Roman Hitting Functions

Henning Fernau
Fachbereich IV, Informatikwissenschaften, Universität Trier, Germany

Kevin Mann
Fachbereich IV, Informatikwissenschaften, Universität Trier, Germany

Abstract

Roman domination formalizes a military strategy going back to Constantine the Great. Here, armies are placed in different regions. A region is secured if there is at least one army in this region or there are two armies in one neighbored region. This simple strategy can be easily translated into a graph-theoretic question. The placement of armies is described by a function which maps each vertex to 0, 1 or 2. Such a function is called *Roman dominating* if each vertex with value 0 has a neighbor with value 2.

Roman domination is one of few examples where the related (so-called) extension problem is polynomial-time solvable even if the original decision problem is NP-complete. This is interesting as it allows to establish polynomial-delay enumeration algorithms for listing minimal Roman dominating functions, while it is open for more than four decades if all minimal dominating sets of a graph or (equivalently) if all hitting sets of a hypergraph can be enumerated with polynomial delay, or even in output-polynomial time. To find the reason why this is the case, we combine the idea of hitting set with the idea of Roman domination. We hence obtain and study a new problem, called Roman Hitting Function, generalizing Roman Domination towards hypergraphs. This allows us to delineate the frontier of polynomial-delay enumerability.

Our main focus is on the extension version of this problem, as this was the key problem when coping with Roman domination functions. While doing this, we find some conditions under which the Extension Roman Hitting Function problem is NP-complete. We then use parameterized complexity as a tool to get a better understanding of why Extension Roman Hitting Function behaves in this way. From an alternative perspective, we can say that we use the idea of parameterization to study the question what makes certain enumeration problems that difficult.

Also, we discuss another generalization of Extension Roman Domination, where both a lower and an upper bound on the sought minimal Roman domination function is provided. The additional upper bound makes the problem hard (again), and the applied parameterized complexity analysis (only) provides hardness results.

Also from the viewpoint of Parameterized Complexity, the studies on extension problems are quite interesting as they provide more and more examples of parameterized problems complete for W[3], a complexity class with only very few natural members known five years ago.

2012 ACM Subject Classification Theory of computation → Problems, reductions and completeness; Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases enumeration problems, polynomial delay, domination problems, hitting set, Roman domination

Digital Object Identifier 10.4230/LIPIcs.IPEC.2024.24

1 Introduction

For more than four decades, the question if all minimal hitting sets can be enumerated with polynomial delay is an open question. This Transversal Hypergraph Problem is equivalent to the question if all minimal dominating sets can be enumerated with polynomial delay. From the point of view of applications, it is quite important to find an affirmative answer: no user likes to wait "forever" to see the next solution, or to get to know that no further solution exist. In order to explore this question, the problem of enumerating minimal

19th International Symposium on Parameterized and Exact Computation (IPEC 2024).
Editors: Édouard Bonnet and Paweł Rzążewski; Article No. 24; pp. 24:1–24:15
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

dominating sets has been investigated in many special graph classes. Several graph classes have been identified where this enumeration problem can be solved with polynomial delay. In this paper, we approach this problem from a different side. Namely, in [4] it was shown that all minimal Roman dominating functions[1] can be enumerated with polynomial delay. This was a rather surprising finding as in most other complexity aspects, Roman Domination and Dominating Set behave quite the same. More specifically, to mention some of these results:

- Roman Domination is NP-complete, even on special graph classes; see [20, 35].
- Minimum Roman Domination can be approximated up to a logarithmic factor but not any better, unless P = NP, confer [1, 35, 34].
- Roman Domination under standard parameterization (by an upper-bound k on the number of armies) is complete for W[2]; see [22]. However, the dual parameterization puts Roman Domination in FPT; see [2, 6, 7].

So, we take this as a basis and try to generalize Roman Domination towards Hitting Set to understand when or where the polynomial delay feature disappears. In passing, we also generalize the enumeration results from [4] considerably. As we will explain in the following, we introduce a generalizations of Roman Domination towards hypergraphs; we can describe the tractability frontier (with respect to polynomial delay) quite accurately. This also adds to the understanding what lets the Roman variation of domination behave that differently from the classical setting when it comes to enumeration.

Apart from quite a number of combinatorial (graph-theoretic) results that have been obtained for Roman domination, nicely surveyed in a 45-page chapter of [27], the decision problem Roman Domination has been studied from various aspects. Even though Roman Domination and Dominating Set behave the same in terms of complexity in a variety of settings this parallelism unexpectedly breaks down for two (mutually related) tasks:

- Can we enumerate all minimal solutions of a given instance with polynomial delay?
- Can we decide, given a certain part of the solution, if there exists a minimal solution that extends the given pre-solution?

The first type of question is also known as an *output-sensitive enumeration* problem. Even the less demanding task to enumerate all minimal dominating sets in output-polynomial time is open. The corresponding enumeration question for Roman dominating functions can be solved with polynomial delay, as proven in [4] and used in [3]. This result is based on another result giving a polynomial-time algorithm for the *extension problem(s)* as described in the second item. The idea is to call an extension test before diving further into branching. This strategy is well-established in the area of enumeration algorithms, dating back to Read and Tarjan [36], but few concrete examples are known; we only refer to the discussion in [33, 37]. This makes Extension Roman Domination one of few examples where the extension problem is polynomial-time solvable, while the original problem is NP-complete. For more details on extension problems, we refer to the survey [13].

The simple scientific question that we want to investigate in this paper is "why": *What causes Roman domination to be feasible with respect to enumeration and extension?* To find out why Extension Roman Domination (Ext RD for short) behaves in this peculiar way and what can be seen as a difference to Extension Dominating Set (Ext DS for short), we are going to generalize the concept of Roman domination and try to find the borderline of tractability. By this, we refer to the question if minimal hitting sets can be enumerated with polynomial delay. This so-called Transversal Hypergraph Problem [21] is open for four decades. It is quite important, as it appears in may application areas, and in particular

[1] These technical notions will be defined below.

in databases, there are quite a number of interesting equivalent problems, or problems that are shown to be *transversal-hard*, as called in [25]; we only mention two recent references and refer to the papers cited therein: [9, 10]. This question is equivalent to several enumeration problems in logic, database theory and also to enumerating minimal dominating sets in graphs, see [17, 21, 24, 29]. Our paper can be read as trying to understand which kind of problems are transversal-hard, and furthermore, to describe situations when polynomial-delay enumeration algorithms exist. Previous research on enumeration algorithms for minimal dominating sets often tried to look into special graph classes where polynomial-delay enumeration could be exhibited, or not; cf. [28, 29] as examples. This approach can be seen as specializing a known (transversal-hard) enumeration problem by studying special graph classes. Our approach is different as we come from a domination-type problem with a known polynomial-delay enumeration algorithm for general graphs and we try to stretch this result by generalization to understand when this enumeration task becomes transversal-hard.

It is well-known that Hitting Set (HS for short) can be viewed as a generalization of Dominating Set (DS for short) by modelling a graph by the closed-neighborhood hypergraph. One of the main differences between Dominating Set (in graphs) and Hitting Set is that in the second setting, there is a clear distinction between the objects that can dominate (the vertices of the hypergraph) and the objects that should be dominated (which are the hyperedges, i.e., sets of vertices). Although HS and Roman Domination are both established concepts that generalize DS, it seems that there is no combination of both concepts published in the literature. Actually, trying to define such a combination comes with some problems. If we want a HS instance to represent a DS instance by the closed-neighborhood hypergraph, the vertex set of the given graph is the vertex set of the hypergraph, and the set of all closed neighborhoods is the (hyper)edge set. Ignoring twins, this implies a bijection between the universe and the (hyper)edge set. But in general hypergraphs, the number of hyperedges and the number of elements in the universe are independent. Therefore, we have to think about how to interpret the “value one setting” such that it is related to the definition of Roman Domination where exactly one army is put on a certain vertex. We suggest modelling this effect in hypergraphs based on the following idea: If a vertex has the value 1 under a Roman dominating function, then it hits only its “own” closed-neighborhood hyperedge. In general hypergraphs, we have to explicitly express how a vertex “owns” a hyperedge. Hence, we need a function, called *correspondence*, which maps a vertex to an incident hyperedge, such that this hyperedge is dominated if the vertex has the value 1. This hypergraph problem is defined more formally in the next section.

