

The Richness of CSP Non-redundancy

Abstract

An instance of a constraint satisfaction problem (CSP) is called non-redundant if no constraint is implied by the rest. The non-redundancy (NRD) of a relation R is the maximum number of constraints in a non-redundant instance of $\text{CSP}(R)$, as a function of the number of variables n . Recent progress has shown that non-redundancy is crucially linked to many important questions concerning sparsification, kernelization, query complexity, universal algebra, logic, and extremal combinatorics. Given that non-redundancy is a nexus for these diverse topics, the central goal of this paper is to more deeply understand non-redundancy.

The NRD of an arity- r relation lies between $\Omega(n)$ and $O(n^r)$, and when $r = 2$, exhibits a sharp linear-versus-quadratic dichotomy. Until recently, all known examples were consistent with NRD exhibiting only polynomial growth n^c for integer exponents c . We show that the general behavior of NRD is substantially richer. We prove that for every rational $s \geq 1$, there exists a relation with $\text{NRD} = \Theta(n^s)$. Using connections to high-girth graphs in extremal combinatorics, we further show that infinitely many exponents arise as the NRD of relations even for arity 3.

Inspired by these concrete results, we develop a general algebraic theory of non-redundancy based on universal algebraic objects known as partial promise pattern polymorphisms. As an application of this algebraic theory, we revisit the notion of *Mal'tsev embeddings*, which is the most general technique known to date for establishing that a relation has linear NRD. We resolve several structural questions about Mal'tsev embeddings, via a powerful new tool we develop called *Catalan polymorphisms*. In particular, we give the first example of a relation that admits no Abelian embedding but does embed into the non-Abelian quantum Pauli group.

Contents

1	Introduction	1
1.1	Our Contributions	2
1.2	Technical Highlights	8
2	Preliminaries	13
2.1	Non-redundancy	13
2.2	Conditional Non-redundancy	14
2.3	Embeddings	15
3	Every Possible Rational Exponent is Attainable	16
3.1	Defining the Relations	16
3.2	Lower Bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$	17
3.3	Upper Bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$	18
3.4	Connections to CSP Kernelization	21
3.5	An Alternative Construction with Smaller Arity	21
4	Graph Girth and Binary Conditional NRD Classification	22
4.1	Classification of Binary Conditional NRD	22
4.2	Infinitely Many NRD Exponents for Ternary Predicates	25
4.3	Ternary Predicates Approaching Linear NRD	26
5	Towards An Algebraic Approach: Partial Promise Polymorphisms and Patterns	28
5.1	A Galois Connection	28
5.2	Polymorphism Patterns	33
6	Catalan Polymorphisms and Linear NRD	37
6.1	Catalan Identities and Polymorphisms	38
6.2	Pattern Polymorphism Perspective: Boolean Mal'tsev Embeddings are Balanced	43
6.3	A Non-Abelian Group Embedding	47
6.4	Application: Excluding Mal'tsev Embeddings	50
7	The Algebraic Structure of Patterns and Pattern Powers	51
7.1	FGPP Definitions and Pattern Powers	52
7.2	Gadget Reductions for Conditional Non-redundancy	58
7.3	The Cube-power Polymorphism Patterns	60
7.4	Connections to Hypergraph Turán	63
7.5	Hypergraph Turán Examples and Applications	67
8	Conclusion	70
A	Omitted Details from Section 3	78
A.1	Connections to Kernelization	78
A.2	A Construction of Lower Arity	84

1 Introduction

The seminal works of Karger [51] and Benczur and Karger [7] showed that any undirected graph can be sparsified to a weighted subgraph with only $\tilde{O}(n)$ edges so that all cuts are preserved within a constant multiplicative factor. An active and influential direction in recent years has extended this cut sparsification result to hypergraphs [57, 24, 56], including spectral variants [77, 6, 76, 50, 64, 46] and Cayley graphs [55, 44, 5].

About a decade ago, Kogan and Krauthgamer [57] posed the question of *CSP sparsification*, allowing for constraint types other than the cut constraint. More formally, for a relation $R \subseteq D^r$ over a finite domain D , given an instance $\mathcal{I}$ of $\text{CSP}(R)$ with variable set V , a sparsified instance $\mathcal{I}'$ consists of a (suitably reweighted) subset of constraints with the property that, for every assignment $\sigma : V \rightarrow D$, the weight of constraints satisfied by σ in $\mathcal{I}'$ is within $(1 \pm \varepsilon)$ of the number of constraints satisfied by σ in $\mathcal{I}$. Although there are often many possible sparsifiers, we seek to identify one whose support size (i.e., number of constraints with positive weight) is as small as possible. Concretely, when $R = \text{NEQ} = \{(0, 1), (1, 0)\} \subseteq \{0, 1\}^2$, the associated CSP sparsification is exactly *cut sparsification* in undirected graphs. Allowing for arbitrary relations gives us a much richer set of sparsification of problems to analyze.

Since the work of Kogan and Krauthgamer, several works have expanded our understanding of CSP sparsification in many cases. For example, the case of binary relations ($R \subseteq D^2$ for any finite domain D) is completely understood [37, 20]. A classification of ternary Boolean relations [54] has also been completed. A recent work on sparsifying linear codes [55] also handled the case in which R can be defined by a system of linear *inequations*.

In this backdrop, a recent work by Brakensiek and Guruswami [14] showed the very general result that *every* CSP can be sparsified down to its *non-redundancy*. Non-redundant instances (to be defined shortly) are obvious obstacles to sparsification, and this result shows that they are the only obstacle, for worst-case CSP instances. This general result motivates the systematic study of non-redundancy of relations, which we undertake in this paper. In addition to its intimate connection to sparsification, non-redundancy also arises in the study of other computational questions about CSPs. One such area is that of *kernelization*, where one seeks to distill an instance $\mathcal{I}$ of $\text{CSP}(R)$ to a smaller instance $\mathcal{I}'$ while preserving satisfiability. In this setting, the study of non-redundancy has greatly aided in the efficient construction of kernels [61, 23, 21].

Another related question is that of the query complexity of CSPs in the partial membership query model [72, 11, 10, 9]. Each query has as input a partial assignment to the variable of the instance, and the output to the query is whether this partial assignment satisfies the clauses induced by this assignment. The study of non-redundancy is crucially¹ tied to this query complexity problem and is in fact equal to the *VC-dimension* of the query complexity problem.

More broadly, the non-redundancy notion has historically given rise to algorithmic problems that have received significant attention. For example, checking whether a given constraint is redundant

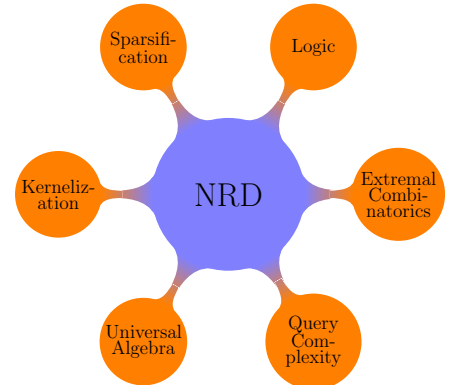


Figure 1: Non-redundancy and its many connections.

¹We remark that this line of work appears to be the first to coin the term *non-redundancy*.

(often formulated as an *implication problem* [74] or *constraint entailment* problem [12]), as well as the task of finding a maximal set of non-redundant constraints of a given instance (computing a *prime* or *minimal* instance—see also [71]). Furthermore, non-redundancy questions have been considered for propositional logic [67, 68], Datalog [65] finite-domain CSPs [31, 49], and various forms of tempo-spatial qualitative reasoning problems [66, 63, 36]. Non-redundancy has also been studied extensively in the context of relational databases where a standard approach is to decompose a database into a suitable *normal form* where one can eliminate redundant entries [27, 1, 29]. Beyond these algorithmic applications, the study of non-redundancy lies at the nexus of extremal combinatorics and universal algebra, with its study revealing many intriguing phenomena and connections [61, 23, 9, 21, 14].

In this paper, we build a comprehensive theory of non-redundancy of relations, and make numerous contributions advancing our understanding of non-redundancy on multiple fronts, including their possible asymptotic behaviors and interrelationships.

With this context, let us now finally define non-redundancy of relations formally. To introduce it, let us ask the simple question: What is the maximum number of possible edges in an n -vertex undirected graph that has no cycle? The answer, of course, is $n - 1$, achieved when the graph is a tree. Viewing a graph as an instance of a constraint satisfaction problem (CSP) with equality constraints imposed on adjacent vertices, an acyclic graph is one where every equality constraint is *non-redundant* in the sense that it is not implied by the collection of all other constraints. A tree is then a maximum-sized collection of non-redundant equality constraints.

Generalizing this, for any relation $R \subseteq D^r$ of arity r over a finite domain D , we say that an instance of $\text{CSP}(R)$, with constraints R applied to certain r -tuples of variables, is non-redundant if each of its constraints is not implied by the others. In other words, for each constraint there is an assignment violating only that constraint, so that dropping any single constraint in a non-redundant instance alters the set of satisfying assignments. The non-redundancy of R , which we denote by $\text{NRD}(R, n)$, is the largest number of possible constraints in a non-redundant instance of $\text{CSP}(R)$, as a function of the number n of variables of the CSP instance. For non-trivial relations $\emptyset \neq R \subsetneq D^r$, $\text{NRD}(R, n)$ lies in the range $[\Omega(n), \mathcal{O}(n^r)]$.

The aforementioned sparsification result [14] shows that for any instance of $\text{CSP}(R)$ on n variables, there is a sub-instance with at most $\tilde{\mathcal{O}}_\varepsilon(\text{NRD}(\bar{R}, n))$ reweighted constraints that preserves the number of original constraints satisfied by all assignments up to $(1 \pm \varepsilon)$ multiplicative factors.² This general result still leaves the question of determining $\text{NRD}(R, n)$ for a given relation R . As we demonstrate in this paper, the NRD of even simple relations can have a surprisingly rich behavior, and the study of NRD is a challenging pursuit with many exciting mathematical connections.

1.1 Our Contributions

We now present the contributions of this paper to the theory of non-redundancy. We divide our contributions into three broad categories. First, we improve our understanding of the possible growth rates of non-redundancy functions. For instance, we show that for any rational number $p/q \geq 1$, there exists a predicate $R_{p,q}$ for which $\text{NRD}(R_{p,q}, n) = \Theta_{p,q}(n^{p/q})$. Second, we shed

²Note that an assignment that witnesses that an $\bar{R}$ -constraint is not redundant also witnesses the corresponding R -constraint cannot be dropped by a sparsifier since an assignment satisfying all but one $\bar{R}$ -constraint satisfies exactly one R -constraint. This makes $\text{NRD}(\bar{R}, n)$ an obvious lower bound for sparsifying worst-case n variable $\text{CSP}(R)$ instances. For $R = \text{NEQ} = \{(0, 1), (1, 0)\} \subseteq \{0, 1\}^2$ (which corresponds to cut sparsification), the relation $\bar{R} = \text{EQ} = \{(0, 0), (1, 1)\}$ has $\text{NRD}(\bar{R}, n) = n - 1$, with trees being extremal non-redundant instances as mentioned earlier.

new light on the nature of predicates R for which $\text{NRD}(R, n)$ grows as a linear function of n . Our contributions here include discovering a new universal algebraic object known as *Catalan polymorphisms* which gives new insights into the nature of Mal'tsev embeddings, a general technique for establishing a predicate has linear non-redundancy. Third, to more deeply understand the logical underpinnings of non-redundancy, we develop a universal algebraic theory of a generalization of non-redundancy known as conditional non-redundancy. As an application, we show that many hypergraph Turán problems in extremal combinatorics (including the notorious Erdős box problem) can be recast in the language of conditional non-redundancy.

1.1.1 Possible Growth Rates for Non-redundancy

We begin by discussing our results on the possible growth rates of non-redundancy. For any non-trivial relation R of arity 2 (over any finite domain), it was known that $\text{NRD}(R, n)$ is either $\Theta(n)$ or $\Theta(n^2)$ [9, 20]. In [14], the authors give the first example of a relation, with arity 3, whose NRD is asymptotically not a polynomial function, and lies between $\Omega(n^{1.5})$ and $\mathcal{O}(n^{1.6})$. This hints at the richness of non-redundancy and raises the question of whether the growth rate of NRD can have a fractional exponent. As our first result, we give a strong positive answer to this question, showing in fact that *every* rational exponent occurs as the NRD of some relation.

Theorem 1.1 (Every fractional exponent as NRD). *For every rational number $p/q \geq 1$, there is a relation $R_{p,q}$ such that $\text{NRD}(R_{p,q}, n) = \Theta_{p,q}(n^{p/q})$.*

The arity of the relation $R_{p,q}$ can be taken to be p . In particular there is an arity 3 relation R with $\text{NRD}(R, n) = \Theta(n^{1.5})$. We discuss this construction in more detail in the technical highlights (Section 1.2). Note that this result may be viewed as a logical analogue of a classical result in extremal combinatorics due to Frankl [39], who showed that every rational p/q appears as the growth rate of some hypergraph Turán problem.

The predicates used to prove Theorem 1.1 also have significant consequences for the concept of *kernelization*, which is a formalization of polynomial-time preprocessing with performance guarantees for NP-hard decision problems [38]. In the context of a CSP over a finite constraint language Γ consisting of one or more finite relations, a kernelization algorithm with size bound $f(n)$ is a polynomial-time algorithm that reduces any n -variable instance of $\text{CSP}(\Gamma)$ into an equisatisfiable instance on at most $f(n)$ constraints. A priori, the NRD growth rate of the predicates in Γ seems unrelated to the best-possible kernelization size bound for $\text{CSP}(\Gamma)$. In one direction, bounds on NRD may be non-constructive and therefore not achievable by a polynomial-time algorithm; in the other direction, a kernelization algorithm is allowed to modify the instance by changing constraints (or even introducing new ones) to arrive at an equisatisfiable instance of small size, while a reduction to a non-redundant instance can only omit constraints. Surprisingly, it turns out that in many cases the best-possible kernelization size for $\text{CSP}(\Gamma)$ coincides with the NRD value of the most difficult predicate in Γ (up to $n^{o(1)}$ factors).

To make this precise, define the *kernelization exponent* of $\text{CSP}(\Gamma)$ as the infimum of all values α for which $\text{CSP}(\Gamma)$ has a kernelization of size $\mathcal{O}(n^\alpha)$. Chen, Jansen, and Pieterse [23] presented a framework based on bounded-degree polynomials to derive kernelization algorithms for various CSPs, which was later generalized by Lagerkvist and Wahlström [61] using algebraic tools. Through matching lower bound constructions, conditional on the complexity-theoretic assumption $\text{NP} \not\subseteq \text{coNP/poly}$, the exact kernelization exponent is known for many CSPs [32, 23, 61]. All kernelization exponents for $\text{CSP}(\Gamma)$ known to date are *integers*. Even for kernelization algorithms outside the realm of CSPs, to the best of our knowledge all NP-hard problems for which matching kerneliza-

tion upper and lower bounds are known (e.g. [32, 33, 48]) have an *integral* kernelization exponent. This raises the question whether $\text{CSP}(\Gamma)$ has an integral kernelization exponent for each choice of Γ . An adaptation of Theorem 1.1 to the setting of kernelization, described in Appendix A.1, provides a strong negative answer to this question. We show that for each choice of integers $p \geq q \geq 1$, there is a constraint language Γ whose kernelization exponent is exactly $\frac{p}{q}$ (assuming $\text{NP} \not\subseteq \text{coNP/poly}$).

As our next result we show that the NRD of arity 3 relations itself has a very rich behavior, and can precisely capture the famous extremal graph theory problem of the largest number of edges in a graph of girth at least $2k$, for any $k \geq 2$. This highlights that the study of non-redundancy may offer powerful new viewpoints on long-standing problems in extremal combinatorics.

Theorem 1.2 (Infinitely many ternary NRD exponents approaching 2). *For every integer $k \geq 2$, there is a ternary relation R_{2k} such that $\text{NRD}(R_{2k}, n) = n \cdot \Theta_k(\text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$ where $\text{ex}(n, \{C_3, \dots, C_{2k-1}\})$ is the largest number of edges in an n -vertex graph with no cycle of length less than $2k$.*

It is believed that $\text{ex}(n, \{C_3, \dots, C_{2k-1}\}) = \Theta_k(n^{1+1/k})$ and it is known that it grows as $n^{1+\Theta(1)/k}$ [40]. In particular, this implies that $\text{NRD}(R_{2k}, n) = n^{2+\Theta(1)/k}$, showing that infinitely many exponents are possible for ternary relations.

Recall that the NRD of ternary relations lies in the range $[\Omega(n), \mathcal{O}(n^3)]$. It is known that for a ternary relation R , either $\text{NRD}(R, n) = \Theta(n^3)$ or $\text{NRD}(R, n) = \mathcal{O}(n^{2.75})$ [21], so the NRD exponent is bounded away from 3. Furthermore, Theorem 1.2 shows that one can approach quadratic NRD arbitrarily closely. Can the NRD exponent of ternary relations approach 1 arbitrarily closely? We will soon discuss a candidate family of predicates to address this question (Theorem 1.5) but first we describe our results for languages with *near-linear* NRD.

1.1.2 On Linear Non-redundancy: Catalan polymorphisms and Embeddings

Recall that by the recent general result of [14] a linear bound on $\text{NRD}(\overline{R})$ implies a sparsifier for the associated $\text{CSP}(R)$ of $n \text{ polylog}(n)$ size. Hence, obtaining near-linear bounds for non-redundancy seems highly desirable, albeit a significant challenge, and has been the primary motivation for earlier works on non-redundancy and sparsification [47, 23, 61, 57, 37, 20, 24, 55, 54, 56].

To hint at the richness of this question, consider the relation $1\text{-in-3} := \{(1, 0, 0), (0, 1, 0), (0, 0, 1)\} \subset \{0, 1\}^3$ whose associated CSP is the classic NP-complete problem 1-in-3-SAT. Note that $1\text{-in-3} = L \cap \{0, 1\}^3$ where $L = \{(a, b, c) \in (\mathbb{Z}/3\mathbb{Z})^3 \mid a + b + c = 1\}$. Thus, a non-redundant instance of $\text{CSP}(1\text{-in-3})$ is a fortiori also a non-redundant instance of $\text{CSP}(L)$, with the same Boolean assignments as non-redundancy witnesses. Being an affine relation, $\text{NRD}(L, n) = \mathcal{O}(n)$ since a maximal non-redundant instance is the same as a basis and a basis must have linear size. Therefore, we also have that $\text{NRD}(1\text{-in-3}, n) = \mathcal{O}(n)$. Such *embeddings* (Section 2.3), and more generally Abelian group embeddings (i.e., embeddings into relations consisting of solutions to an Abelian group equation), have been *the* principal tool for establishing (near) linear bounds in both the NRD [47, 23, 61] and the sparsification context [55, 54]. Curiously, it is additionally known that one can deduce linear NRD for a relation R from so-called *Mal'tsev embeddings* [70, 61], i.e., when $R \subseteq D^r$ is the projection of a relation $S \subseteq E^r$ that is closed under a Mal'tsev term, i.e., a ternary function $\phi : E^3 \rightarrow E$ satisfying the identities $\phi(x, x, y) = \phi(y, x, x) = y$. In this context ϕ is also called a *polymorphism*. The map $\phi(x, y, z) = x \cdot y^{-1} \cdot z$ over a group $(G, \cdot)$ is the quintessential example of a Mal'tsev term.

However, a fundamental open question regarding Mal'tsev embeddings was whether they are more powerful than Abelian group embeddings, and all known Mal'tsev embeddings prior to our

work coincided with Abelian embeddings. As one of our main results, we show that non-Abelian embeddings are indeed more powerful than Abelian embeddings via a concrete predicate.

Theorem 1.3. *There is a relation $\text{PAULI} \subseteq D^6$ with $|D| = 3$ that admits a Mal'tsev embedding, in fact a non-Abelian group embedding, but no Abelian group embedding.*

The non-Abelian group hosting the claimed embedding is the 16-element Pauli group generated by the single qubit unitaries X, Y, Z , hence our choice of name PAULI for the relation. The proof of the above hinges on a key concept that we introduce toward a better understanding of Mal'tsev terms, namely a higher arity generalization which we call *Catalan polymorphisms*. In the group example above of a Mal'tsev term, the higher order alternating product $\psi_m(x_1, x_2, \dots, x_m) = x_1 \cdot x_2^{-1} \cdot x_3 \cdot x_4^{-1} \cdots x_{m-1} \cdot x_m$ for odd $m \geq 3$ also satisfies a “cancellation rule” $\psi_m(x_1, x_2, \dots, x_m) = \psi_{m-2}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m)$ whenever $x_i = x_{i+1}$. We call these *Catalan identities* due to a close relationship with Catalan numbers. As our main technical tool, we show (Theorem 6.4) that higher order polymorphisms obeying such Catalan identities exist on the basis of *any* Mal'tsev term, not just in the group product case. In an Abelian group, the alternating product has extra non-contiguous cancellations (e.g., when $x_i = x_{i+3}$). Exploiting this, we define a relation over $\{x, y, z\}$ based on Catalan identities involving three variables x, y, z that fails to be closed under any commutative alternating product and thus lacks an Abelian embedding. However, the map $x \mapsto X, y \mapsto Y$ and $z \mapsto Z$ embeds the relation into the non-Abelian Pauli group.

In contrast to the above separation, we show that over Boolean domains, Mal'tsev and Abelian group embeddings coincide, answering an open question of [23]. This proof also crucially uses the Catalan polymorphisms as a tool.

Theorem 1.4. *Let $P \subseteq \{0, 1\}^r$ be a relation which embeds into a relation $A \subseteq D^r$ that admits a Mal'tsev polymorphism. Then, P also embeds into a finite Abelian group.*

One of the main remaining open questions about Mal'tsev embeddings for non-Boolean domains is whether the existence of a Mal'tsev embedding over an infinite domain always implies the existence of a Mal'tsev embedding over a finite domain. While this question remains open, we do manage to close the gap between Mal'tsev and group embeddings and prove that a Mal'tsev embedding (over a finite or infinite domain) always implies an embedding into an infinite *Coxeter* group. This, again, uses the novel description of Mal'tsev embeddings via Catalan polymorphisms and represents an important step towards fully characterizing Mal'tsev embeddings. In particular, the question of whether an embedding over an infinite group can always be brought down to the finite has a higher chance of being answerable by existing techniques in group theory than the corresponding question for arbitrary Mal'tsev terms.

While Mal'tsev embeddings suffice for linear NRD, it is not clear if they are the only reason governing linear NRD. We define a family of ternary relations, alluded to earlier, which demonstrate that if Mal'tsev embeddings are necessary for linear NRD, they are “barely so,” in the sense that one can obtain NRD at most $n^{1+\varepsilon}$ for any $\varepsilon > 0$ while lacking a Mal'tsev embedding.

Theorem 1.5. *For every odd m , there is a ternary predicate CYC_m^* over domain size m such that $\text{NRD}(\text{CYC}_m^*, n) \leq \mathcal{O}_m(n^{\frac{m}{m-1}})$. Further, CYC_m^* does not admit a Mal'tsev embedding.*

Here, $\text{CYC}_m^* := \{(x, y, z) \in (\mathbb{Z}/m\mathbb{Z})^3 : x + y + z = 0 \wedge y - x \in \{0, 1\}\} \setminus \{(0, 0, 0)\}$. Once again, the Catalan terms that we introduced and related to Mal'tsev terms play a key role in proving the above statement. We show that CYC_m^* does not admit a Catalan polymorphism of arity $2m - 1$ and thus also lacks a Mal'tsev embedding. This can be viewed as some evidence that CYC_m^* might have super-linear NRD. Establishing such a lower bound remains an open question.

