} + 2 - |\{v\} \cap N(u) \cap N(w)| + |Y_{w,u,v}|$ and $f_{w,u,v}(x) = f'_{w,u}(x) \in \{0, 1, 2\}$ for $x \in V \setminus (\{v\} \cup Y_{w,u,v})$. In particular, $f_{w,u,v}(u) = f'_{w,u}(u) = 2$. (More generally, $V_2(f_{w,u,v}) \supseteq V_2(f'_{w,u})$.) Now we consider $x \in Y_{w,u,v}$. For $x \in L_v \setminus (L_u \cup N(v)) \subseteq V \setminus L_u$, $f'_{w,u}(x) = 0$. Otherwise, $f'_{w,u} - \chi_{\{x\}}$ would belong to $\Gamma_{w,u}$ with $\omega(f'_{w,u} - \chi_{\{x\}}) < \omega(f'_{w,u})$. So assume $x \in (N(v) \cup N(u)) \setminus N(w)$. Since $w \in P_u$, $r_w < l_x < r_u < r_v$. If $\emptyset \neq (V_2(f'_{w,u}) \setminus \{u\}) \cap N(x)$, this would contradict the fact that $f'_{w,u}(z) \neq 2$ for all $z \in W_w \setminus \{u\}$. Hence, $f'_{w,u}(z) = 0$ because of minimality. Hence, $Y_{w,u,v} \subseteq V_0(f'_{w,u})$ and $f_{w,u,v}(x) \in \{0, 1, 2\}$ for $x \in Y_{w,u,v}$.

Now consider $f_{w,u,v}(v)$. As $f'_{w,u}(v) \neq 2$, $v \notin Y_{w,u,v}$ and $f'_{w,u}(v) + 1 \leq f_{w,u,v}(v) \leq f'_{w,u}(v) + 2$, i.e., $1 \leq f_{w,u,v}(v) \leq 3$. We will next show that $1 \leq f_{w,u,v}(v) \leq 2$. For the sake of contradiction, assume $f_{w,u,v}(v) = 3$. Then, $f'_{w,u}(v) = 1$ and $v \notin N(u) \cap N(w)$. As (in addition) $r_w < r_u < r_v$ and $l_w \leq l_u \leq l_v$, we must have $r_w < l_v$. Hence, $\{w, v\} \notin E$.

If $N(v) \cap V_2(f'_{w,u}) \subseteq \{u\}$, then $f'_{w,u} - \chi_{\{v\}} \in \Gamma_{w,u}$, contradicting minimality of $f'_{w,u}$ in $\Gamma_{w,u}$. If $\emptyset \neq (V_2(f'_{w,u}) \cap N(v)) \setminus \{u, w\}$, this would contradict the construction of $\Gamma_{w,u}$, as $r_w < l_v < r_x$. Hence, $f_{w,u,v}(v) \in \{1, 2\}$. If $f_{w,u,v}(v) = 1$, then $v \in N(u) \cap N(w) \subseteq L_u$ and $f'_{w,u}(v) = 0$. But then $f'_{w,u}|_{L_u}$ would not be a perfect Roman dominating function. Therefore, $f_{w,u,v}(v) = 2$.

Now we want show that $f_{w,u,v}|_{L_v}$ is a perfect Roman dominating function on $G[L_v]$. Since $f'_{w,u}$ is a Roman dominating function on $G[L_u]$ and $L_v = L_u \cup N[v] \cup (L_v \setminus (L_u \cup N[v])) \subseteq L_u \cup N[v] \cup Y_{w,u,v}$, $f_{w,u,v}|_{L_v}$ is a Roman dominating function on $G[L_v]$. Assume there is an $x \in V_0(f_{w,u,v}|_{L_v})$ with $|V_2(f_{w,u,v}|_{L_v}) \cap N(x)| > 1$. Since $V_2(f_{w,u,v}) = V_2(f'_{w,u}) \cup \{v\}$ (also implying that for all $y \in W_u \setminus \{v\}$, $f_{w,u,v}(y) \neq 2$), $x \in N(v)$. For each $x \in N(v) \cap N(u)$, either $x \in Y_{w,u,v}$ or $x \in N(u) \cap N(w) \subseteq V_1(f'_{w,u}) \cup V_2(f'_{w,u})$. Hence, $x \in N(v) \setminus N(u)$ and $N(x) \subseteq W_u \subseteq W_w$ (otherwise $u \notin P_v$). By definition of $\Gamma_{w,u}$, $N(x) \cap V_2(f_{w,u,v}) = \{v\}$. In total, $f_{w,u,v} \in \Gamma_{u,v}$.

By the definition of $\Gamma_{u,v}$, $\{u, v\} \subseteq V_2(f'_{u,v})$. (a) Assume $\{u, v\} = V_2(f')$. Since $f'_{u,v}|_{L_v}$ is a perfect Roman dominating function on $G[L_v]$, $L_v \setminus (N[u] \cup N[v]) \subseteq V_1(f'_{u,v})$, as these vertices are not dominated. Further, $N(u) \cap N(v) \subseteq V_1(f'_{u,v})$ because these vertices have two neighbors in $V_2(f'_{u,v})$. So $f := 2 \cdot \chi_{\{u,v\}} + \chi_{Z_{u,v}} \leq f'_{u,v}$. As $f \in \Gamma_{u,v}$, $f = f'_{u,v}$ with $\omega(f) = 4 + |Z_{u,v}|$. (b) Assume $\{u, v\} \subsetneq V_2(f'_{u,v})$. Then, let $w \in V_2(f'_{u,v}) \setminus \{u, v\}$ with maximal r_w . Since $w \notin W_u \setminus \{v\}$, $r_w < r_u$. If $l_u < l_w$, $f'_{u,v} - \chi_{\{w\}} \in \Gamma_{u,v}$ with $\omega(f'_{u,v} - \chi_{\{w\}}) < f'_{u,v}$. Therefore, $w \in P_u$ and for each $y \in V_2(f'_{u,v}) \setminus \{v\}$, $N(y) \subseteq L_u$. Hence, $x \in L_v \setminus (L_u \cup N[v])$ has no neighbor in $V_2(f'_{u,v})$ and $L_v \setminus (L_u \cup N[v]) \subseteq V_1(f'_{u,v})$. Consider a $x \in (N(u) \cap N(v)) \setminus N(w)$. Since x has more than one neighbor in $V_2(f'_{u,v})$, $f'_{u,v}(x) > 0$. Clearly, $r_w < l_x < r_u$ holds. By the definition of w , $f'_{u,v}(x) \neq 2$. Define $f'' := f'_{u,v} - \chi_{Y_{w,u,v}} - (2 - \chi_{N(u) \cap N(w)}) \cdot \chi_{\{v\}}$. Clearly, $\omega(f'') = \omega(f'_{u,v}) - 2 + |\{v\} \cap N(u) \cap N(w)| - |Y_{w,u,v}|$. We want to show that $f'' \in \Gamma_{w,u}$ to obtain $\gamma_{w,u} < \omega(f'')$ and

$$\min\{4 + |Z_{u,v}|, 2 - |\{v\} \cap N(u) \cap N(w)| + \gamma_{w,v} + |Y_{w,u,v}| \mid w \in P_u\} \leq \omega(f'_{u,v}).$$

By the definition of w , $f''(y) \neq 2$ for all $y \in W_w \setminus \{u\}$. Further, $f''(w) = f''(u) = 2$. This leaves us to show that $\tilde{f} = f''|_{L_u} \in \mathcal{PRDF}(G[L_u])$. Assume there is a $x \in V_0(\tilde{f})$ with $|N(x) \cap V_2(\tilde{f})| \neq 1$. As $V_2(f'') = V_2(f'_{u,v}) \setminus \{v\}$, for $x \in V_0(f'_{u,v}) \setminus (N[v] \cup Y_{u,v,w})$, $|N(x) \cap V_2(\tilde{f})| = 1$. Since $L_u \cap (L_v \setminus (L_u \cup N[v])) = \emptyset$, we can assume $x \in N[v] \cap L_u$. For $y \in N(v) \cup N(u) \cup N(w)$, $\tilde{f}(y) = f''(y) = 1$ and $w \in V_2(\tilde{f})$. Let $x \in N(u) \setminus N[w]$. Hence, $|N(x) \cap V_2(\tilde{f})| > 1$. This contradicts the definition of w , as there is a $z \in V_2(\tilde{f}) \setminus \{u\} = V_2(f'_{u,v}) \setminus \{u, v\}$ with $r_w < l_x < r_z$. If $x \in N(v) \setminus N(u)$, then $r_u < l_x$ and $x \in L_u$, since $l_u < l_v$ and $r_u < r_v$. Now consider $x = v$. For $v \in N(u) \cap N(w)$, $\tilde{f}(v) = 1$. If $v \notin N(u)$, then $r_u < l_v$ and $v \notin L_u$. Since the case $v \in N(u) \setminus N(w)$ is discussed above, this completes the proof of the claim. \square

Let f be a minimum perfect Roman dominating function on G . If there is a $u \in V$ with $V_2(f) = \{u\}$, then $\omega(u) = |V \setminus N[u]| + 2$. So assume $|V_2(f)| > 1$. Then, let $u, v \in V_2(f)$ such that for all $w \in V_2(f) \setminus \{v, u\}$, $r_w < r_u < r_v$. Therefore, $N(x) \setminus L_v = \emptyset$ for each $x \in V_2(f)$. So f is perfect Roman dominating function on L_v . Further $V \setminus L_v \subseteq V_1(f)$. Because f is also minimal, $l_u < l_v$. Hence, $\gamma_{u,v} + |V \setminus L_v| \leq \omega(f)$. Since $f_{u,v} + \chi_{V \setminus L_v}$ is also a perfect Roman dominating function for $f_{u,v} \in \Gamma_{u,v}$, equality holds.

In total, an algorithm only has to compute $\gamma_{u,v} + |V \setminus L_v|$ for each $v \in V$ and $u \in P_v$ as explained above. Clearly, $\gamma_{u,v}$ can be computed in polynomial time, and as there are at most $|V|^2$ many such pairs, the whole algorithm runs in polynomial time. This concludes the proof of Theorem 6. \square

3.2. Split Graphs

A split graph $G = (V, E)$ is a graph in which the vertex set V can be partitioned into C, I , where C is a clique and I is an independent set of G . It seems to be open as to which complexity class PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMI-

NATION belong on split graphs as explicitly asked in [19]. We will answer this question in a somewhat surprising manner: while UNIQUE RESPONSE ROMAN DOMINATION can be solved in linear time, PERFECT ROMAN DOMINATION turns out to be NP-complete. First, we will consider unique response Roman dominating functions on split graphs and prove a combinatorial result.