2 Definitions and Notation

Throughout this paper, we will freely use standard notions from complexity theory without defining them here. This includes notions from parameterized complexity, concerning FPT and the further lower levels of the W-hierarchy up to W[3], as described in textbooks like [23, 19].

Let $\mathbb{N}$ denote the set of all nonnegative integers (including 0). For $n \in \mathbb{N}$, we will use the notation $[n] := \{1, \ldots, n\}$. For a finite set A and some $n \in \mathbb{N}$ with $n \leq |A|$, the cardinality of A, $\binom{A}{n}$ denotes the set of all subsets of A of cardinality n, while 2^A denotes the power set of A. For two sets A, B, B^A denotes the set of all mappings $f : A \to B$. If $C \subseteq A$, then $f(C) = \{f(x) \mid x \in C\} \subseteq B$. We denote by $\chi_C \in \{0,1\}^A$ the characteristic function, where $\chi_C(x) = 1$ holds iff $x \in C$. For two functions $f, g \in \mathbb{N}^A$, we write $f \leq g$ iff $f(a) \leq g(a)$ holds for all $a \in A$. Further, we define the *weight* of f by $\omega(f) = \sum_{a \in A} f(a)$.

We focus on not necessarily simple hypergraphs $H = \left(X, \hat{S} = (s_i)_{i \in I}\right)$ with a finite universe X, also called *vertex* set, and a finite index set I, where for each $i \in I$, $s_i \subseteq X$ is a *hyperedge*[2]. With the set S we denote the set which includes all hyperedges of the family $\hat{S}$, i.e., $S = \{s_i \mid i \in I\}$. Note that the same hyperedge may appear multiple times in the family $\hat{S}$. If this is forbidden, we speak of a *simple hypergraph.* For all $x \in X$, define $\mathbf{S}(x) = \{s_i \in S \mid x \in s_i\}$ as the set of hyperedges that is *hit* by the vertex x, hence defining a function $\mathbf{S} : X \to 2^S$. A set $D \subseteq X$ is a *hitting set* (*hs* for short) iff $\mathbf{S}(D) = S$, where $\mathbf{S}(D) = \bigcup_{x \in D} \mathbf{S}(x)$. Similarly, define $\mathbf{I} : X \to 2^I$ by $\mathbf{I}(x) = \{i \in I \mid x \in s_i\}$, extending to $A \subseteq X$ by $\mathbf{I}(A) = \bigcup_{x \in A} \mathbf{I}(x)$. We call $\tau : X \to I$ a *correspondence* if $x \in s_{\tau(x)}$ for all $x \in X$ or if, in other words, $\tau(x) \in \mathbf{I}(x)$ for all $x \in X$.

Consider a (simple undirected) graph $G = (V, E)$ as a hypergraph $G = (V, \hat{E})$, where each hyperedge contains exactly two elements. Now, we call the hyperedges just edges. Talking about simple graphs, we can consider E as the index set. For each vertex $v \in V$, define its neighborhood as $N(v) = \{u \mid \{v, u\} \in E\}$ and the closed neighborhood as $N[v] = \{v\} \cup N(v)$. For vertex sets $U \subseteq V$, we use $N[U] = \bigcup_{v \in U} N[v]$ for the closed neighborhood of U. A *dominating set* (*ds* for short) of a graph $G = (V, E)$ is a set $D \subseteq V$ such that $N[D] = V$. A function $f : V \to \{0, 1, 2\}$ is a *Roman dominating function* (Rdf for short) iff, for each vertex $v \in V$ with $f(v) = 0$, there exists a $u \in N(v)$ with $f(u) = 2$. A Rdf f is *minimal* if for each Rdf g with $g \leq f$, $f = g$ holds, leading to the following problem.

Problem name: Roman Domination (RD)
Given: A graph $G = (V, E)$ and $k \in \mathbb{N}$
Question: Is there a Rdf f with $\omega(f) \leq k$?

Problem name: Extension Roman Domination (Ext RD)
Given: A graph $G = (V, E)$ and a function $f : V \to \{0, 1, 2\}$
Question: Is there a minimal rdf g for G with $f \leq g$?

Somewhat surprisingly, the extension problem in the second box was proven to be polynomial-time solvable in [4]. This implies that minimal Rdf can be enumerated with polynomial delay. The basis of this algorithm is a combinatorial characterization of minimal Rdfs. To be able to formulate the combinatorial characterization of minimal Rdf, we need a further notion. For $D \subseteq V$ and $v \in D$, define the *private neighborhood* of $v \in V$ with respect to D as $P_{G,D}(v) \coloneqq N[v] \setminus N[D \setminus \{v\}]$ whose elements are the private neighbors of v.

▶ **Theorem 2.1** ([4]). *Let $G = (V, E)$ be a graph and $f : V \to \{0, 1, 2\}$ be a function. Abbreviate $G' \coloneqq G\left[f^{-1}(0) \cup f^{-1}(2)\right]$. Then, f is a minimal Rdf iff we find:*

1. $N\left[f^{-1}(2)\right] \cap f^{-1}(1) = \emptyset$,
2. $\forall v \in f^{-1}(2) : P_{G', f^{-1}(2)}(v) \not\subseteq \{v\}$, *also called* privacy condition, *and*
3. $f^{-1}(2)$ *is a minimal ds of* G'.

In order to explore why these results were possible, we generalize these notions and problems for hypergraphs in two ways, with a clear focus on the second possibility.

The first one is probably the most natural one, formed in analogy to the notion of a ds in a (simple) hypergraph; see [5]. Let $H = (V, E)$ be a (simple) hypergraph, i.e., $E \subseteq 2^V$. Then, $f : V \to \{0, 1, 2\}$ is a Rdf if, for all $v \in V$ with $f(v) = 0$, there is a vertex $u \in V$ with $f(u) = 2$ that is a neighbor of v, i.e., it shares an edge with v, which means, more formally,

[2] For our proofs, we found this index notation more convenient than a multiset notation.

that there exists some $e \in E$ with $\{u, v\} \subseteq e$. However, as we show next, we can transfer all interesting properties of Roman domination from the graph case to the hypergraph setting by using the same reduction: if $G = (V, E)$ is a simple hypergraph, we can construct a graph $G' = (V, E')$ by setting $\{x, y\} \in E'$ iff there is a hyperedge $e \in E$ with $\{x, y\} \subseteq e$. This construction is also known as the *Gaifman graph* of G. Then, $f : V \to \{0, 1, 2\}$ is a Rdf of the hypergraph G iff f is a Rdf of the graph G'. Conversely, we just have to interpret a given graph as a simple hypergraph.

Therefore, we propose a further generalization of Roman domination towards hypergraphs that allow us to study why Roman domination shows such a peculiar behavior when it comes to its extension version, as well as concerning enumerating minimal Rdfs:

Let $H = (X, \hat{S} = (s_i)_{i \in I})$ be a hypergraph and $\tau : X \to I$ be a correspondence. We call a function $f : X \to \{0, 1, 2\}$ a *Roman hitting function* (Rhf for short) if, for each $i \in I$, there is an $x \in s_i$ with $f(x) = 2$ or if there exists an $x \in X$ with $\tau(x) = i$ and $f(x) = 1$. In these scenarios, we say that x hits i or s_i. For a function $f : X \to \{0, 1, 2\}$, we define the partition $P(f) = \{f^{-1}(0), f^{-1}(1), f^{-1}(2)\}$. We now define two decision problems related to Rhf.

Problem name: Roman Hitting Function, or RHF for short
Given: A finite set X, a hyperedge family $\hat{S} = (s_i)_{i \in I}$, forming the hypergraph $(X, \hat{S})$, a correspondence $\tau : X \to I$, and $k \in \mathbb{N}$
Question: Is there a Rhf f with $\omega(f) \leq k$?