1.1.3 A Theory of Conditional NRD with Connections to Hypergraph Turán

An important tool to study NRD, including establishing aforementioned Theorems 1.1, 1.2, and 1.5, is the concept of *conditional non-redundancy* defined in [14] (although implicit in [9]). Here, when studying $\text{NRD}(R, n)$ for $R \subsetneq D^r$, we focus on a relation $S \supsetneq R$ (often with just one more tuple) and require the non-redundant instance of $\text{CSP}(R)$ to have the property that for every constraint, there is an assignment violating only that constraint by landing within $S \setminus R$ (as opposed to anywhere in $D^r \setminus R$). We denote such conditional non-redundancy by $\text{NRD}(R \mid S, n)$. It is useful since it can be much easier to upper bound $\text{NRD}(R \mid S, n)$ (which is a priori at most $\text{NRD}(R, n)$), and for judicious choices of S , one can also deduce a good upper bound on $\text{NRD}(R, n)$ via a “triangle inequality” $\text{NRD}(R, n) \leq \text{NRD}(R \mid S, n) + \text{NRD}(S, n)$. En route to Theorem 1.2 we give a full *characterization* of all possible behaviors of conditional non-redundancy for binary relations over arbitrary finite domains: For every $R \subsetneq S \subseteq D^2$, either $\text{NRD}(R \mid S, n) = O_D(n)$, or for some $k \geq 2$, $\text{NRD}(R \mid S, n) = \Theta(\text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$.

Conditional NRD appears to be a powerful tool for more fine-grained NRD results (necessary by Theorem 1.1) and it seems apt to study it in the *universal algebraic* setting. Here, the basic idea is to forget about concrete relations and focus on algebraic conditions that can be derived from “higher order” homomorphisms (*polymorphisms*), of which the previously mentioned Mal’tsev term is a notable special case. However, in universal algebra one often assumes *total* polymorphisms that, roughly, correspond to classical gadget reductions between problems. For example, 1-in-3-SAT and 3-SAT are polynomial-time interreducible, and in fact have the same polymorphisms, but different NRD (linear versus cubic). A second complication in the conditional $\text{NRD}(R \mid S, n)$ setting is that we need to take the relationship between R and S into account. We prove that *partial promise polymorphisms*, applied componentwise on tuples from R and producing a tuple in $\tilde{S} = D^r \setminus (S \setminus R)$, govern $\text{NRD}(R \mid S, n)$. However, describing $\text{NRD}(R \mid S, n)$ by explicitly enumerating all possible partial promise polymorphisms is not realistic. As a comparison, imagine a world where we tried to understand calculus through large tables of function values, rather than using equations or further abstractions such as only looking at the degree or the derivative of a polynomial. The same applies here: we need abstractions that allow us to concentrate on certain key properties. We develop a general theory and prove that the most significant algebraic properties come from (restricted) systems of linear identities/equations which we refer to as *polymorphism patterns* and let $\text{Pattern}(R, \tilde{S})$ be the set of patterns satisfied by partial promise polymorphisms of $(R, \tilde{S})$. See Figure 2 for an example of a pattern and a satisfying partial function. We then manage to prove the following general statement.

$U_3(x, x, x, y, y, y, y) = x$	$u_3(0, 0, 0, 1, 1, 1, 1) = 0$	$u_3(1, 1, 1, 0, 0, 0, 0) = 1$
$U_3(x, y, y, x, x, y, y) = x$	$u_3(0, 1, 1, 0, 0, 1, 1) = 0$	$u_3(1, 0, 0, 1, 1, 0, 0) = 1$
$U_3(y, x, y, x, y, x, y) = x$	$u_3(1, 0, 1, 0, 1, 0, 1) = 0$	$u_3(0, 1, 0, 1, 0, 1, 0) = 1$
	$u_3(0, 0, 0, 0, 0, 0, 0) = 0$	$u_3(1, 1, 1, 1, 1, 1, 1) = 1$

Figure 2: The 3-universal/cube pattern U_3 and the resulting partial Boolean function (named u_3). Each Boolean tuple is colored with the color of the identity from which it can be derived. The two constant Boolean tuples can be produced by any identity and are colored in black.

Theorem 1.6. *Let $S_1 \subseteq T_1$ and $S_2 \subseteq T_2$ be pairs of relations. If*

$$\text{Pattern}(S_2, \widetilde{T_2}) \subseteq \text{Pattern}(S_1, \widetilde{T_1})$$

then

$$\text{NRD}(S_1 \mid T_1, n) = \mathcal{O}(n + \text{NRD}(S_2 \mid T_2, n)).$$

The theorem is proven by first establishing a suitable *Galois correspondence* between $\text{Pattern}(\cdot, \cdot)$ and sets of relations closed under conjunctive formulas with functional, unary terms (*functionally guarded promise primitive positive definitions*, fgppp-definitions). This generalizes *fgpp-definitions* (see Carbonnel [21]) to the promise setting. To extend this even further we consider a stronger operation that maps c -tuples of variables in one relation to variables in another, allowing us to, e.g., reduce a problem $\text{CSP}(S_1 \mid T_1)$ on n variables to a problem $\text{CSP}(S_2 \mid T_2)$ on $\mathcal{O}(n^2)$ variables. This naturally gives rise to the kind of fractional non-redundancy exponents seen in Theorem 1.1. The algebraic counterpart is a novel c -ary *power* operator P^c on a pattern P where we group together c -tuples of variables, and view them as a single variable. See Figure 3 for an example when $c = 2$. This pattern is produced by selecting pairs of variables from the columns of any ordered selection of two equations, and viewing it as a pattern over V^2 instead of V . In the figure $(x, x), (x, y), (y, x), (y, y)$ have been renamed to a, b, c, d . By first relating c -ary powers of patterns ($\text{Pattern}^c(S, T)$) to c -tupled fgppp-definitions we obtain the following link to conditional NRD.

Theorem 1.7. *Let (S_1, T_1) and (S_2, T_2) be promise relations over D_1 , respectively, D_2 , and let $c \geq 1$. If*

$$\text{Pattern}^c(S_2, \tilde{T}_2) \subseteq \text{Pattern}(S_1, \tilde{T}_1)$$

then

$$\text{NRD}(S_1 \mid T_1, n) = O_{D_1, D_2, r_1, r_2}(n + \text{NRD}(S_2 \mid T_2, n^c)).$$

These notions generalize and unify all known algebraic reductions in the CSP literature that involve restricted existential quantification [14, 21, 23, 62]. In particular, the power operation makes it possible to tightly relate $\text{NRD}(S_1 \mid T_1, n)$ and $\text{NRD}(S_2 \mid T_2, n)$ even when they have different exponents. This generalizes and unifies techniques implicit in Theorem 1.1 and appears to be a generally useful method for bounding NRD. For example, if $U_3 \notin \text{Pattern}(S, \tilde{T})$ then $\text{NRD}(S \mid T, n) = \Omega(n^3)$ and if $U_3^2 \notin \text{Pattern}(S, \tilde{T})$ then $\text{NRD}(S \mid T, n) = \Omega(n^{3/2})$.

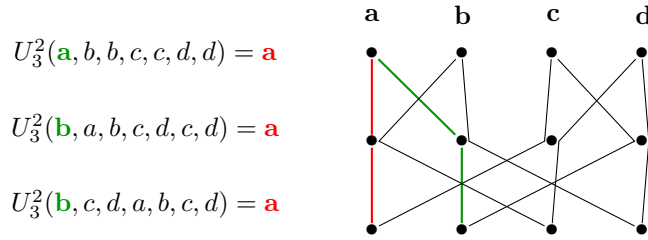


Figure 3: The polymorphism pattern U_3^2 corresponding to the square of U_3 from Figure 2. For example, the first equation is obtained by combining the orange and blue equation in U_3 . The corresponding 3-partite hypergraph is obtained by letting each partite set correspond to variables a, b, c, d in the corresponding equation, and adding a hyperedge for each “column” of the equations.

Given that polymorphism patterns strongly correspond to conditional non-redundancy, an interesting algebraic question is to fix a set of r -ary patterns Q and study $\text{NRD}(S \mid T, n)$ if r -ary (S, T) is preserved by Q . We relate this to hypergraph Turán problems in the following way: by associating Q to a set of hypergraphs $\mathcal{H}(Q)$ one can prove that the maximum conditional non-redundancy possible exactly coincides with the size of the largest hypergraph on n vertices excluding $\mathcal{H}(Q)$ as subhypergraphs. This greatly expands on connections between non-redundancy and hypergraph

Turán problems observed by previous works [9, 21, 14]. See Figure 3 for an example of this translation. To state the theorem formally we let $\text{Inv}^=r(Q)$ be the set of relation pairs of arity r preserved by Q , which we assume is equipped with r disjoint domain types (*sorts*) corresponding to the r partite sets in a hypergraph.

Theorem 1.8. *Let Q be a finite set of (r -ary multisorted) polymorphism patterns. Then there is a finite set $\mathcal{H}(Q)$ of r -partite hypergraphs with the following property: for all $n \in \mathbb{N}$,*

$$\max_{(R,S) \in \text{Inv}^=r(Q)} \text{NRD}(R \mid S, n) = \text{ex}_r(n, \mathcal{H}(Q)),$$

where $\text{ex}_r(n, \mathcal{H}(Q))$ is the size of the largest r -partite r -uniform hypergraph on n vertices excluding $\mathcal{H}(Q)$.

For example, if $U_3^2 \in \text{Pattern}(R, \tilde{S})$ then any conditionally non-redundant instance (X, Y) of $\text{CSP}(R \mid S)$ is free of the hypergraph in Figure 3 (as well as other hypergraphs obtained by restricting the pattern U_3^2). One interpretation of this result is that *finite* conditions Q , which algebraically seem rather weak when interpreted over a specific language in the promise setting, are best studied in the Turán setting, and contain notoriously hard open problems. For example, the cube pattern U_3 from Figure 2 results in the Turán problem that is known as *Erdős box problem* [35, 53, 41], with well-studied generalizations of the box problem corresponding to U_k for arbitrary $k > 3$ [35, 40, 28]. As such, resolving conditional non-redundancy in the algebraically weak setting of finite sets of patterns cannot be done without simultaneously advancing other areas of mathematics. More optimistically, we hope these connections between hypergraph Turán problems and universal algebra could eventually lead to progress on these long-standing problems.

1.2 Technical Highlights

We now discuss the techniques utilized to prove the main results in this paper. Given the breadth of results we prove about NRD, we highlight a couple of key ideas from each area of contribution.

1.2.1 NRD Exponents

All Rational Exponents. Our (first) proof of Theorem 1.1 considers the predicate $\text{OR-DP}_{p,q}$ on $p + p^q$ variables over the domain $\{0, 1\} \cup \{0, 1\}^q$. Intuitively, a tuple $(x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q})$ belongs to $\text{OR-DP}_{p,q}$ when $(x_1, \dots, x_p) \in \text{OR}_p$ (the arity- p OR predicate) and the following *direct product* relation holds: for each $i \in [p^q]$, the value of the *padding* variable $\tilde{x}_i$ equals the bitstring of length q obtained by concatenating the values of $x_{t_1}, \dots, x_{t_q}$, where $(t_1, \dots, t_q) \in [p]^q$ corresponds to the base- p representation of i . This property ensures that whenever two constraints y, y' over $\text{OR-DP}_{p,q}$ use the *same* variable $\tilde{x}_i$ at the same position i , then to satisfy these constraints the values at positions $(t_1, \dots, t_q)$ of the first constraint must match the values at positions $(t_1, \dots, t_q)$ in the second. Intuitively, this allows us to identify the pairs of variables used on these q positions without changing the set of satisfying assignments. After applying all such identifications, each q -tuple of OR-variables appearing on the first p positions of a constraint, is governed without loss of generality by a *unique* padding variable $\tilde{x}_i$. Conversely, we can model any collection of OR_p -constraints over n variables by introducing n^q fresh variables to be used as padding and plugging these into the later positions of $\text{OR-DP}_{p,q}$ constraints. The latter implies that $\text{NRD}(\text{OR-DP}_{p,q}, n + n^q) \geq \text{NRD}(\text{OR}_p, n)$, which is easily seen to be $\Omega_p(n^p)$ since any instance of $\text{CSP}(\text{OR}_p)$ is non-redundant if no two clauses involve the same set of p variables. This leads to

the lower bound $\text{NRD}(\text{OR-DP}_{p,q}, n) \geq \Omega_{p,q}(n^{\frac{p}{q}})$. However, the upper-bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$ requires more work.

The basic idea for the upper bound is that given a non-redundant instance of $\text{OR-DP}_{p,q}$ on variable set X and constraint set Y that has been simplified by the identifications described above, we can derive two set families. First, let $\mathcal{A} \subseteq \binom{X}{p}$ be the p -uniform³ set family containing $\{y_1, \dots, y_p\}$ for $y \in Y$. Furthermore, let $\mathcal{B} \subseteq \binom{X}{q}$ be the q -uniform set family consisting of sets T which are a subset of some $S \in \mathcal{A}$. By the Kruskal-Katona theorem [59, 52], we have that $|\mathcal{B}| \geq \Theta_{p,q}(|\mathcal{A}|^{q/p})$. By utilizing the concept of *conditional* non-redundancy we may assume that $|\mathcal{A}|$ equals the number of constraints $|Y|$ (in a suitable conditional setting, two constraints that employ the *same* p variables in their OR_p part cannot be violated separately from each other) while $|\mathcal{B}|$ equals the number of padding variables in the instance (since each q -tuple of x_j 's has a unique $\tilde{x}_i$ associated to them). Therefore, $\text{NRD}(\text{OR-DP}_{p,q}, n) = \mathcal{O}_{p,q}(n^{p/q})$, as desired.

Note that the arity of this construction is rather large. We are able to reduce the arity of the construction all the way down to p by *puncturing* the construction. In particular, we consider a carefully-chosen subset of the coordinates of $\text{OR-DP}_{p,q}$ and argue that the asymptotic bound still holds. In particular, we are able to drop the first p coordinates of $\text{OR-DP}_{p,q}$ corresponding to OR_p . The main technical difference is instead of Kruskal-Katona we use *Shearer's inequality* to prove the bound. See Section 3.5 for more details.

Graph Girth. Unlike Theorem 1.1, the methods behind Theorem 1.2 are connected to Turán theory. More precisely, for $k \geq 2$, we employ the following bipartite *shoelace* representation (see Figure 4) of the cycle C_{2k} over two copies of the elements $\{0, \dots, k-1\}$ as

$$C_{2k} := \{(0,0), (0,1), (1,0), (1,2), (2,1), \dots, (k-2, k-1), (k-1, k-2), (k-1, k-1)\} \subseteq \{0, \dots, k-1\}^2.$$

That is, $(x, y) \in C_{2k}$ if $(x, y) \in \{(0,0), (k-1, k-1)\}$ or $|x-y| = 1$. Also let $C_{2k}^* := C_{2k} \setminus \{(0,0)\}$ be the cycle minus an edge. A crucial observation we make is that *bipartite* graphs are conditionally non-redundant instances of $\text{CSP}(C_{2k}^* \mid C_{2k})$ if and only if they have girth at least $2k$ (see Figure 4). Since there is a constant-factor reduction from general to bipartite instances (see Lemma 2.3), we have that $\text{NRD}(C_{2k}^* \mid C_{2k}, n) = \Theta(\text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$. In fact, for general binary predicates $R \subsetneq S \subseteq D^2$, we have that $\text{NRD}(R \mid S, n) = \Theta(\text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$, where $2k-1$ is the minimum length of a path between any pair of points $(x_0, y_0) \in S \setminus R$ within the bipartite graph on $2|D|$ vertices whose edges come from R . The special case of no path existing (i.e., infinite girth) corresponds to $\text{NRD}(R \mid S, n) = \mathcal{O}(n)$.

We can lift bounds on the conditional NRD of binary predicates into the non-conditional NRD of ternary predicates by considering the following family of ternary predicates. For any $k \geq 2$, define

$$R_{2k} := (C_{2k} \times \{0,1\}) \setminus \{(0,0,0)\}, \quad S_{2k} := C_{2k} \times \{0,1\}.$$

Using techniques similar to those for analyzing $\text{NRD}(C_{2k}^* \mid C_{2k}, n)$, we can prove that $\text{NRD}(R_{2k} \mid S_{2k}, n) = \Theta(n \cdot \text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$. The key observation though is that S_{2k} can be decomposed into a binary relation (C_{2k}) and a unary relation ($\{0,1\}$). As a result, we have that $\text{NRD}(S_{2k}, n) = \Theta(n^2)$. Therefore, by the triangle inequality, we have that $\text{NRD}(R_{2k}, n) \leq \text{NRD}(R_{2k} \mid S_{2k}, n) + \text{NRD}(S_{2k}, n) = \Theta(n \cdot \text{ex}(n, \{C_3, \dots, C_{2k-1}\}))$. Note this technique does not work for analyzing $\text{NRD}(C_{2k}^* \mid C_{2k}, n)$ because for $k \geq 3$, we have that $\text{NRD}(C_{2k}, n) = \Theta(n^2) > \text{NRD}(C_{2k}^* \mid C_{2k})$, so the triangle inequality would give an upper bound of n^2 instead of $n^{1+\Theta(1)/k}$.

³We can avoid repeated variables by assuming without loss of generality that our instance is *multipartite*. See Section 2 for more details.

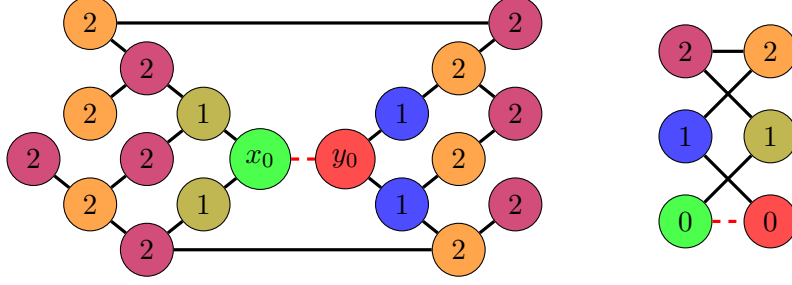


Figure 4: This figure illustrates how a bipartite graph G with girth 6 (left) is a non-redundant instance of $\text{CSP}(C_6^* \mid C_6)$ (right). In particular, the colors illustrate a homomorphism from G to C_6 such that only the target edge (x_0, y_0) is mapped to $(0, 0)$. This coloring is generalized to bipartite graphs of arbitrary girth in Lemma 4.3.

1.2.2 Linear NRD and Catalan Polymorphisms

We now dive deeper into the structure of the *Catalan polymorphisms* which drive our main results on linear non-redundancy and Mal'tsev embeddings (Theorems 1.3, 1.4, and 1.5). In particular for any domain D , we say that a sequence of operators $\{\psi_m : D^m \rightarrow D \mid m \in \mathbb{N} \text{ odd}\}$ are *Catalan polymorphisms* (Definition 6.2) if they have the following properties: (1) for all $x \in D$, $\psi_1(x) = x$, and (2) for all odd $m \in \mathbb{N}$ at least 3 and all $i \in [m - 1]$, we have that for all $x_1, \dots, x_m \in D$,

$$x_i = x_{i+1} \implies \psi_m(x_1, \dots, x_m) = \psi_{m-2}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m). \quad (1)$$

Our critical technical result (Theorem 6.4) is that given *any* operator $\varphi : D^3 \rightarrow D$ satisfying the Mal'tsev identities that $\varphi(x, x, y) = \varphi(y, x, x) = y$ for all $x, y \in D$, we can generate *all* the Catalan polymorphisms by (1) composing φ with itself and (2) identifying input coordinates as equal. As base cases, we have that $\psi_1(x) = \varphi(x, x, x)$ and $\psi_3(x, y, z) = \varphi(x, y, z)$. For the first nontrivial case of ψ_5 , we have that $\psi_5(x_1, x_2, x_3, x_4, x_5)$ is

$$\varphi(\varphi(x_1, x_2, \varphi(x_3, x_4, x_5)), \varphi(\varphi(x_1, x_2, x_3), x_3, \varphi(x_3, x_4, x_5)), \varphi(\varphi(x_1, x_2, x_3), x_4, x_5)). \quad (2)$$

One can verify (1) holds for (2) with some casework (see Example 6.7). This construction is generalized to all odd m by using a recursive formula reminiscent of the recursive formula for the Catalan numbers. Importantly, Theorem 6.4 may be seen as a “group theoretic” analogue of the classical complexity theory result of Valiant (see e.g., [43] and references therein) that a majority gate on three bits can simulate an arbitrary majority gate. That is, we prove that a “Mal'tsev gate” can simulate an arbitrarily large “Catalan gate,” although the depth of the reduction is exponential in m . Note that Valiant's reduction is of much importance in the current study of promise CSPs (e.g., [2, 13, 3]), perhaps suggesting Catalan polymorphisms could also be used to advance the theory of promise CSPs. The Catalan polymorphisms also have natural partial analogues which turn out to coincide with the so-called *universal partial Mal'tsev polymorphisms* [61] arising from any Mal'tsev embedding. However, the latter are poorly understood even for small arities, and have no clear connection to group theory.

The true power of the Catalan polymorphisms is revealed in its applications. In particular, the cancellation laws of the Catalan polymorphisms show that any Mal'tsev embedding “acts like” an embedding into an infinite group. More precisely, we prove that every predicate with a Mal'tsev embedding also embeds into an infinite group known as the Coxeter group. This immediately

resolves Theorem 1.4, as [23] already showed that the result is true when a Boolean predicate has an infinite group embedding.⁴

As previously discussed, the identities underlying Catalan polymorphisms were crucial in inspiring the construction of PAULI in Theorem 1.3. In particular, for any operator $f: D^5 \rightarrow D$, consider the following six identities involving arbitrary $x, y, z \in D$.

$$\begin{array}{lll} f(x, x, y, y, z) = z, & f(x, x, z, y, y) = z, & f(z, x, x, y, y) = z, \\ f(x, y, y, x, z) = z, & f(z, x, y, y, x) = z, & f(x, y, z, x, y) = z. \end{array}$$

The first five identities are satisfied by any Catalan polymorphism ψ_5 , but the last one can only be guaranteed when the operator arises from an Abelian group. As such, if we define $\text{PAULI} \in \{x, y, z\}^6$ to consist of 5 tuples corresponding to each argument of the identities above (e.g., the first argument implies $(x, x, z, x, z, x) \in \text{PAULI}$), then we can deduce that PAULI does not embed into an Abelian group as that would imply that $(z, z, z, z, z, z) \in \text{PAULI}$. However, as previously mentioned, PAULI embeds into the Pauli group by mapping each of x, y, z to its corresponding generator. Thus, the insights from Catalan polymorphisms allow us to separate Abelian embeddings from non-Abelian embeddings and conclude that no analogue of Theorem 1.4 exists even when the domain size is 3.

Finally, we note that Catalan polymorphisms greatly streamline proofs showing that a predicate lacks a Mal'tsev embedding. In particular, previous proofs that predicates lack Mal'tsev embeddings (in e.g., [23, 9, 61]) relied on constructing complicated Mal'tsev gadgets similar to that of (2), but the gadgets were found in an ad-hoc manner for the particular problem. With the Catalan polymorphisms being “universal” in the sense that they alone imply an infinite group embedding, to rule out an explicit predicate P having a Mal'tsev embedding, we just need to check if there is an application of the Catalan polymorphisms to P which results in a tuple not in P . This is precisely how we prove the lack of Mal'tsev embeddings in Theorem 1.5, while the upper bound follows from techniques developed to prove Theorem 1.2.