Lemma 1. *Let $G = (V, E)$ be a connected split graph with a clique C and an independent set I such that $V = C \cup I$ and $C \cap I = \emptyset$. For each unique response Roman dominating function $f : V \rightarrow \{0, 1, 2\}$, one of the following conditions holds:*

1. $V_2(f) \cap I = \emptyset$ and $|V_2(f) \cap C| \leq 1$;
2. $V_2(f) \subseteq I$.

Proof. Suppose that $V_2(f) \not\subseteq I$. By the definition of unique response Roman dominating function, two vertices of C cannot have the value 2 since C is a clique. Hence, $|V_2(f) \cap C| \leq 1$. For the sake of contradiction, assume there are $v \in I$ and $u \in C$ with $f(u) = f(v) = 2$. As the function f is a unique response Roman dominating function, u and v cannot be neighbors. Since G is connected, $\emptyset \subsetneq N_G(v) \subseteq C \setminus \{u\} \subseteq N_G(u)$. This contradicts the assumption that f is a unique response Roman dominating function. \square

This lemma implies that we can solve UNIQUE RESPONSE ROMAN DOMINATION by looking for a vertex v of maximum degree (which must be in an inclusion-wise maximal clique) and setting the value of v to two and the values of the other vertices accordingly. This implies the following.

Theorem 7. *UNIQUE RESPONSE ROMAN DOMINATION can be solved in linear time on split graphs.*

Proof. How can we find a unique response Roman dominating function of the smallest weight? Let us take a look at the unique response Roman dominating function for the two cases of Lemma 1. Without loss of generality, we can assume that for the given split graph $G = (V, E)$, we have a decomposition $V = C \cup I$ into a clique C and an independent set I such that C is inclusion-wise maximal. Consider the unique response Roman dominating function $f : V \rightarrow \{0, 1, 2\}$.

First, assume f fulfills $V_2(f) \cap I = \emptyset$ and $|V_2(f) \cap C| \leq 1$. Since $V_2(f) \cap C = \emptyset$ would imply $f = 1$ (constant) and $\omega(f) = |C| + |I|$, let us assume $\{v\} = V_2(f) \cap C$. Therefore, $C \cap V_1(f) = \emptyset$ holds. Hence, $\omega(f) = 2 + |I \setminus N_G(v)|$. This implies $u_R(G) \leq \min_{v \in C} (2 + |I \setminus N_G(v)|) \leq 2 + |I|$.

Secondly, let us assume $V_2(f) \subseteq I$. As I is an independent set and there is no vertex in C with the value 2, $f(v) \geq 1$ for each $v \in I$. This implies

$$\omega(f) = |I| + |V_2(f)| + |C \setminus N_G(V_2(f))| \geq |I| + 1.$$

Equality holds if and only if there exists a vertex $v \in I$ with $N_G(v) = C$ and, for all $u \in I$, $f(u) = 2$ if and only if $u = v$. In this case, C is not maximal, contradicting our assumption. Thus, this second case can never give a smaller value than the first one, i.e., $u_R(G) = \min_{v \in C} (2 + |I \setminus N_G(v)|)$. Hence, in order to solve the UNIQUE RESPONSE ROMAN DOMINATION on G , we only have to find the vertex with the highest degree, as it will be in a maximal clique. \square

Interestingly, the seemingly very similar problem of PERFECT ROMAN DOMINATION on split graphs is harder to solve. To see this, we will use a problem called PERFECT CODE that is formally defined next.

Problem name: PERFECT CODE (PC for short)
Given: A graph $G = (V, E)$ and $k \in \mathbb{N}$
Question: Is there a perfect code D on G with $|D| \leq k$?

It is known that PERFECT CODE is NP-complete, also in restricted graph classes like cubic planar graphs and if $k \geq |V|$; see [24,31,32].

Theorem 8. PERFECT ROMAN DOMINATION on split graphs is NP-complete.

Proof. Membership follows from PERFECT ROMAN DOMINATION on general graphs. To prove NP-hardness, we will make use of the NP-hardness of PERFECT CODE. Let $G = (V, E)$ be a graph and $k \in \mathbb{N}$. To avoid trivialities, assume $k \leq |V|$.

Define $t = 3 \cdot |V| + 4$, $k' = k + |V| + 4$ and $G' = (V', E')$ with

$$V' = \{x, y\} \cup \{v' \mid v \in V\} \cup \{v_i \mid v \in V, i \in [t]\} \cup \{a_1, b_1, \dots, a_t, b_t\},$$

$$E' = \{\{x, a_i\}, \{y, b_i\} \mid i \in [t]\} \cup \{\{v', u_i\} \mid v, u \in V, u \in N[v], i \in [t]\} \cup \left(\frac{\{x, y\} \cup \{v' \mid v \in V\}}{2} \right).$$

The instance (G', k') of PERFECT ROMAN DOMINATION can be constructed in polynomial time. Moreover, G' is easily seen to be a split graph. We still have to prove that (G, k) is a yes-instance of PERFECT CODE if and only if (G', k') is a yes-instance of PERFECT ROMAN DOMINATION.

Let $S \subseteq V$ be a perfect code of G with $|S| \leq k$. Define $f \in \{0, 1, 2\}^V$ by the sets

$$V_0(f) = \{v_i \mid i \in [t], v \in V\} \cup \{a_i, b_i \mid i \in [t]\},$$

$$V_1(f) = \{v' \mid v \notin S\},$$

$$V_2(f) = \{x, y\} \cup \{v' \mid v \in S\}.$$

This implies $\omega(f) = |V \setminus S| + 4 + 2 \cdot |S| = |V| + 4 + |S| \leq k'$. The set $\{a_1, \dots, a_t\} \cup \{v' \mid f(v) \neq 2\}$ is only dominated by x and the vertices b_1, \dots, b_t are only dominated by y . Assume there are $v \in V$ and $i \in [t]$ such that $|N_{G'}(v_i) \cap V_2(f)| \neq 1$. By $N_{G'}(v_i) = \{u' \mid u \in N_G[v]\}$, this implies $|N_{G'}(v_i) \cap V_2(f)| = |N_G(v) \cap S| \neq 1$. This contradicts that S is a perfect code.

Assume that $f \in \{0, 1, 2\}^V$ is a minimum perfect Roman dominating function (so also minimal) with $\omega(f) \leq k'$.

Claim 3. For each $z \in \{v_i, a_i, b_i \mid i \in [t], v \in V\}$, it holds that $f(z) = 0$.

Proof. For the sake of contradiction, assume that there is some $v \in V$ and $i \in [t]$ with $f(v_i) = 2$. This implies with Theorem 1 that $\{x, y\} \cup \{z' \mid z \in V\} \subseteq N_{G'}(u')$ has no vertex with f -value 2. Thus, $N_{G'}(v_i) \cap V_2(f) = \emptyset$. Therefore, $f(v_j) \neq 0$ for each $j \in [t]$. Hence, $\omega(f) \geq t + 1 > k'$, contradicting the choice of f .

Assume that there are some $v \in V$ and $i \in [t]$ with $f(v_i) = 1$. Furthermore, we can assume $|N_{G'}(v_i) \cap V_2(f)| \neq 1$, by Theorem 1. As by construction $N_{G'}(v_i) = N_{G'}(v_j)$ for each $j \in [t]$, we find $|N_{G'}(v_i) \cap V_2(f)| = |N_{G'}(v_j) \cap V_2(f)|$. If $|N_{G'}(v_i) \cap V_2(f)| = 0$, then $f(v_j) \neq 0$ holds for each $j \in [t]$ since v_j is not dominated. Hence, $\omega(f) \geq t \cdot |V| \geq t > k'$, contradicting the choice of f . Assume $|N_{G'}(v_i) \cap V_2(f)| \geq 2$. Thus, for each $j \in [t]$, $f(v_j) \neq 0$, as, otherwise, f is not perfect. This contradicts the construction as above.

We can discuss the claims concerning a_i and b_i along very similar lines. \square

Further, we know $f(x) = f(y) = 2$. Define $S = \{v \mid f(v') = 2\}$. As $f(v_i) = 0$ for each $v \in V, i \in [t]$, $|N_{G'}(v_i) \cap V_2(f)| = 1$. This implies that $|N_G[v] \cap S| = 1$ for each $v \in V$. Therefore, S is a perfect code.

Since $f(x) = f(y) = 2, f(v') \geq 1$ for all $v \in V$. This implies that $\omega(f) = |V| + |V_2(f)| + 4 = |V| + |S| + 4 \leq k'$. Hence, $|S| \leq k' - 4 - |V| = k$. This concludes the proof of Theorem 8. \square

We contrast this by giving a branching algorithm for PERFECT ROMAN DOMINATION that yields the following result. Our algorithm relies on combinatorial insights from Theorem 4 and [29], where Roman dominating functions have been discussed on split graphs so that $V_2(f') \cap I = \emptyset$ or $V_2(f') \cap C = \emptyset$ for minimum f' .

Theorem 9. $\omega(f)$ -PERFECT ROMAN DOMINATION can be solved in time $\mathcal{O}^*(\varphi^{\omega(f)})$ on split graphs, where φ is the golden ratio.

Proof. Let $G = (V, E)$ be a split graph with the vertex set partition $V = C \cup I$, where C is a clique and I an independent set. Let k be the input parameter. Further, let $f' : V \rightarrow \{0, 1, 2\}$ be a minimum perfect Roman dominating function on G . Then, by Theorem 4 and [29], where Roman dominating functions have been discussed on split graphs, $V_2(f') \cap I = \emptyset$ or $V_2(f') \cap C = \emptyset$. We have to consider these two possibilities for the perfect Roman dominating functions f that we will consider in our algorithm.

For $V_2(f) \cap C = \emptyset$, we just branch on each vertex in I if it obtains the value 1 or 2. The vertices cannot obtain the value 0, as they cannot be dominated by any other vertex. After deciding this, we can delete the vertex, and the parameter goes down by either 1 or 2. This gives us the branching vector $(1, 2)$. When we have branched on each vertex in I , then we assign 0 to each $c \in C$ with $|V_2(f) \cap N(c)| = 1$, and we assign 1 otherwise (this is important to obtain the perfect Roman dominating function property). Now, we only need to check if any $\omega(f)$ is at most the given parameter value k .

Now, we consider the perfect Roman dominating functions $f \in \{0, 1, 2\}^V$ with $V_2(f) \cap I = \emptyset$. As C is a clique, if $|V_2(f) \cap C| \geq 2$, then $C \subseteq V_1(f) \cup V_2(f)$. First, we consider the polynomially many perfect Roman dominating functions with $|V_2(f) \cap C| = 1$. Since these functions are unique response Roman dominating functions, this can be performed in polynomial time. After this, we guess two vertices from C and assign the value 2 to these vertices. Again, there are polynomially many possibilities here. From now on, we can branch on the remaining vertices in C and decide if these will be assigned to 1 or 2. Analogously to $V_2(f) \cap C = \emptyset$, then we assign 0 to each $v \in I$ with $|V_2(f) \cap N(v)| = 1$, and we assign 1 otherwise. Again, we also have to check if $\omega(f)$ is at most the given parameter value k .