Problem name: Extension Roman Hitting Function, or Ext RHF for short
Given: A finite set X, a hyperedge family $\hat{S} = (s_i)_{i \in I}$, forming the hypergraph $(X, \hat{S})$, a correspondence $\tau : X \to I$, and $f : X \to \{0, 1, 2\}$
Question: Is there a minimal Rhf g with $f \leq g$?

To understand in which way this setting generalizes RD, recall that there are alternative ways to specify a graph as a hypergraph; namely, the *closed-neighborhood hypergraph* G_{nb} associated to a graph $G = (V, E)$ can be described as $G_{nb} = (V, (N[v])_{v \in V})$. Clearly, $D \subseteq V$ is a ds iff D is a hs of G_{nb}. As $v \in N[v]$, the identity can be viewed as a correspondence. In this interpretation, $f : V \to \{0, 1, 2\}$ is a Rdf of G iff it is a Rhf of G_{nb}.

Organization of the Paper and Main Results. In Section 3, we will prove that our optimization problem is NP-complete and, more interestingly, k-RHF is W[2]-complete. Then, we turn our attention to the extension problem. Recall that the algorithmic results in the case of Roman domination were based on some basic combinatorial insights. Following this logic, first we show in Section 4 a combinatorial characterization of minimal Rhf that we can make use of in Section 5 where we prove that Ext RHF with surjective correspondences can be solved in polynomial time. In Section 6, we return to Roman domination and consider a variant of the extension problem where we give both lower and upper bound conditions to the minimal Rdf that we are looking for. In contrast to the original problem (that only provides a lower bound), this two-sided extension problem turns out to be NP-complete as we show. Furthermore, we identify two natural parameters under which bounded-Ext RD is W[3]-complete. In Section 7, we further discuss different parameterization of Ext RHF. Again, we obtain some parameterizations for Ext RHF where the problem is W[3]-complete. To save space, we will mark theorems with $(*)$ if the proof is in the long version of the paper.

3 The Optimization Problem RHF

In this section, we will discuss the (parameterized) complexity of the optimization problem RHF. The probably most natural parameterization for the problem is by an upper bound k on the weight of the Rhf. For our results, we will use the W[2]-completeness of RD (with k as parameter) shown in [22]. The hardness follows, as Rdf can be interpreted as Rhf. For the membership, we construct a split graph, where the clique represents the elements of the universe and the independent set form the hyperedges. Here, the hyperedges are added twice to the independent set if the inverse image of the hyperedge with respect to τ is empty. In this way, it is better to put a neighbor to 2 than to put these two vertices to 1.

▶ **Theorem 3.1.** (∗) *RHF is NP-complete. k-RHF is W[2]-complete.*

The fact that RD is NP-complete even on split graphs was mentioned repeatedly in the literature, for instance, in [16, 30], but to the best of our knowledge, no proof of this fact has been published. We will provide a strengthened assertion in the following. Recall that two vertices u, v in a graph are called *true twins* if $N[u] = N[v]$.

▶ **Lemma 3.2.** (∗) *RD is NP-complete even on true-twin-free split graphs. Likewise, k-RD is W[2]-complete on true-twin-free split graphs.*

The hardness part of the proof of Theorem 3.1 (implicitly) uses the fact that the family of hyperedges could include a hyperedge multiple times as there could be twins (vertices with the same closed neighborhoods) in the original graph. Consider a complete graph $K_n = ([n], E_n)$ with $n \geq 2$ vertices. Since the closed neighborhoods are always equal to $[n]$, if we would use a normal set for the hyperedges instead of a family of hyperedges, then the best solution would be $\chi_{\{v\}}$ for any vertex v. This would even not be a Rdf. Namely, minimal Rdf would be of the form $2 \cdot \chi_{\{v\}}$ for any vertex v, or they would be constant 1. Nevertheless, the following holds, revisiting Lemma 3.2.

▶ **Corollary 3.3.** *RHF is NP-complete even on simple hypergraphs. Furthermore, k-RHF is W[2]-complete.*

The NP-completeness of the optimization problem also motivates our analysis of the extension problem; it could help speed up an exact branching algorithm for solving this decision problem. In the long version, we also consider approximation complexity for RHF.

4 Combinatorial Properties of Minimal Rhf

In this section, we will prove combinatorial properties of minimal Rhf. This will help us analyze the complexity of Ext RHF.

▶ **Theorem 4.1.** *Let X be a vertex set, $\hat{S} = (s_i)_{i \in I}$ be a hyperedge family and $\tau : X \to I$ be a correspondence. Then, a function $f : X \to \{0, 1, 2\}$ is a minimal Rhf iff the following constraint items hold:*

0. $\forall x, y \in f^{-1}(1) : x \neq y \Rightarrow \tau(x) \neq \tau(y)$,

1. $\forall x \in f^{-1}(1) : s_{\tau(x)} \cap f^{-1}(2) = \emptyset$,

2. $\forall x \in f^{-1}(2) \exists i \in I \setminus \{\tau(x)\} : s_i \cap f^{-1}(2) = \{x\}$, *and*

3. $f^{-1}(2)$ *is a minimal hs on* $\{s_i \in S \mid i \in I, \tau^{-1}(i) \cap f^{-1}(1) = \emptyset\}$.

Proof. Let f be a minimal Rhf on X, $\hat{S}$ and τ. Assume there are $x, y \in f^{-1}(1)$ with $x \neq y$ but $\tau(x) = \tau(y) = i$. Define $\tilde{f} = f - \chi_{\{y\}}$. Trivially, $\tilde{f} \leq f$ and $\tilde{f} \neq f$. Since $f^{-1}(2) = \tilde{f}^{-1}(2)$ and $\tau(\tilde{f}^{-1}(1)) = \tau(f^{-1}(1) \setminus \{y\}) = \tau(f^{-1}(1))$ hold, $\tilde{f}$ is a Rhf. Thus, f is not minimal, which is a contradiction.

Now assume there is an $x \in f^{-1}(1)$ with $s_{\tau(x)} \cap f^{-1}(2) \neq \emptyset$. Define $\tilde{f} = f - \chi_{\{x\}}$. Trivially, $\tilde{f} \leq f$, $f \neq \tilde{f}$ and $f^{-1}(2) = \tilde{f}^{-1}(2)$. Hence, $s_{\tau(x)} \in \mathbf{S}(\tilde{f}^{-1}(2))$. Since s_i ,for $i \in I \setminus \{\tau(x)\}$, is hit by $\tilde{f}$ in the same way as by f, $\tilde{f}$ is a Rhf. This contradicts the minimality of f.

Assume $f^{-1}(2)$ is not a minimal hs on S'. If there is an $s \in S'$ that is not hit by $f^{-1}(2)$, then f is no Rhf, contradicting our assumption. Hence, we can assume $f^{-1}(2)$ is not minimal. More explicitly we assume there is an $x \in f^{-1}(2)$ such that for each $i \in I \setminus \{\tau(x)\}$, $s_i \cap f^{-1}(2) \neq \{x\}$. Then there is an $x \in f^{-1}(2)$ with, for each $s \in \mathbf{S}'(x) \setminus \{\tau(x)\}$, there exists a $y \in \left(f^{-1}(2) \setminus \{x\}\right) \cap s$. Define $\tilde{f} = f - \chi_{\{x\}}$. Let $i \in I$. As $\tau(f^{-1}(1)) \cup \{\tau(x)\} = \tau(\tilde{f}^{-1}(1))$ holds by definition, we only need to consider $i \in I \setminus (\tau(f^{-1}(1)) \cup \tau(x))$. For i with $s_i \in \mathbf{S}(f^{-1}(2)) \setminus \mathbf{S}(x)$, trivially $s_i \cap \tilde{f}^{-1}(2) \neq \emptyset$. If $s_i \in \mathbf{S}(x) \setminus \{\tau(x)\}$, as we mentioned before, there is a $y \in \left(f^{-1}(2) \setminus \{x\}\right) \cap s_i = \tilde{f}^{-1}(2) \cap s_i$. Thus, $\tilde{f}$ is a Rhf and f is not minimal, contradicting our assumption. Hence, the four conditions hold.