1.2.3 Theory of conditional NRD

Let us highlight some of the most important techniques in our algebraic approach for studying conditional non-redundancy. First, for a pair of relations $S \subseteq T \subseteq D^r$ we say that an n -ary partial function $f: X \rightarrow D$ (for some $X \subseteq D^n$ called the *domain* of f) is a *partial promise polymorphism* of (S, T) if

$$(f(s_1^{(1)}, \dots, s_1^{(n)}), \dots, f(s_r^{(1)}, \dots, s_r^{(n)})) \in T$$

for each sequence $(s_1^{(1)}, \dots, s_r^{(1)}), \dots, (s_1^{(n)}, \dots, s_r^{(n)}) \in S$ where each involved application is defined. If we let $\text{pPol}(S, T)$ be the set of all such partial promise polymorphisms we obtain a well-behaved algebraic object which is closed under two natural algebraic notions: identifying arguments (*variable identification minors*) and weakening a function by making it less defined (taking *subfunctions*). We call these objects (strong) *partial minions* since the corresponding algebraic objects in the total function setting are known as minions [3]. Here, it is interesting to observe that both *promise polymorphisms* and *partial polymorphisms* are well-studied algebraic objects: the former underpins the algebraic approach for studying *promise CSPs* [2, 13, 3], and the latter has seen important applications in fine-grained complexity of CSPs [62] and kernelization/non-redundancy [9, 21, 23, 61]. Curiously, the combination is virtually unexplored and only briefly appears in a technical lemma by Pippenger [73].

⁴Independently, [54] proved a similar result to [23] when the predicate has an *Abelian* group embedding.

While it is then relatively straightforward to establish that $\text{pPol}(S, \tilde{T})$ (recall that $\tilde{T} = D^r \setminus (T \setminus S)$) governs the conditional non-redundancy $\text{NRD}(S \mid T, n)$, obtaining concrete bounds based on completely classifying all partial promise polymorphisms seems daunting. Here, inspired by Lagerkvist & Wahlström [62] and Carbonnel [21] we instead concentrate on the aforementioned *identities* $\text{Pattern}(S, \tilde{T})$ satisfied by the partial functions in $\text{pPol}(S, \tilde{T})$. This abstraction allows us to concentrate on the functions in $\text{pPol}(S, \tilde{T})$ that actually affect $\text{NRD}(S \mid T, n)$ to a significantly large degree and avoids exhaustive characterizations of partial promise polymorphisms.

In the algebraic approach for studying CSPs one often seeks a way to relate functional objects (such as patterns, or polymorphisms) with relational objects that can be used to obtain reductions [4]. Often, one wants a *Galois connection* between these sets which implies that the objects are dually related, in the sense that small functional objects correspond to large relational objects, and vice versa. Hence, we set forth to develop a Galois connection between patterns and sets of relations closed under certain restricted first-order formulas. Here, on the relational side we consider conjunctive formulas where atoms are allowed to use functional expressions of bounded arity $c \geq 1$ (c -fgppp-definitions). The reason for allowing such expressions is that they relate NRD within a power of c , which, in the light of Theorem 1.1, seems necessary to capture even a fragment of the rich non-redundancy landscape. On the operational side we then introduce a c -ary *power* operation on a pattern P (P^c). This operation is by design very similar to the power operation in universal algebra, well-known from Birkhoff’s HSP theorem [19]. But applying it to patterns is strikingly different, and seemingly much more powerful, than to directly apply it to partial functions. We study powers of the cube pattern (from Figure 2) in extra detail since the *absence* of this pattern gives a general and useful method for proving lower bounds on NRD.

In order to relate invariance under a finite set of patterns to hypergraph Turán problems we first extend the algebraic machinery to *multisorted* patterns in order to apply it to r -partite CSP instances. The key idea is then to interpret the patterns over the r different sorts in an r -partite CSP instance, since the application of such a pattern to constraints in a CSP instance can be used to witness redundancy. For each finite set of r -sorted patterns Q we then construct a corresponding set of hypergraphs $\mathcal{H}(Q)$ (recall Figure 3) whose Turán number closely matches the maximum possible conditional non-redundancy of all r -ary relation pairs $S \subseteq T$ invariant under Q . Choosing the correct set $\mathcal{H}(Q)$ is not completely trivial and requires subtle algebraic properties of Q . We provide several concrete examples of patterns and what this implies for the corresponding Turán problems and note that the assumption that Q is finite is, algebraically speaking, a rather weak condition. This is in contrast to the CSP dichotomy theorem where tractability boiled down to the existence of a single condition, but more in line with tractability results for PCSPs which generally require an infinite family of polymorphisms [13]. Fundamentally, the technical reason is that there is no general way to *compose* functions in this setting while preserving a given relation pair. Hence, algebraically oriented methods for analyzing conditional NRD generally require more than a finite set of conditions.

Organization

In Section 2, we list some definitions and known results for non-redundancy and related problems. In Section 3, we prove Theorem 1.1 which shows that non-redundancy can capture any rational exponent at least one. In Section 4, we classify the conditional non-redundancy of every binary promise language and as corollaries prove Theorem 1.2 and the upper bound for Theorem 1.5. In Section 5, we extend the techniques in Section 3 and Section 4 and formulate the essential components of a comprehensive algebraic theory of non-redundancy, based around promise partial

polymorphisms definable by patterns. In Section 6, we build on Section 5 to give a number of new insights on the structure of Mal'tsev embeddings, including the theory of Catalan polymorphisms. Here, Theorems 1.3, 1.4, and 1.5 are proved. In Section 7 we complete the algebraic theory with a new power operator and describe the resulting gadget reductions that preserve conditional NRD, thus proving Theorem 1.6 and Theorem 1.7, among other results. We also apply the algebraic theory to better understand the connections between non-redundancy and hypergraph Turán theory, and prove e.g. Theorem 1.8, as well for consequences for relations invariant under a well-known pattern in universal algebra. In Section 8, we give some concluding remarks and discuss directions for future research. In Appendix A, we give some proofs omitted from Section 3.

2 Preliminaries

We begin the main part of the paper by introducing important notation. Throughout, for an r -ary tuple t and $i \in [r]$ we write either t_i or $t[i]$ for the i th component of t , depending on the context. We defer most of the algebraic notions to Section 5 where we also restate some basic notions in an algebraic context.

2.1 Non-redundancy

We now formally define the main notation we need to discuss non-redundancy. In particular, we adapt the notation used in previous works [9, 21, 14]. We use the terms *predicate* and/or *relation* to describe a set $R \subseteq D^r$, where D , the *domain* is finite unless otherwise specified, and $r \geq 1$ is a positive integer. We call the elements $t \in R$ *tuples* and say that R is *non-trivial* if it contains at least one tuple and does not contain all tuples in D^r . We consider a *structure* to be a pair⁵ of the form (D, R) . A *homomorphism* between structures $(D, R \subseteq D^r)$ and $(E, S \subseteq E^r)$ is a map $\sigma : D \rightarrow E$ such that for every $t \in R$, we have that $\sigma(t) := (\sigma(t_1), \dots, \sigma(t_r)) \in S$. If D and E are clear from context, then we say that σ is a homomorphism from R to S .

Like in [14], we view instances of a CSP as structures themselves. In particular, given a relation $R \subseteq D^r$, we can view any structure $\Psi := (X, Y \subseteq X^r)$ as an *instance* of $\text{CSP}(R)$. We usually call X the *variables* and Y the *clauses*, but we may also refer to X as *vertices* and Y as *(hyper)edges*. We also usually reserve the symbol $n := |X|$. We let $\text{sat}(R, Y)$ denote the set of homomorphisms from (X, Y) to (D, R) (note that D and X can be inferred from context). More traditionally, $\text{sat}(R, Y)$ can be thought of as the set of assignments $\sigma : X \rightarrow D$ such that each clause $y \in Y$ has that $\sigma(y) \in R$.

We say that a clause $y \in Y$ is *non-redundant* if $\text{sat}(R, Y) \subsetneq \text{sat}(R, Y \setminus \{y\})$. In other words, there exists a homomorphism σ from $Y \setminus \{y\} \rightarrow R$ which does not extend to Y . We call such a map σ a *witness* for y . Semantically, this means that the clause y cannot be logically inferred from $Y \setminus \{y\}$. We say that (X, Y) is a non-redundant instance of $\text{CSP}(R)$ if every $y \in Y$ is non-redundant. This leads to our definition of non-redundancy of a predicate.

Definition 2.1 (Non-redundancy of a predicate [9]). Given a relation $R \subseteq D^r$, the maximum number of clauses of a non-redundant instance of $\text{CSP}(R)$ on n variables is denoted by $\text{NRD}(R, n)$.

⁵Usually in the context of CSPs, a structure can have many more symbols, corresponding to a multitude of relations. However, in the context of non-redundancy and related questions one can always without loss of generality consider a single relation, since the non-redundancy of a language Γ is within a constant factor of the largest non-redundancy of a single relation $R \in \Gamma$; see for example [21, 55, 14].

Often, it is more convenient to work with instances of $\text{CSP}(R)$ which are *multipartite*. More formally, for $R \subseteq D^r$, we say that an instance (X, Y) of $\text{CSP}(R)$ is r -partite if there is a partition of the variable set as $X := X_1 \dot{\cup} \dots \dot{\cup} X_r$, such that for each clause $(y_1, \dots, y_r) \in Y$ and each $i \in [r]$ we have $y_i \in X_i$. To avoid having to keep track of the exact arity, we say an instance of $\text{CSP}(R)$ is *multipartite* if it is r -partite where r is the arity of R . As we show below, we can obtain bounds on general non-redundancy by analyzing the non-redundancy of multipartite instances. Therefore it will be convenient to have notation for this quantity.

Definition 2.2 (Non-redundancy of multipartite instances). Given a relation $R \subseteq D^r$ and $n \in \mathbb{N}$, we define $\text{NRD}^*(R, n)$ to be the maximum number of clauses of any *multipartite* non-redundant instance of $\text{CSP}(R)$ on n variables.

Clearly $\text{NRD}(R, n) \geq \text{NRD}^*(R, n)$. As observed by [14], the reverse inequality is approximately true.

Lemma 2.3 (Implicit in [14]). Let R be a relation of arity r over domain D and let $\Psi = (X, Y)$ be a non-redundant instance of $\text{CSP}(R)$. Then there is a non-redundant r -partite instance of $\text{CSP}(R)$ on $r|X|$ variables and $|Y|$ clauses. Consequently, $\text{NRD}(R, n) \leq \text{NRD}^*(R, r \cdot n)$.

Proof. Let (X, Y) be a non-redundant instance of $\text{CSP}(R)$. Define $X' := X \times [r]$ and map each $(y_1, \dots, y_r) \in Y$ to $((y_1, 1), \dots, (y_r, r)) \in Y' \subseteq (X')^r$. It remains to argue that the instance (X', Y') is non-redundant. For any clause $y' \in Y'$ corresponding to a clause $y \in Y$, by the non-redundancy of (X, Y) there is an assignment $\sigma_y: X \rightarrow D$ that satisfies all clauses $Y \setminus \{y\}$ but not y . It is easy to verify that $\sigma'_y: X' \rightarrow D$ defined via $\sigma'_y((x, i)) = \sigma(x)$ satisfies all clauses of $Y' \setminus \{y'\}$ but not y' . Hence (X', Y') is a non-redundant instance on $r|X|$ variables and $|Y|$ clauses. It follows that the non-redundancy of general instances on n variables is upper-bounded by the non-redundancy of multipartite instances on rn variables, so that $\text{NRD}(R, n) \leq \text{NRD}^*(R, r \cdot n)$. $\square$

We also now observe that if $c > 1$ is a constant, then $\text{NRD}(R, n)$ and $\text{NRD}(R, cn)$ are also related by a constant.

Lemma 2.4 (Implicit in e.g., [21]). For $R \subseteq D^r$ and for any $c > 1$, we have that

$$\text{NRD}(R, cn) = O_c(\text{NRD}(R, n)).$$

Proof. Let (X, Y) be a non-redundant instance of $\text{CSP}(R)$ with $|X| = cn$ and $|Y| = \text{NRD}(R, cn)$. Let $X' \subseteq X$ be a random subset of X of size n , and let $Y' := Y \cap X'^r$. If n is sufficiently large, the probability that any $y \in Y$ is also in Y' approximately $(1 - o(1))/c^r$. Thus, in expectation we have that $\text{NRD}(R, n) \geq (1 - o(1)) \text{NRD}(R, cn)/c^r$. This proves the asymptotic bound. $\square$

We remark that the analogous bound holds for multipartite instances.

2.2 Conditional Non-redundancy

An important concept introduced in [14] is the notion of *conditional* non-redundancy, which refines the notion of non-redundancy (Definition 2.1) to impose more structure on the witness to a clause being non-redundant. More formally, given relations $P \subsetneq Q \subseteq D^r$ and a structure $(X, Y \subseteq X^r)$, we say that a clause $y \in Y$ is (*conditionally*) *non-redundant* for $\text{CSP}(P \mid Q)$ if there is a witness $\sigma \in \text{sat}(P, Y \setminus \{y\}) \setminus \text{sat}(P, Y)$ which also satisfies $\sigma \in \text{sat}(Q, Y)$. In other words, although our

witness σ cannot satisfy y (i.e., $\sigma(y) \notin P$), we want to make sure that $\sigma(y) \in Q$. As such, we call Q the *scaffolding* of P .

Analogous to (non-conditional) non-redundancy, we say that an instance (X, Y) of $\text{CSP}(P \mid Q)$ is conditionally non-redundant if every $y \in Y$ is (conditionally) non-redundant. Often we drop “conditionally” if it is clear from context (e.g., “Let (X, Y) be a non-redundant instance of $\text{CSP}(P \mid Q)$ ”). Analogous to Definition 2.1, we let $\text{NRD}(P \mid Q, n)$ denote the size of the largest non-redundant instance of $\text{CSP}(P \mid Q)$ on n variables. The multipartite version $\text{NRD}^*(P \mid Q, n)$ can be defined analogously, and we remark that Lemma 2.3 and Lemma 2.4 can be extended to the conditional NRD setting with the same proof methods.

The key utility in studying conditional non-redundancy is that it satisfies a triangle inequality, which we quote as follows.

Lemma 2.5 (Triangle inequality [14]). *For any predicates $R \subsetneq S \subsetneq T \subseteq D^r$ and $n \in \mathbb{N}$, we have that*

$$\text{NRD}(R \mid T, n) \leq \text{NRD}(R \mid S, n) + \text{NRD}(S \mid T, n).$$

In particular, if $T = D^r$, we have that

$$\text{NRD}(R, n) \leq \text{NRD}(R \mid S, n) + \text{NRD}(S, n).$$

We note that the analogous bounds for multipartite instances, obtained by replacing NRD by NRD^* , also hold by the exact same proof.

2.3 Embeddings

Given two structures (D, R) and (E, S) , we say that R *embeds* into S if there exists an injective homomorphism $\sigma : D \rightarrow E$ from R to S which satisfies the additional property that $\sigma(R) = \sigma(D)^r \cap S$. As noticed by prior works (e.g., [61, 9]), if an embedding exists, then any non-redundant instance of $\text{CSP}(R)$ is a non-redundant instance of $\text{CSP}(S)$, so $\text{NRD}(R, n) \leq \text{NRD}(S, n)$. As such, we can use embeddings to prove non-redundancy upper bounds.

Often, one seeks to find embeddings into structures (E, S) admitting a special algebraic operation known as a *Mal'tsev polymorphism* (or Mal'tsev term). That is, there is a map $\varphi : E^3 \rightarrow E$ such that for any $x, y \in E$, we have that $\varphi(x, y, y) = \varphi(y, y, x) = x$ and further for any $t_1, t_2, t_3 \in S$ we have that $\varphi(t_1, t_2, t_3) \in S$ (where φ on tuples is applied componentwise). A canonical example is when $(E, \cdot)$ is a group and $S \subseteq E^r$ is a coset. Then, we can define $\varphi(x, y, z) = x \cdot y^{-1} \cdot z$. As we shall see much later (Section 6), this canonical example is in a strong sense the *only* example.

Mal'tsev polymorphisms and embeddings imply linear NRD upper bounds, as formalized below. Such a bound follows from a characterization made by Bulatov and Dalmau [16] of structures with a Mal'tsev polymorphism.

Theorem 2.6 ([61, 9]). *If $R \subseteq D^r$ is a predicate with a Mal'tsev polymorphism, then $\text{NRD}(R, n) \leq |D|n$. Likewise, if R has an embedding into a structure (E, S) with a Mal'tsev polymorphism then $\text{NRD}(R, n) \leq |E|n$.*

We also note that embeddings are themselves a special case of a much broader class of reductions known as *fgpp*-definitions which were first studied by Carbonnel [21] and extended by us in Section 5 in the context of conditional non-redundancy.

3 Every Possible Rational Exponent is Attainable

A key question about non-redundancy is the set of possible exponents which can appear. More precisely, for a finite domain D and an arity $r \in \mathbb{N}$, we define the *exponent* $\alpha(R)$ of a non-trivial relation $R \subseteq D^r$ to be the least $\alpha \geq 1$ such that $\text{NRD}(R, n) = \mathcal{O}(n^{\alpha+o(1)})$. The main result of this section is that any rational number at least 1 can be the exponent of a finite predicate.

Theorem 3.1. *For every rational number $p/q \geq 1$, there exists a nontrivial relation $\text{OR-DP}_{p,q}$ for which $\alpha(\text{OR-DP}_{p,q}) = p/q$. In fact, we have that $\text{NRD}(\text{OR-DP}_{p,q}, n) = \Theta_{p,q}(n^{p/q})$.*

Remark 3.2. This result is similar in spirit to the hypergraph Turán result of Frankl [39].

3.1 Defining the Relations

We start by defining the predicates with the prescribed exponents and their domains.

Definition 3.3. For $p, q \in \mathbb{N}$ with $p \geq q$, we define $D_{p,q} := \{0, 1\} \cup \{(b) \mid b \in \{0, 1\}^q\}$.

Hence the domain $D_{p,q}$ consists of the Boolean values 0 and 1, together with one element (b) for each bitstring of length q . Note that when $q = 1$, the elements (0) and (1) corresponding to bitstrings of length 1 are distinct from the elements 0 and 1 denoting *true* and *false*. We now define two relations over $D_{p,q}$: $\text{OR-DP}_{p,q}$ and $\text{DP}_{p,q}$. Here, $\text{OR-DP}_{p,q}$ is the relation referred to in Theorem 3.1 and $\text{DP}_{p,q}$ is a scaffolding for $\text{OR-DP}_{p,q}$ which will assist in upper-bounding $\text{NRD}(\text{OR-DP}_{p,q}, n)$ via Lemma 2.5.

Definition 3.4. For $p, q \in \mathbb{N}$ with $p \geq q$, we define relations $\text{OR-DP}_{p,q} \subsetneq \text{DP}_{p,q} \subseteq D_{p,q}^{p+p^q}$ as follows. Let $\text{id}_{x_{p,q}}: [p^q] \rightarrow [p]^q$ be a bijection between the integers $\{1, \dots, p^q\}$ and q -tuples over $\{1, \dots, p\}$. The only tuples contained in $\text{OR-DP}_{p,q}$ and $\text{DP}_{p,q}$ are of the form $T = (x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q})$ where $x_i \in \{0, 1\}$ for each $i \in [p]$ and $\tilde{x}_i \in \{(b) \mid b \in \{0, 1\}^q\}$ for each $i \in [p^q]$. A tuple T of this form is contained in $\text{DP}_{p,q}$ if and only if the following condition is met:

1. For each $i \in [p^q]$, let $S^{(i)} = (s_1^{(i)}, \dots, s_q^{(i)}) = \text{id}_{x_{p,q}}(i) \in [p]^q$ be the q -tuple indexed by i . The value of $\tilde{x}_i$ in tuple T should be equal to the bitstring $(x_{s_1^{(i)}}, \dots, x_{s_q^{(i)}})$. This constraint acts like a “direct product,” hence the name $\text{DP}_{p,q}$.

We say that a tuple $T \in \text{DP}_{p,q}$ is also in $\text{OR-DP}_{p,q}$ if and only if the following additional condition is met:

2. At least one of the values x_i for $i \in [p]$ is equal to 1. (So the x_i ’s are not all 0; this part of the constraint acts as OR_p .)

In other words, a tuple T is contained in $\text{OR-DP}_{p,q}$ if its first p values are Boolean and at least one of them is 1, while the remaining p^q variables each take a bitstring as value, and the bitstring of variable $\tilde{x}_i$ is exactly the concatenation of the values of the x -variables in the positions encoded by $\text{id}_{x_{p,q}}(i)$. Note that for each tuple $T \in \text{OR-DP}_{p,q}$, the values of the first p entries of the tuple uniquely determine the value of the remaining p^q entries. We leverage this “redundancy” introduced by the $\tilde{x}_i$ variables in order to precisely connect the non-redundancy of the predicate OR_p with the non-redundancy of our constructed predicate $\text{OR-DP}_{p,q}$. In particular, by utilizing Kruskal-Katona theorem, we will be able to tightly quantify the impact these $\tilde{x}_i$ variables have on the redundancy.

Example 3.5. For $p = 3$ and $q = 2$, the relation $\text{OR-DP}_{3,2}$ has arity $3 + 3^2 = 12$. If we project it down to a more manageable arity by only keeping the “essential” padding variables for tuples $(1, 2)$, $(1, 3)$ and $(2, 3)$ we get the 6-ary relation

$$\begin{aligned} \text{OR-DP}'_{3,2} = \{ & (1, 0, 0, (10), (10), (00)), \\ & (0, 1, 0, (01), (00), (10)), \\ & (0, 0, 1, (00), (01), (01)), \\ & (1, 1, 0, (11), (10), (10)), \\ & (1, 0, 1, (10), (11), (01)), \\ & (0, 1, 1, (01), (01), (11)), \\ & (1, 1, 1, (11), (11), (11)) \} \end{aligned}$$

when using the bijection $\text{id}_{x_{p,q}}$ given by listing the size-2 subsets of $[3]$ in lexicographic order. The relation $\text{DP}_{3,2}$ has the additional tuple $(0, 0, 0, (00), \dots, (00))$.

We proceed by analyzing the non-redundancy exponent of $\text{OR-DP}_{p,q}$.

3.2 Lower Bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$.

We start with proving a lower bound for the non-redundancy of $\text{OR-DP}_{p,q}$, as the result is a bit easier to obtain and illustrates the intent of the construction.

Lemma 3.6. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$, we have that*

$$\text{NRD}(\text{OR-DP}_{p,q}, n + n^q) \geq \text{NRD}(\text{OR}_p, n) \geq \binom{n}{p}.$$

Therefore, $\text{NRD}(\text{OR-DP}_{p,q}, n) \geq \Omega_{p,q}(n^{p/q})$.

Proof. For arbitrary n , let (X, Y) be the instance of $\text{CSP}(\text{OR}_p)$ with $|X| = \{x_1, \dots, x_n\}$ and $Y = \{(x_{i_1}, \dots, x_{i_p}) \mid i_1 < \dots < i_p\}$. Then $|Y| = \binom{n}{p}$ and the instance (X, Y) is non-redundant, since for any choice of $i_1 < \dots < i_p$ the assignment that maps $x_{i_1}, \dots, x_{i_p}$ to 0 and all other variables to 1, satisfies all clauses except $(x_{i_1}, \dots, x_{i_p})$. Consider $X' := X \cup X^q$. We define a set $Y' \subseteq (X')^{p+p^q}$. For each $y \in Y$, we construct a corresponding $y' \in Y'$ as follows.

- if $i \in [p]$, then $y'_i = y_i$.
- if $i > p$, then $y'_i = (y_{t_1}, \dots, y_{t_q}) \in X^q$, where $t_1, \dots, t_q \in [p]$ are chosen such that

$$(t_1, \dots, t_q) = \text{id}_{x_{p,q}}(i - p),$$

where we recall that $\text{id}_{x_{p,q}}$ is a bijection between $[p^q]$ and $[p]^q$.