Our algorithm obviously runs in time $\mathcal{O}^*(\varphi^k)$, where φ is the golden ratio and k the parameter. \square

3.3. Cobipartite Graphs

Now, we consider *cobipartite graphs*, which are the complementary graphs of bipartite graphs. These graphs can be also characterized as graphs for which the vertex set can be partitioned into two cliques. On this graph class, PERFECT ROMAN DOMINATION is solvable in polynomial time.

Theorem 10. PERFECT ROMAN DOMINATION is polynomial-time solvable on cobipartite graphs.

Proof. Let $G = (V, E)$ be a cobipartite graph, with V partitioned into the two cliques $C_1, C_2 \subseteq V$. For $v \in V$, define the perfect Roman dominating function

$$g_v : V \rightarrow \{0, 1, 2\}, x \mapsto \begin{cases} 0, & x \in N(v), \\ 1, & x \in V \setminus N[v], \\ 2, & x = v. \end{cases}$$

Let $f \in \{0, 1, 2\}^V$ be a perfect Roman dominating function with $|V_2(f) \cap C_1| \geq 2$ and $v \in C_2$. Next, we prove that $\omega(g_v) \leq \omega(f)$. For each $u \in C_1 \setminus V_2(f), |V_2(f) \cap N(u)| \geq 2$.

Hence, $C_1 \subseteq V_1(f) \cup V_2(f)$. Since $C_2 \subseteq N[v] = V_0(g_v) \cup \{v\}$, $\omega(g_v) \leq |C_1| + 2 \leq \omega(f)$. Symmetrically, $\omega(g_v) \leq |C_1| + 2 \leq \omega(f)$ for $v \in C_1$ and a perfect Roman dominating function f with $|V_2(f) \cap C_2| \geq 2$.

Let f be a minimum perfect Roman dominating function on G . By the previous considerations, we can assume $|V_2(f) \cap C_i| \leq 1$ for each $i \in \{1, 2\}$. This leaves only $|V|^2 + |V|$ many possibilities to check. \square

As mentioned before, Cabrera et al. [18] proved that UNIQUE RESPONSE ROMAN DOMINATION is NP-complete on bipartite graphs. For cobipartite graphs, this problem is polynomial-time solvable. For this result, we use the idea of 2-packings; we also refer to the discussion in [33]. Let $G = (V, E)$ be a graph. A set $S \subseteq V$ is called a 2-packing if the distance between two vertices in S is at least 3. Targhi et al. [34] presented a proof for $u_r(G) = \min\{2|S| + |V(G) \setminus N_G[S]| \mid S \text{ is a 2-packing}\}$. Analogously to this, we can prove that, for each graph $G = (V, E)$, there exists a bijection ψ_G between all 2-packings of the graph and all unique response Roman dominating functions. Here, for a 2-packing S and for $x \in V$,

$$\psi_G(S)(x) = \begin{cases} 0, & x \in N(S) \setminus S, \\ 1, & x \in V \setminus N(S), \\ 2, & x \in S. \end{cases}$$

This also yields that, for each unique response Roman dominating function f , $V_2(f)$ is a 2-packing. This is the main idea of our algorithm.

Lemma 2. UNIQUE RESPONSE ROMAN DOMINATION is polynomial-time solvable on cobipartite graphs. Furthermore, there are at most $\frac{|V|^2}{4}$ many 2-packings, or unique response Roman dominating functions, in $G = (V, E)$.

Proof. Let $G = (V, E)$ be a cobipartite graph, together with the partition into the two cliques $C_1, C_2 \subseteq V$. Two vertices from the same clique cannot be in a 2-packing at the same time. Therefore, from each clique, there can be at most one vertex in a 2-packing. Hence, there are at most $|C_1| \cdot |C_2|$ many 2-packings, or unique response Roman dominating functions. As C_1, C_2 is a partition of V , this leads to $|C_1| \cdot |C_2| = |C_1| \cdot |V \setminus C_1| = |C_1| \cdot (|V| - |C_1|)$ many 2-packings. This expression is maximal if $|C_1| = |C_2|$, which implies that there are at most $\frac{|V|^2}{4}$ many 2-packings, or unique response Roman dominating functions, in G . \square

As indicated in the proof, the upper bound $\frac{|V|^2}{4}$ is tight: consider the complement of the complete bipartite graph $K_{t,t}$, where both classes have the same size t . This is interesting, as there could be exponentially many unique response Roman dominating functions even on connected split graphs, which is quite a related class of graphs that we consider next.

3.4. Remarks on Enumeration of Unique Response Roman Dominating Functions in Split Graphs

Enumerating 2-packings (corresponding to unique response Roman dominating functions) in general graphs in time $\mathcal{O}(1.5399^n)$ was shown in [35].

Corollary 2. All unique response Roman dominating functions of a connected split graph of order n can be enumerated in time $\mathcal{O}(\sqrt[3]{3}^n) \subseteq \mathcal{O}(1.4423^n)$.

Proof. Non-trivial connected split graphs have no isolates. By Observation 3, we can hence use the enumeration algorithm for minimal Roman dominating functions on split graphs of order n from [29] which runs in time $\mathcal{O}(\sqrt[3]{3}^n)$. \square

$\Omega(\sqrt[3]{3}^{|V|})$ is even a tight lower bound for connected split graphs as shown next.

Theorem 11. *There is an infinite family of connected split graphs $G_t = (V_t, E_t)$ with $\Omega(\sqrt[3]{3}^{|V_t|})$ many unique response Roman dominating functions.*

Proof. We only need to consider the split graph family $G_t = (V_t, E_t)$ with $V_t := C_t \cup I_t$, $C_t := \{c_1, \dots, c_t\}$, $I_t := \{v_1, \dots, v_{2t}\}$ and

$$E_t := \binom{C_t}{2} \cup \{\{c_i, v_{2i-1}\}, \{c_i, v_{2i}\} \mid i \in \{1, \dots, t\}\}.$$

Clearly, $|V_t| = 3t$. By the arguments from Theorem 7, for each unique response Roman dominating function f on G_t with $|V_2(f) \cap C_t| = 1$, in fact $|V_2(f)| = 1$. There are t many such unique response Roman dominating functions. Let f be a unique response Roman dominating function with $V_2(f) \cap C_t = \emptyset$. In this situation, for each $i \in \{1, \dots, t\}$, there are three ways to dominate c_i :

- $f(c_i) = f(v_{2i-1}) = f(v_{2i}) = 1$;
- $f(c_i) = f(v_{2i}) = 0$ and $f(v_{2i-1}) = 2$;
- $f(c_i) = f(v_{2i-1}) = 0$ and $f(v_{2i}) = 2$.

Hence, there are $t + 3^t = \frac{|V|}{3} + \sqrt[3]{3}^{|V|}$ many unique response Roman dominating functions. \square

For general graphs, the best lower bound [35] is $\mathcal{O}(1.4970^n)$.

It should be mentioned that the polynomial-delay property of the Roman dominating function-enumeration algorithm from [29] is not inherited for enumerating the unique response Roman dominating function, as each minimal Roman dominating function is not a unique response Roman dominating function. Nonetheless, we present a sketch of a direct polynomial-delay branching enumeration algorithm for enumerating unique response Roman dominating functions that is optimal due to Theorem 11. For this purpose, we consider each 2-packing on a connected split graph and use the bijection between unique response Roman dominating functions and 2-packings. Our algorithm is a modification of the recursive backtracking algorithm of [36] and has polynomial delay with the same arguments. The algorithm of [36] is a general approach to enumerate all sets of $\mathcal{F} \subseteq 2^U$ for a universe U , where \mathcal{F} fulfills the following downward closure property: if $X \in \mathcal{F}$ and $Y \subseteq X$, then $Y \in \mathcal{F}$.

Theorem 12. *All unique response Roman dominating functions of a connected split graph of order n can be enumerated in time $\mathcal{O}(\sqrt[3]{3}^n)$ with polynomial delay and polynomial space.*

One of the rather surprising aspects of the proof is that we do not need any sophisticated measure-and-conquer methodology to analyze the run-time of the branching algorithm by a simple case analysis concerning the vertex degrees. In the core part of the algorithm, we decide if a vertex in the independent set is put into the related 2-packing or not; also cf. [36] concerning polynomial delay.

Proof. Let $G = (V, E)$ be a connected split graph with the partition $V = C \cup I$, where C is a clique and I is an independent set. With the arguments from Theorem 7, we can first enumerate each of the $|C|$ many 2-packings S with $S \cap C \neq \emptyset$. From now on, we branch on the vertices in I , deciding if they will be in the 2-packing S or not. If the vertex from I is not put into S , then we delete this vertex. If we put a vertex $v \in I$ into S , then we delete, for each $u \in N(v)$, the whole set $\{u\} \cup (N(u) \cap I)$. Furthermore, we delete vertices from C which have no neighbors in I . We always branch on a vertex $v \in I$ with the highest degree. If $\deg(v) \geq 3$, then by putting the vertex in S , we delete 4 vertices. This leads to a branching vector $(4, 1)$ (the branching value is below 1.3804). For $\deg(v) = 2$, we have to make a case distinction in the analysis: if v is the only neighbor in I for each of the neighbors of v , then in both cases, these vertices will be deleted after this branch. This implies the branching

vector (3,3) (the branching value is $\sqrt[3]{2} < \sqrt[3]{3}$). Otherwise, at least 4 vertices will be deleted after putting v into S . This would again lead to the branching vector (4,1).

Therefore, we can assume that each vertex in I has exactly one neighbor. Let c be this neighbor. If $|N(c) \cap I| \geq 3$, after putting v into S , we would delete at least 4 vertices and the branching would be again (4,1). For $N(c) \cap I = \{v\}$, we would delete c in each case. This would lead to the branching vector (2,2) (the branching value is $\sqrt{2} \leq \sqrt[3]{3}$). The remaining case would be $|N(c) \cap I| = 2$. This case is considered in the $\Omega(\sqrt[3]{3}^n)$ -example in Theorem 11. Its analysis implies the claimed running time of $\mathcal{O}^*(\sqrt[3]{3}^n)$. \square

3.5. Chordal Bipartite Graphs

Chordal bipartite graphs are bipartite graphs where each induced cycle has length 4. Before working on the NP-hardness, we prove the following lemma.