For the if-part assume f fulfills the constraints. By Constraint 3, for all $i \in I$, either $s_i \cap f^{-1}(2) \neq \emptyset$, or there is an $x \in f^{-1}(1)$ with $\tau(x) = i$. Hence, f is a Rhf. Let $g : X \to \{0,1,2\}$ be a minimal Rhf with $g \leq f$. Thus, $g^{-1}(2) \subseteq f^{-1}(2)$ and $\mathbf{S}(g^{-1}(2)) \subseteq \mathbf{S}(f^{-1}(2))$ hold. Furthermore, $g^{-1}(1) \subseteq f^{-1}(1) \cup f^{-1}(2)$. Since for each $x \in X$, $\{s_{\tau(x)}\} \subseteq \mathbf{S}(x)$, for each $i \in \tau(g^{-1}(1))$, $i \in \tau(f^{-1}(1))$ or $s_i \in \mathbf{S}(f^{-1}(2))$. Let $x \in X$ be an element with $g(x) < f(x)$.

Case 1: $g(x) = 0 < 2 = f(x)$. This implies that, for each $i \in \mathbf{I}(x)$, there exists a $y \in s_i$ with $2 = g(y) \leq f(y) = 2$ or $y \in \tau^{-1}(i) \cap g^{-1}(1) \subset \tau^{-1}(i) \cap (f^{-1}(1) \cup f^{-1}(2))$. This either contradicts Constraint 1 or Constraint 2.

Case 2: $g(x) = 1 < 2 = f(x)$. This case works analogously, somehow simpler. We only need to exclude $i = \tau(x)$.

Case 3: $f(x) = 1$. This implies $g(x) = 0$. Since g is a Rhf, either $s_{\tau(x)} \cap g^{-1}(2)$ is not empty or there exists a $y \in g^{-1}(1)$ with $\tau(x) = \tau(y)$.

Case 3.1: $\tau(x) \cap g^{-1}(2) \neq \emptyset$. As $g^{-1}(2) \subseteq f^{-1}(2)$, this contradicts Constraint 1.

Case 3.2: There is a $y \in g^{-1}(1)$ with $\tau(x) = \tau(y)$. Thus, either there is a $y \in f^{-1}(1) \setminus \{x\}$ with $\tau(x) = \tau(y)$ (this contradicts Constraint 0) or $f(y) = 2$ (this contradicts Constraint 1).

Thus, $g = f$ holds. Therefore, f is minimal. ◀

▶ Remark 4.2. One can compare Theorem 4.1 with Theorem 2.1. For a graph $G = (V, E)$, let $G_{nb} = (V, (N[v])_{v \in V})$ be the closed-neighborhood hypergraph. Here, for each $f : V \to \{0,1,2\}$ and $i \in \{1,2,3\}$, f fulfills Constraint i of Theorem 4.1 with respect to G_{nb} iff f fulfills Constraint i of Theorem 2.1 with respect to G.

We call a $f : X \to \{0,1,2\}$ *extensible* on the hypergraph $H = (X, \hat{S})$ with correspondence τ if there is a minimal Rhf g with $f \leq g$. The following two results are basically implied by Theorem 4.1.

▶ **Lemma 4.3.** (∗) *Let $H = (X, \hat{S} = (s_i)_{i \in I})$ be a hypergraph with correspondence τ and $f : X \to \{0,1,2\}$ be a function with $x \in f^{-1}(2)$, $y \in f^{-1}(1)$ and $x \in s_{\tau(y)}$. Then, f is extensible iff $f + \chi_{\{y\}}$ is extensible.*

▶ **Lemma 4.4.** (∗) *Let $H = (X, \hat{S} = (s_i)_{i \in I})$ be a hypergraph with correspondence τ and $f : X \to \{0,1,2\}$ be a function with $x, y \in f^{-1}(1)$, $x \neq y$ and $\tau(x) = \tau(y)$. Then, f is extensible iff $f + \chi_{\{x,y\}}$ is extensible.*

▶ **Theorem 4.5.** *Let $H = (X, \hat{S} = (s_i)_{i \in I})$ be a hypergraph with correspondence τ, $\tau : X \to I$. Let $f : X \to \{0,1,2\}$ be a function such that $x \neq y$ implies $\tau(x) \neq \tau(y)$ for each $x, y \in f^{-1}(1)$. Then, f is extensible iff there exist a set R_2 with $f^{-1}(2) \subseteq R_2 \subseteq f^{-1}(1) \cup f^{-1}(2)$ and a mapping $\rho : R_2 \to I$, satisfying the following constraints.*

1. $\forall x \in R_2 : \rho(x) \neq \tau(x)$.
2. $\forall x \in R_2 : s_{\rho(x)} \cap R_2 = \{x\}$.
3. $\forall x \in f^{-1}(1) \setminus R_2 : s_{\tau(x)} \cap R_2 = \emptyset$.
4. $\forall i \in I$ *such that* $\tau^{-1}(i) = \emptyset$:
   $s_i \subseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right) \implies s_i \cap R_2 \neq \emptyset$.

Proof. Define $I' := \{i \in I \mid \tau^{-1}(i) = \emptyset\}$. First, we assume that f is extensible. Let $g : X \to \{0,1,2\}$ be a minimal Rhf with $f \leq g$. By Constraint item 3 of Theorem 4.1, $g^{-1}(2)$ is a minimal hs on $\{s_i \in S \mid i \in I, \tau^{-1}(i) \cap f^{-1}(1) = \emptyset\}$ $(*)$. With Theorem 4.1, Constraint 2, this implies that, for each $x \in g^{-1}(2)$, there exists an $\rho(x) \in I \setminus \{\tau(x)\}$ such that $s_{\rho(x)} \cap g^{-1}(2) = \{x\}$. Define $R_2 = (f^{-1}(1) \cup f^{-1}(2)) \cap g^{-1}(2)$. Clearly, $f^{-1}(2) \subseteq R_2 \subseteq f^{-1}(1) \cup f^{-1}(2)$. We have to check the four constraints claimed for R_2. The first two are even true in a slightly more general fashion by $(*)$. If there would exist a $y \in f^{-1}(1) \setminus R_2 \subseteq g^{-1}(1)$ with $\emptyset \neq s_{\tau(y)} \cap R_2 \subseteq s_{\tau(y)} \cap g^{-1}(2)$, then this would contradict Theorem 4.1, Constraint 2, showing the third constraint of this theorem. We now turn to the fourth and last constraint. Let $i \in I'$. Since $\tau^{-1}(i) = \emptyset$, there has to be a $y \in g^{-1}(2) \cap s_i$. If $y \in R_2$, then the constraint is satisfied. Hence, we can assume that $y \in g^{-1}(2) \setminus R_2 = g^{-1}(2) \cap f^{-1}(0)$. Consider $x \in f^{-1}(1) \setminus R_2$. As $f \leq g$ and $x \notin R_2$, we have $g(x) = 1$. By Constraint 1 of Theorem 4.1, $y \notin s_{\tau(x)}$. If $s_i \subseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right)$ and $s_i \cap R_2 = \emptyset$ hold, then this would contradict $s_{\rho(x)} \cap R_2 = \{x\}$. Therefore, all the constraints hold.

Let now f, R_2 and $\rho : R_2 \to I$ fulfill the constraints of this theorem. For this part of the proof we will define a hypergraph H' that includes each edge where $\tau(X)$ does not include its index and the edge is not hit, yet. We will show that there is a minimal hs D on H' which does not include any vertex of $s_{\rho(x)}$ for $x \in R_2$ or $s_{\tau(x)}$ for $f^{-1}(1) \setminus R_2$. $R_2 \cup D$ will describe the set of vertices with value 2. We will hit the remaining vertices by assigning the value 1 to some vertices. Therefore, we define the hypergraph $H' = (X', (s'_i)_{i\in I''})$ with

$$I'' := I' \cap \{i \in I \mid s_i \cap R_2 = \emptyset\},$$

$$X' := \left(\bigcup_{i\in I''} s_i\right) \setminus \left(\left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right)\right)$$

and $s'_i := s_i \cap X'$. If s'_i is empty for an $i \in I''$, then there would not exist any hs on H'. Therefore, we need to ensure that such an index does not exist. Let $i \in I''$. Hence, $\tau^{-1}(i) = \emptyset$ and $s_i \cap R_2 = \emptyset$. The contraposition of the implication of Constraint 4 implies

$$s_i \nsubseteq \left(\bigcup_{x\in f^{-1}(1)\setminus R_2} s_{\tau(x)}\right) \cup \left(\bigcup_{x\in R_2} s_{\rho(x)}\right).$$

Hence, $s'_i \neq \emptyset$ for each $i \in I''$. Thus, there is a minimal hs D on H'. The construction of H' and D implies that $\tau^{-1}(i) \neq \emptyset$ for each $i \in I \setminus \mathbf{I}(R_2 \cup D)$. For each $i \in I \setminus \mathbf{I}(R_2 \cup D)$, x_i will describe an arbitrary vertex in $\tau^{-1}(i)$, unless there exists an $x_i \in \left(f^{-1}(1) \setminus R_2\right) \cap \tau^{-1}(i)$ (by assumption on f, there is at most one such element). In this case, we choose this x_i.