It remains to argue that (X', Y') is a non-redundant instance of $\text{CSP}(\text{OR-DP}_{p,q})$. Consider each $y \in Y$ and its corresponding $y' \in Y'$. Since (X, Y) is a non-redundant instance of $\text{CSP}(\text{OR}_p)$, there exists an assignment $\sigma_y : X \rightarrow \{0, 1\}$ which satisfies every clause $y' \in Y$ except y itself. We now lift this to a map $\sigma'_{y'} : X' \rightarrow D_{p,q}$ as follows:

- For each $x \in X' \cap X$, we set $\sigma'_{y'}(x) := \sigma_y(x) \in D_{p,q} \cap \{0, 1\}$.

- For each $\tilde{x} := (x_1, \dots, x_q) \in X' \cap X^q$, we set $\sigma'_{y'}(\tilde{x}) := (\sigma_y(x_1), \dots, \sigma_y(x_q)) \in D_{p,q} \cap \{(b) \mid b \in \{0, 1\}^q\}$.

For each $z' \in Y'$ with corresponding $z \in Y$, observe that $\sigma'_{y'}$ satisfies Property 1 of Definition 3.4 for z' since for every $i \in [p^q]$ with $(t_1, \dots, t_q) = \text{id}_{p,q}(i)$, we have that

$$\sigma'_{y'}(z'_{i+p}) = (\sigma_y(z'_{t_1}), \dots, \sigma_y(z'_{t_q})) = (\sigma'_{y'}(z'_{t_1}), \dots, \sigma'_{y'}(z'_{t_q})).$$

However, $\sigma'_{y'}$ satisfies Property 2 of Definition 3.4 for z' if and only if σ_y satisfies z . In other words, $\sigma'_{y'}$ satisfies z' if and only if $z' \neq y'$. Thus, since the choice of y' was arbitrary, we have that (X', Y') is a non-redundant instance of $\text{CSP}(\text{OR-DP}_{p,q})$ of size $|Y'| = |Y| = \binom{n}{p}$, as desired. The asymptotic bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$ then immediately follows. $\square$

Remark 3.7. This lower bound technique can be made much more general using the theory of *c-fgpp reductions*. See Section 5 for more details.

3.3 Upper Bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$

In this section we prove an upper bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$ using the Kruskal-Katona theorem. We first introduce some relevant terminology.

A *set system* over a universe U is a collection $\mathcal{F}$ of subsets of U . A set system is p -uniform for $p \in \mathbb{N}$ if all sets in $\mathcal{F}$ have cardinality p . The q -shadow of a p -uniform set family $\mathcal{F}$ over U is the q -uniform set family $\{X' \in \binom{U}{q} \mid \exists X \in \mathcal{F}: X' \subseteq X\}$. So the sets in the shadow of $\mathcal{F}$ are precisely those sets that can be obtained from a member of $\mathcal{F}$ by removing $p - q$ elements. Kruskal and Katona proved lower bounds on the size of the q -shadow of a uniform set family.

Theorem 3.8 (Kruskal-Katona theorem, simplified form [59, 52]). *Let $p, q \in \mathbb{N}$ with $p \geq q$. Let U be a finite set, let $\mathcal{A}$ be a p -uniform set family over U , and let $\mathcal{B}$ be the q -shadow of $\mathcal{A}$. If $|\mathcal{A}| = \binom{x}{p}$, then $|\mathcal{B}| \geq \binom{x}{q}$.*

Using this ingredient, we prove an upper bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$.

Theorem 3.9. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$, there exist a constant $C_{p,q} > 0$ such that $\text{NRD}(\text{OR-DP}_{p,q}, n) \leq C_{p,q} \cdot n^{\frac{p}{q}}$.*

We achieve this upper bound by applying the triangle inequality of Lemma 2.5 with $\text{DP}_{p,q}$ as scaffolding. By Lemma 2.3 and Lemma 2.4, it suffices to consider multipartite instances. The bulk of the technical work is then devoted to bounding $\text{NRD}^*(\text{OR-DP}_{p,q} \mid \text{DP}_{p,q}, n)$.

Lemma 3.10. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$ it holds that $\text{NRD}^*(\text{OR-DP}_{p,q} \mid \text{DP}_{p,q}, n) \leq \mathcal{O}_{p,q}(n^{\frac{p}{q}})$.*

Proof. Let $c_{p,q} \geq 1$ be any constant such that $\binom{m}{p} \leq c_{p,q} \cdot \binom{m}{q}^{\frac{p}{q}}$ for all $m \in \mathbb{N}$ with $m \geq p$, which exists since $\binom{m}{q} \geq 1$ and $\binom{m}{r} \in \Theta(m^r)$ for all $r \in \mathbb{N}$. To prove the lemma, we will use induction on the number of variables n to show that $\text{NRD}^*(\text{OR-DP}_{p,q} \mid \text{DP}_{p,q}, n) \leq 2p \cdot c_{p,q} \cdot n^{\frac{p}{q}}$. So consider a multipartite instance (X, Y) of $\text{CSP}(\text{OR-DP}_{p,q})$ on n variables that is a non-redundant instance of $\text{CSP}(\text{OR-DP}_{p,q} \mid \text{DP}_{p,q})$. Choose $X_1 \dot{\cup} X_2 \dot{\cup} \dots \dot{\cup} X_{p+q} = X$ such that $Y \subseteq X_1 \times X_2 \times \dots \times X_{p+q}$. We will refer to the variables $\bigcup_{i>p} X_i$ as *padding variables*.

The base case deals with instances on $n \leq p + p^q$ variables. Since each variable can only appear on a single coordinate in a multipartite instance, such instances have at most one clause. Hence $|Y| \leq 1 \leq 2p \cdot c_{p,q} \cdot n^{\frac{p}{q}}$.

For the induction step, assume the instance has $n > p + p^q$ variables and assume that the bound holds for multipartite non-redundant instances with fewer variables. We say that our instance (X, Y) is *reduced* if it has the following property:

- **Reduced:** for any $i \in [p^q]$ with its associated tuple $(t_1, \dots, t_q) = \text{id}_{p,q}(i)$, for any $y, y' \in Y$, the following implication holds: if $y_{i+p} = y'_{i+p}$, then $y_{t_j} = y'_{t_j}$ for all $j \in [q]$.

Next, we show how to derive the desired bound on $|Y|$ by induction if (X, Y) is not reduced. Afterwards, we will derive the bound for reduced instances.

Non-reduced instances. Suppose there exists an index $i \in [p^q]$ and clauses $y, y' \in Y$ such that $y_{i+p} = y'_{i+p}$ while $y_{t_j} \neq y'_{t_j}$ for some $j \in [p]$. So clauses y, y' use the same padding variable $y_{i+p} = y'_{i+p}$ that governs a tuple of positions including t_j , while the variables y_{t_j} and y'_{t_j} appearing at that position differ. Consider the instance $(\tilde{X}, \tilde{Y})$ of $\text{CSP}(\text{OR-DP}_{p,q})$ derived from (X, Y) as follows. Let $\tilde{X} := X \setminus \{y'_{t_j}\}$ be obtained by removing variable y'_{t_j} and let $\tilde{Y}$ contain a copy $\tilde{y}$ of each $y \in Y$, in which the occurrence of variable y'_{t_j} is replaced by y_{t_j} (if it occurs at all).

It is clear that $|\tilde{X}| = n - 1$. Next we argue that $|\tilde{Y}| = |Y|$. Assume for a contradiction that $|\tilde{Y}| < |Y|$, which implies that Y contains two distinct clauses y^1, y^2 such that y^2 is obtained from y^1 by replacing variable y'_{t_j} (which occurs at position t_j due to the multipartite property) by variable y_{t_j} . Note in particular that this implies $y^1_{p+i} = y^2_{p+i}$, so that the two clauses use the same variable at the padding position for tuple $\text{id}_{p,q}(i)$. Since (X, Y) is $\text{DP}_{p,q}$ -conditionally non-redundant, there is an assignment σ that $\text{DP}_{p,q}$ -satisfies all clauses of Y , while it $\text{OR-DP}_{p,q}$ satisfies $Y \setminus \{y^1\}$ but not y^1 . We now use the structure of the scaffolding predicate $\text{DP}_{p,q}$ to reach a contradiction. Since σ $\text{DP}_{p,q}$ -satisfies y^1 , by Property 1 of Definition 3.4 the j -th bit of $\sigma(y^1_{p+i})$ equals the value of $\sigma(y^1_{t_j})$. Analogously, the j -th bit of $\sigma(y^2_{p+i})$ equals the value of $\sigma(y^2_{t_j})$. Since $y^1_{p+i} = y^2_{p+i}$ we find $\sigma(y^1_{t_j}) = \sigma(y^2_{t_j})$. As $\sigma(y^1) \in \text{DP}_{p,q} \setminus \text{OR-DP}_{p,q}$, by Property 2 of Definition 3.4 the first p variables of y^1 evaluate to 0 under σ . Since y^2 can be obtained from y^1 by replacing y_{t_j} by y'_{t_j} , which has the same value under σ , we find that the first p variables of y^2 also evaluate to 0 under σ . But then $\sigma(y^2) \notin \text{OR-DP}_{p,q}$, a contradiction.

From the previous argument we conclude that $|\tilde{Y}| = |Y|$. To bound $|Y|$ in terms of $|\tilde{X}|$, it therefore suffices to bound $|\tilde{Y}|$ in terms of $|\tilde{X}|$ via induction. To that end, we argue that $(\tilde{X}, \tilde{Y})$ is again $\text{DP}_{p,q}$ -conditionally non-redundant. So consider an arbitrary clause $\tilde{y}^* \in \tilde{Y}$ that was obtained from some $y^* \in Y$ by replacing y'_{t_j} (if it occurs) by y_{t_j} . Since the original instance (X, Y) was non-redundant, there is an assignment σ^* that satisfies $Y \setminus \{y^*\}$ but not y^* . Moreover, since (X, Y) is $\text{DP}_{p,q}$ -conditionally non-redundant, we know σ^* $\text{DP}_{p,q}$ -satisfies all clauses in Y . In particular, this means that Property 1 of Definition 3.4 holds for clauses y and y' under σ^* . Recall that our choice of y, y' as a violation of being reduced ensured that $y_{t_j} \neq y'_{t_j}$ while $y_{p+i} = y'_{p+i}$, with $t_j \in \text{id}_{p,q}(i)$. Property 1 therefore ensures that $\sigma^*(y_{t_j}) = \sigma^*(y'_{t_j})$. Consequently, replacing an occurrence of y'_{t_j} in a clause of Y by y_{t_j} to obtain a clause of $\tilde{Y}$ does not affect to which tuple σ^* maps the clause. Since $\tilde{\sigma}^*$ is just σ^* restricted to the remaining variables, it follows that $\tilde{\sigma}^*$ is a witness for $\tilde{y}^*$. This establishes that $(\tilde{X}, \tilde{Y})$ is $\text{DP}_{p,q}$ -conditionally non-redundant. Since it has $n - 1$ variables, we may apply induction to conclude that $|Y| = |\tilde{Y}| \leq 2p \cdot c_{p,q} \cdot |\tilde{X}|^{\frac{p}{q}} \leq 2p \cdot c_{p,q} \cdot n^{\frac{p}{q}}$.

Reduced instances. Finally, we show how to derive the desired bound on $|Y|$ when (X, Y) is reduced. Since (X, Y) is $\text{DP}_{p,q}$ -conditionally non-redundant, for each clause $y \in Y$ its projection onto the first p coordinates $(y_1, \dots, y_p)$ is unique: if two tuples $y, y' \in Y$ have the same variables at the first p positions, it is impossible to $\text{DP}_{p,q}$ -satisfy both y, y' while $\text{OR-DP}_{p,q}$ -satisfying exactly one of them. The only way to falsify $\text{OR-DP}_{p,q}$ while satisfying $\text{DP}_{p,q}$ is to have the first p variables map to 0. That would, however, also $\text{OR-DP}_{p,q}$ -violate any other clause with the same first p variables.

Since (X, Y) is multipartite, each variable $x \in X_j$ can only appear at one position j in clauses of Y . Therefore, the set system $\mathcal{A} := \{A_y = \{y_1, \dots, y_p\} \mid y \in Y\}$ is p -uniform and contains a unique set for each $y \in Y$. We may therefore bound $|Y|$ via $|\mathcal{A}|$. To do so, we consider the q -shadow $\mathcal{B}$ of $\mathcal{A}$. Let $X' := X_{p+1} \dot{\cup} X_{p+2} \dot{\cup} \dots \dot{\cup} X_{p+p^q}$ be the padding variables of X . We argue that $|\mathcal{B}| \leq |X'|$ by showing that there is an injection $f: \mathcal{B} \rightarrow X'$ from $\mathcal{B}$ to X' . For each set $B \in \mathcal{B}$, we define its value $f(B)$ as follows. Since $\mathcal{B}$ is the q -shadow of $\mathcal{A}$, there exists $A_y = \{y_1, \dots, y_p\} \in \mathcal{A}$ with $B \subseteq A_y$. In fact, there might be multiple such A_y ; fix one arbitrarily. Let $t_1, \dots, t_q$ be the q indices for which $y_{t_j} \in B$, which correspond to the elements of A_y that generated its q -shadow B . Let $i := \text{id}_{p,q}^{-1}(t_1, \dots, t_q) \in [p^q]$ be the index of the q -tuple. We set $f(B) := y_{p+i}$, i.e., we map the q -set B in the shadow to the padding variable in the set X' that appears at the position of clause y controlling the tuple of variables that form the shadow. The fact that f is an injection now follows directly from the fact that (X, Y) is multipartite and reduced.

Since f is an injection from $\mathcal{B}$ to $|X'|$ we infer $|\mathcal{B}| \leq |X'|$: the number of distinct padding variables in the instance is at least as large as the q -shadow $\mathcal{B}$ of $\mathcal{A}$. Let $m \geq p$ be the largest integer such that $|\mathcal{A}| \geq \binom{m}{p}$. This implies that $|\mathcal{A}| < \binom{m+1}{p} \leq 2p \binom{m}{p}$. By the Kruskal-Katona theorem (Theorem 3.8) we have $|X'| \geq |\mathcal{B}| \geq \binom{m}{q}$. Hence we can now derive:

$$|Y| = |\mathcal{A}| \leq 2p \cdot \binom{m}{p} \leq 2p \cdot c_{p,q} \cdot \binom{m}{q}^{\frac{p}{q}} \leq 2p \cdot c_{p,q} \cdot |X'|^{\frac{p}{q}} \leq 2p \cdot c_{p,q} \cdot n^{\frac{p}{q}},$$

which concludes the proof. $\square$

To facilitate a bound on $\text{NRD}^*(\text{OR-DP}_{p,q}, n)$ via conditional non-redundancy, we now derive a bound on $\text{NRD}^*(\text{DP}_{p,q}, n)$ by essentially showing it is an affine constraint.

Lemma 3.11. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$ it holds that $\text{NRD}^*(\text{DP}_{p,q}, n) \leq \mathcal{O}_{p,q}(n)$.*

Proof. We prove this with an application of Theorem 2.6. More concretely, the projection constraints defining $\text{DP}_{p,q}$ can be viewed as projection constraints inside of the Abelian group $G := ((\mathbb{Z}/2\mathbb{Z})^{q+2}, +)$. We define a coset $H \subseteq G^{p+p^q}$ as follows. We define H to be the set of $(h^1, \dots, h^{p+p^q}) \in G^{p+p^q}$ satisfying the following conditions:

- (i) If $i \in [p]$, then $h^i = (0, 0, \dots, 0)$ or $h^i = (1, 0, \dots, 0)$.
- (ii) If $i \in [p^q]$, then $h^{i+p} = (0, 1, x_1, \dots, x_q)$ for some $x_1, \dots, x_q \in \mathbb{Z}/2\mathbb{Z}$.
- (iii) For any $i \in [p^q]$ and $(t_1, \dots, t_q) = \text{id}_{p,q}(i)$, we have for every $j \in [q]$ that $h_{j+2}^{i+p} = h_1^{t_j}$.

It is straightforward to verify that H is a coset. Consider the map $\sigma: D_{p,q} \rightarrow (\mathbb{Z}/2\mathbb{Z})^{q+2}$ defined as follows:

- For $b \in D_{p,q}$, we set $\sigma(b) = (b, 0, 0, \dots, 0)$.

- For $(b_1, \dots, b_q) \in D_{p,q}$, we set $\sigma(b_1, \dots, b_q) = (0, 1, b_1, \dots, b_q)$.

By inspection, properties (i)–(iii) ensure that $\sigma(\text{DP}_{p,q}) = H \cap \sigma(D_{p,q})^{p+p^q}$. In particular, for any $T = (x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q}) \in H \cap \sigma(D_{p,q})^{p+p^q}$, we have that (i) ensures that $x_1, \dots, x_p$ correspond to $\sigma(0)$ or $\sigma(1)$, (ii) ensures that $\tilde{x}_1, \dots, \tilde{x}_{p^q}$ each correspond to some $\sigma(b_1, \dots, b_q)$, and (iii) ensures Property 1 is satisfied in the definition of $\text{DP}_{p,q}$ (Definition 3.4). Therefore, σ is an embedding of $\text{DP}_{p,q}$ into H .

Thus, since H is an Abelian coset (and thus has a Mal'tsev polymorphism), by Theorem 2.6 we have that $\text{NRD}^*(\text{DP}_{p,q}, n) \leq \text{NRD}^*(H, n) \leq \mathcal{O}_{p,q}(n)$, as desired. $\square$

By using the established upper-bounds together with the triangle inequality for conditional non-redundancy and the reduction to multipartite instances, the desired bound on $\text{NRD}(\text{OR-DP}_{p,q}, n)$ now follows easily.

Lemma 3.12. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$ it holds that $\text{NRD}(\text{OR-DP}_{p,q}, n) \leq \mathcal{O}_{p,q}(n^{\frac{p}{q}})$.*

Proof. The lemma follows from the following derivation:

$$\begin{aligned}
& \text{NRD}(\text{OR-DP}_{p,q}, n) \\
& \leq \text{NRD}^*(\text{OR-DP}_{p,q}, (p + p^q)n) && \text{By Lemma 2.3.} \\
& \leq \text{NRD}^*(\text{OR-DP}_{p,q} \mid \text{DP}_{p,q}, (p + p^q)n) + \text{NRD}^*(\text{DP}_{p,q}, (p + p^q)n) && \text{By Lemma 2.5.} \\
& \leq \mathcal{O}_{p,q}(((p + p^q)n)^{\frac{p}{q}}) + \mathcal{O}_{p,q}((p + p^q)n) && \text{Lemma 3.10 and 3.11.} \\
& \leq \mathcal{O}_{p,q}(n^{\frac{p}{q}}). && \square
\end{aligned}$$

The proof of Theorem 3.1 now follows directly from the lower bound in Lemma 3.6 with the matching upper bound in Lemma 3.12.

3.4 Connections to CSP Kernelization

The methods used to prove Theorem 3.1 can be used to prove an even stronger fact: for every $p \geq q$, there exists a constraint language $\text{SAT-DP}_{p,q}$ such that instances of $\text{CSP}(\text{SAT-DP}_{p,q})$ on n variables can be efficiently kernelized to have at most $\mathcal{O}(n^{p/q})$ clauses. The constraint language $\text{SAT-DP}_{p,q}$ consists of two relations: the aforementioned $\text{OR-DP}_{p,q}$ as well as $\text{CUT} := \{(0, 1), (1, 0)\} \subseteq \{0, 1\}^2 \subseteq D_{p,q}^2$ which is simply the negation relation on the Boolean values 0 and 1.

The reason we need to add this extra relation is that $\text{CSP}(\text{OR-DP}_{p,q})$ has a near-trivial polynomial time algorithm: if the Boolean variables and the padding variables (as defined in the proof of Theorem 3.9) overlap, we know the instance is unsatisfiable. Otherwise, we can assign 1 to each Boolean variable and $(1, \dots, 1)$ to each padding variable. Note that instances of $\text{CSP}(\text{OR}_p)$ have the analogous issue.

By introducing CUT , one can then reduce $\text{CNF-}p\text{-SAT}$ to $\text{CSP}(\text{SAT-DP}_{p,q})$, allowing the application of the kernelization lower bound for $\text{CNF-}p\text{-SAT}$ for $p \geq 3$ by Dell and van Melkebeek proved [33]. See Appendix A.1 for the full details on porting Theorem 3.1 to CSP kernelization.

3.5 An Alternative Construction with Smaller Arity

We conclude this section by noting that Theorem 3.1 can be proved for predicates of (much) smaller arity by *puncturing* $\text{OR-DP}_{p,q}$ (i.e., dropping coordinates). For any $p, q \in \mathbb{N}$ with $p \geq q$.

Let $E_{p,q} := \{0,1\}^q$. Let $\mathcal{F} \subseteq \binom{[p]}{q}$ be a family of sets. Define the operation $\text{proj}_{\mathcal{F}} : \{0,1\}^p \rightarrow E_{p,q}^{\mathcal{F}}$ as follows

$$\text{proj}_{\mathcal{F}}(x) := (x|_S : S \in \mathcal{F}).$$

Given this operation, we can define a pair of predicates analogous to $\text{OR-DP}_{p,q}$ and $\text{DP}_{p,q}$ from Definition 3.4. We define the following pair of predicates.

$$\begin{aligned} \text{OR-DP}_{\mathcal{F}} &:= \{\text{proj}_{\mathcal{F}}(x) : x \in \text{OR}_p\} \subset E_{p,q}^{\mathcal{F}} \\ \text{DP}_{\mathcal{F}} &:= \{\text{proj}_{\mathcal{F}}(x) : x \in \{0,1\}^p\} \subset E_{p,q}^{\mathcal{F}}. \end{aligned}$$

Unlike $\text{OR-DP}_{p,q}$ and $\text{DP}_{p,q}$, $\text{OR-DP}_{\mathcal{F}}$ and $\text{DP}_{\mathcal{F}}$ no longer have Boolean variables keeping track of the underlying Boolean assignment. This allows for the predicates to have even smaller arity (with a tradeoff of a somewhat trickier proof).

We derive non-redundancy bounds for a specific class of set families $\mathcal{F}$. We say that $\mathcal{F}$ is q/p -regular if there exists a probability distribution over $\mathcal{F}$ such that for every $i \in [p]$, the probability that a sample S from $\mathcal{F}$ satisfies $i \in S$ is exactly q/p . Our bound is as follows.

Theorem 3.13. *For any $p \geq q \geq 1$ and for any q/p -regular $\mathcal{F} \subseteq \binom{[p]}{q}$, $\text{NRD}(\text{OR}_{\mathcal{F}}, n) = \Theta_q(n^{p/q})$.*

The most notable difference between the proof of Theorem 3.1 and Theorem 3.13 is that the former uses Kruskal-Katona in the upper bound while the latter uses Shearer’s inequality. Details are available in Appendix A.2.

4 Graph Girth and Binary Conditional NRD Classification

One of the first classification theorems in the context of non-redundancy is the classification of *binary* relations. In particular, given any relation $R \subseteq D^2$, we either have that $\text{NRD}(R, n) = O_D(n)$ or $\text{NRD}(R, n) = \Omega_D(n^2)$. This was proved in the context of non-redundancy by [9] and an analogous result was proved in the context of sparsification by [20].

In this section, we seek to generalize these results to the setting of *conditional* non-redundancy. In particular, given a domain D and $R \subsetneq S \subseteq D^2$, we seek to classify the possible asymptotics of $\text{NRD}(R \mid S, n)$. As we shall see, unlike the dichotomy theorem for ‘unconditional’ non-redundancy, there are infinitely many possible exponents possible for $\text{NRD}(R \mid S, n)$, corresponding to the possible girths a bipartite graph may have. This classification is given in Section 4.1.