Lemma 3. *Let $G = (V, E)$ be a graph with $v, u \in V$ with $v \neq u$ and $N(v) = N(u)$. G is a chordal bipartite graph if and only if $G[V \setminus \{u\}]$ is one.*

Proof. Let $G' := G[V \setminus \{u\}]$. If G is a chordal bipartite graph, G' is clearly also one. Now assume G' is a chordal bipartite graph with the partition $V \setminus \{u\} = A \cup B$. W.l.o.g., let $v \in A$. As $N(u) = N(v)$, G is a bipartite graph with the partition $V = (A \cup \{u\}) \cup B$. So let u, x_1, \dots, x_ℓ, u be a cycle with $\ell > 4$. If $v \notin \{x_1, \dots, x_\ell\}$, then v, x_1, \dots, x_ℓ, v is a cycle in G' . Hence, there is a chord in u, x_1, \dots, x_ℓ, u . Assume there is an $i \in [\ell]$ with $x_i = v$. Clearly, $i \notin \{1, \ell\}$. Thus, u, x_1, v, x_{i+1}, u a cycle. So, G is chordal bipartite. \square

To prove the NP-hardness on PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION on chordal bipartite graphs, we use the problem ONE-IN-THREE 3SAT, with a reduction inspired by [32] (Theorem 2.2) for the NP-hardness of PERFECT CODE. However, our proof is much more sophisticated in order to deal with the weights of the perfect Roman dominating functions.

Let us define this problem more formally.

Problem name: ONE-IN-THREE 3SAT

Given: Set X of variables, set C of clauses over X with at most three literals.

Question: Is there a truth assignment such that each clause in C has exactly one true literal?

Theorem 13. PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION are NP-complete even on chordal bipartite graphs.

Proof. The NP-membership follows as in the general case by guess-and-check. The following NP-hardness proof is inspired by the NP-hardness proof for PERFECT CODE in [32]. Let $\mathcal{X} = \{x_1, \dots, x_n\}$ be a set of Boolean variables, and $\mathcal{C} = \{C_1, \dots, C_m\}$ be a set of clauses over \mathcal{X} . Define $k := 4mn + 2n(3m - 1) + 2m$, $M' := [n] \times [m]$, $M := M' \times [k]$, $\mathcal{A} := \{a_{i,j}, \bar{a}_{i,j} \mid (i, j) \in M'\}$, $\mathcal{B} := \{b_{i,j}, \bar{b}_{i,j} \mid (i, j) \in M'\}$. Further define for $(i, j) \in M'$:

- $\mathcal{E}_{i,j} := \{e_{i,j,p} \mid p \in [k]\};$
- $\mathcal{F}_{i,j} := \{f_{i,j,p} \mid p \in [k]\};$
- $\mathcal{U}_{i,j} := \{u_{i,j,p} \mid p \in [k]\}$ for $j \neq m$.

Also, let $\mathcal{W}_j := \{w_{j,p} \mid p \in [k]\}$ for $j \in [m]$. We group these sets into the following:

- $\mathcal{E} := \bigcup_{(i,j) \in M'} \mathcal{E}_{i,j};$
- $\mathcal{F} := \bigcup_{(i,j) \in M'} \mathcal{F}_{i,j};$
- $\mathcal{U} := \bigcup_{(i,j) \in M', j \neq m} \mathcal{U}_{i,j};$
- $\mathcal{W} := \bigcup_{j=1}^m \mathcal{W}_j;$
- $\mathcal{V} := \mathcal{E} \cup \mathcal{F} \cup \mathcal{U} \cup \mathcal{W};$

- $\mathcal{V}' := \{\mathcal{E}_{i,j}, \mathcal{F}_{i,j} \mid (i, j) \in M'\} \cup \{\mathcal{U}_{i,j} \mid (i, j) \in M', j \neq m\} \cup \{\mathcal{W}_j \mid j \in [k]\}$.

For $T \in \mathcal{V}'$, define

- $V_T := \{r[T]\} \cup \{s[T, p], q[T, p] \mid p \in [k]\}$;
- $E_T := \{\{r[T], t\}, \{q[T, p], s[T, p]\}, \{r[T], q[T, p]\} \mid t \in T, p \in [k]\}$.

Define $G := (V, E)$ with

$$V := \mathcal{A} \cup \mathcal{B} \cup \mathcal{V} \cup \left(\bigcup_{T \in \mathcal{V}'} V_T \right),$$

$$E := \{\{a_{i,j}, \bar{b}_{i,j}\}, \{a_{i,j}, e_{i,j,p}\}, \{\bar{a}_{i,j}, b_{i,j}\}, \{\bar{a}_{i,j}, e_{i,j,p}\} \mid i \in [n], j \in [m], p \in [k]\}$$

$$\cup \{\{f_{i,j,p}, \bar{b}_{i,j}\}, \{f_{i,j,p}, b_{i,j}\} \mid (i, j, p) \in M\} \cup \left(\bigcup_{T \in \mathcal{V}'} E_T \right)$$

$$\cup \{\{a_{i,j}, w_{j,p}\} \mid (i, j, p) \in M, x_i \in C_i\} \cup \{\{\bar{a}_{i,j}, w_{j,p}\} \mid (i, j, p) \in M, \bar{x}_i \in C_i\}$$

$$\cup \{\{u_{i,j,p}, b_{i,j}\}, \{u_{i,j,p}, \bar{b}_{i,j+1}\} \mid (i, j, p) \in M, j \neq m\}$$

$$\cup \{\{t_1, t_2\} \mid t_1 \in \mathcal{F} \cup \mathcal{U}, t_2 \in \mathcal{E} \cup \mathcal{W}\}.$$

Claim 4. G is a chordal bipartite graph.

Proof. G is bipartite with the partition $A \cup B = V$, where

$$A := \mathcal{A} \cup \mathcal{F} \cup \mathcal{U} \cup \{r[\mathcal{U}_{i,j}] \mid (i, j) \in M', j \neq m\}$$

$$\cup \{r[\mathcal{F}_{i,j}], q[\mathcal{E}_{i,j}, p], s[\mathcal{E}_{i,j}, p], q[\mathcal{W}_j, p], s[\mathcal{W}_j, p] \mid (i, j, p) \in M\},$$

$$B := \mathcal{B} \cup \mathcal{E} \cup \mathcal{W} \cup \{q[\mathcal{U}_{i,j}, p], s[\mathcal{U}_{i,j}, p] \mid (i, j, p) \in M, j \neq m\}$$

$$\cup \{r[\mathcal{E}_{i,j}], r[\mathcal{W}_j], q[\mathcal{F}_{i,j}, p], s[\mathcal{F}_{i,j}, p] \mid (i, j, p) \in M\}.$$

To make it easier to recognize each edge (besides the ones in E_T), we mention the vertex in A first when talking about edges; see Figure 4 for an illustration of the important aspects of the construction. Observe that for $t_1, t_2 \in T'$, with

$$T' \in \{T, \{s[T, p] \mid p \in [k]\}, \{q[T, p] \mid p \in [k]\} \mid T \in \mathcal{V}'\},$$

$N(t_1) = N(t_2)$. Hence, we can use Lemma 3 exhaustively and consider $G' := G[V']$ with $V' := \mathcal{A} \cup \mathcal{B} \cup \mathcal{V}_1 \cup \bigcup_{T \in \mathcal{V}'} V'_T$ with $\mathcal{V}_1 := \{e_{i,j,1}, f_{i,j,1}, u_{i,j',1}, w_{j,1} \mid (i, j) \in M', j' \in [m-1]\}$ and $V'_T = \{r[T], q[T, 1], s[T, 1]\}$. By construction, no vertex in $\bigcup_{T \in \mathcal{V}'} V'_T$ can be part of an induced cycle. Thus, we can also delete these vertices. The remaining graph is $G'' = G[V'']$ with $V'' := \mathcal{A} \cup \mathcal{B} \cup \mathcal{V}_1$.

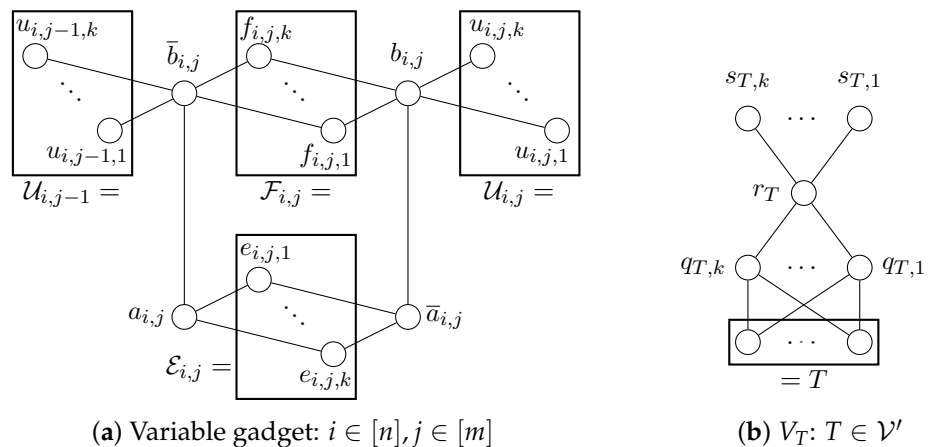


Figure 4. Constructions for Theorem 13.

Let $c = x_1, \dots, x_\ell, x_1$ be a chordless cycle in G'' with smallest length $\ell > 4$. If $a_{i,j}, \bar{b}_{i,j}$ are on c for $(i, j) \in M'$, then either $e_{i,j,1}$ or $w_{j,1}$ and either $f_{i,j,1}$ or $u_{i,j,1}$ are on c . This would give a chord. Hence, $a_{i,j}, \bar{b}_{i,j}$ cannot be on c at the same time. Analogously, this holds for $\bar{a}_{i,j}, b_{i,j}$. Therefore, if there are $a \in A$ and $b \in B$ on c , then the neighbors y_1, y_2 of a in c are in $\mathcal{E} \cup \mathcal{W}$ and the neighbors z_1, z_2 of b in c are in $\mathcal{F} \cup \mathcal{U}$. Hence, a, y_1, z_1, y_2, a is a cycle included in c . If there is an $a \in A$ on c , then its neighbors x, y on the cycle are in $\mathcal{E} \cup \mathcal{W}$. The neighbors of x, y on c are in $\mathcal{F} \cup \mathcal{U}$. Therefore, there also exists a chord between x and the neighbor of y on c and there is no vertex of \mathcal{A} on c . Analogously, this holds for B . So, c is a cycle in $G[\mathcal{V}]$, which is a complete bipartite graph, contradicting our assumption that c is chordless and of length $\ell > 4$. Thus, there exists no such c and G'' , as well as G' and G being chordal bipartite. \square

For each $T \in \mathcal{V}'$, $|V_T| = 2k + 1$. Further $|\mathcal{V}| = k(m + n(m - 1) + 2nm)$ and $|\mathcal{A}| = |\mathcal{B}| = 2nm$. Thus, the graph has

$$\begin{aligned} &|A| + |B| + |\mathcal{V}| + |\{z_T \mid T \in \mathcal{V}', z_T \in V_T\}| \\ &= 4nm + k(m + n(m - 1) + 2nm) + |\mathcal{V}'| \cdot (2k + 1) \\ &= 4nm + (3k + 1)(m + n(m - 1) + 2nm) \end{aligned}$$

many vertices. Since k is polynomial in n, m , the graph can be constructed in polynomial time. This leaves us to show that $(\mathcal{X}, \mathcal{C})$ is a yes-instance of the ONE-IN-THREE 3SAT if and only if there is a perfect Roman dominating function f on G with $\omega(f) \leq k$. To prove the NP-hardness for UNIQUE RESPONSE ROMAN DOMINATION, we show that if there is such an f , then there is even a unique response Roman dominating function g with $\omega(g) \leq k$.