Define $g : X \to \{0,1,2\}$ with $g^{-1}(1) = \{x_i \mid i \in I \setminus \mathbf{I}(R_2 \cup D)\}$ and $g^{-1}(2) = D \cup R_2$. We will now use Theorem 4.1 to show that g is a minimal Rhf. By the construction of $g^{-1}(1)$, Constraints 0 and 1 of Theorem 4.1, are fulfilled. Since $g^{-1}(1)$ hits each edge in $I \setminus \mathbf{I}(D \cup R_2)$, each hyperedge in $\{s_i \mid i \in I, \tau^{-1}(i) \cap g^{-1}(1) = \emptyset\}$ is hit by $g^{-1}(2)$. As D is minimal and $D \cap \left(\cup_{x\in R_2} s_{\rho(x)}\right) = \emptyset$, Constraint 2 implies that is $D \cup R_2$ also a minimal hs on $\mathbf{I}(D \cup R_2)$. The remaining constraint of Theorem 4.1 follows by the first two constraints of f together with definition of H' and D as H' only contains hyperedges s'_i where $\tau^{-1}(i) = \emptyset$. ◀

We will use Lemma 4.4 and Theorem 4.5 to show a W[3]-membership.

▶ Remark 4.6. Theorem 4.5 already gives an idea why Ext RHF with surjective correspondence (and therefore also Ext RD) runs in polynomial time. Let $H = (X, (s_i)_{i \in I})$ be a hypergraph and $f : X \to \{0, 1, 2\}$ be a function with the surjective correspondence τ. Hence, we can disregard Constraint 4, as $\tau^{-1}(i) \neq \emptyset$ for each $i \in I$. Then, we could use Lemmata 4.3 and 4.4. We set $R_2 := f^{-1}(2)$. By Constraint 3, we have to add each $x \in f^{-1}(1)$ with $s_{\tau(x)} \cap R_2 \neq \emptyset$ to R_2. Now we can check if for each $x \in R_2$ there is a $i_x \in I$ that fulfills Constraints 1 and 2. This will be our strategy in the next section.

5 Ext RHF with Surjective Correspondence and Ext RD

In this section, we present a polynomial-time algorithm for Ext RHF instances with a surjective correspondence function τ. At the end of this section, we explain how this algorithm can be viewed as a natural generalization of Ext RD that was studied before in [4].

Algorithm 1 ExtRHF Algorithm.

procedure ExtRHF Solver$(X, \hat{S}, \tau, f)$
  Input: set X, $\hat{S} := (s_i)_{i \in I}$, τ correspondence function, $f : X \to \{0, 1, 2\}$ with $\{i \in I \mid \tau^{-1}(i) = \emptyset\} \subseteq \mathbf{I}\left(f^{-1}(2)\right)$.
  Output: Is there a minimal Rhf g with $f \leq g$?
  $M_2 := f^{-1}(2)$, $M_1 := f^{-1}(1)$
  for $x \in M_1$ **do**
    for $y \in M_1 \setminus \{x\}$ **do**
      if $\tau(x) = \tau(y)$ **then**
        Add x, y to M_2 and delete them in M_1.
        Continue with the next x.
  $M := M_2$ { All $x \in g^{-1}(2)$ are considered in the following. }
  while $M \neq \emptyset$ **do**
    Choose $x \in M$.
    for $y \in \tau^{-1}(\mathbf{I}(x))$ **do**
      if $y \in M_1$ **then** Add y to M_2 and M. Delete y in M_1.
    Delete x from M.
  for $x \in M_2$ **do**
    if $\mathbf{I}(x) \setminus \mathbf{I}(M_2 \setminus \{x\}) \subseteq \{\tau(x)\}$ **then Return no**
  for $i \in I \setminus (\mathbf{I}(M_2) \cup \tau(M_1))$ **do**
    Add one arbitrary element $x \in \tau^{-1}(i)$ to M_1.
  Return yes { $g^{-1}(0) = X \setminus (M_1 \cup M_2), g^{-1}(1) = M_1, g^{-1}(2) = M_2$ }

▶ **Theorem 5.1.** (∗) *Algorithm 1 solves Ext RHF for instances $(X, \hat{S}, \tau, f)$ satisfying $\{i \in I \mid \tau^{-1}(i) = \emptyset\} \subseteq \mathbf{I}\left(f^{-1}(2)\right)$ in polynomial time.*

A special case of this theorem entails: Ext RHF with surjective τ is polynomial-time solvable. In the long version, we discuss the connections between Algorithm 1 and Algorithm 1 in [4].

6 Bounded Extension Roman Domination

In this section, we will discuss a two-sided bounded version of Extension Roman Domination which was also suggested by a colleague of ours.

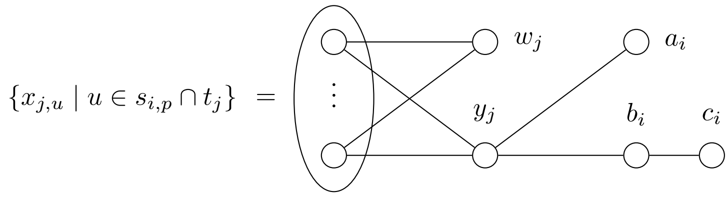

Figure 1 Construction for Theorem 6.1, for $i \in [k]$, $p \in [\ell_i]$ and $[j \in \ell_T]$.

Problem name: Bounded Extension Roman Domination (bounded-Ext RD)
Given: A graph $G = (V, E)$ and functions $f, h : V \to \{0, 1, 2\}$.
Question: Is there a minimal Rdf $g : V \to \{0, 1, 2\}$ with $f \leq g \leq h$?

We show in Corollary 6.4 that bounded-Ext RD is NP-complete. Thus, we look for FPT-algorithms. One natural parameterization for this problem could be $\omega(2 - h)$, because for $2 - h = 0$, we are back to Ext RD as a special case, which is known to be solvable in polynomial time. Hence, this parameterization can be viewed as a "distance-from-triviality" parameter [26]. However, as we will prove in Theorem 6.1, this parameterization strategy fails. We employ the well-known W[3]-completeness of Multicolored Independent Family (MultIndFam), parameterized by k, in the reduction presented in the proof of Theorem 6.1.

Problem name: Multicolored Independent Family (MultIndFam)
Given: A $(k+1)$-tuple $(S_1, \ldots, S_k, T)$ of subsets of 2^U on the common universe U, i.e., $(U, S_1), \ldots, (U, S_k), (U, T)$ are $k + 1$ many simple hypergraphs.
Question: Are there hyperedges $s_1 \in S_1, \ldots, s_k \in S_k$ such that no $t \in T$ is a subset of $\bigcup_{i=1}^{k} s_i \subseteq U$?

In that theorem, we actually discuss a slightly different parameterization, namely $\kappa_{h^{-1}(0)}(G, f, h) := |h^{-1}(0)|$.