As an application, we construct two families of (unconditional) ternary predicates. The first family of predicates constructed in Section 4.2 proves that infinitely many NRD exponents are possible for ternary predicates—in contrast to the constructions in Section 3 which only produce finitely many examples for each arity. The second family of predicates, constructed in Section 4.3, shows that ternary predicates which lack of Mal’tsev embedding can have non-redundancies getting arbitrarily close to linear.

4.1 Classification of Binary Conditional NRD

Recall that $\text{NRD}(R \mid S, n)$ captures the maximum possible size of a non-redundant instance of $\text{CSP}(R)$ where the non-satisfying assignments must come from $S \setminus R$ (i.e., all witnessing solutions are satisfying assignments to $\text{CSP}(S)$). First, we reduce to the case that $|S \setminus R| = 1$.

Proposition 4.1. *For any $R \subsetneq S \subseteq D^r$, we have that*

$$\max_{t \in S \setminus R} \text{NRD}(R \mid R \cup \{t\}, n) \leq \text{NRD}(R \mid S, n) \leq |D|^r \max_{t \in S \setminus R} \text{NRD}(R \mid R \cup \{t\}, n).$$

Proof. The lower bound is trivial. For the upper bound, consider any non-redundant instance $(X, Y \subseteq X^r)$ of $\text{CSP}(R \mid S)$ and note that we can partition the $y \in Y$ based on which $t_y \in S \setminus R$ is assigned by the witness corresponding to y . If we partition Y into $|S \setminus R|$ parts according to the value of t_y , then (X, Y) can be viewed as the disjoint union of non-redundant instances of $\text{CSP}(R \mid R \cup \{t\})$ for $t \in S \setminus R$. $\square$

We now work toward describing the binary characterization. Given an integer $k \geq 2$, we let $C_{2k} \subseteq \{0, 1, \dots, k-1\}^2$ denote the $2k$ -cycle. More explicitly, we use a bipartite representation where the two copies of the domain $\{0, \dots, k-1\}$ represent different vertices, and define

$$C_{2k} := \{(0, 0), (0, 1), (1, 0), (1, 2), (2, 1), \dots, (k-2, k-1), (k-1, k-2), (k-1, k-1)\}.$$

We also let $C_{2k}^* := C_{2k} \setminus \{(0, 0)\}$.

Remark 4.2. Equivalently, the $2k$ -cycle can be represented by the following predicate:

$$\tilde{C}_{2k} := \{(0, 0), (0, 1), (1, 1), (1, 2), (2, 2), \dots, (k-1, k-1), (k-1, 0)\}$$

with $\tilde{C}_{2k}^* = \tilde{C}_{2k} \setminus \{(0, 0)\}$. This latter representation will be of use in Section 4.3. Since the bipartite graphs C_{2k} and $\tilde{C}_{2k}$ are isomorphic, one can show that $\text{NRD}(C_{2k}^* \mid C_{2k}, n) = \Theta(\text{NRD}(\tilde{C}_{2k}^* \mid \tilde{C}_{2k}, n))$. Such equivalences are formalized in the theory of fgppp-definitions in Section 5.

We now classify the (bipartite) non-redundant instances of $\text{CSP}(C_{2k}^* \mid C_{2k})$. See Figure 4 for an illustration of the proof technique.

Lemma 4.3. *For any $k \geq 2$, a bipartite graph $G = (A, B, E \subseteq A \times B)$ is a non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$ iff G has girth at least $2k$.*

Remark 4.4. Note that since $C_4 = \{0, 1\}^2$, a non-redundant instance of $\text{CSP}(C_4^* \mid C_4)$ is precisely a non-redundant instance of $\text{CSP}(\text{OR}_2)$. Hence, that particular case was already handled by previous works (e.g., [37, 9]).

Proof. First assume that G is a non-redundant instance but has a cycle of length $\ell < 2k$. Since G is bipartite, ℓ is even. Thus, there exists an injective homomorphism $f : C_\ell \rightarrow G$. Since G is non-redundant, for each $e \in C_\ell$, there exists a map $g_e : G \rightarrow C_{2k}$ such that $(g_e \circ f)^{-1}((0, 0)) = \{(0, 0)\}$. However, since $\ell < 2k$, every homomorphism from C_ℓ to C_{2k} is constant, so we have a contradiction. Thus, every non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$ has girth at least $2k$.

Now, assume that G has girth at least $2k$. Pick any edge $e := (x_0, y_0)$ of G , we seek to construct a homomorphism $f_e : G \rightarrow C_{2k}$ such that $f_e^{-1}(0, 0) = \{e\}$. For every $v \in A \cup B$ (the vertex set of G), let $d_e(v)$ be the distance from v to $\{x_0, y_0\}$ in G . In particular, $d_e(x_0) = d_e(y_0) = 0$. Any vertex in a different connected component than $\{x_0, y_0\}$ is assigned a distance of ∞ . Our homomorphism f_e is then

$$f_e(v) := \min(d_e(v), k-1).$$

To see why this is valid, consider any edge (x, y) of G . If $d_e(x), d_e(y) \geq k-1$, we are fine as $(f_e(x), f_e(y)) = (k-1, k-1) \in C_{2k}$. Otherwise, note that by the triangle inequality $|d_e(x) - d_e(y)| \leq 1$. If $|d_e(x) - d_e(y)| = 1$, then we have that $(f_e(x), f_e(y)) \in C_{2k}$. Otherwise, assume for sake of contradiction that $d_e(x) = d_e(y) \leq k-2$. In that case, there must be a trail from x to $\{x_0, y_0\}$ to y back to x of length $d_e(x) + d_e(y) + 2 \leq 2k-2$. Since $d_e(x) = d_e(y)$, the shortest paths from x and y to $\{x_0, y_0\}$ cannot use the edge (x, y) . Thus, this trail must include a simple cycle of length at most $2k-2$, a contradiction. $\square$

As an immediate corollary of Lemma 4.3 and Lemma 2.3, we have that the non-redundancy of $\text{CSP}(C_{2k}^* \mid C_{2k})$ is asymptotically equal to the maximum size of bipartite graphs of girth at least $2k$.

Corollary 4.5. *For all $k \geq 2$,*

$$\text{NRD}(C_{2k}^* \mid C_{2k}, n) = \Theta_D(\text{ex}(n, \{C_3, C_4, C_5, \dots, C_{2k-1}\})).$$

In particular, by the estimates of [40], we have that there exists a universal constant $c > 0$ such that.

$$\text{NRD}(C_{2k}^* \mid C_{2k}, n) \in [\Omega_D(n^{1+\frac{c}{k-1}}), O_D(n^{1+\frac{1}{k-1}})].$$

Remark 4.6. It is a long-standing conjecture in combinatorics that the constant c can be taken to be 1 for all $k \geq 2$, but this is only known to be true for $k \in \{2, 3, 4, 6\}$. See Section 4 of [40] for a detailed discussion.

We now complete the classification by showing that Lemma 4.3 captures (essentially) all cases.

Theorem 4.7. *For every $R \subsetneq S \subseteq D^2$, we either have that $\text{NRD}(R \mid S, n) = O_D(n)$, or there exists $k \geq 2$ such that*

$$\text{NRD}(R \mid S, n) = \Theta_D(\text{NRD}(C_{2k}^* \mid C_{2k}, n)).$$

Proof. By Proposition 4.1, we may assume without loss of generality that $S \setminus R = \{t_0\}$. Let $(x_0, y_0) := t_0$ and let ℓ be the length of the shortest path from x_0 to y_0 in R (when viewed as the adjacency matrix of a bipartite graph). Since $t_0 \notin R$, we either have that $\ell = \infty$ (i.e., a path does not exist), or ℓ is an odd integer at least 3.

If $\ell = \infty$, consider a bipartite non-redundant instance $G := (A, B, E \subseteq A \times B)$ of $\text{CSP}(R \mid S)$. For each edge $e := (x, y) \in E$, there must exist a homomorphism $f_e : A \cup B \rightarrow D$ from G to S for which $f_e^{-1}((x_0, y_0)) = \{(x, y)\}$. Since x_0 and y_0 are not connected in R , we have that if we delete e from G , then x and y are in separate connected components. Therefore, G must be acyclic. By Lemma 2.3, we have that $\text{NRD}(R \mid S, n) = \mathcal{O}(n)$.

Otherwise, assume that ℓ is an odd integer at least 3. Thus, there exists an integer $k \geq 2$ such that $\ell = 2k - 1$. We claim that $\text{NRD}(R \mid S, n) = \Theta_D(\text{NRD}(C_{2k}^* \mid C_{2k}, n))$. By definition of k , there exists an injective homomorphism $h : C_{2k} \rightarrow S$ such that $h(0, 0) = t_0$. Thus, every non-redundant instance of $\text{NRD}(C_{2k}^* \mid C_{2k}, n)$ is a non-redundant instance of $\text{NRD}(R \mid S, n)$.

Conversely, let $G := (A, B, E \subseteq A \times B)$ be a bipartite non-redundant instance of $\text{NRD}(R \mid S, n)$. For any $e := (x, y) \in G$, consider a homomorphism $f_e : G \rightarrow S$ such that $f_e^{-1}(t_0) = \{e\}$. Since the shortest path from x_0 to y_0 in R is at most $\ell = 2k - 1$, any path from x to y in $G \setminus \{e\}$ must also have length at least $2k - 1$. That is, every cycle of G containing e has length at least $2k$. Since e is arbitrary, G has girth at least $2k$.

Therefore, by Lemma 4.3, every *bipartite* non-redundant instance of $\text{CSP}(R \mid S)$ is a non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$. Therefore, by Lemma 2.3, we have that $\text{NRD}(R \mid S, n) = \Theta(\text{NRD}(C_{2k}^* \mid C_{2k}, n))$. $\square$

Remark 4.8. The correspondence between (R, S) and (C_{2k}^*, C_{2k}) in the proof of Theorem 4.7 can be viewed as an *fgppp-equivalence*. See Section 5 for more details.

4.2 Infinitely Many NRD Exponents for Ternary Predicates

As an immediate application of Lemma 4.3, we can prove that infinitely many exponents are possible for (non-conditional) ternary predicates. For $k \geq 2$, let $S_{2k} := C_{2k} \times \{0, 1\} \subseteq \{0, 1, \dots, k-1\}^3$ and $R_{2k} := S_{2k} \setminus \{(0, 0, 0)\}$. We now prove the following

Lemma 4.9. *For all $k \geq 2$, $\text{NRD}(R_{2k} \mid S_{2k}, n) = \Theta(n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n))$.*

Proof. We divide the proof into the lower bound and the upper bound on $\text{NRD}(R_{2k} \mid S_{2k}, n)$.

Lower Bound. First, we show that $\text{NRD}(R_{2k} \mid S_{2k}, 2n) \geq \Omega(n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n))$ – note that $\text{NRD}(R_{2k} \mid S_{2k}, 2n) = \Theta(\text{NRD}(R_{2k} \mid S_{2k}, n))$ by Lemma 2.4. In particular, by Lemma 2.3 there exists a bipartite non-redundant instance $G = (A_1, A_2, E)$ of $\text{CSP}(C_{2k}^* \mid C_{2k})$ with $\Omega(\text{NRD}(C_{2k}^* \mid C_{2k}, n))$ edges. Let A_3 be a set of size n disjoint from A_1 and A_2 . Let $H = (A_1, A_2, A_3, E \times A_3)$ be a tripartite hypergraph with $\Omega(n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n))$ hyperedges. We claim that H is a non-redundant instance of $R_{2k} \mid S_{2k}$.

Fix a hyperedge $(x_1, x_2, x_3) \in E \times A_3$. Since G is a non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$, we have that there exists a map $f_{x_1, x_2} : A_1 \cup A_2 \rightarrow \{0, 1, \dots, k-1\}$ such that for all $e \in E$, $f_{x_1, x_2}(e) \in C_{2k}$ and $f_{x_1, x_2}(e) = (0, 0)$ iff $e = (x_1, x_2)$. Consider then the map $g_{x_1, x_2, x_3} : A_1 \cup A_2 \cup A_3 \rightarrow \{0, 1, \dots, k-1\}$ defined by

$$g_{x_1, x_2, x_3}(y) = \begin{cases} f_{x_1, x_2}(y) & y \in A_1 \cup A_2 \\ 0 & y = x_3 \\ 1 & y \in A_3 \setminus \{x_3\} \end{cases}.$$

In particular, for any $e := (y_1, y_2, y_3) \in E \times A_3$, we have that $(g_{x_1, x_2, x_3}(y_1), g_{x_1, x_2, x_3}(y_2)) = (f_{x_1, x_2}(y_1), f_{x_1, x_2}(y_2)) \in C_{2k}$ and $g_{x_1, x_2, x_3}(y_3) \in \{0, 1\}$ so $g_{x_1, x_2, x_3}(e) \in S_{2k}$. Furthermore, $g_{x_1, x_2, x_3}(e) = (0, 0, 0)$ if and only if $(y_1, y_2) = (x_1, x_2)$ and $y_3 = x_3$. Thus, H is indeed a non-redundant instance of $\text{CSP}(R_{2k} \mid S_{2k})$.

Upper Bound. Next, we show that $\text{NRD}(R_{2k} \mid S_{2k}, n) \leq n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n)$. By Lemma 2.3, it suffices consider a tripartite non-redundant instance $H := (A_1, A_2, A_3, F \subseteq A_1 \times A_2 \times A_3)$ of $\text{CSP}(R_{2k} \mid S_{2k})$. Fix $x_3 \in A_3$ and let $E_{x_3} := \{(x_1, x_2) : (x_1, x_2, x_3) \in F\}$. We seek to prove that $G_{x_3} := (A_1, A_2, E_{x_3})$ is a non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$. This suffices to prove the upper bound as then

$$|F| \leq \sum_{x_3 \in A_3} |E_{x_3}| \leq \sum_{x_3 \in A_3} \text{NRD}(C_{2k}^* \mid C_{2k}, n) = n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n).$$

In particular, consider $(x_1, x_2) \in E_{x_3}$. Since H is a tripartite non-redundant instance of $\text{CSP}(R_{2k} \mid S_{2k})$, there exists a homomorphism $f_e : H \rightarrow S_{2k}$ which sends only (x_1, x_2, x_3) to $(0, 0, 0)$. In particular, $f_e(x_3) = 0$. Therefore, for any other $(x, y) \in E_{x_3}$, we must have $(f_e(x), f_e(y)) \in C_{2k}^*$. Thus, G_{x_3} is a non-redundant instance of $\text{CSP}(C_{2k}^* \mid C_{2k})$. $\square$

Furthermore, it is not hard to bound the non-redundancy of S_{2k} .

Proposition 4.10. *For all $k \geq 2$, $\text{NRD}(S_{2k}, n) = \mathcal{O}(n^2)$.*

Proof. Let $B = \{0, 1\}$ be the unary relation restricting a predicate to be 0 or 1. By definition of S_{2k} , we have the logical equivalence

$$S_{2k}(x_1, x_2, x_3) \equiv C_{2k}(x_1, x_2) \wedge B(x_3).$$

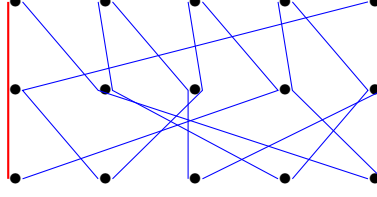


Figure 5: A hypergraph representation of CYC_5 , with the red hyperedge being the one to delete to get CYC_5^* . Note that if we delete any row from this figure, we get a graph isomorphic to C_{10} .

That is, $\{C_{2k}, B\}$ fgpp-defines S_{2k} . By a bound of Carbonnel (cf. [21, Prop. 2]), we have that $\text{NRD}(S_{2k}, n) \leq \text{NRD}(C_{2k}, n) + \text{NRD}(B, n) = \mathcal{O}(n^2)$, as C_{2k} is a binary constraint and B is a unary constraint. $\square$

Combining Lemma 4.9 and Proposition 4.10 using the triangle inequality (Lemma 2.5) and invoking Corollary 4.5, we can deduce the main result of this section.

Theorem 4.11. *For all $k \geq 2$, $\text{NRD}(R_{2k}, n) = \Theta(n \cdot \text{NRD}(C_{2k}^* \mid C_{2k}, n)) \in [n^{2+\frac{\Omega(1)}{k-1}}, n^{2+\frac{1}{k-1}}]$.*

Remark 4.12. Note that $\text{NRD}(R_6, n) = \Theta(n^{2.5})$. In particular, 2.5 is currently the largest known non-redundancy exponent less than 3 for a ternary predicate. By a result of Carbonnel [21], such an exponent must be less than 2.75.

4.3 Ternary Predicates Approaching Linear NRD

To conclude this section, we give another ternary adaptation of $\text{CSP}(C_{2k}^* \mid C_{2k})$. However, this time the non-redundancies are approaching linear. For odd $m \in \mathbb{N}$ define

$$\text{CYC}_m := \{(x, y, z) \in (\mathbb{Z}/m\mathbb{Z})^3 : x + y + z = 0 \wedge y - x \in \{0, 1\}\}.$$

Further define $\text{CYC}_m^* := \text{CYC}_m \setminus \{(0, 0, 0)\}$. See Figure 5 for an illustration. We seek to prove the following

Theorem 4.13. *For all odd $m \geq 3$, we have that $\text{NRD}(\text{CYC}_m^*, n) \leq \mathcal{O}_m(n^{\frac{m}{m-1}})$.*

Remark 4.14. This proof never uses the fact that m is odd. We state it this way because the corresponding “lower bound” of Theorem 4.17 only holds for odd m . In fact, for even $m \geq 2$, one can show that $\text{NRD}(\text{CYC}_m^*) = \mathcal{O}_m(n)$ (e.g., CYC_2^* is 2-in-3-SAT).

Proof. Similar to the proof of Theorem 4.11, we split the proof of our upper bound into two parts. First, we show that $\text{NRD}(\text{CYC}_m^* \mid \text{CYC}_m, n) = \mathcal{O}_m(n^{\frac{m}{m-1}})$. Second, we show that $\text{NRD}(\text{CYC}_m, n) = \mathcal{O}_m(n)$. These imply the upper bound on $\text{NRD}(\text{CYC}_m^*, n)$ by Lemma 2.5.

Step 1. By Lemma 2.3, it suffices to show that a tripartite non-redundant instance $H := (A_1, A_2, A_3, F \subseteq A_1 \times A_2 \times A_3)$ of $\text{CSP}(\text{CYC}_m^* \mid \text{CYC}_m)$ has at most $\mathcal{O}_m(n^{\frac{m}{m-1}})$ hyperedges.

First, notice that H is a *linear* hypergraph; that is, no two hyperedges overlap in exactly two vertices. Assume for sake of contradiction that $e = (x_1, x_2, x_3)$ and $f = (y_1, y_2, y_3)$ have two vertices in common. Since H is non-redundant, there must be a map that sends e to $(0, 0, 0)$ but f to some element of CYC_m^* . However, CYC_m^* lacks any elements with two or more zeros, a contradiction.

Since H is a linear hypergraph, we may define $E := \{(x_1, x_2) : \exists x_3 \in A_3, (x_1, x_2, x_3) \in F\}$ and note that $|E| = |F|$. Thus, by Corollary 4.5 and Remark 4.2, it suffices to prove that $G := (A_1, A_2, E)$ is a non-redundant instance of $\text{CSP}(\tilde{C}_{2m}^* \mid \tilde{C}_{2m})$.

For any $e := (x_1, x_2, x_3) \in F$ let $f_e : A_1 \cup A_2 \cup A_3 \rightarrow \{0, 1, \dots, m-1\}$ be the homomorphism which maps H to CYC_m but e is the only hyperedge mapped to $(0, 0, 0)$. Note that $\tilde{C}_{2m}^* = \{(z_1, z_2) : \exists z_3, (z_1, z_2, z_3) \in \text{CYC}_m\}$. Therefore, f_e restricted to $A_1 \cup A_2$ is a homomorphism from G to $\tilde{C}_{2m}$ which only maps (x_1, x_2) to $(0, 0)$. Thus, G is a non-redundant instance of $\text{CSP}(\tilde{C}_{2m}^* \mid \tilde{C}_{2m})$, as desired.

Step 2. To finish, we show that $\text{NRD}(\text{CYC}_m, n) = \mathcal{O}_m(n)$. We do this by constructing an explicit Abelian embedding of CYC_m . Let $D := \mathbb{Z}/m\mathbb{Z}$ and consider the group $G = (\mathbb{Z}/2\mathbb{Z})^D$ which we identify with the set of subsets of D equipped with the symmetric difference operator $\oplus$. Furthermore, we identify each element $i \in D$ with its singleton $\{i\} \in G$.

Let Q be the subgroup of G^3 generated by the elements of CYC_m . By definition, $\text{CYC}_m \subseteq Q \cap D^3$. We claim that $\text{CYC}_m = Q \cap D^3$. Consider any triple $(a_1, a_2, a_3) \in D^3$ such that there exists a subset $S \subseteq \text{CYC}_m$ such that

$$(\{a_1\}, \{a_2\}, \{a_3\}) = \bigoplus_{(b_1, b_2, b_3) \in S} (\{b_1\}, \{b_2\}, \{b_3\}).$$

A key observation is the following. For any $T \subseteq \tilde{C}_{2m}$, consider the analogous expression $(B_1, B_2) := \bigoplus_{(b_1, b_2) \in T} (\{b_1\}, \{b_2\})$. If we interpret T as a collection of paths in $\tilde{C}_{2m}$, then B_1 and B_2 are precisely the sets of endpoints of the paths (restricted to the two halves of the bipartite graph). Thus, since CYC_m restricted to its first two coordinates is precisely $\tilde{C}_{2m}$, the projection of S to its first two coordinates must correspond precisely to a path between a_1 (on the left) and a_2 (on the right) in $\tilde{C}_{2m}$. Since each tuple of CYC_m is uniquely determined by its first two coordinates, we have narrowed down S to the following two possibilities:

$$\begin{aligned} S_0 &:= \{(a_1, a_1, -2a_1), (a_1 - 1, a_1, -2a_1 + 1), (a_1 - 1, a_1 - 1, -2a_1 + 2), \dots, (a_2, a_2, -2a_2)\} \\ S_1 &:= \{(a_1, a_1 + 1, -2a_1 - 1), (a_1 + 1, a_1 + 1, -2a_1 - 2), \dots, (a_2 - 1, a_2, -2a_2 + 1)\} \end{aligned}$$

One can verify that $\bigoplus_{(b_1, b_2, b_3) \in S_i} (\{b_1\}, \{b_2\}, \{b_3\})$ gives the same value for $i \in \{0, 1\}$. Note that $S_0 \sqcup S_1$ is a partition of CYC_m . Thus, at least one of S_0 or S_1 has cardinality at most m . Notice that the third coordinate of S_0 increases by 1 for each successive tuple, and the third coordinate of S_1 decreases by 1 for each successive tuple. Thus, for the set S_i of size at most m , the expression $\bigoplus_{(b_1, b_2, b_3) \in S_i} (\{b_1\}, \{b_2\}, \{b_3\})$ will sum $|S_i|$ distinct values in the third coordinate. The only way this can equal $\{a_3\}$ is for $|S_i| = 1$ and $S_i = \{(a_1, a_2, a_3)\}$. Thus, $(a_1, a_2, a_3) \in \text{CYC}_m$, as desired.

Thus, we have proven that CYC_m has an Abelian group embedding, so $\text{NRD}(\text{CYC}_m, n) \leq \mathcal{O}_m(n)$. $\square$

Remark 4.15. The argument used in Step 1 can be interpreted as a special case of Theorem 7.14 with the fgppp-definition of $(\text{CYC}_m^* \mid \text{CYC}_m)(x_1, x_2, x_3) \equiv (\tilde{C}_{2m}^* \mid \tilde{C}_{2m})(x_1, x_2)$.