Let ϕ be a solution of the ONE-IN-THREE 3SAT instance. Then, define $f \in \{0, 1, 2\}^V$ with

$$v \mapsto \begin{cases} 2, & \exists T \in \mathcal{V}' : v = r_T \\ 2, & \exists (i, j) \in M' : v \in \{a_{i,j}, b_{i,j}\} \wedge \phi(x_i) = 1 \\ 2, & \exists (i, j) \in M' : v \in \{\bar{a}_{i,j}, \bar{b}_{i,j}\} \wedge \phi(x_i) = 0 \\ 0, & \text{else} \end{cases}$$

Clearly, $\omega(f) = 2 \cdot (m + n(m - 1) + 2nm) + 4nm = k$. For each $T \in \mathcal{V}'$ and $p \in [k]$, $q[T, p], s[T, p]$ are only dominated by r_T . By the definition of f , for each $i \in [n]$, the vertices in $\{a_{i,j}, b_{i,j}, \bar{a}_{i,j}, \bar{b}_{i,j}, e_{i,j,p}, f_{i,j,p} \mid j \in [m], p \in [k]\} \cup \{u_{i,j,p} \mid j \in [m - 1], p \in [k]\}$ are dominated exactly once by either $a_{i,j}, b_{i,j}$, or $\bar{a}_{i,j}, \bar{b}_{i,j}$. Let $j \in [m]$ and $p \in [k]$. Since ϕ is a solution, there is exactly one literal which fulfills C_j . Hence, there is exactly one neighbor of $w_{j,p}$ in of value 2. Thus, f is a perfect Roman dominating function. Furthermore, it is even a unique response Roman dominating function.

Now, let f be a minimal perfect Roman dominating function with $\omega(f) = k$. Let $T \in \mathcal{V}'$ and $p \in [k]$. Since $s[T, p]$ is pendant, $f(s[T, p]) \neq 0$ implies $f(r[T]) \neq 2$. Otherwise, f would not be minimal (see Theorem 1, conditions 2 and 3). Hence, for each $p' \in [k]$, $f(s[T, p']) \neq 0$ and $\omega(f) > k$, as $r[T]$ has to be dominated. Thus, for each $T' \in \mathcal{V}'$ and $p' \in [k]$, $f(s[T', p']) = 0$ and $f(r[T']) = 2$. If there is a $v \in T$ with $f(v) = 2$, then $f(q[T, p']) \neq 0$ for each $p' \in [k]$ and $\omega(f) > k$. Hence, $T \cap V_2(f) = \emptyset$. For each $t_1, t_2 \in T$, $N(t_1) = N(t_2)$ and conditions 1 and 2 of Theorem 1 imply $f(t_1) = f(t_2)$. Hence, $N(\mathcal{A} \cup \mathcal{B}) \setminus (\mathcal{A} \cup \mathcal{B}) \subseteq \mathcal{V} \subseteq V_0(f)$ as otherwise $\omega(f) > k$. Since for each $i \in [n], j \in [m]$, $N[a_{i,j}] \setminus \mathcal{V} = \{a_{i,j}, \bar{b}_{i,j}\} = N[\bar{b}_{i,j}] \setminus \mathcal{V}$ and $N[\bar{a}_{i,j}] \setminus \mathcal{V} = \{\bar{a}_{i,j}, b_{i,j}\} = N[b_{i,j}] \setminus \mathcal{V}$ and we have $2nm$ such pairs, $\omega(f|_{\mathcal{A} \cup \mathcal{B}}) \leq 4nm$. Hence,

$$\omega(f|_{\mathcal{A} \cup \mathcal{B} \cup \{r[T] \mid T \in \mathcal{V}'\}}) \leq 4nm + 2(m + n(m - 1) + 2nm) = k.$$

Thus, equality has to hold and $\{q_{T,p} \mid T \in \mathcal{V}', p \in [k]\} \subseteq V_0(f)$. Further, for each $i \in [n], j \in [m]$, $f(a_{i,j}) + f(\bar{b}_{i,j}) = 2 = f(\bar{a}_{i,j}) + f(b_{i,j})$. If there exist $i \in [n], j \in [m]$ such that $f(a_{i,j}) = 1 = f(\bar{b}_{i,j})$, then $f(\bar{a}_{i,j}) = 2 = f(b_{i,j})$ to dominate $\mathcal{E}_{i,j} \cup \mathcal{F}_{i,j}$. Hence, $V_1(f) = \emptyset$. For

all $i \in [n], j \in [m]$, to dominate $\mathcal{E}_{i,j} \cup \mathcal{F}_{i,j}, |V_2(f) \cap \{a_{i,j}, \bar{a}_{i,j}\}| = 1 = |V_2(f) \cap \{b_{i,j}, \bar{b}_{i,j}\}|$ and $f(a_{i,j}) = f(b_{i,j}) \neq f(\bar{a}_{i,j}) = f(\bar{b}_{i,j})$. Let $i \in [n], j \in [m - 1]$. Since $u_{i,j,p}$ must be dominated exactly once, $f(\bar{b}_{i,j+1}) \neq f(b_{i,j})$. Also, $f(b_{i,j}), f(b_{i,j+1}), f(\bar{b}_{i,j+1}) \in \{0, 2\}$, together with $f(\bar{b}_{i,j+1}) \neq f(b_{i,j+1})$, implies $f(b_{i,j}) = f(b_{i,j+1})$. So, for each $i \in [n]$ and $j_1, j_2 \in [m]$, $f(a_{i,j_1}) = f(a_{i,j_2}) \neq f(\bar{a}_{i,j_1}) = f(\bar{a}_{i,j_2})$.

Define $X_1 := \{x_i \mid i \in [n], \forall j \in [m] : f(a_{i,j}) = 2\}$ and the assignment $\phi := \chi_{X_1} \in \{0, 1\}^{\mathcal{X}}$. Let $j \in [m]$. Since $j \in [m], w_{j,p}, q[\mathcal{W}_j, p] \in V_0(f)$ for all $p \in [k], N(w_{j,p}) \cap \mathcal{A} \cap V_2(f)$ is not empty. Hence, there is a true literal satisfying C_j . So, the given ONE-IN-THREE 3SAT instance is a yes-instance. This completes the proof of Theorem 13. \square

4. Parameterized Complexity

In this section, we consider the parameterized complexity aspects of PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION. More precisely, we consider the parameters *clique width* (cwd for short, originally defined in [37]), *treewidth* (tw for short, originally defined in several papers at about the same time but nowadays standard; see [38]), solution size, and its dual order minus solution size.

4.1. Clique Width and Treewidth

The clique width of a graph G ($\text{cwd}(G)$ for short) denotes the minimum number of labels needed to construct G with the following operations:

1. Create a graph of one isolate vertex.
2. Build the disjoint union of two labeled graphs.
3. Connect each vertex with label i with each vertex of label j by an edge.
4. Relabel vertices with label i to j .

Recall that $\text{LinEMSOL}(\tau_1)$ refers to the possibility to formulate an optimization problem using a linear function of the cardinalities of sets that realize the free set variables, while τ_1 restricts the vocabulary of the monadic second-order logic to vertex set variables, with adjacency as the only relation. We want to employ this logic next. First, we give a $\text{LinEMSOL}(\tau_1)$ formula for the optimization problems corresponding to PERFECT ROMAN DOMINATION and to UNIQUE RESPONSE ROMAN DOMINATION in order to show that these are linear-time FPT if parameterized by cwd , using now classical results from [39,40]. cwd is an interesting parameter since an FPT algorithm with cwd as the parameter implies FPT algorithms with respect to tw , nd , or vc as parameters. However, the algorithmic results of the previous section do not follow from these considerations, as it is known that even unit interval graphs have unbounded clique width; see [40].

Lemma 4. MINIMUM PERFECT ROMAN DOMINATION on the graph $G = (V, E)$ can be expressed as

$$\operatorname{argmin}_{X_1 \subseteq V, X_2 \subseteq V} \{|X_1| + 2|X_2| \mid \langle G, X_1, X_2 \rangle \models \theta(X_1, X_2)\}, \text{ with } \theta(X_1, X_2) \text{ equal to}$$

$$\forall v(X_1(v) \vee X_2(v) \vee \exists u[X_2(u) \wedge E(v, u) \wedge \forall w([X_2(w) \wedge E(w, v)] \Rightarrow u = w)]).$$

Proof. Let $G = (V, E)$ be a graph for which we want to determine the perfect Roman dominating function of the smallest weight. First, if f is some perfect Roman dominating function of G , then $\theta(V_1(f), V_2(f))$ is obviously true. Now, we want to show why $X_1, X_2 \subseteq V$, as a minimum solution would describe a perfect Roman dominating function. Define $f := \chi_{X_1} + 2\chi_{X_2}$. We can then see that for all $v \in V$, we have either $v \in X_1$, or $v \in X_2$, or there exists a $u \in X_2 \cap N(v)$ such that there is no other neighbor of v in X_2 . Therefore, we only need to show $X_1 \cap X_2 = \emptyset$. So, assume there is a $v \in X_1 \cap X_2$. Then, for $X'_1 := X_1 \setminus \{v\}$, (X'_1, X_2) is a smaller solution: v is still in X_2 , which does not change. Hence, f is a perfect Roman dominating function. \square

Theorem 14. *cwd*-PERFECT ROMAN DOMINATION is in FPT if we assume the clique decomposition given as part of the input. Furthermore, PERFECT ROMAN DOMINATION can be solved in linear time on distance-hereditary graphs.

Proof. As already noted, the first claim follows from Lemma 4. Golumbic and Rotics showed in [40] that distance-hereditary graphs have a clique width bounded by three, which gives the second claim. \square

Theorem 15. *tw*-PERFECT ROMAN DOMINATION is in FPT.

Proof. First, given a graph $G = (V, E)$, it is possible to compute an optimal tree decomposition in FPT-time, parameterized by the treewidth of the graph. Then, we can apply Courcelle’s theorem that allows to conclude our result, as we have provided an MSO formula for our property. \square

With similar techniques, we can also prove an analogous result for UNIQUE RESPONSE ROMAN DOMINATION.