▶ **Theorem 6.1.** *$\kappa_{h^{-1}(0)}$-bounded-Ext RD is W[3]-hard even on bipartite graphs.*

Proof. Let $k \in \mathbb{N}$ and $(S_1, \ldots, S_k, T)$ be a $(k+1)$-tuple of subsets of 2^U with a common universe U. To simplify the notation, let $T = \{t_1, \ldots, t_{\ell_T}\}$ and $S_i = \{s_{i,1}, \ldots s_{i,\ell_i}\}$ for $i \in [k]$. Define $X_j := \{x_{j,u} \mid u \in t_j\}$ for $j \in [\ell_T]$, $Y_i := \{y_{i,1}, \ldots, y_{i,\ell_i}\}$ and $G = (V, E)$ with

$$V := \left(\bigcup_{i=1}^{k} \{a_i, b_i, c_i\} \cup Y_i\right) \cup \left(\bigcup_{j=1}^{\ell_T} \{w_j\} \cup X_j\right),$$

$$E := \{\{a_i, y_{i,p}\}, \{b_i, y_{i,p}\}, \{b_i, c_i\} \mid i \in [k], p \in [\ell_i]\} \cup \{\{w_j, x_{j,u}\} \mid j \in [\ell_T], u \in t_j\}$$
$$\cup \{\{y_{i,p}, x_{j,u}\} \mid i \in [k], p \in [\ell_i], j \in [\ell_T], u \in s_{i,p} \cap t_j\}.$$

Clearly, G is bipartite, as $V = A \cup B$ decomposes V into two disjoint independent sets, with $A = \left(\bigcup_{i=1}^{k} \{a_i, b_i\}\right) \cup \left(\bigcup_{j=1}^{\ell_T} X_j\right)$ and $B = \left(\bigcup_{i=1}^{k} \{c_i\} \cup Y_i\right) \cup \left(\bigcup_{j=1}^{\ell_T} \{w_j\}\right)$. Furthermore, we need the maps $f, h \in \{0, 1, 2\}^V$ with $f = 2\chi_{\{w_1, \ldots, w_{\ell_T}\} \cup \{b_1, \ldots, b_k\}}$ and $h = 2(1 - \chi_{\{a_1, \ldots, a_k\}})$.

▷ Claim 6.2. $(*)$ $S_1, \ldots, S_k, T$ is a yes-instance of the MultIndFam problem iff there exists a minimal Rdf g on G with $f \leq g \leq h$.

Since $k = |\{a_1, \ldots, a_k\}| = |h^{-1}(0)|$, this is an FPT-reduction. ◀

Since h maps no vertex to 1 in the reduction presented in the proof of Theorem 6.1, bounded-EXT RD is W[3]-hard, parameterized by $\kappa_{2-h}(G,f,h) := \sum_{v\in V}(2-h(v))$. Namely, in the construction of Theorem 6.1, $\kappa_{2-h}(G,f,h) = 2\cdot\kappa_{|h^{-1}(0)|}(G,f,h)$.

Another parameterization could be $\omega(f)$: If $\omega(f)=0$, there is a minimal Rdf $g: V \to \{0,1,2\}$ with $f \leq g \leq h$ iff h is a Rdf (this can be checked in polynomial time). If there is such a g, then h is also a Rdf. If h is a Rdf, then we can decrease the value of the vertices until we can no longer decrease the value of any vertices without losing the Rdf property. To understand the complexity of this parameter, we need the following extension version of HS.

Problem name: EXTENSION HITTING SET (EXT HS)
Given: A simple hypergraph $H=(X,S)$, $S \subseteq 2^X$, and a set $U \subseteq X$.
Question: Is there a minimal hs $T \subseteq X$ with $U \subseteq T$?

In [8], it was proven that EXT HS is W[3]-complete when parameterized by $|U|$. We use this result in the proof of the following theorem.

▶ **Theorem 6.3.** (∗) *$\omega(f)$-bounded-EXT RD is W[3]-hard on split graphs.*

Since the reductions in this section are polynomial-time reductions and since membership in NP is easily seen using guess-and-check, we can conclude:

▶ **Corollary 6.4.** *bounded-EXT RD is NP-complete even on split graphs or bipartite graphs.*

We will make use of the W[3]-hardness in the next section, when we turn to discuss the complexity of EXT RHF. The reductions provided there will also show that bounded-EXT RD is W[3]-complete for some parameterizations.

Finally, let us mention that, in any given bounded-EXT RD instance (G,f,h), we can always assume (1) $f \leq h$ and, moreover, (2) $h(v)=0$ implies $h(u)=2$ for some $u \in N(v)$. Otherwise, there cannot exist a Rdf g with $f \leq g \leq h$. Both conditions are easy to check.

7 Complexity of Ext RHF

In this section, we will show that there are instances of EXT RHF which are W[3]-complete, considering their different parameterizations. For the W[3]-membership, we make again use of MULTINDFAM, as there is no further W[3]-complete problems that we find suitable for a reduction. Unfortunately, this reduction is quite technical.

▶ **Theorem 7.1.** *$\omega(f)$-EXT RHF is in W[3].*

Proof. Let $H=(X,\hat{S}=(s_i)_{i\in I})$ be a hypergraph with correspondence $\tau: X \to I$ and let $f: X \to \{0,1,2\}$ be some function, comprising an instance of $\omega(f)$-EXT RHF. (∗) We can assume that there are not two elements $x,y \in X$ such that $f(x)=f(y)=1$ with $\tau(x)=\tau(y)$ or $f(x)=2, f(y)=1$ with $x \in s_{\tau(y)}$. Otherwise, we could use Corollaries 4.3 and 4.4.

We will construct an equivalent MULTINDFAM instance next. We define its universe as $U := X \cup \{r_{i,x}, x' \mid x \in f^{-1}(\{1,2\}), i \in I\} \cup \{\tau_x \mid x \in f^{-1}(1)\}$. For the construction of the hypergraphs, we need to define some additional (auxiliary) sets:

- For $x \in f^{-1}(\{1,2\})$, $i \in \mathbf{I}(x)$ abbreviate $\tilde{s}_{x,i} := s_i \cup \{r_{i,x}, x'\}$.
- Define $t_i := s_i \cup \{\tau_x \mid x \in f^{-1}(1) \cap s_i\}$ for $i \in I$ with $\emptyset = \tau^{-1}(i) = s_i \cap f^{-1}(2)$.
- For each $x \in f^{-1}(2)$, let $S_x := \{\tilde{s}_{x,i} \mid \tau(x) \neq i\}$ and
  for each $x \in f^{-1}(1)$, let $S_x := \{s_{\tau(x)} \cup \{\tau_x\}\} \cup \{\tilde{s}_{x,i} \mid \tau(x) \neq i\}$.

- Furthermore, we need the target set $T = T' \cup T''$, where

$$T' \coloneqq \{\, t_i \mid i \in I \wedge \tau^{-1}(i) = \emptyset \wedge s_i \cap f^{-1}(2) = \emptyset \,\} \quad \text{and}$$
$$T'' \coloneqq \{\, \{r_{i,x}, y'\} \mid i \in I \wedge \{x, y\} \subseteq (s_i \cap f^{-1}(\{1,2\})) \wedge x \neq y \,\} \cup \{\, \{\tau_x, y'\} \mid x \in g^{-1}(1) \wedge y \in s_{\tau(x)} \wedge x \neq y \,\} .$$

Now we will explain the idea of each element. It is important to keep in mind that we want to use Theorem 4.5: If x' is in a chosen hyperedge, then we assign the value 2 to x in the minimal Rhf. The element $r_{i,x}$ gives us information about the mapping ρ. $r_{i,x}$ is in one of the chosen edges iff $\rho(x) = i$ holds. Therefore, the sets $\{r_{i,x}, y'\}$ verify the Constraint 2 of Theorem 4.5. τ_x will only be in a set we chose if we assign the value 1 to x. Hence, Constraint 3 will be checked by the sets $\{\tau_x, y\}$. The sets in T' correspond to the sets which we consider in Constraint 4. This is also the reason why τ_x is in included in t_i. Since there exists a τ_x, $f(x) = 1$. If $x \in R_2$, $s_i \cap R_2 \neq \emptyset$. In the MultIndFam instance, this corresponds to: τ_x will not be in our sets, which implies that t_i will not be covered completely.