Remark 4.16. One can check that $\text{CYC}_3^* = \{111, 222, 012, 120, 201\}$ which is equal to the BCK predicate first considered by [9]. The best upper bound on $\text{NRD}(\text{BCK}, n)$ in the literature is $\mathcal{O}(n^{1.5} \log n)$ by [14]. Theorem 4.13 mildly improves this upper bound to $\mathcal{O}(n^{1.5})$.

Sadly, we do not know of a nontrivial lower bound for the non-redundancy CYC_m^* . However, we give evidence that a nontrivial lower bound should exist by proving that CYC_m^* lacks a Mal'tsev embedding.

Theorem 4.17. *For all odd $m \geq 3$, CYC_m^* lacks a Mal'tsev embedding.*

The proof of Theorem 4.17 is rather technical as it first requires developing new properties about the operators in an algebra with a Mal'tsev term. As such, we defer the proof of Theorem 4.17 to Section 6.4.

5 Towards An Algebraic Approach: Partial Promise Polymorphisms and Patterns

In this section we take the first steps for developing a novel algebraic theory for studying conditional non-redundancy. It turns out that the fundamental algebraic object are *partial polymorphisms* between relation pairs (S, T) . This polymorphism notion merges and generalizes the following two notions.

1. Polymorphisms between relation pairs (A, B) that have been fundamental in studying the approximation variant of CSP known as *promise CSP* [13, 3].
2. Partial polymorphisms of relations that have been used to study fine-grained complexity questions [62], kernelization [61], and non-redundancy [9, 21].

We begin in Section 5.1 by proving that our partial polymorphism notion on the functional side yields sets of partial functions closed under the formation of *minors*. On the relational side we instead get *quantifier-free* primitive positive definitions, and we obtain a full Galois correspondence between these notions. Via this connection it is straightforward to show that partial promise polymorphisms govern conditional non-redundancy. Moving up one abstraction level we in Section 5.2 concentrate on *identities*, or *patterns*, satisfied by partial promise polymorphisms.

5.1 A Galois Connection

We begin by developing a Galois connection between sets of partial functions closed under the formation of *minors* and sets of relations closed under a certain type of conjunctive formulas. Throughout, we assume that all domains D_1 and D_2 are finite, although many concepts generalize to e.g. ω -categorical structures. A *relation pair* over finite domains D_1 and D_2 is a tuple (S, T) where $S \subseteq D_1^r$ and $T \subseteq D_2^r$ for some $r \geq 0$ (note that $D_1^0 = D_2^0 = \emptyset$). If, in addition, there exists a homomorphism $h: D_1 \rightarrow D_2$, i.e., $h(t) \in T$ for each $t \in S$ then (S, T) is said to be a *promise relation*, and is sometimes written (S, T, h) . A *promise language* is a set of relation pairs where there exist $h: D_1 \rightarrow D_2$ such that (S, T, h) is a promise relation for each relation pair (S, T) , and a *language* is simply a set of relation pairs. Note that all relation pairs in a language might not have the same arity. In the forthcoming applications we often assume that $D_1 \subseteq D_2$ in which case h can always be defined as the identity function $h(x) = x$ over D_1 . If the domain for a promise relation (S, T) is not specified we always assume that $S \subseteq T$. We often write (Γ_1, Γ_2) for a language and thus write $(S, T) \in (\Gamma_1, \Gamma_2)$ to denote a specific relation pair.

Let D_1 and D_2 be two finite sets. An n -ary *partial function* from D_1 to D_2 is a map $f: X \rightarrow D_2$ for $X \subseteq D_1^n$, and we write $\text{dom}(f) = X$ for the set of values where f is defined. We frequently express $x \notin \text{dom}(f)$ by $f(x) = \perp$. A *subfunction* of f is a partial function g where $\text{dom}(g) \subseteq \text{dom}(f)$ and such that $g(\mathbf{x}) = f(\mathbf{x})$ for every $\mathbf{x} \in \text{dom}(g)$. We write π_i^n for the i -ary *projection*, in approximation contexts often called *dictator* satisfying $\pi_i^n(\mathbf{x}) = \mathbf{x}_i$. The domain is usually

taken from the context and not explicitly denoted. A *partial projection* (or *partial dictator*) is a subfunction of a projection. When $D_1 = D_2 = D$ then it is natural to combine functions in the following sense: for an n -ary partial function f and m -ary partial functions $g_1, \dots, g_n$ over the same universe D we define composition of $f, g_1, \dots, g_n$ as

$$(f \circ (g_1, \dots, g_n))(x_1, \dots, x_m) = f(g_1(x_1, \dots, x_m), \dots, g_n(x_1, \dots, x_m))$$

for all sequences of arguments $x_1, \dots, x_m \in A$ such that (1) each $(x_1, \dots, x_m) \in \text{dom}(g_i)$ and (2) $g_1(x_1, \dots, x_m), \dots, g_n(x_1, \dots, x_m) \in \text{dom}(f)$. Composition of total functions is then just a special case when all involved functions $f, g_1, \dots, g_n$ are total, i.e., $\text{dom}(f) = D^n$ and each $\text{dom}(g_i) = D^m$. For a set of (total or partial) functions F over A we write $[F]$ for the smallest set of (total or partial) functions over A which contains F , is closed under composition, and which contains all (total or partial) projections. Such sets are sometimes called (strong partial) *clones* and naturally correspond to the term algebra induced by a given algebra.

When working with partial functions from D_1 to D_2 it is generally not possible to compose functions, and the relevant algebraic closure condition is instead formation of *partial minors*.

Definition 5.1. A n -ary (variable-identification) *minor* $f_{/h}$ of an m -ary partial function $f: X \rightarrow D_2$, where $h: [m] \rightarrow [n]$, is a function of the form $f_{/h}(x_1, \dots, x_n) = f(x_{h(1)}, \dots, x_{h(m)})$.

We now define a *partial minor*, often just called a *minor*, g of f as a subfunction of g' for a variable identification minor g' of f (observe that g' is technically a subfunction of itself, so any variable identification minor is also a minor).

Definition 5.2. A set of partial functions F from D_1 to D_2 is said to be a *strong partial minion* if it is closed under minors, i.e., if $f \in F$ and g is a minor of f then $g \in F$. We write $[F]_{\min}$ for the smallest strong partial minion that contains F .

Note that the subfunction condition in the definition of a partial minor clearly sets apart strong partial minions from minions of total functions and prevents the former to be defined under the unifying umbrella of *abstract minions* [15]. However, we will see that many notions carry over to the partial setting and that strong partial minions can be described via relations. Thus, we say that an n -ary partial function $f: \text{dom}(f) \rightarrow D_2$ is a *partial polymorphism* of an r -ary relation pair (S, T) if, for each sequence $x^{(1)}, \dots, x^{(n)} \in S$, either

1. there exists $1 \leq i \leq r$ such that $(x_i^{(1)}, \dots, x_i^{(n)}) \notin \text{dom}(f)$, or
2. $(f(x_1^{(1)}, \dots, x_1^{(n)}), \dots, f(x_r^{(1)}, \dots, x_r^{(n)})) \in T$.

Similarly, f is a partial polymorphism of a language (Γ_1, Γ_2) if it is a partial polymorphism of every $(S, T) \in (\Gamma_1, \Gamma_2)$, in which case we also say that (Γ_1, Γ_2) is *invariant* under f .

Definition 5.3. For a language (Γ_1, Γ_2) we let $\text{pPol}(\Gamma_1, \Gamma_2)$ be the set of all partial polymorphisms from Γ_1 to Γ_2 .

If $(\Gamma_1, \Gamma_2) = \{(S, T)\}$ is singleton then we write $\text{pPol}(S, T)$ rather than $\text{pPol}(\{(S, T)\})$, and if $\Gamma_1 = \Gamma_2$, simply $\text{pPol}(\Gamma_1)$. Let us also remark that sets of the form $\text{pPol}(\Gamma_1)$ are closed under functional composition as well as subfunctions.

Definition 5.4. For a set of partial functions F from D_1 to D_2 we let

$$\text{Inv}(F) = \{(S, T) \mid r \geq 0, S \subseteq D_1^r, T \subseteq D_2^r, F \subseteq \text{pPol}(S, T)\}.$$

We observe that $\text{Inv}(F)$ is a promise language if $[F]_{\min}$ contains at least one unary, total function.

Let us now relate the $\text{Inv}(\cdot)$ and $\text{pPol}(\cdot)$ operators together as follows. To simplify the definition we temporarily work in the setting of relational structures in a prescribed signature τ containing relation symbols and their associated arities. A *quantifier-free primitive positive formula* (qfpp-formula) over a relational signature τ is a conjunctive formula $\varphi(x_1, \dots, x_r)$ with free variables $x_1, \dots, x_r$ where each atom is of the form $R_i(\mathbf{x}^i)$ (for some $(R_i, r_i) \in \tau$ and r_i tuple of variables $\mathbf{x}^i$), an equality of the form $x = y$, for $x, y \in \{x_1, \dots, x_r\}$, or $\mathbf{f}(x_i)$ for some $x_i \in \{x_1, \dots, x_r\}$, where $\mathbf{f}$ is a special notation for the empty unary relation. Each combination of a qfpp-formula $\varphi(x_1, \dots, x_r)$ and a τ -structure Γ_1 induces a relation $\{(a_1, \dots, a_r) \mid \varphi^{\Gamma_1}(a_1, \dots, a_r)\}$, where equality $=$ is interpreted in the obvious way, and where $\mathbf{f}$ is interpreted as $\emptyset$. The sole reason for introducing $\mathbf{f}$ is that the relation pair $(\emptyset, \emptyset)$ is invariant under every partial polymorphism and thus should always be definable. In the literature one sometimes assumes the existence of a symbol $\mathbf{t}$ always interpreted as the full relation, but note that we can simulate a constraint $\mathbf{t}$ via an equality constraint $x = x$.

We lift this notion to relation pairs as follows. Let (Γ_1, Γ_2) be a language over D_1, D_2 and (Γ'_1, Γ'_2) a language over D_1, D_2 . We say that (Γ'_1, Γ'_2) is

1. *quantifier-free primitive positive definable* (qfpp-definable) in (Γ_1, Γ_2) if, for each relational symbol R (of arity r) of (Γ'_1, Γ'_2) , there is a qfpp-formula φ_R such that

$$R^{\Gamma'_1} = \{(a_1, \dots, a_r) \in D_1^r \mid \varphi_R^{\Gamma_1}(a_1, \dots, a_r)\},$$

and, similarly,

$$R^{\Gamma'_2} = \{(a_1, \dots, a_r) \in D_2^r \mid \varphi_R^{\Gamma_2}(a_1, \dots, a_r)\},$$

2. a *strict relaxation* of (Γ_1, Γ_2) if $\Gamma_1, \Gamma'_1, \Gamma_2, \Gamma'_2$ are similar relational structures and, for every relational symbol R ,

$$R^{\Gamma'_1} \subseteq R^{\Gamma_1} \quad \text{and} \quad R^{\Gamma_2} \subseteq R^{\Gamma'_2}.$$

More generally (Γ_1, Γ_2) is *quantifier-free primitive positive promise definable* (qfpppp-definable) in (Γ'_1, Γ'_2) if the former can be obtained by a sequence of qfpp-definitions and strict relaxations of the latter. We write $\langle \Gamma_1, \Gamma_2 \rangle_{\overline{\mathcal{A}}}$ for the smallest set of relation pairs containing (Γ_1, Γ_2) and which is closed under qfpppp-definitions, and similarly to the $\text{pPol}(\cdot, \cdot)$ operator write $\langle S, T \rangle_{\overline{\mathcal{A}}}$ for a singleton language $(\Gamma_1, \Gamma_2) = \{(S, T)\}$.

Lemma 5.5. *Let (Γ_1, Γ_2) be a language. Then $\text{pPol}(\Gamma_1, \Gamma_2)$ is a strong partial minion ($\text{pPol}(\Gamma_1, \Gamma_2) = [\text{pPol}(\Gamma_1, \Gamma_2)]_{\min}$).*

Proof. The inclusion $\text{pPol}(\Gamma_1, \Gamma_2) \subseteq [\text{pPol}(\Gamma_1, \Gamma_2)]_{\min}$ is trivial since every partial function is a minor of itself. For the other direction, let g be an n -ary minor of an m -ary $f \in \text{pPol}(\Gamma_1, \Gamma_2)$, i.e., g is a subfunction of a variable-identification minor $f_{/h}$ of f for some $h: [m] \rightarrow [n]$. For each $(S, T) \in (\Gamma_1, \Gamma_2)$ it follows that each application $f_{/h}(t^{(1)}, \dots, t^{(m)})$ for $t^{(1)}, \dots, t^{(m)} \in S$ is equivalent to f applied as $f(t^{h(1)}, \dots, t^{h(m)})$, which is either undefined or included in T . Hence, $f_{/h}$ preserves (S, T) . Similarly, it is easy to see that any subfunction g of $f_{/h}$ also preserves any relation invariant under $f_{/h}$, and we conclude that $\text{pPol}(\Gamma_1, \Gamma_2) = [\text{pPol}(\Gamma_1, \Gamma_2)]_{\min}$. $\square$

Note that if (Γ_1, Γ_2) over D_1, D_2 is a promise language, witnessed by a function $h: D_1 \rightarrow D_2$, then $h \in \text{pPol}(\Gamma_1, \Gamma_2)$, but if (Γ_1, Γ_2) is not a promise language then we are not guaranteed the existence of a total, unary function in $\text{pPol}(\Gamma_1, \Gamma_2)$.

Lemma 5.6. *Let D_1 and D_2 be two finite sets. Then $\langle \text{Inv}(F) \rangle_{\exists} = \text{Inv}(F)$ for any set F of partial functions from D_1 to D_2 .*

Proof. The inclusion $\text{Inv}(F) \subseteq \langle \text{Inv}(F) \rangle_{\exists}$ is trivial since any relation can qfpp-define itself. For the other direction, consider relation symbols R of arity r and R' of arity r' , interpreted by $R^{\Gamma_1}, R^{\Gamma_1}$ in Γ_1 over D_1 and $R^{\Gamma_2}, R^{\Gamma_2}$ over D_2 , and the following qfpp-definitions φ .

1. $R(x_1, \dots, x_r) \wedge f(x_i)$ for some $x \in [r]$. In this case the statement is trivial.
2. $R(x_1, \dots, x_r) \wedge (x_i = x_j)$ for $i, j \in [r]$. Then $\varphi^{\Gamma_1} = \{(d_1, \dots, d_r) \mid (d_1, \dots, d_r) \in S, d_i = d_j\}$ and $\varphi^{\Gamma_2} = \{(d_1, \dots, d_r) \mid (d_1, \dots, d_r) \in R^{\Gamma_2}, d_i = d_j\}$. Consider an n -ary $f \in F$ and an application $f(t^{(1)}, \dots, t^{(n)}) = t$ for $t^{(1)}, \dots, t^{(n)} \in \varphi^{\Gamma_1}$ where every application $f(t_k^{(1)}, \dots, t_k^{(n)})$, $1 \leq k \leq r$, is defined. Observe that $t_i = t_j$, i.e., the i th element of t equals the j th element of t , which implies that $t \in \varphi^{\Gamma_2}$ (since $t \in R^{\Gamma_2}$).
3. $R(x_{h(1)}, \dots, x_{h(r)})$ for $h: [r] \rightarrow [s]$ for some $s \geq 1$. We sketch the most important arguments. Note that one for each $(d_1, \dots, d_s) \in \varphi^{\Gamma_1}$ (respectively, φ^{Γ_2}) can define a corresponding tuple $(d_{h(1)}, \dots, d_{h(r)}) \in \varphi^{\Gamma_1}$ (respectively, φ^{Γ_2}). Thus, if $f(t^{(1)}, \dots, t^{(n)}) = t$ for some n -ary $f \in F$ and $t^{(1)}, \dots, t^{(n)} \in \varphi^{\Gamma_1}$ then one can define tuples which when applied to f yields a tuple in φ^{Γ_2} .
4. $R(x_{h(1)}, \dots, x_{h(r)}) \wedge R'(x_{h'(1)}, \dots, x_{h'(r')})$ for $h: [r] \rightarrow X$ and $h': [r'] \rightarrow Y$. First, assume that $X \cap Y = \emptyset$, i.e., the two constraints have no overlapping variables. Then φ^{Γ_1} (respectively, φ^{Γ_2}) is simply the Cartesian product of two relations that are invariant under F from the preceding case, and the claim easily follows. Similarly, if $X = Y$ then the resulting relation is the intersection of two invariant constraints. The general case follows via similar arguments.

This generalizes to arbitrary qfpp-definitions via a simple inductive argument. Last, consider a strict relaxation (S', T') of some $(S, T) \in \text{Inv}(F)$, i.e., $S' \subseteq S$ and $T' \supseteq T$. Assume there exists an n -ary $f \in F$ and tuples $t^{(1)}, \dots, t^{(n)} \in S' \subseteq S$ such that $f(t^{(1)}, \dots, t^{(n)}) = t \notin T'$. It follows that f does not preserve (S, T) , either, since $t^{(1)}, \dots, t^{(n)} \in S \supseteq S'$ and since $t \notin T \subseteq T'$. $\square$

With similar arguments we obtain the following two characterizations.

Lemma 5.7. *Let (Γ_1, Γ_2) be a language. Then $\text{pPol}(\langle \Gamma_1, \Gamma_2 \rangle_{\exists}) = \text{pPol}(\Gamma_1, \Gamma_2)$.*

Lemma 5.8. *Let F be a set of partial functions over A, B . Then $\text{Inv}(F) = \text{Inv}([F]_{\min})$.*

To complete our description we additionally need the following two lemmas.

Lemma 5.9. *Let F be a strong partial minion over domains D_1, D_2 . For any partial function $g: X \rightarrow D_2$ (for some $n \geq 1$ and $X \subseteq D_1^n$) where $g \notin F$ there*

1. *exists a relation pair (S, T) such that $F \subseteq \text{pPol}(S, T)$ and $g \notin \text{pPol}(S, T)$, and*
2. *if F contains a unary, total function h then (S, T) is a promise relation.*

Proof. First, consider the relation $S = \{t^{(1)}, \dots, t^{(n)}\}$ whose arguments (in, say, lexicographical order) enumerates the domain X of g , i.e., for the i th tuple $t \in X$ we have that $(t_i^{(1)}, \dots, t_i^{(n)}) = t$. Second, consider the relation T formed by, for every $f \in F$ of arity m , and all tuples $t^{(i_1)}, \dots, t^{(i_m)} \in S$, adding $f(t^{(i_1)}, \dots, t^{(i_m)})$ whenever each application is defined.

By definition, g cannot preserve (S, T) since it would then be equivalent to a partial function in F . Similarly, by construction, (S, T) is invariant under every function in F . Note that if F contains a unary, total function h , then by construction, h is a homomorphism from S to T , and (S, T) is thus a promise relation. $\square$

Lemma 5.10. *Let (Γ_1, Γ_2) be a language. Let (S, T) be a relation pair over A, B such that $(S, T) \notin \langle \Gamma_1, \Gamma_2 \rangle_{\exists}$. Then there exists $f \in \text{pPol}(\Gamma_1, \Gamma_2)$ such that $f \notin \text{pPol}(S, T)$.*

Proof. This follows from Proposition 2.5 in Pippenger [73] since the construction does not use existential quantification. $\square$

We may now conclude that sets of the form $\text{Inv}(\text{pPol}(\Gamma_1, \Gamma_2))$ are precisely the sets containing (Γ_1, Γ_2) closed under qfppp-definitions, and, similarly, that all strong partial minions $[F]_{\min}$ can be described as the set of polymorphisms of the relations invariant under F ($\text{pPol}(\text{Inv}(F))$).

Theorem 5.11. *Let (Γ_1, Γ_2) be a language over D_1, D_2 . Then $\langle \Gamma_1, \Gamma_2 \rangle_{\exists} = \text{Inv}(\text{pPol}(\Gamma_1, \Gamma_2))$.*

Proof. The first direction follows from Lemma 5.7. Second, let $(S, T) \in \text{Inv}(\text{pPol}(\Gamma_1, \Gamma_2))$. If $(S, T) \notin \langle \Gamma_1, \Gamma_2 \rangle_{\exists}$ then Lemma 5.10 implies that there is $f \in \text{pPol}(\Gamma_1, \Gamma_2)$ such that $f \notin \text{pPol}(S, T)$, contradicting that (S, T) is invariant under every partial function in $\text{pPol}(\Gamma_1, \Gamma_2)$. $\square$

Theorem 5.12. *Let F be a set of partial functions from D_1 to D_2 . Then $[F]_{\min} = \text{pPol}(\text{Inv}(F))$.*

Proof. First, let $f \in [F]_{\min}$. Then f preserves every relation in $\text{Inv}(F)$ (by Lemma 5.8). Second, let $f \in \text{pPol}(\text{Inv}(F))$. If $f \notin [F]_{\min}$ then Lemma 5.9 implies that there is (S, T) such that $[F]_{\min} \subseteq \text{pPol}(S, T)$ but $f \notin \text{pPol}(S, T)$, i.e., $(S, T) \in \text{Inv}(F)$, contradicting the assumption that $f \in \text{pPol}(\text{Inv}(F))$. $\square$

By combining everything together thus far we obtain a Galois connection between $\text{Inv}(\cdot)$ and $\text{pPol}(\cdot, \cdot)$ which implies the following duality.

Theorem 5.13. *Let (Γ_1, Γ_2) and (Γ'_1, Γ'_2) be languages over D_1, D_2 . Then*

$$\text{pPol}(\Gamma_1, \Gamma_2) \subseteq \text{pPol}(\Gamma'_1, \Gamma'_2)$$

if and only if

$$(\Gamma'_1, \Gamma'_2) \subseteq \langle \Gamma_1, \Gamma_2 \rangle_{\exists}.$$

Proof. First, assume that $\text{pPol}(\Gamma_1, \Gamma_2) \subseteq \text{pPol}(\Gamma'_1, \Gamma'_2)$. Assume there exists $(S, T) \in (\Gamma'_1, \Gamma'_2)$ such that $(S, T) \notin \langle \Gamma_1, \Gamma_2 \rangle_{\exists}$. By Lemma 5.10 there exists $f \in \text{pPol}(\Gamma_1, \Gamma_2)$ such that $f \notin \text{pPol}(S, T) \supseteq \text{pPol}(\Gamma'_1, \Gamma'_2)$, contradicting the assumption that $\text{pPol}(\Gamma_1, \Gamma_2) \subseteq \text{pPol}(\Gamma'_1, \Gamma'_2)$.

For the other direction, assume that $(\Gamma'_1, \Gamma'_2) \subseteq \langle \Gamma_1, \Gamma_2 \rangle_{\exists}$. Assume there exists $f \in \text{pPol}(\Gamma_1, \Gamma_2)$ such that $f \notin \text{pPol}(\Gamma'_1, \Gamma'_2)$, and let $(S, T) \in (\Gamma'_1, \Gamma'_2)$ be a promise relation witnessing this, i.e., $f \notin \text{pPol}(S, T)$. By assumption, $(S, T) \in \langle \Gamma_1, \Gamma_2 \rangle_{\exists}$, and Lemma 5.7 then implies that $\text{pPol}(\Gamma_1, \Gamma_2) = \text{pPol}(\langle \Gamma_1, \Gamma_2 \rangle_{\exists}) \subseteq \text{pPol}(S, T)$, contradicting the assumption that $f \in \text{pPol}(\Gamma_1, \Gamma_2)$. $\square$

Recall that for a promise relation (S, T) over D we write $\tilde{T}$ for $D^r \setminus (T \setminus S)$. We can now prove that the conditional non-redundancy of a relation is determined by the partial polymorphisms from S to $\tilde{T}$, i.e., we are interested in partial polymorphisms f such that if $f(t_1, \dots, t_n) = t$ is defined then $t \in S$ if $t \in T$.