Theorem 16. *cwd*-UNIQUE RESPONSE ROMAN DOMINATION is in FPT if we assume the clique decomposition given as part of the input.

Proof. For UNIQUE RESPONSE ROMAN DOMINATION, we have to make sure that there is no vertex of value 1 or 2 in the neighborhood of any vertex of value 2. We do this by adding the following part, connected with a logical conjunction to the formula $\theta(X_1, X_2)$ in Lemma 4: $\forall v(X_2(v) \Rightarrow \forall u[E(v, u) \Rightarrow (\neg X_1(u) \wedge X_2(u))])$. \square

Theorem 17. *tw*-UNIQUE RESPONSE ROMAN DOMINATION is in FPT.

4.2. Solution Size

Besides the structural parameters, also the solution size is an interesting, often-used parameter. We show next that $\omega(f)$ -UNIQUE RESPONSE ROMAN DOMINATION and $\omega(f)$ -PERFECT ROMAN DOMINATION are $W[1]$ -complete. We do this by using Steps-SHORT NON-DETERMINISTIC TURING MACHINE COMPUTATION (SNTMC) which is known to be $W[1]$ -complete (see [38,41,42]), as well as PERFECT CODE, parameterized by the solution size. So, we cannot expect FPT-results with this natural parameterization. We start our presentation with a $W[1]$ -membership proof, which is, in a mathematical sense, the most interesting proof. As a side product, we also obtain membership in XP. To this end, we need the following interesting structural characterization lemma that is proven with some double-counting arguments.

Lemma 5. Let $G = (V, E)$ be a graph and $f \in \{0, 1, 2\}^V$ be a function such that $N(u) \cap N(v) \cap V_0(f) = \emptyset$ for all $u, v \in V_2(f)$, $u \neq v$. Then, f is a perfect Roman dominating function if and only if

$$|V| = \left(\sum_{u \in V_1(f)} (1 - |N(u) \cap V_2(f)|) \right) + \left(\sum_{v \in V_2(f)} (|N[v]| - |N(v) \cap V_2(f)|) \right).$$

Proof. On $G = (V, E)$, let $f \in \{0, 1, 2\}^V$ be a function such that $N(u) \cap N(v) \cap V_0(f) = \emptyset$ for all $u, v \in V_2(f)$, $u \neq v$. First, consider

$$\begin{aligned} \sum_{u \in V_1(f)} (1 - |N(u) \cap V_2(f)|) &= |V_1(f)| - \sum_{u \in V_1(f)} \sum_{v \in V_2(f)} \chi_E(\{u, v\}) \\ &= |V_1(f)| - \sum_{v \in V_2(f)} |N(v) \cap V_1(f)|. \end{aligned}$$

Furthermore, we can re-write the second sum as follows:

$$\sum_{v \in V_2(f)} (|N[v]| - |N(v) \cap V_2(f)|) = |V_2(f)| + \sum_{v \in V_2(f)} |N(v) \setminus V_2(f)|.$$

Putting these thoughts together, we observe

$$\begin{aligned} & \left(\sum_{u \in V_1(f)} (1 - |N(u) \cap V_2(f)|) \right) + \left(\sum_{v \in V_2(f)} (|N[v]| - |N(v) \cap V_2(f)|) \right) \\ &= |V_1(f)| + |V_2(f)| + \sum_{v \in V_2(f)} (|N(v) \setminus V_2(f)| - |N(v) \cap V_1(f)|) \\ &= |V_1(f)| + |V_2(f)| + \sum_{v \in V_2(f)} |N(v) \setminus (V_1(f) \cup V_2(f))| \\ &= |V_1(f)| + |V_2(f)| + \sum_{v \in V_2(f)} |N(v) \cap V_0(f)|. \end{aligned}$$

By the assumption on f , $\{N(v) \cap V_0(f) \mid v \in V_2(f)\}$ is a collection of pairwise disjoint sets, i.e., $\sum_{v \in V_2(f)} |N(v) \cap V_0(f)| = |\bigcup_{v \in V_2(f)} (N(v) \cap V_0(f))| \leq |V_0(f)|$. This leads us to

$$|V| \geq \left(\sum_{u \in V_1(f)} (1 - |N(u) \cap V_2(f)|) \right) + \left(\sum_{v \in V_2(f)} (|N[v]| - |N(v) \cap V_2(f)|) \right).$$

Equality holds if and only if $|\bigcup_{v \in V_2(f)} (N(v) \cap V_0(f))| = |V_0(f)|$. Since $(N(v) \cap V_0(f))$ are disjoint and subsets of $V_0(f)$, this is equivalent to the fact that f is a Roman dominating function. With the assumption on f , f would even be a perfect Roman dominating function. \square

Theorem 18. $\omega(f)$ -PERFECT ROMAN DOMINATION belongs to $W[1]$.

Proof. (Sketch) We will use Steps-SNTMC. So let $G = (V, E)$ be a graph of order n and $k \in \mathbb{N}$. The reduction will compute a Turing machine M with a tape alphabet containing $V \cup \{\zeta\}$, with $\zeta \notin V$. In its internal memory, M stores a table T that contains, for $u, v \in V$, $u \neq v$, either $N(u) \cap N(v)$ (if $|N(u) \cap N(v)| \leq k$) or ∞ . Moreover, the reduction machine also computes addition and subtraction tables for numbers up to n and the alphabet contains also these numbers so that M can perform basic arithmetic, like reading two number symbols from the tape and printing the result, in constant time. Finally, M has access to a table of the degrees of all vertices in its internal memory. The Turing machine M will work in five phases as follows:

1. Guess the vertices in $V_2(f)$ and then $V_1(f)$ and write them (as symbols) on the tape. We separate the vertices via the symbol ζ . This takes $\leq k + 1$ steps.
2. Also, in time $\mathcal{O}(k)$, M can check if twice the number of V_2 -symbols (left of the separator ζ) plus the number of V_1 -symbols (right to ζ) is at most k . If not, M stops in a non-final state.
3. Check if we enumerated any vertex in $V_1(f) \cup V_2(f)$ twice (in $\mathcal{O}(k^2)$ steps). If this is the case, then the machine stops in a non-final state. Otherwise, we know now that the V_1 -symbols together with the V_2 -symbols that we guess do define a function f of weight $\omega(f) \leq k$.
4. Check for each pair of vertices $u, v \in V_2(f)$ ($u \neq v$) if $N(u) \cap N(v) \subseteq V_1(f) \cup V_2(f)$. This condition is clearly necessary if f should be a perfect Roman dominating function. For this check, we need table T . If $T[u, v] = \infty$, $N(u) \cap N(v) \subseteq V_1(f) \cup V_2(f)$ is impossible and we can stop in a non-final state. Otherwise, M checks element by element of $N(u) \cap N(v)$ if it is in $V_1(f) \cup V_2(f)$. Altogether, this needs at most $\mathcal{O}(k^3)$ many steps.

- For each $v \in V_1(f) \cup V_2(f)$, M computes $|N(v) \cap V_2(f)|$ (at most $\mathcal{O}(k^2)$ steps). Use these numbers to compute (with basic arithmetics)

$$n' := \left(\sum_{u \in V_1(f)} (1 - |N(u) \cap V_2(f)|) \right) + \left(\sum_{v \in V_2(f)} (|N[v]| - |N(v) \cap V_2(f)|) \right)$$

Finally, M uses Lemma 5 to check if f is a perfect Roman dominating function, i.e., if $n = n'$. \square

Using a construction simpler than the previous one for $W[1]$ -membership together with a relatively straightforward reduction from PERFECT CODE for the $W[1]$ -hardness that basically introduces $2k$ twins per vertex, we can prove the following.

Theorem 19. $\omega(f)$ -UNIQUE RESPONSE ROMAN DOMINATION is $W[1]$ -complete.

Proof. For the $W[1]$ -hardness proof, we use PERFECT CODE. Let $G = (V, E)$ be a graph and $k \in \mathbb{N}$. Define $V_i := \{v_i \mid v \in V\}$ for $i \in [2k + 1]$ and $\tilde{G} := (\tilde{V}, \tilde{E})$ with

$$\tilde{V} := \bigcup_{i=1}^{2k+1} V_i, \quad \tilde{E} := \{ \{v_i, u_j\} \mid i, j \in [2k + 1], v, u \in V, v \in N[u] \}.$$

This graph has $(2k + 1) \cdot |V|$ many vertices.

Claim 5. G has a perfect code D of size $|D| \leq k$ if and only if \tilde{G} has a perfect Roman dominating function f with $\omega(f) \leq 2k$.

Proof. Let $D \subseteq V$ be a perfect code. Define $D_1 := \{v_1 \mid v \in D\}$, $f_D := 2 \cdot \chi_{D_1}$. Assume there exists a $v_i \in V_0(f_D)$ with $N_{\tilde{G}}(v_i) \cap D_1 = N_{\tilde{G}}(v_i) \cap V_2(f_D) = \emptyset$. Thus, v_i has no neighbor in D_1 and $N_G[v] \cap D = \emptyset$. Hence, D is no dominating set, which is a contradiction. Assume there is a $v_i \in V_0(f_D)$ with $|N_{\tilde{G}}(v_i) \cap D_1| = |N_{\tilde{G}}(v_i) \cap V_2(f_D)| \geq 2$. In this case, $|N_G[v] \cap D| = 2$, contradicting the fact that D is a perfect code. If there exists a $v_1 \in D_1 = V_2(f_D)$ with $N_{\tilde{G}}(v_1) \cap V_2(f_D) \neq \emptyset$, then $v \in D$ would have a neighbor in D , and D would not be a perfect code in G . As even $V_1(f_D) = \emptyset$, f_D is also a unique response Roman dominating function with $\omega(f) = 2 \cdot |D| \leq 2k$.

Let $f \in \{0, 1, 2\}^V$ be a unique response Roman dominating function with $\omega(f) \leq 2k$ and $v \in V$. Since $C_v := \{v_i \in i \in [2k + 1]\}$ is a clique, we can assume $|C_v \cap V_2(f)| \leq 1$. Further, for all $v_i, v_j \in C_v$, $N_{\tilde{G}}[v_i] = N_{\tilde{G}}[v_j]$. Therefore, we can assume, without loss of generality, that if $C_v \cap V_2(f) \neq \emptyset$, then $f(v_1) = 2$. Assume there is a $v_i \in V_1(f)$. Thus, $N_{\tilde{G}}(v_i) \cap V_2(f) = \emptyset$. Since C_v is a clique of true twins, $N_{\tilde{G}}(v_j) \cap V_2(f) = \emptyset$ and $f(v_j) = 1$ for each $j \in [2k + 1]$. Hence, $|V_1(f)| > 2k$. This contradicts $\omega(f) \leq 2k$. Therefore, $V_1(f) = \emptyset$.