▷ Claim 7.2. $(*)$ (H, τ, f) is a yes-instance of Ext RHF iff $(U, (S_x)_{x\in f^{-1}(\{1,2\})}, T)$ is a yes-instance of MultIndFam.

As $|U| \leq |X| \cdot (|I| + 4)$, $|T| \leq |I| + |X|^2 \cdot (|I| + 1)$ and $|S_x| \leq |I|$, the MultIndFam can be constructed in polynomial time. Furthermore, the parameter of the constructed instance of MultIndFam is $|f^{-1}(1) \cup f^{-1}(2)| \leq \omega(f)$. Hence, Ext RHF belongs to W[3]. ◀

As mentioned in the previous proof, the described reduction is also a polynomial-time reduction. Hence, Ext RHF is a member of NP, but this is also observed by the guess-and-check characterization of NP. For the hardness results, we will use bounded-Ext RD.

▶ **Theorem 7.3.** *$\omega(f)$-Ext RHF is W[3]-hard even if the correspondence function is injective. Furthermore, $\omega(f)$-bounded-Ext RD is W[3]-complete.*

Proof. We will make use of Theorem 6.3, reducing from bounded-Ext RD. Let (G, f, h) be a instance of the bounded-Ext RD, with $G = (V, E)$. We can assume (1) $f \leq h$ and, moreover, (2) $h(v) = 0$ implies $h(u) = 2$ for some $u \in N(v)$. We parameterize by $\omega(f)$. For $v \in X \coloneqq V \setminus h^{-1}(0)$, define $T_v \coloneqq \big(N(v) \setminus h^{-1}(\{0,1\})\big) \cup \{v\}$, and for $v \in h^{-1}(0)$, define $T_v \coloneqq \big(N(v) \setminus h^{-1}(\{0,1\})\big)$. Further, we set $\hat{S} \coloneqq (T_v)_{v\in V}$ and we define τ as the correspondence satisfying $\tau(v) = v$ and we let $\overline{f} : X \to \{0,1,2\}$, $v \mapsto f(v)$, i.e., $\overline{f} = f|_X$. Let $H = (X, \hat{S})$. Then, $(H, \tau, \overline{f})$ describes an instance of Ext RHF. The parameter is $\omega\left(\overline{f}\right)$ for this instance. As $f \leq h$, $h(v) = 0$ implies $f(v) = 0$. Thus, $\omega(f) = \omega\left(\overline{f}\right)$, so that the parameter value does not change when moving from the bounded-Ext RD instance to the $\omega(f)$-Ext RHF instance. Clearly, the described construction can be carried out in polynomial time. Trivially, τ is injective. What remains to be shown is the following claim.

▷ Claim 7.4. $(*)$ (G, f, h) is a yes-instance of bounded-Ext RD iff $(H, \tau, \overline{f})$ is a yes-instance of Ext RHF. ◀

As this is also a polynomial-time reduction, it implies following corollary.

▶ **Corollary 7.5.** *Ext RHF is NP-complete.*

We know that Ext RHF is polynomial-time solvable if the correspondence function is surjective. This leads to the question if $\kappa_1(\mathcal{I}) = |\{i \in I | \tau^{-1}(i) = \emptyset\}|$ is a good parameter for this problem for each instance $\mathcal{I} = (H, \tau, f)$ with $H = (V, (s_i)_{i\in I})$, $\tau : V \to I$, $f : V \to \{0,1,2\}$, somehow measuring the *distance from triviality* again. In other words, we try to use parameterized complexity to study the phenomenon that classical function properties as surjectivity seem to be crucial for finding polynomial-time algorithms for Ext RHF.

▶ **Theorem 7.6.** $(*)$ *κ_1-Ext RHF is W[3]-complete.*

In the long version, we consider an explicit XP-algorithm for κ_1-Ext RHF; membership in XP already follows from Theorem 7.6. We are discussing other parameterizations that either lead to para-NP-hardness results (Theorem 7.7) or to FPT-results (Theorem 7.9).

▶ **Theorem 7.7.** (∗) *κ_ζ-Ext RHF is para-NP-hard for each parameterization described by $\zeta \in \{|f^{-1}(0)|, |f^{-1}(1)|, |f^{-1}(2)|, |f^{-1}(\{0,2\})|\}$.*

We can use one of the reductions of Theorem 7.7 to prove transversal-hardness.

▶ **Theorem 7.8.** (∗) *If there would be a algorithm to enumerate all minimal Rhf of an instance (H, τ) with polynomial delay, then there is an algorithm that enumerates all minimal hitting sets of a hypergraph H' with polynomial delay.*

▶ **Theorem 7.9.** *κ_ζ-Ext RHF $\in$ FPT for $\zeta \in \{\omega(2-f), |f^{-1}(\{0,1\})|\}$.*

Proof (Sketch). Let $(H = (X, \hat{S}), \tau, f)$ be an instance. The idea is to walk through all functions $g : X \to \{0,1,2\}$ with $f \le g$ and test if g is a minimal Rhf. This runs in FPT-time. Furthermore, we can check in polynomial time if a function is a minimal Rhf (by modifying Algorithm 1). Hence, there are $2^{\omega(2-f(x))}$ or $3^{|f^{-1}(\{0,1\})|}$ many possibilities for g. ◀

8 Conclusions

We have generalized the notion of Roman domination towards hypergraphs by introducing the definition of Rhf. We have proven that all minimal Rhf can be enumerated with polynomial delay if the correspondence function is surjective. This can be seen as a technical answer to our question what causes Roman domination to behave different from classical domination with respect to polynomial-delay enumerability. When the correspondence is not surjective, RHF rather behaves like DS; in particular, its extension problem is W[3]-complete when parameterized by the given pre-solution's weight, and we observe that it is transversal-hard to enumerate all minimal Rhf.

The main open problems in the context of this paper are the following ones:

- How tight is transversal hardness linked to the NP-hardness of a related extension problem? In the line of the studies in this paper, these links were pretty tight. But in general, only one direction is clear: if extensibility can be decided in polynomial time, then enumeration is possible with polynomial delay.For a even more general discussion in this direction, cf. [12, 17, 32, 37]. Also, in [31] graph problems related to Roman domination were studied and there, both polynomial-delay enumeration was shown and NP-hardness of the corresponding extension problem.
- We also do not know if the polynomial-delay enumerability questions that we discussed are really equivalent to the polynomial-delay enumerability of minimal hitting sets.
- We mentioned in the introduction that RD is in FPT, when parameterized in a dual way, meaning, in this case, by $n - k$, where n is the number of vertices of the graph and k is an upper-bound on the weight of the Rdf. It might be interesting to have similar results for the two generalizations of Roman domination introduced in this paper. However, now it is not very clear what the "dual" of the $\omega(f)$-parameterization should be.

We are currently looking for non-trivial graph-classes where bounded-Ext RD is solvable in polynomial time, hence looking onto another tractability frontier.

Notice that up to quite recently, only a handful of (natural) problems have been known to be complete for W[3]. Even today, apart from the extension problems and their relatives that we mentioned throughout this paper, we only know of the problems shown in [11, 14, 15]. Seeing more and more problems these days that are complete for W[3] adds new interest to W[3]. It might be time to attack the 25-years-old open question if W[3] = W*[3], see [18]. According to [15], the current status is: W[3] ⊆ W*[3] ⊆ W[4].

References

1 A. Aazami, J. Cheriyan, and K. R. Jampani. Approximation algorithms and hardness results for packing element-disjoint Steiner trees in planar graphs. *Algorithmica*, 63(1-2):425–456, 2012. doi:10.1007/S00453-011-9540-3.

2 F. N. Abu-Khzam, C. Bazgan, M. Chopin, and H. Fernau. Data reductions and combinatorial bounds for improved approximation algorithms. *Journal of Computer and System Sciences*, 82(3):503–520, 2016. doi:10.1016/J.JCSS.2015.11.010.

3 F. N. Abu-Khzam, H. Fernau, and K. Mann. Roman census: Enumerating and counting Roman dominating functions on graph classes. In J. Leroux, S. Lombardy, and D. Peleg, editors, *48th International Symposium on Mathematical Foundations of Computer Science, MFCS*, volume 272 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.MFCS.2023.6.