Proposition 5.14. *Let (S_1, T_1) and (S_2, T_2) be promise relations over a domain D . If $\text{pPol}(S_2, \widetilde{T_2}) \subseteq \text{pPol}(S_1, \widetilde{T_1})$ then $\text{NRD}(S_1 \mid T_1, n) = \mathcal{O}(\text{NRD}(S_2 \mid T_2, n))$.*

Proof. We provide a proof sketch since this proposition is superseded by Theorem 7.14 in Section 7.2, and ignore handling relaxations and equality constraints. First, via Theorem 5.13 the condition that $\text{pPol}(S_1, \widetilde{T_1}) \subseteq \text{pPol}(S_2, \widetilde{T_2})$ implies that (S_2, T_2) qfpp-defines (S_1, T_1) . Consider a conditional non-redundant instance (X, Y) of $\text{CSP}(S_1 \mid T_1)$ on n variables. We replace every constraint of S_1 with the corresponding constraints of S_2 , as prescribed by the witnessing qfpp-definition. For each constraint $y \in Y$, there then exists an assignment $\sigma_y : X \rightarrow D$ for which $\sigma_y(y') \in T_1$ for all $y' \in Y \setminus \{y\}$ but $\sigma_y(y) \in T_1 \setminus S_1$.

In particular, σ_y will S_2 -satisfy every constraint which replaces each $y' \in Y \setminus \{y\}$. However, for the clauses replacing y , we have that

$$0 = (D^r \setminus (T_1 \setminus S_1))(\sigma_y(y)) = \bigwedge_{i=1}^t (D^r \setminus (T_2 \setminus S_2))(\sigma_y(y|_{S_i})).$$

Thus, complementing both sides, we then have that

$$1 = (T_2 \setminus S_2)(\sigma_y(y)) = \bigvee_{i=1}^t (T_1 \setminus S_1)(\sigma_y(y|_{S_i}))$$

and at least one of the constraints replacing y will have σ_y assign it to some value in $T_1 \setminus S_1$. Pick one such clause arbitrarily for each $y \in Y$. The σ_y 's witness that the resulting instance is non-redundant for $\text{CSP}(S_1 \mid T_1)$. $\square$

From now on, we primarily work in the signature-free setting and when specifying that an r -ary relation R over D is definable via a combination of a qfpp-formula φ_R and a structure Γ over D as $R = \{(a_1, \dots, a_r) \in D^r \mid \varphi_R^\Gamma(a_1, \dots, a_r)\}$ we express this as

$$R(x_1, \dots, x_r) \equiv \varphi_R^\Gamma(x_1, \dots, x_r)$$

and do not make a sharp distinction between structures Γ and sets of relations Γ .

5.2 Polymorphism Patterns

Fully describing $\text{pPol}(\Gamma_1, \Gamma_2)$ even for Boolean promise languages (Γ_1, Γ_2) appears to be out of reach with current algebraic methods. Fortunately, we can abstract away specific operations and concentrate on *identities* satisfied by the operations instead, similar to the universal algebraic approach for studying classical complexity of (P)CSPs where one typically looks only at *linear* identities $f(x_1, \dots, x_n) \approx g(y_1, \dots, y_m)$ [4]. However, the identities relevant in our setting have an even more restricted form where the right-hand side consists of a single variable, and can concisely be defined as follows in the signature-free setting.

Definition 5.15. An n -ary *polymorphism pattern* over a set of variables V for $n \geq 1$ is a subset of $\{(t, x) \mid t = (x_1, \dots, x_n) \in V^n, x \in V\}$.

Example 5.16. A *Mal'tsev* term ϕ can be defined by $P = \{((x, x, y), y), ((y, x, x), y)\}$ over $V = \{x, y\}$ and a *majority* term can be defined by $P' = \{((x, x, y), x), ((x, y, x), x), ((y, x, x), x)\}$.

Definition 5.17. Let D be a set and let P be an n -ary polymorphism pattern. We define $I_D(P)$ as the set of n -ary partial functions f such that:

1. if $((x_1, \dots, x_n), x) \in P$ where $x \in \{x_1, \dots, x_n\}$, then for every $g: \{x_1, \dots, x_n\} \rightarrow D$ we have

$$f(g(x_1), \dots, g(x_n)) = g(x),$$

2. if $((x_1, \dots, x_n), y) \in P$ where $y \notin \{x_1, \dots, x_n\}$ then $(g(x_1), \dots, g(x_n)) \in \text{dom}(f)$ for every $g: \{x_1, \dots, x_n\} \rightarrow D$, and

which is undefined otherwise.

Functions in $I_D(P)$ are sometimes called *pattern polymorphisms*.

Example 5.18. If we revisit Example 5.16 then for the Mal'tsev term $P = \{((x, x, y), y), (y, x, x), y)\}$ we have $I_{\{0,1\}}(P) = \{m\}$ for

1. $m(1, 1, 0) = m(0, 1, 1) = m(0, 0, 0) = 0$, and
2. $m(0, 0, 1) = m(1, 0, 0) = m(1, 1, 1) = 1$,

and which is undefined for $(0, 1, 0)$ and $(1, 0, 1)$. Note that this function is *idempotent* and *self-dual* in the sense that negating the input variables corresponds to negating the output.

In contrast, for $P' = \{((x, x, y), x), ((x, y, x), x), ((y, x, x), x)\}$ we get $I_{\{0,1\}}(P') = \{m'\}$ where m' is the totally defined majority function on the Boolean domain. However, for any D with at least three elements $I_D(P')$ will contain a properly partial function.

If $I_D(P) = \emptyset$ then P is *inconsistent* over D and if $I_D(P)$ contains all n -ary functions over D then P is said to be *trivial* over A . A pattern which is not inconsistent is simply said to be *consistent*. We observe that for a consistent pattern P we have $|I_D(P)| = 1$ *unless* P contains an identity $((x_1, \dots, x_n), y)$ for $y \notin \{x_1, \dots, x_n\}$, in which case this identity induces $|D|$ different partial functions $f_1, \dots, f_{|D|}$ where $f_i(g(x_1), \dots, g(x_n)) = i$ for every $g: \{x_1, \dots, x_n\} \rightarrow D$. In other words the specific value for this identity is arbitrary, and can be uniquely defined as a *multifunction* f such that $f(g(x_1), \dots, g(x_n)) = D$. However, to avoid overloading the technical machinery more than needed, we stick with the viewpoint that $I_A(P)$ is a set of functions. We remark that patterns P such that $|I_D(P)| > 1$ are typically too restrictive to be interesting in the general setting, but they are relevant in the multisorted setting discussed below.

We proceed by relating $I_D(P)$ to the property of *commuting* with all maps which, intuitively, shows that the domain of any function in $I_D(P)$ is highly symmetric.

Definition 5.19. Let f be an n -ary partial function over D and let $g: D \rightarrow D$ be a unary map. We say that f *commutes* with g if $f(g(x_1), \dots, g(x_n)) = g(f(x_1, \dots, x_n))$ for all $(x_1, \dots, x_n) \in \text{dom}(f)$.

Example 5.20. Recall that the partial Boolean Mal'tsev operation m was idempotent and self-dual. One alternative way to define this notion is that m commutes with the two Boolean constant maps and with Boolean negation $g(x) = 1 - x$.

We lift this notion to pairs of functions (possibly defined over different domains) as follows.

Definition 5.21. Let f and f' be n -ary partial functions over D_1 and D_2 , respectively. We say that f, f' *commute* with $g: D_1 \rightarrow D_2$ if

$$f'(g(x_1), \dots, g(x_n)) = g(f(x_1, \dots, x_n))$$

and is defined for all $(x_1, \dots, x_n) \in \text{dom}(f)$.

Proposition 5.22. *Let P be an n -ary polymorphism pattern and let D_1 and D_2 be finite sets. For every $p \in I_{D_1}(P)$ there exists $p' \in I_{D_2}(P)$ such that p, p' commutes with every unary $g: D_1 \rightarrow D_2$.*

Proof. Let $p \in I_{D_1}(P)$. We observe that if $((x_1, \dots, x_n), x) \in P$ where $x \in \{x_1, \dots, x_n\}$ then, for every $p' \in I_{D_2}(P)$, we must have that $p'(g(b_1), \dots, g(b_n)) = g(p(b_1, \dots, b_n))$ for every $g: D_1 \rightarrow D_2$ and $b_1, \dots, b_n$ such that there is $h: \{x_1, \dots, x_n\} \rightarrow D_1$, where $b_i = h(x_i)$.

On the other hand, assume that $((x_1, \dots, x_n), y) \in P$ for $y \notin \{x_1, \dots, x_n\}$. Let $b_1, \dots, b_n \in D_1$ such that there is $h: \{x_1, \dots, x_n\} \rightarrow D_1$ where $h(x_i) = b_i$ for every $1 \leq i \leq n$, i.e., $b_1, \dots, b_n$ matches the pattern, as witnessed by the function h . There then exists $p' \in I_{D_2}(P)$ such that $g(p(b_1, \dots, b_n)) = p'(g(b_1), \dots, g(b_n))$ for every $g: D_1 \rightarrow D_2$. $\square$

We will often work in the following setting: consider finite, disjoint sets $D_1, \dots, D_k$ for some $k \geq 1$, and a relation $R \subseteq (D_1 \cup \dots \cup D_k)^r$ with an associated signature $s: [k] \rightarrow [r]$ which for each argument returns the corresponding sort, i.e., if $h(i) = j$ then $t_i \in D_j$ for each $t \in R$. Note that we for each D and $r \geq 1$ can construct r disjoint copies $D_1, \dots, D_r$ of R and thus simply speak about r -ary multisorted relations over D . To ease the notation we thus sometimes simply say that R is an r -ary multisorted relation over D , rather than over disjoint copies $D_1, \dots, D_r$. It is then convenient not only to let the pattern act independently on each domain, but to allow different patterns for each domain type, which is in line with multisorted algebras in the CSP literature [18].

Definition 5.23. A *multisorted pattern* is a tuple $(P_1, \dots, P_r)$ where P_i is an n -ary polymorphism pattern for each $i \in [r]$. We say that P preserves an m -ary multisorted promise relation (S, T) over $D_1 \cup \dots \cup D_r$ with signature $s: [m] \rightarrow [r]$ if

$$(p_{s(1)}(x_1^{(1)}, \dots, x_1^{(n)}), \dots, p_{s(m)}(x_m^{(1)}, \dots, x_m^{(n)})) \in T$$

for all $p_{s(1)} \in I_{D_{s(1)}}(P_{s(1)}), \dots, p_{s(m)} \in I_{D_{s(m)}}(P_{s(m)})$ and $x^{(1)}, \dots, x^{(n)} \in S$ where each $(x_i^{(1)}, \dots, x_i^{(n)}) \in \text{dom}(p_{s(i)})$.

This naturally allows us to speak about the set of patterns preserving a given promise relation.

Definition 5.24. For a (multisorted) promise relation (S, T) over D ($D_1 \cup \dots \cup D_r$) we let

$$\text{Pattern}(S, T) = \{P \mid P \text{ is an } n\text{-ary pattern, } n \geq 1, (S, T) \in \text{Inv}(\{P\})\}$$

and

$$\text{mPattern}(S, T) = \{(P_1, \dots, P_r) \mid (P_1, \dots, P_r) \text{ is a multisorted pattern, } (S, T) \in \text{Inv}(\{P\})\}.$$

We are now interested in properties of $\text{Pattern}(\cdot)$ (and $\text{mPattern}(\cdot)$) when viewed as algebraic objects themselves. With the aim of showing that these sets form minor like objects we define the following closure operator for (multisorted) patterns.

Definition 5.25. Let $P = (P_1, \dots, P_r)$ be a multisorted pattern where each P_i has arity n . A *variable identification minor* of P is a multisorted pattern $P_{/h} = (P_{1/h}, \dots, P_{r/h})$ where $P_{i/h} = \{(x_1, \dots, x_m), x\} \mid (x_{h(1)}, \dots, x_{h(n)}), x \in P_i\}$ for some $h: [n] \rightarrow [m]$ and $m \geq 1$.

Similarly to minors of partial functions we then say that P' is a *minor* of a multisorted pattern P if it is a subset of a variable identification minor of P .

Definition 5.26. For a set of multisorted patterns Q we let $[Q]_{\min} = \{P' \mid P' \text{ is a minor of } P \in Q\}.$

Simultaneously, it then also makes sense to describe the relations invariant under a set of multisorted patterns.

Definition 5.27. For a set of (multisorted) polymorphism patterns Q and D we write $\text{Inv}_D(Q)$ for the set of all (multisorted) (S, T) over D preserved by each $P \in Q$, and $\text{Inv}(Q) = \bigcup_{i \geq 1, |D|=i} \text{Inv}_D(Q)$ for the set of all (multisorted) relations, over some finite domain, preserved by Q .

We can relate these operators together as follows.

Proposition 5.28. *For any set Q of (multisorted) polymorphism patterns $\text{Inv}(Q) = \text{Inv}([Q]_{\min})$.*

Proof. Since $Q \subseteq [Q]_{\min}$ it trivially follows that $\text{Inv}([Q]_{\min}) \subseteq \text{Inv}(Q)$. For the other direction, pick $(S, T) \in \text{Inv}(Q)$ over some $D = D_1 \times \dots \times D_r$. Let $P' \in [Q]_{\min}$ be a multisorted polymorphism pattern. First, assume that P' is a variable identification minor of some $P \in Q$. Then P' must preserve (S, T) , too, since each function in $I_D(P')$ is a variable identification minor of a function in $I_D(P)$, and we may simply apply Proposition 5.8. Similarly, if $P'' \subseteq P'$, then P'' preserves (S, T) since every partial function in $I_D(P'')$ is a subfunction of a partial function in $I_D(P)$. $\square$

Similarly, we obtain the following proposition via an abstraction of Lemma 5.5 on the pattern level.

Proposition 5.29. $\text{mPattern}(S, T) = [\text{mPattern}(S, T)]_{\min}$ for any multisorted promise relation (S, T) .

Thus, patterns can be used to define partial promise polymorphisms and form reasonable algebraic objects in their own right. It is natural to then ask what happens with the general algebraic picture if we *only* consider partial promise polymorphisms definable by patterns. A limited case of this setting was considered by Carbonnel [21] who introduced *functionally guarded pp-definitions* (fgpp-definitions) as a relational counter part to patterns. Here, the basic building blocks over some r -ary relation R over a domain D are atoms of the form $R(g_1(x_1), \dots, g_r(x_r))$ where $x_1, \dots, x_r$ are variables and $g_1, \dots, g_r : D \rightarrow D$ are unary functions (maps). Each combination of a map and a variable can then be seen as a generalization of a Boolean literal to arbitrary maps over arbitrary finite domains. This notion generalizes to maps between arbitrary finite domains and we say that (S_1, T_1) over D_1 has a *functionally guarded promise pp-definition* (fgppp-definition) over (S_2, T_2) over D_2 if there are maps $g_1, \dots, g_m : D_1 \rightarrow D_2$ and qfpp-formulas φ^{S_2} and φ^{T_2} (differing only in the sense that the former uses S_2 and the latter T_2) such that

$$S(x_1, \dots, x_r) \equiv \varphi^{\Gamma_1}(\ell_1, \dots, \ell_k)$$

and

$$T(x_1, \dots, x_r) \equiv \varphi^{\Gamma_2}(\ell_1, \dots, \ell_k)$$

where each ℓ_i is a combination of a map $g_i \in \{g_1, \dots, g_m\}$ and a variable from $\{x_1, \dots, x_r\}$. In the multisorted setting we associate each sort $1 \leq i \leq r$ with a set of maps and only allow maps in position i to use maps of that type. Concretely, we obtain the following links between (multisorted) patterns and fgppp-definitions.

Theorem 7.9. *Let (S_1, T_1) be a promise relation over D_1 and (S_2, T_2) a promise relation over D_2 . Then $\text{Pattern}(S_2, T_2) \subseteq \text{Pattern}(S_1, T_1)$ if and only if (S_2, T_2) fgppp-defines (S_1, T_1) .*

Corollary 7.12. *Let (S_1, T_1) be a multisorted promise relation over disjoint $D_1, \dots, D_r$ and (S_2, T_2) a multisorted promise relation over disjoint $E_1, \dots, E_s$. Then we have $\text{mPattern}(S_2, T_2) \subseteq \text{mPattern}(S_1, T_1)$ if and only if (S_2, T_2) fgppp-defines (S_1, T_1) .*

We defer the reader to Section 7.1 for the proofs since fgppp-definitions turn out to be a special case of a more powerful concept where non-unary functions are allowed.

6 Catalan Polymorphisms and Linear NRD

In this section, we use pattern polymorphisms to study the properties of predicates which have *linear* non-redundancy. In other words, $\text{NRD}(R, n) = O_R(n)$. In the literature, the primary known reason for a predicate $P \subseteq D^r$ having linear redundancy is the existence of a *Mal'tsev embedding* [61, 9]. To recall Section 2.3, we say that P has a Mal'tsev embedding if there exists a (finite) domain $E \supseteq D$ and a predicate $Q \subseteq E^r$ such that

- $Q \cap D^r = P$, and
- There exists a *Mal'tsev term* $\varphi \in \text{Pol}(Q)$. That is, φ is ternary and satisfies the identity $\varphi(x, x, y) = \varphi(y, x, x) = y$.

For any predicate R admitting a Mal'tsev embedding over a finite domain E one can prove that $\text{NRD}(R, n) = O_E(n)$ by applying the simple algorithm for Mal'tsev constraints [16] in order to compute a sufficiently compact basis [61].

Remark 6.1. As pointed out in [9], a slightly larger class of predicates has linear redundancy: the class of predicates which can be fgpp-defined from predicates with Mal'tsev embeddings. However, they prove that any such predicate has what is called an *infinite* Mal'tsev embedding (i.e., the domain E is infinite).

A special class of Mal'tsev embeddings is the *group embeddings*. In particular, imagine that $(E, \cdot, {}^{-1})$ is a group. Then, $\varphi(x, y, z) = x \cdot y^{-1} \cdot z$ is a Mal'tsev term. Under this stipulation, the property of $\varphi \in \text{Pol}(Q)$ is equivalent to Q being a coset of the group. We call such embeddings *group embeddings*. If the group happens to be an Abelian group, we call it an *Abelian embedding*. For these special classes of Mal'tsev embeddings we can also make the distinction between embeddings over a finite or infinite domain. In particular, it is known that Abelian embeddings over an infinite domain (e.g. $(\mathbb{Z}, +)$) always imply an Abelian embedding over a finite domain—see for instance Section 6.1 of [54].

The goal of this section is to advance the theory of Mal'tsev embeddings by deriving new properties. In particular, we make the following contributions:

- We introduce a new family of polymorphism identities known as the *Catalan identities*. In particular, we show that every clone with a Mal'tsev term has *Catalan terms* (or *Catalan polymorphisms*) which satisfy these identities.
- As an immediate corollary, we show the Catalan identities imply that every finite predicate with an infinite Mal'tsev embedding has an embedding into an infinite Coxeter group.
- Furthermore, we can prove that every Boolean predicate with an infinite Mal'tsev embedding is in fact *balanced* (i.e., has an Abelian embedding). This resolves an open question of [23].
- Conversely, we construct a predicate over domain size three with a finite group embedding but no Abelian embedding. Combining this result with the main theorem of [14], this gives the first explicit example of a predicate with near-linear sparsifiability that doesn't derive from the framework of [54].

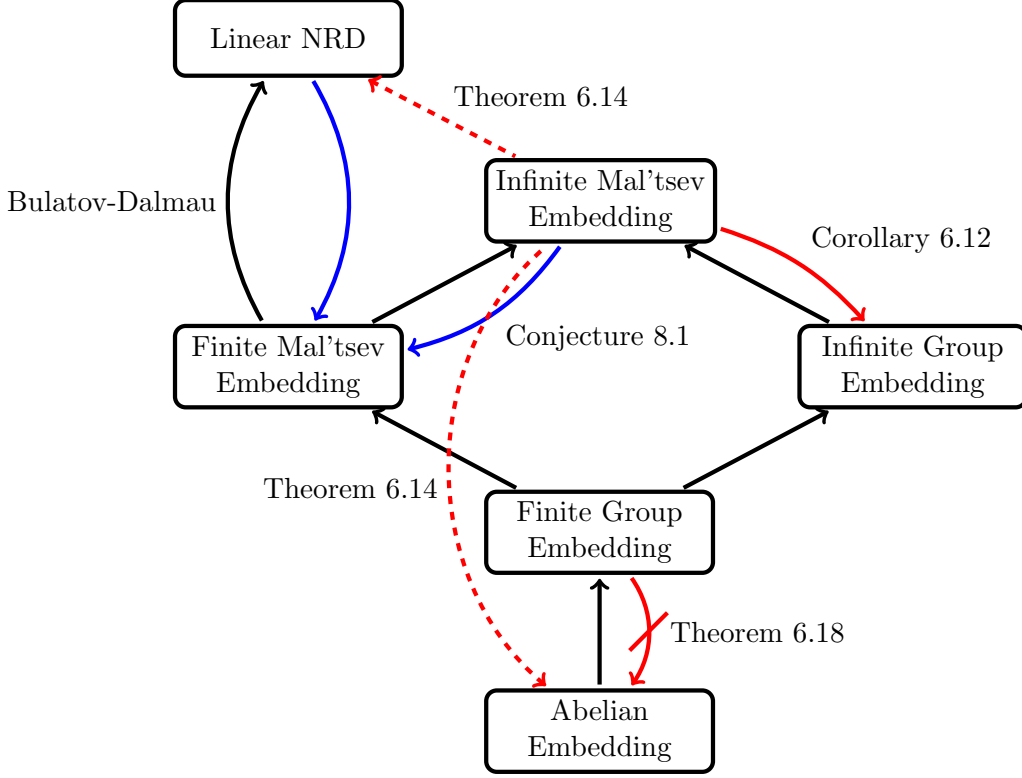


Figure 6: Black edges are known results. Red edges are new results proved in this paper. Blue edges are conjectures. Dashed edges apply to Boolean predicates only. Note that the notion of an “Abelian embedding” is the same for finite and infinite groups due to a result of [54].

- Furthermore, the Catalan identities streamline existing proofs in the literature that certain predicates lack infinite Mal’tsev embeddings.

See Figure 6 for an overview of existing results alongside our new results.

6.1 Catalan Identities and Polymorphisms

Recall that a ternary operator $\varphi : D^3 \rightarrow D$ is a Mal’tsev term if it satisfies the identities $\varphi(x, x, y) = \varphi(y, x, x) = y$ for all $x, y \in D$. One can generalize the Mal’tsev identity to arbitrary odd arity as follows.

Definition 6.2 (Catalan identities). We say that a sequence of terms $\{\psi_1, \psi_3, \psi_5, \dots\}$ such that ψ_m has arity m for all odd $m \in \mathbb{N}$ satisfies the *Catalan identities* if the following properties hold.

- For all x , $\psi_1(x) = x$.
- For all odd $m \in \mathbb{N}$ at least 3 and all $i \in [m - 1]$, we have that

$$\text{for all } x_1, \dots, x_m, x_i = x_{i+1} \implies \psi_m(x_1, \dots, x_m) = \psi_{m-2}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m). \quad (3)$$

Such terms are called *Catalan terms* or *Catalan polymorphisms*.

Note that Definition 6.2 applies equally well to total and partial functions. In the latter case we naturally still require that ψ_1 is totally defined and the condition $\psi_m(x_1, \dots, x_m) = \psi_{m-2}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m)$ then means that ψ_m and ψ_{m-2} are both defined and agree or that both equal $\perp$.