Define $D := \{v \in V \mid v_1 \in V_2(f)\}$. If there is a $v \in V$ with $N[v] \cap D = \emptyset$, f would not be a Roman dominating function, as v_1 is not dominated. The existence of a $v \in V$ with $|N[v] \cap D| \geq 2$ would contradict the fact that f is unique response Roman dominating function, as either $v_1 \in V_2(f)$ and v_1 has a neighbor in $V_2(f)$, or v_1 has two neighbors in $V_2(f)$. In total, D is a perfect code. \square

Now, we will sketch the $W[1]$ -membership proof by using Steps-STMC. To this end, let $G = (V, E)$ be a graph of order n and $k \in \mathbb{N}$. The Turing machine will work in four phases as follows, assuming it was provided with some basic arithmetic capabilities by the reduction machine as detailed in the proof of Theorem 18:

- Guess the vertices in $V_2(f)$ and then $V_1(f)$. We split the vertices via a special symbol ζ (this takes at most $2k + 1$ steps).

2. Check if vertices in $V_2(f)$ have distance 2 to the vertices in $V_1(f)$ and distance 3 to the vertices in $V_2(f)$ (in at most $\frac{k(k+1)}{2}$ steps). If this is the case, then the machine stops in a non-final state.
3. Check if we enumerated vertices in $V_1(f)$ twice (in at most $\frac{k(k+1)}{2}$ steps). If this is the case, then the machine stops in a non-final state.
4. If $|V_1(f)| + \sum_{v \in V_2(f)} |N[v]| = |V|$, then f is a minimal unique response Roman dominating function (as each vertex is only dominated once). This can be computed in k steps.

This concludes the proof of Theorem 19. \square

The previous $W[1]$ -hardness result could also be used to show that $\omega(f)$ -PERFECT ROMAN DOMINATION is $W[1]$ -complete. We can show this type of result even on bipartite graphs.

Theorem 20. $\omega(f)$ -PERFECT ROMAN DOMINATION is $W[1]$ -complete even on bipartite graphs.

Proof. For membership, we refer to Theorem 18. For the hardness, we use PERFECT CODE. Let $G = (V, E)$ be a graph and $k \in \mathbb{N}$. Define $\tilde{G} := (\tilde{V}, \tilde{E})$ with $V' := \{v'_i \mid v \in V, i \in [2k + 3]\}, Y := \{y_1, \dots, y_{2k+3}\}, Z := \{z_1, \dots, z_{2k+3}\}$, giving

$$\begin{aligned} \tilde{V} &:= V \cup V' \cup \{x\} \cup Y \cup Z, \\ \tilde{E} &:= \{\{v, u'_i\} \mid \{v, u\} \in E, i \in [2k + 3]\} \cup \{\{y_i, x\}, \{z_i, x\} \mid i \in [2k + 3]\} \cup \\ &\quad \{\{v, x\} \mid v \in V\} \cup \{\{y_i, v'_i\} \mid v \in V, i \in [2k + 3]\}. \end{aligned}$$

\tilde{G} is visualized in Figure 5. \tilde{G} is bipartite with the classes $A := V \cup Y \cup Z$ and $B := \{x\} \cup V'$ (the edges are written in such a way that the vertices from A are mentioned first). Furthermore, \tilde{G} has $(2k + 3) \cdot (|V| + 2) + |V| + 1$ many vertices. This leaves us to show that G has a perfect code of size at most k if and only if \tilde{G} has a perfect Roman dominating function with weight at most $2k + 2$.

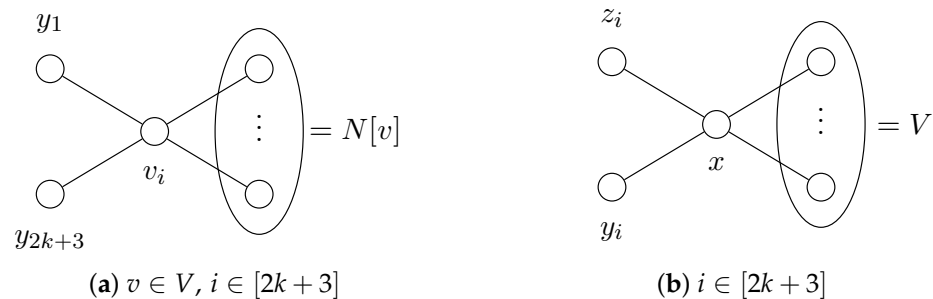


Figure 5. Construction for Theorem 20.

Assume there is a perfect code $D \subseteq V$ with $|D| \leq k$. Then, define $f := 2 \cdot \chi_{D \cup \{x\}}$. $V \cup \{x\} \cup Y \cup Z$ is dominated by x . If there were a $v'_i \in V'$ which is not dominated by f , then v would not be dominated in G by D . Therefore, f is a Roman dominating function. Further $N(Y \cup Z) \cap V_2(f) = \{x\}$. For any $i \in [2k + 3]$ and $v \in V$, v'_i has only one neighbor in $V_2(f)$, as $Y \subseteq V_0(f)$ and, otherwise, v would be dominated twice by D . Thus, f is a perfect Roman dominating function with $\omega(f) = 2 \cdot |D| + 2 \leq 2k + 2$.

Let $f \in \{0, 1, 2\}^{\tilde{V}}$ be a minimal perfect Roman dominating function with $\omega(f) \leq 2k + 2$. Assume $f(x) \neq 2$. As $N(z_i) = \{x\}$ for each $i \in [2k + 3]$, $Z \cap V_0(f) = \emptyset$. So, $2k + 2 < 2k + 3 \leq \omega(f)$. Hence, $f(x) = 2$. Since $N(v'_i) \subseteq N(x)$ for each $i \in [2k]$ and $v \in V$, $V_2(f) \cap V' = \emptyset$. Otherwise, f is not minimal. Assume there is an $i \in [2k + 3]$ with $f(y_i) = 2$. As f is minimal, there must exist a private neighbor in $V_0(f)$. As $f(x) = 2$, there exists a $v \in V$ such that v'_i is the private neighbor. Thus, $(N(v'_i) \setminus \{y_j\}) \cap V_2(f) =$

$N[v] \cap V_2(f) = (N(v'_j) \setminus \{y_j\}) \cap V_2(f) = \emptyset$ for $j \in [2k + 3]$. Thus, for each $j \in [2k + 3]$, $f(v'_j) \neq 0$ or $f(y_j) = 2$. Hence, $2k + 2 < 2k + 3 + 2 \leq \omega(f)$. So, $Y \subseteq V_0(f)$.

Define $D = V_2(f) \cap V$. Clearly, $|D| \leq k$. Assume that D is not a dominating set. Then, there is a $v \in V$ that is not dominated. Therefore, $N(v'_i) \cap V_2(f) = \emptyset$ and $f(v_i) = 1$ for each $j \in [2k + 3]$. Hence, $\omega(f) \geq 2k + 3$. Therefore, D is a dominating set. Assume D is not perfect. So, there is a $v \in V$ with $2 \leq |N[v] \cap D| = |V_2(f) \cap N(v'_i)|$ for each $i \in [2k + 1]$. So, $\omega(f) \geq 2k + 3$. Hence, D is a perfect code. \square

4.3. Dual Parameterization

Different from PERFECT CODE, it makes sense to consider the dual parameterization $(|V| - k)$, as it is even NP-hard to find out if the graph has any perfect code (see [24]). So, the problem is para-NP-hard for the dual parameterization. This is not the case for PERFECT ROMAN DOMINATION and UNIQUE RESPONSE ROMAN DOMINATION since the constant 1 function is always a perfect Roman dominating function/unique response Roman dominating function. Therefore, these problems are trivially solvable if $n - k = 0$. We can even see this parameter as a *distance from triviality* parameter in the sense of [43].

Theorem 21. $(n - k)$ -PERFECT ROMAN DOMINATION, $(n - k)$ -UNIQUE RESPONSE ROMAN DOMINATION \in FPT.

Proof. Let $G = (V, E)$ be a graph and $f \in \{0, 1, 2\}^V$. Then,

$$\begin{aligned} n - \omega(f) &= \sum_{v \in V} (1 - f(v)) \\ &= \left(\sum_{v \in V_0(f)} (1 - 0) \right) + \left(\sum_{v \in V_1(f)} (1 - 1) \right) + \left(\sum_{v \in V_2(f)} (1 - 2) \right) \\ &= |V_0(f)| - |V_2(f)|. \end{aligned}$$

We will use this equality to bound the diameter and degree by $n - k$.

Assume there is a vertex v with $\deg(v) \geq n - k + 1$. Then, $f_v := \chi_V + \chi_{\{v\}} - \chi_{N(v)}$ is clearly a unique response Roman dominating function, and therefore a perfect Roman dominating function with

$$n - \omega(f_v) = |V_0(f_v)| - |V_2(f_v)| = \deg(v) - 1 \geq n - k.$$

Hence, $\omega(f_v) \leq k$, and we observe a yes-instance and can hence use the consideration of a vertex of maximum degree as a reduction rule.

Assume there is a shortest path $p = v_1, \dots, v_{3 \cdot (n-k)}$ of length $3 \cdot (n - k)$. Define $f_p := \chi_V + \sum_{i=0}^{n-k-1} (\chi_{\{v_{3i+2}\}} - \chi_{N(v_{3i+2})})$. Clearly, each vertex in $V_0(f)$ has at least one neighbor in $V_2(f)$. Assume there exists a vertex $u \in V_0(f)$ with $v_{3i+2}, v_{3j+2} \in V_2(f) \cap N(u)$ with $i \neq j$ (without loss of generality, $i < j$). This would imply that $v_1, \dots, v_{3i+2}, u, v_{3j+2}, \dots, v_{3 \cdot (n-k)}$ is a shorter path. Hence, f_p is a perfect Roman dominating function. Analogously, $V_2(f_p)$ is an independent set. Therefore, f_p is even a unique response Roman dominating function. Further,

$$n - \omega(f_p) = |V_0(f_p)| - |V_2(f_p)| \geq 2 \cdot (n - k) - n - k \geq n - k.$$

So, $\omega(f) \leq k$, and we observe a yes-instance and can hence use the consideration of the graph diameter as a reduction rule.

As the maximum degree and diameter are bounded in $n - k$, the number of vertices is bounded by the Moore bound, giving $\mathcal{O}\left((n - k)^{3 \cdot (n-k)}\right)$ as the kernel size; see [44]. \square

5. Final Remarks

We have enhanced the understanding of the combinatorial structure, the algorithmic nature, and the complexity of PERFECT ROMAN DOMINATION and UNIQUE RESPONSE

ROMAN DOMINATION in many ways: on certain graph classes, with respect to several parameterizations, and concerning enumeration. But many aspects are still open for exploration. For instance, we did not touch the question of kernel sizes in any of our FPT results. Also, the fact that the ‘bijection approach’ that we took for enumeration necessitates parsimonious reductions, and this could be used towards investigating counting complexities, again, an untouched ground in this area.