4 F. N. Abu-Khzam, H. Fernau, and K. Mann. Minimal Roman dominating functions: Extensions and enumeration. *Algorithmica*, 86:1862–1887, 2024. doi:10.1007/s00453-024-01211-w.

5 B. D. Acharya. Domination in hypergraphs. *AKCE International Journal of Graphs and Combinatorics*, 4(2):117–126, 2007.

6 S. Bermudo and H. Fernau. Combinatorics for smaller kernels: The differential of a graph. *Theoretical Computer Science*, 562:330–345, 2015. doi:10.1016/J.TCS.2014.10.007.

7 S. Bermudo, H. Fernau, and J. M. Sigarreta. The differential and the Roman domination number of a graph. *Applicable Analysis and Discrete Mathematics*, 8:155–171, 2014.

8 T. Bläsius, T. Friedrich, J. Lischeid, K. Meeks, and M. Schirneck. Efficiently enumerating hitting sets of hypergraphs arising in data profiling. In *Algorithm Engineering and Experiments (ALENEX)*, pages 130–143. SIAM, 2019. doi:10.1137/1.9781611975499.11.

9 T. Bläsius, T. Friedrich, J. Lischeid, K. Meeks, and M. Schirneck. Efficiently enumerating hitting sets of hypergraphs arising in data profiling. *Journal of Computer and System Sciences*, 124:192–213, 2022. doi:10.1016/J.JCSS.2021.10.002.

10 T. Bläsius, T. Friedrich, and M. Schirneck. The complexity of dependency detection and discovery in relational databases. *Theoretical Computer Science*, 900:79–96, 2022. doi:10.1016/J.TCS.2021.11.020.

11 H. L. Bodlaender, C. Groenland, and M. Pilipczuk. On the complexity of problems on tree-structured graphs. Technical Report arXiv:2208.12543v3, ArXiv, Cornell University, 2022. doi:10.48550/arXiv:2208.12543v3.

12 F. Capelli and Y. Strozecki. Incremental delay enumeration: Space and time. *Discrete Applied Mathematics*, 268:179–190, 2019. doi:10.1016/J.DAM.2018.06.038.

13 K. Casel, H. Fernau, M. Khosravian Ghadikolaei, J. Monnot, and F. Sikora. On the complexity of solution extension of optimization problems. *Theoretical Computer Science*, 904:48–65, 2022. doi:10.1016/j.tcs.2021.10.017.

14 J. Chen and F. Zhang. On product covering in 3-tier supply chain models: Natural complete problems for W[3] and W[4]. *Theoretical Computer Science*, 363(3):278–288, 2006. doi:10.1016/J.TCS.2006.07.016.

15 Y. Chen, J. Flum, and M. Grohe. An analysis of the W^*-hierarchy. *The Journal of Symbolic Logic*, 72(2):513–534, 2007. doi:10.2178/JSL/1185803622.

16 E. J. Cockayne, P. A. Dreyer Jr., S. M. Hedetniemi, and S. T. Hedetniemi. Roman domination in graphs. *Discrete Mathematics*, 278:11–22, 2004. doi:10.1016/J.DISC.2003.06.004.

17 N. Creignou, M. Kröll, R. Pichler, S. Skritek, and H. Vollmer. A complexity theory for hard enumeration problems. *Discrete Applied Mathematics*, 268:191–209, 2019. doi:10.1016/J.DAM.2019.02.025.

18 R. G. Downey and M. R. Fellows. Threshold dominating sets and an improved characterization of $W[2]$. *Theoretical Computer Science*, 209(1–2):123–140, 1998. doi:10.1016/S0304-3975(97)00101-1.

19 R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

20 P. A. Dreyer. *Applications and Variations of Domination in Graphs.* PhD thesis, Rutgers University, New Jersey, USA, 2000.

21 T. Eiter and G. Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24(6):1278–1304, 1995. `doi:10.1137/S0097539793250299`.

22 H. Fernau. Roman Domination: a parameterized perspective. *International Journal of Computer Mathematics*, 85:25–38, 2008. `doi:10.1080/00207160701374376`.

23 J. Flum and M. Grohe. *Parameterized Complexity Theory.* Springer, 2006.

24 A. Gainer-Dewar and P. Vera-Licona. The minimal hitting set generation problem: Algorithms and computation. *SIAM Journal of Discrete Mathematics*, 31(1):63–100, 2017. `doi:10.1137/15M1055024`.

25 G. Gogic, C. H. Papadimitriou, and M. Sideri. Incremental recompilation of knowledge. *Journal of Artificial Intelligence Research*, 8:23–37, 1998. `doi:10.1613/JAIR.380`.

26 J. Guo, F. Hüffner, and R. Niedermeier. A structural view on parameterizing problems: distance from triviality. In R. Downey, M. Fellows, and F. Dehne, editors, *International Workshop on Parameterized and Exact Computation IWPEC 2004*, volume 3162 of *LNCS*, pages 162–173. Springer, 2004. `doi:10.1007/978-3-540-28639-4_15`.

27 T. W. Haynes, S.T. Hedetniemi, and M. A. Henning, editors. *Topics in Domination in Graphs*, volume 64 of *Developments in Mathematics.* Springer, 2020.

28 M. M. Kanté, V. Limouzy, A. Mary, and L. Nourine. Enumeration of minimal dominating sets and variants. In O. Owe, M. Steffen, and J. A. Telle, editors, *Fundamentals of Computation Theory — 18th International Symposium, FCT*, volume 6914 of *LNCS*, pages 298–309. Springer, 2011. `doi:10.1007/978-3-642-22953-4_26`.

29 M. M. Kanté, V. Limouzy, A. Mary, and L. Nourine. On the enumeration of minimal dominating sets and related notions. *SIAM Journal of Discrete Mathematics*, 28(4):1916–1929, 2014. `doi:10.1137/120862612`.

30 M. Liedloff, T. Kloks, J. Liu, and S.-L. Peng. Efficient algorithms for Roman domination on some classes of graphs. *Discrete Applied Mathematics*, 156(18):3400–3415, 2008. `doi:10.1016/J.DAM.2008.01.011`.

31 K. Mann and H. Fernau. Perfect Roman domination: Aspects of enumeration and parameterization. In A. A. Rescigno and U. Vaccaro, editors, *Combinatorial Algorithms (Proceeding 35th International Workshop on Combinatorial Algorithms IWOCA)*, volume 14764 of *LNCS*, pages 354–368. Springer, 2024. `doi:10.1007/978-3-031-63021-7_27`.

32 A. Mary. *Énumération des dominants minimaux d'un graphe.* PhD thesis, LIMOS, Université Blaise Pascal, Clermont-Ferrand, France, November 2013.

33 A. Mary and Y. Strozecki. Efficient enumeration of solutions produced by closure operations. *Discrete Mathematics & Theoretical Computer Science*, 21(3), 2019. `doi:10.23638/DMTCS-21-3-22`.

34 C. Padamutham and V. S. R. Palagiri. Algorithmic aspects of Roman domination in graphs. *Journal of Applied Mathematics and Computing*, 64:89–102, 2020. `doi:10.1007/S12190-020-01345-4`.

35 A. Pagourtzis, P. Penna, K. Schlude, K. Steinhöfel, D. S. Taylor, and P. Widmayer. Server placements, Roman domination and other dominating set variants. In R. A. Baeza-Yates, U. Montanari, and N. Santoro, editors, *Foundations of Information Technology in the Era of Networking and Mobile Computing, IFIP 17th World Computer Congress – TC1 Stream / 2nd IFIP International Conference on Theoretical Computer Science IFIP TCS*, pages 280–291. Kluwer, 2002. Also available as Technical Report 365, ETH Zürich, Institute of Theoretical Computer Science, 10/2001. `doi:10.1007/978-0-387-35608-2_24`.

36 R. C. Read and R. E. Tarjan. Bounds on backtrack algorithms for listing cycles, paths, and spanning trees. *Networks*, 5:237–252, 1975. `doi:10.1002/NET.1975.5.3.237`.

37 Y. Strozecki. Enumeration complexity. *EATCS Bulletin*, 129, 2019.