Author Contributions: Conceptualization and methodology, K.M. and H.F.; proof writing and draft preparation, K.M.; writing—final version, K.M. and H.F.; visualization, K.M.; supervision, H.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: There is no data involved in our article.

Acknowledgments: We are grateful to reviewers of the conference version who helped improve this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Dreyer, P.A. Applications and Variations of Domination in Graphs. Ph.D. Thesis, Rutgers University, New Brunswick, NJ, USA, 2000.
- Fernau, H. Roman Domination: A parameterized perspective. *Int. J. Comput. Math.* **2008**, *85*, 25–38. [\[CrossRef\]](#)
- Liedloff, M.; Kloks, T.; Liu, J.; Peng, S.L. Efficient algorithms for Roman domination on some classes of graphs. *Discret. Appl. Math.* **2008**, *156*, 3400–3415. [\[CrossRef\]](#)
- Liu, C.H.; Chang, G.J. Roman domination on strongly chordal graphs. *J. Comb. Optim.* **2013**, *26*, 608–619. [\[CrossRef\]](#)
- Peng, S.L.; Tsai, Y.H. Roman Domination on Graphs of Bounded Treewidth. In Proceedings of the 24th Workshop on Combinatorial Mathematics and Computation Theory, Nantou, Taiwan, 27–28 April 2007; pp. 128–131.
- Shang, W.; Wang, X.; Hu, X. Roman domination and its Variants in Unit Disk Graphs. *Discret. Math. Algorithms Appl.* **2010**, *2*, 99–106. [\[CrossRef\]](#)
- Stewart, I. Defend the Roman Empire. *Sci. Am.* **1999**, *281*, 136,137,139. [\[CrossRef\]](#)
- Chellali, M.; Haynes, T.W.; Hedetniemi, S.T.; McRae, A.A. Roman $\{2\}$ -domination. *Discret. Appl. Math.* **2016**, *204*, 22–28. [\[CrossRef\]](#)
- Abdollahzadeh Ahangar, H.; Chellali, M.; Sheikholeslami, S.M. On the double Roman domination in graphs. *Discret. Appl. Math.* **2017**, *232*, 1–7. [\[CrossRef\]](#)
- Banerjee, S.; Henning, M.A.; Pradhan, D. Algorithmic results on double Roman domination in graphs. *J. Comb. Optim.* **2020**, *39*, 90–114. [\[CrossRef\]](#)
- Beeler, R.A.; Haynes, T.W.; Hedetniemi, S.T. Double Roman domination. *Discret. Appl. Math.* **2016**, *211*, 23–29. [\[CrossRef\]](#)
- Hennings, M.; Yeo, A. *Total Domination in Graphs*; Springer: Berlin/Heidelberg, Germany, 2013.
- Haynes, T.W.; Hedetniemi, S.; Henning, M.A. (Eds.) *Topics in Domination in Graphs*; Developments in Mathematics Series; Springer: Berlin/Heidelberg, Germany, 2020; Volume 64.
- Haynes, T.W.; Hedetniemi, S.T.; Henning, M.A. *Structures of Domination in Graphs*; Developments in Mathematics Series; Springer: Berlin/Heidelberg, Germany, 2021; Volume 66.
- Henning, M.A.; Klostermeyer, W.F.; MacGillivray, G. Perfect Roman domination in trees. *Discret. Appl. Math.* **2018**, *236*, 235–245. [\[CrossRef\]](#)
- Rubalcaba, R.R.; Slater, P.J. Roman dominating influence parameters. *Discret. Math.* **2007**, *307*, 3194–3200. [\[CrossRef\]](#)
- Mann, K.; Fernau, H. Perfect Roman Domination: Aspects of Enumeration and Parameterization. In Proceedings of the Combinatorial Algorithms (Proceeding 35th International Workshop on Combinatorial Algorithms IWOCOA), Ischia, Italy, 1–3 July 2024; Rescigno, A.A., Vaccaro, U., Eds.; Springer: Berlin/Heidelberg, Germany, 2024; LNCS volume 14764, pp. 354–368.
- Cabrera Martínez, A.; Puertas, M.; Rodríguez-Velázquez, J. On the 2-Packing Differential of a Graph. *Results Math.* **2021**, *76*, 157. [\[CrossRef\]](#)
- Banerjee, S.; Chaudhary, J.; Pradhan, D. Unique Response Roman Domination: Complexity and Algorithms. *Algorithmica* **2023**, *85*, 3889–3927. [\[CrossRef\]](#)
- Banerjee, S.; Keil, J.M.; Pradhan, D. Perfect Roman domination in graphs. *Theor. Comput. Sci.* **2019**, *796*, 1–21. [\[CrossRef\]](#)
- Fernau, H.; Golovach, P.A.; Sagot, M. Algorithmic Enumeration: Output-sensitive, Input-Sensitive, Parameterized, Approximative (Dagstuhl Seminar 18421). *Dagstuhl Rep.* **2018**, *8*, 63–86.
- Wasa, K. Enumeration of Enumeration Algorithms. *arXiv* **2016**, arXiv:1605.05102. [\[CrossRef\]](#)
- Abu-Khzam, F.N.; Fernau, H.; Mann, K. Minimal Roman Dominating Functions: Extensions and Enumeration. *Algorithmica* **2024**, *86*, 1862–1887. [\[CrossRef\]](#)

24. Kratochvíl, J.; Krivánek, M. On the Computational Complexity of Codes in Graphs. In Proceedings of the Mathematical Foundations of Computer Science (MFCS), Carlsbad, Czech Republic, 29 August–2 September 1988; Chytil, M., Janiga, L., Koubek, V., Eds.; Springer: Berlin/Heidelberg, Germany, 1988; LNCS volume 324, pp. 396–404.
25. Casel, K.; Fernau, H.; Ghadikolaie, M.K.; Monnot, J.; Sikora, F. Abundant Extensions. In Proceedings of the Algorithms and Complexity—12th International Conference (CIAC), Virtual, 10–12 May 2021; Calamoneri, T., Corò, F., Eds.; Springer: Berlin/Heidelberg, Germany, 2021; LNCS volume 12701, pp. 3–17. [[CrossRef](#)]
26. Chellali, M.; Haynes, T.W.; Hedetniemi, S.M.; Hedetniemi, S.T.; McRae, A.A. A Roman Domination Chain. *Graphs Comb.* **2016**, *32*, 79–92. [[CrossRef](#)]
27. Döcker, J. Monotone 3-Sat-(2,2) is NP-complete. *arXiv* **2019**, arXiv:1912.08032. [[CrossRef](#)]
28. Strozecki, Y. Enumeration Complexity. *EATCS Bull.* **2019**, *129*.
29. Abu-Khzam, F.N.; Fernau, H.; Mann, K. Roman Census: Enumerating and Counting Roman Dominating Functions on Graph Classes. In Proceedings of the 48th International Symposium on Mathematical Foundations of Computer Science (MFCS), Bordeaux, France, 28 August–1 September 2023; Leroux, J., Lombardy, S., Peleg, D., Eds.; Leibniz International Proceedings in Informatics (LIPIcs); Schloss Dagstuhl—Leibniz-Zentrum für Informatik: Wadern, Germany, 2023; Volume 272, pp. 1–15.
30. Golumbic, M.C. Interval Graphs. In *Annals of Discrete Mathematics*; Elsevier: Amsterdam, The Netherlands, 2004; Volume 57, Chapter 8, pp. 171–202.
31. Fellows, M.R.; Hoover, M.N. Perfect domination. *Australas. J. Comb.* **1991**, *3*, 141–150.
32. Lu, C.L.; Tang, C.Y. Weighted efficient domination problem on some perfect graphs. *Discret. Appl. Math.* **2002**, *117*, 163–182. [[CrossRef](#)]
33. Trejo-Sánchez, J.A.; Madera-Ramírez, F.A.; Fernández-Zepeda, J.A.; López-Martínez, J.L.; Flores-Lamas, A. A fast approximation algorithm for the maximum 2-packing set problem on planar graphs. *Optim. Lett.* **2023**, *17*, 1435–1454. [[CrossRef](#)]
34. Targhi, E.E.; Rad, N.J.; Volkmann, L. Unique response Roman domination in graphs. *Discret. Appl. Math.* **2011**, *159*, 1110–1117. [[CrossRef](#)]
35. Junosza-Szaniawski, K.; Rzażewski, P. On the number of 2-packings in a connected graph. *Discret. Math.* **2012**, *312*, 3444–3450. [[CrossRef](#)]
36. Marino, A. *Analysis and Enumeration. Algorithms for Biological Graphs*; Atlantis Studies in Computing; Atlantis Press: Paris, France, 2015; Volume 6.
37. Courcelle, B.; Engelfriet, J.; Rozenberg, G. Handle-rewriting hypergraph grammars. *J. Comput. Syst. Sci.* **1993**, *46*, 218–270. [[CrossRef](#)]
38. Downey, R.G.; Fellows, M.R. *Fundamentals of Parameterized Complexity*; Texts in Computer Science; Springer: Berlin/Heidelberg, Germany, 2013.
39. Courcelle, B.; Makowsky, J.; Rotics, U. Linear time solvable optimization problems on Graphs of Bounded Clique-Width. *Theory Comput. Syst.* **2000**, *33*, 125–150. [[CrossRef](#)]
40. Golumbic, M.C.; Rotics, U. On the Clique-Width of Some Perfect Graph Classes. *Int. J. Found. Comput. Sci.* **2000**, *11*, 423–443. [[CrossRef](#)]
41. Cattaneo, D.; Perdrix, S. The Parameterized Complexity of Domination-Type Problems and Application to Linear Codes. In Proceedings of the Theory and Applications of Models of Computation (TAMC), Singapore, 11–13 April 2014; Gopal, T.V., Agrawal, M., Li, A., Cooper, S.B., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; LNCS volume 8402, pp. 86–103.
42. Cesati, M. The Turing way to parameterized complexity. *J. Comput. Syst. Sci.* **2003**, *67*, 654–685. [[CrossRef](#)]
43. Guo, J.; Hüffner, F.; Niedermeier, R. A structural view on parameterizing problems: Distance from triviality. In Proceedings of the International Workshop on Parameterized and Exact Computation IWPEC 2004, Bergen, Norway, 14–17 September 2004; Downey, R., Fellows, M., Dehne, F., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; LNCS volume 3162, pp. 162–173.
44. Fernau, H. Extremal Kernelization: A Commemorative Paper. In Proceedings of the Combinatorial Algorithms, IWOCA 2017, Newcastle, Australia, 17–21 July 2017; Brankovic, L., Ryan, J., Smyth, W.F., Eds.; Springer: Berlin/Heidelberg, Germany, 2018; LNCS volume 10765, pp. 24–36.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.