Remark 6.3. We coin the name *Catalan identities* because of the close relationship between these identities and the Catalan numbers (see, e.g., Chapter 14 of [79]). In particular, one definition of the n th Catalan number c_n is the number of syntactically valid strings consisting of n pairs of left and right parentheses. If one “flattens” the Catalan identities by recursively applying (3) $(m-1)/2$ times, we get exactly $c_{(m+1)/2}$ inequivalent identities. For example, when $m = 5$ we get $c_3 = 5$ identities:

$$\begin{aligned}\psi_5(x, x, y, y, z) &= z \\ \psi_5(x, x, z, y, y) &= z \\ \psi_5(z, x, x, y, y) &= z \\ \psi_5(x, y, y, x, z) &= z \\ \psi_5(z, x, y, y, x) &= z\end{aligned}$$

Furthermore, the derivation of Catalan terms from a Mal'tsev term (equations (4) and (5)) follows a recursive structure similar to that of the recursion defining the Catalan numbers: $c_n = c_0 c_{n-1} + c_1 c_{n-2} + \dots + c_{n-1} c_0$.

The main result of this subsection is to show that any clone (over a finite or infinite domain) with a Mal'tsev term also has all the Catalan terms. Recall from Section 5.1 that we for a set of functions F let $[F]$ be the clone generated by F , which in particular contains all possible term operations over F . To ease the notation we write $[f]$ rather than $[\{f\}]$ for a singleton set $F = \{f\}$.

Theorem 6.4. *Let D be a (possibly infinite) domain and let $\varphi : D^3 \rightarrow D$ be a Mal'tsev term. Then there exist $f_1, f_3, f_5, \dots \in [\varphi]$ satisfying the Catalan identities. Conversely, if a sequence of terms $f_1, f_3, \dots$ satisfies the Catalan identities, then f_3 is a Mal'tsev term.*

Consider the following sequence of operators $\{f_{2i+1}\}_{i \geq 0}$ such that f_{2i+1} has arity $2i + 1$.

$$\begin{aligned}f_1(x_1) &:= x_1 \\ f_3(x_1, x_2, x_3) &:= \varphi(x_1, x_2, x_3) \\ f_{2k+1}(x_1, \dots, x_{2k+1}) &:= f_{2k-1}(g_2^{2k+1}(x_1, \dots, x_{2k+1}), \dots, g_{2k}^{2k+1}(x_1, \dots, x_{2k+1})),\end{aligned}\tag{4}$$

where for $k \geq 2$ and $j \in [2k + 1]$ we have that

$$g_j^{2k+1}(x_1, \dots, x_{2k+1}) := \begin{cases} \varphi(f_{j-1}(x_1, \dots, x_{j-1}), x_j, f_{2k+1-j}(x_{j+1}, \dots, x_{2k+1})) & j \text{ even} \\ \varphi(f_j(x_1, \dots, x_j), x_j, f_{2k+2-j}(x_{j+1}, \dots, x_{2k+1})) & j \text{ odd.} \end{cases}\tag{5}$$

Remark 6.5. Note that for $j \in \{1, 2k + 1\}$, we define g_j^{2k+1} , even though these do not appear in (4). This is to facilitate the inductive argument in the proof of Lemma 6.6. Note by (5) we have that

$$g_1^{2k+1}(x_1, \dots, x_{2k+1}) = \varphi(x_1, x_1, f_{2k+1}(x_1, \dots, x_{2k+1})) = f_{2k+1}(x_1, \dots, x_{2k+1})\tag{6}$$

$$g_{2k+1}^{2k+1}(x_1, \dots, x_{2k+1}) = \varphi(f_{2k+1}(x_1, \dots, x_{2k+1}), x_{2k+1}, x_{2k+1}) = f_{2k+1}(x_1, \dots, x_{2k+1})\tag{7}$$

As a concrete example, when $k = 2$, we have that (4) becomes

$$\begin{aligned} f_5(x_1, x_2, x_3, x_4, x_5) &= \varphi(\varphi(x_1, x_2, \varphi(x_3, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_2, x_3), x_3, \varphi(x_3, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_2, x_3), x_4, x_5)). \end{aligned}$$

Theorem 6.4 immediately follows by proving the following lemma.

Lemma 6.6 (Cancellation Lemma). *For all $k \geq 1$ and all $i \in [2k]$, we have that*

$$x_i = x_{i+1} \implies f_{2k+1}(x_1, \dots, x_{2k+1}) = f_{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}). \quad (8)$$

Example 6.7. To help get intuition for the proof, we include a couple of worked examples.

If $k = 2$ and $i = 1$, we have that

$$\begin{aligned} f_5(x_1, x_1, x_3, x_4, x_5) &= \varphi(\varphi(x_1, x_1, \varphi(x_3, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_1, x_3), x_3, \varphi(x_3, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_1, x_3), x_4, x_5)). \\ &= \varphi(\varphi(x_3, x_4, x_5), \varphi(x_3, x_3, \varphi(x_3, x_4, x_5)), \varphi(x_3, x_4, x_5)) \\ &= \varphi(\varphi(x_3, x_4, x_5), \varphi(x_3, x_4, x_5), \varphi(x_3, x_4, x_5)) \\ &= \varphi(x_3, x_4, x_5) = f_3(x_3, x_4, x_5). \end{aligned}$$

If $k = 2$ and $i = 2$, we have that

$$\begin{aligned} f_5(x_1, x_2, x_2, x_4, x_5) &= \varphi(\varphi(x_1, x_2, \varphi(x_2, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_2, x_2), x_2, \varphi(x_2, x_4, x_5)), \\ &\quad \varphi(\varphi(x_1, x_2, x_2), x_4, x_5)). \\ &= \varphi(\varphi(x_1, x_2, \varphi(x_2, x_4, x_5)), \varphi(x_1, x_2, \varphi(x_2, x_4, x_5)), \varphi(x_1, x_4, x_5)) \\ &= \varphi(x_1, x_4, x_5) = f_3(x_1, x_4, x_5), \end{aligned}$$

where the last line uses that that first two arguments of the outer φ are equal.

Proof of Lemma 6.6. We prove this by strong induction on k . The case $k = 1$ asserts that $\varphi(x, x, y) = \varphi(y, x, x) = y$, which is precisely the definition of a Mal'tsev term. Now assume $k \geq 2$ and fix $i \in [2k]$ (i.e., the assumption $x_i = x_{i+1}$). To start, we seek to compute what happens for any $j \in \{2, \dots, 2k\}$ how g_j^{2k+1} simplifies when $x_i = x_{i+1}$. We break this into cases.

Case 1, $j \leq i - 1$. In this case, we seek to show that

$$x_i = x_{i+1} \implies g_j^{2k+1}(x_1, \dots, x_{2k+1}) = g_j^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}). \quad (9)$$

We split into cases based on the parity of j . First, if j is even, $x_i = x_{i+1}$ implies that

$$\begin{aligned} g_j^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_{j-1}(x_1, \dots, x_{j-1}), x_j, f_{2k+1-j}(x_{j+1}, \dots, x_{2k+1})) \\ &= \varphi(f_{j-1}(x_1, \dots, x_{j-1}), x_j, f_{2k-1-j}(x_{j+1}, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1})) \\ &= g_j^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \end{aligned}$$

where the second equality uses the strong induction hypothesis on f_{2k+1-j} . Second, if j is odd, then $x_i = x_{i+1}$ implies that

$$\begin{aligned} g_j^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_j(x_1, \dots, x_j), x_j, f_{2k+2-j}(x_j, \dots, x_{2k+1})) \\ &= \varphi(f_j(x_1, \dots, x_j), x_j, f_{2k-j}(x_j, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1})) \\ &= g_j^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \end{aligned}$$

where the second equality uses the strong induction hypothesis on f_{2k+2-j} (note that $2k+2-j \leq 2k-1$ since $j \geq 2$ is odd). This proves (9).

Case 2, $j \geq i+2$. In this case, we seek to show that

$$x_i = x_{i+1} \implies g_j^{2k+1}(x_1, \dots, x_{2k+1}) = g_{j-2}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}). \quad (10)$$

Like in Case 1, we do casework based on the parity of j . The relevant calculations for j even are

$$\begin{aligned} g_j^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_{j-1}(x_1, \dots, x_{j-1}), x_j, f_{2k+1-j}(x_{j+1}, \dots, x_{2k+1})) \\ &= \varphi(f_{j-3}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{j-1}), x_j, f_{2k-1-(j-2)}(x_{j+1}, \dots, x_{2k+1})) \\ &= g_{j-2}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \end{aligned}$$

and the calculations for j odd are

$$\begin{aligned} g_j^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_j(x_1, \dots, x_j), x_j, f_{2k+2-j}(x_j, \dots, x_{2k+1})) \\ &= \varphi(f_{j-2}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_j), x_j, f_{2k-(j-2)}(x_j, \dots, x_{2k+1})) \\ &= g_{j-2}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}). \end{aligned}$$

Case 3, $j \in \{i, i+1\}$ and $i \notin \{1, 2k\}$. In this case, instead of simplifying g_j^{2k+1} , we seek to show an equality which will help for proving (8):

$$x_i = x_{i+1} \wedge i \notin \{1, 2k\} \implies g_i^{2k+1}(x_1, \dots, x_{2k+1}) = g_{i+1}^{2k+1}(x_1, \dots, x_{2k+1}). \quad (11)$$

If i is even, then $x_i = x_{i+1}$ implies that

$$\begin{aligned} g_i^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_{i-1}(x_1, \dots, x_{i-1}), x_i, f_{2k+1-i}(x_{i+1}, \dots, x_{2k+1})) \\ &= \varphi(f_{i+1}(x_1, \dots, x_{i-1}, x_i, x_{i+1}), x_{i+1}, f_{2k+1-i}(x_{i+1}, \dots, x_{2k+1})) \\ &= g_{i+1}^{2k+1}(x_1, \dots, x_{2k+1}), \end{aligned}$$

where we use $i \neq 2k$ so that $i+1 \leq 2k-1$, allowing for us to invoke the induction hypothesis. If i is odd, then $x_i = x_{i+1}$ implies that

$$\begin{aligned} g_i^{2k+1}(x_1, \dots, x_{2k+1}) &= \varphi(f_i(x_1, \dots, x_i), x_i, f_{2k+2-i}(x_i, x_{i+1}, x_{i+2}, \dots, x_{2k+1})) \\ &= \varphi(f_i(x_1, \dots, x_i), x_{i+1}, f_{2k-i}(x_{i+2}, \dots, x_{2k+1})) \\ &= g_{i+1}^{2k+1}(x_1, \dots, x_{2k+1}), \end{aligned}$$

where we use $i \neq 1$ so that $2k+2-i \leq 2k-1$, allowing for us to invoke the induction hypothesis.

Finishing the proof. We now prove (8). We first consider the boundary cases that $i \in \{1, 2k\}$. If $i = 1$ (that is $x_1 = x_2$), note that

$$g_2^{2k+1}(x_1, \dots, x_{2k+1}) = \varphi(x_1, x_2, f_{2k-1}(x_3, \dots, x_{2k+1})) = f_{2k-1}(x_3, \dots, x_{2k+1}). \quad (12)$$

By the definition of f_{2k+1} , we have that

$$\begin{aligned} x_1 = x_2 &\implies f_{2k+1}(x_1, \dots, x_{2k+1}) \\ &= f_{2k-1}(g_2^{2k+1}(x_1, \dots, x_{2k+1}), g_3^{2k+1}(x_1, \dots, x_{2k+1}), \\ &\quad g_4^{2k+1}(x_1, \dots, x_{2k+1}), \dots, g_{2k}^{2k+1}(x_1, \dots, x_{2k+1})) \\ &= f_{2k-1}(f_{2k-1}(x_3, \dots, x_{2k+1}), g_1^{2k-1}(x_3, \dots, x_{2k+1}), \\ &\quad g_2^{2k-2}(x_3, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k-1}(x_3, \dots, x_{2k+1})) \text{ by (12) and (10)} \\ &= f_{2k-1}(f_{2k-1}(x_3, \dots, x_{2k+1}), f_{2k-1}(x_3, \dots, x_{2k+1}), \\ &\quad g_2^{2k-1}(x_3, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k-1}(x_3, \dots, x_{2k+1})) \text{ by (6)} \\ &= f_{2k-3}(g_2^{2k-1}(x_3, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k-1}(x_3, \dots, x_{2k+1})) \text{ by induction hypothesis} \\ &= f_{2k-1}(x_3, \dots, x_{2k+1}), \end{aligned}$$

as desired. If $i = 2k$ (that is $x_{2k} = x_{2k+1}$), note that

$$g_{2k}^{2k+1}(x_1, \dots, x_{2k+1}) = \varphi(f_{2k-1}(x_1, \dots, x_{2k-1}), x_{2k}, x_{2k+1}) = f_{2k-1}(x_1, \dots, x_{2k-1}). \quad (13)$$

Likewise, we have that

$$\begin{aligned} x_{2k} = x_{2k+1} &\implies f_{2k+1}(x_1, \dots, x_{2k+1}) \\ &= f_{2k-1}(g_2^{2k+1}(x_1, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k+1}(x_1, \dots, x_{2k+1}), \\ &\quad g_{2k-1}^{2k+1}(x_1, \dots, x_{2k+1}), g_{2k}^{2k+1}(x_1, \dots, x_{2k+1})) \\ &= f_{2k-1}(g_2^{2k-1}(x_1, \dots, x_{2k-1}), \dots, g_{2k-2}^{2k-1}(x_1, \dots, x_{2k-1}), \\ &\quad g_{2k-1}^{2k-1}(x_1, \dots, x_{2k-1}), f_{2k-1}(x_1, \dots, x_{2k-1})) \text{ by (9) and (13)} \\ &= f_{2k-1}(g_2^{2k-1}(x_1, \dots, x_{2k-1}), \dots, g_{2k-2}^{2k-1}(x_1, \dots, x_{2k-1}), \\ &\quad f_{2k-1}(x_1, \dots, x_{2k-1}), f_{2k-1}(x_1, \dots, x_{2k-1})) \text{ by (7)} \\ &= f_{2k-3}(g_2^{2k-1}(x_1, \dots, x_{2k-1}), \dots, g_{2k-2}^{2k-1}(x_1, \dots, x_{2k-1})) \text{ by induction hypothesis} \\ &= f_{2k-1}(x_1, \dots, x_{2k-1}), \end{aligned}$$

as desired. Now assume that $i \in \{2, \dots, 2k-1\}$. By (9), (10), and (11), we have that

$$\begin{aligned}
x_i = x_{i+1} &\implies f_{2k+1}(x_1, \dots, x_{2k+1}) \\
&= f_{2k-1}(g_2^{2k+1}(x_1, \dots, x_{2k+1}), \dots, g_{i-1}^{2k+1}(x_1, \dots, x_{2k+1}), \\
&\quad g_i^{2k+1}(x_1, \dots, x_{2k+1}), g_{i+1}^{2k+1}(x_1, \dots, x_{2k+1}), \\
&\quad g_{i+2}^{2k+1}(x_1, \dots, x_{2k+1}), \dots, g_{2k}^{2k+1}(x_1, \dots, x_{2k+1})) \\
&= f_{2k-1}(g_2^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \dots, g_{i-1}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \\
&\quad g_i^{2k+1}(x_1, \dots, x_{2k+1}), g_{i+1}^{2k+1}(x_1, \dots, x_{2k+1}), \\
&\quad g_i^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1})) \text{ by (9) and (10).} \\
&= f_{2k-3}(g_2^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \dots, g_{i-1}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \\
&\quad g_i^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}), \dots, g_{2k-2}^{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1})) \text{ by (11) and IH} \\
&= f_{2k-1}(x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_{2k+1}),
\end{aligned}$$

as desired. This completes the proof of (8) assuming the induction hypothesis. $\square$

6.2 Pattern Polymorphism Perspective: Boolean Mal'tsev Embeddings are Balanced

Theorem 6.4 has a number of implications on the structure of Mal'tsev embeddings. To develop this theory properly, we first consider the Catalan identities from the perspective of partial pattern polymorphisms. In particular, we shall view the Catalan partial polymorphisms as special cases of *group pattern polymorphisms*.

Definition 6.8 (Group Pattern Polymorphisms). Let V be a (possibly infinite) set of variables, and let G be a (possibly infinite) group with a corresponding map $\eta : V \rightarrow G$. For every odd $m \in \mathbb{N}$, we define the pattern polymorphism $\text{Grp}_m^{V,G,\eta}$ to contain all patterns $((x_1, \dots, x_m), y)$ such that

$$\eta(x_1) \cdot \eta(x_2)^{-1} \cdot \eta(x_3) \cdot \dots \cdot \eta(x_{m-1})^{-1} \cdot \eta(x_m) = \eta(y).$$

In other words $\text{Grp}_m^{V,G,\eta}$ captures the arity- m patterns which arise from a group embedding into G via η .

Group pattern polymorphisms can capture many properties of predicates found in the literature. For example, [23] consider the notion of a *balanced* Boolean predicate as a potential characterization of Boolean predicates with linear non-redundancy. We say that a Boolean predicate $P \subseteq \{0, 1\}^r$ is *balanced* if for any odd $m \in \mathbb{N}$ and tuples $t^1, \dots, t^m \in P$, we have the following property: if for each $i \in [r]$ we have $t_i^1 - t_i^2 + t_i^3 - \dots + t_i^m \in \{0, 1\}$, then $t^1 - t^2 + t^3 - \dots + t^m \in P$. In other words, the operation defined by

$$f(x_1, x_2, \dots, x_m) = \begin{cases} x_1 - x_2 + \dots + x_m & \text{if in } \{0, 1\} \\ \perp & \text{otherwise} \end{cases}$$

is a partial polymorphism of P . We now demonstrate that this property of being balanced can be captured by the following group pattern polymorphism

$$\text{Bal}_m := \text{Grp}_m^{\{x,y\}, \mathbb{Z}, \eta}$$

where $\eta(x) = 0$ and $\eta(y) = 1$.

Proposition 6.9. *Any $P \subseteq \{0, 1\}^r$ is balanced if and only if $I_{\{0,1\}}(\text{Bal}_m) \subseteq \text{pPol}(P)$ for all odd $m \in \mathbb{N}$.*

Proof. The condition $I_{\{0,1\}}(\text{Bal}_m) \subseteq \text{pPol}(P)$ is equivalent to the following condition: for all $t^1, \dots, t^m \in P$ satisfying

$$I_{\{0,1\}}(\text{Bal}_m)(t_i^1, t_i^2, \dots, t_i^m) \in \{0, 1\}$$

for each $i \in [r]$, we have $I_{\{0,1\}}(\text{Bal}_m)(t^1, t^2, \dots, t^m) \in P$. However, by definition of Bal_m , we have that $I_{\{0,1\}}(\text{Bal}_m)(t_i^1, t_i^2, \dots, t_i^m) = t_i^1 - t_i^2 + t_i^3 - \dots + t_i^m$ whenever the latter is contained in $\{0, 1\}$. Therefore, this condition is equivalent to P being balanced. $\square$

6.2.1 Alternating Sums and Abelian Embeddings

However, since Bal_m only considers patterns with two symbols, there are multiple natural generalizations into more general patterns. One such generalization, roughly equivalent to the approach taken by [54], is to let G be the free Abelian group generated by V , which we will denote as $\mathbb{Z}^V$ with coordinate-wise addition as the group operation. In other words, the elements of G consist of all (finite) formal sums of the form $z_1 \cdot x_1 + \dots + z_\ell \cdot x_\ell$, where $x_1, \dots, x_\ell \in V$ and $z_1, \dots, z_\ell \in \mathbb{Z}$, with addition and subtraction defined in the obvious manner. We let η be the “identity” map from V to $\mathbb{Z}^V$ and call the resulting group pattern polymorphisms *alternating sum patterns*.⁶ That is, for all odd $m \in \mathbb{N}$, we define $\text{AS}_m^V := \text{Grp}_m^{V, \mathbb{Z}^V, \eta}$. The most general such alternating sum pattern is when V has countably infinite domain. We denote this family of pattern polymorphisms by $\{\text{AS}_m\}$. We now show this family of pattern polymorphisms exactly captures Abelian embeddings.

Proposition 6.10. *For any $P \subseteq D^r$, the following are equivalent.*

- P has a finite Abelian embedding.
- P has an embedding into the free Abelian group $\mathbb{Z}^D$ via $\eta(d) := e_d$.
- $I_D(\text{AS}_m) \subseteq \text{pPol}(P)$ for all odd $m \in \mathbb{N}$.

Proof. First, we show that P has an Abelian embedding if and only if it has an embedding into $\mathbb{Z}^D$ in the canonical manner (i.e., each element of D maps to its corresponding generator). Assume that P has an embedding into some group H via a map $\eta : D \rightarrow H$. By the fundamental property of the free group, there exists a homomorphism $\sigma : \mathbb{Z}^D \rightarrow H$ such that $\eta = \sigma \circ \iota$, where $\iota : D \rightarrow \mathbb{Z}^D$ is the canonical embedding. As such, σ implies that η being an embedding of P into H implies that ι is an embedding of P into $\mathbb{Z}^D$. Note that P must also embed into a finite Abelian group using the lattice-based techniques developed in Section 6.1 of [54]. We omit further details on how this finite group is constructed.

We now proceed to prove the equivalence. Let $Q \subseteq (\mathbb{Z}^D)^r$ be the subgroup of $(\mathbb{Z}^D)^r$ generated by P . We claim that $Q \cap D^r = P$ if and only if $I_D(\text{AS}_m) \subseteq \text{pPol}(P)$.

We first prove the “if” direction. Consider any $t \in Q \cap D^r$. Since Q is generated by P , we have there exists nonzero $z_1, \dots, z_\ell \in \mathbb{Z}$ and corresponding $t^1, \dots, t^\ell \in P$ such that $t = z_1 \cdot t^1 + \dots + z_\ell \cdot t^\ell$. Note that the sum of the coordinates of each t_i (and t) is equal to 1. Thus, we must have that $z_1 + \dots + z_\ell = 1$. Let $m = |z_1| + \dots + |z_\ell|$. We can see that the positive z_i ’s sum to precisely $(m+1)/2$ and negative z_i ’s sum to precisely $-(m-1)/2$. Thus, there exists a tuple $s \in [\ell]^m$ such

⁶We chose this name to be analogous to the “alternating threshold” polymorphisms considered in the theory of promise CSPs [13].

that each $j \in [\ell]$ appears in exactly $|z_j|$ positions of s . Furthermore, j only appears in odd positions if z_j is positive and only in even positions if z_j is negative.

For each $i \in [r]$, we consider $s^i \in D^m$ where $s_j^i = t_i^{s_j}$. Then, since

$$t_i = z_1 \cdot t_i^1 + \cdots + z_\ell \cdot t_i^\ell = \sum_{j=1}^m (-1)^{j-1} t_i^{s_j} = \sum_{j=1}^m (-1)^{j-1} s_j^i.$$

Therefore, $(s_j^i, t_i) \in \text{AS}_m$ for all $i \in [r]$. In particular, this means that $I_D(\text{AS}_m) \subseteq \text{pPol}(P)$ implies $t \in P$. Therefore, $Q \cap D^r = P$.

For the “only if” direction, assume that $I_D(\text{AS}_m) \not\subseteq \text{pPol}(P)$. In particular, this means there exists $p \in I_D(\text{AS}_m)$ and $t^1, \dots, t^m \in P$ but $t \notin P$ such that $p(t_i^1, t_i^2, \dots, t_i^m) = t_i$ for all $i \in [r]$. Since $p \in I_D(\text{AS}_m)$, we must have that $t_i^1 - t_i^2 + \cdots - t_i^m = t_i$ for all $i \in$, where the sum is taken in $\mathbb{Z}^D$. Therefore, $t \in Q \cap D^r$, so $Q \cap D^r \neq P$. By taking the contrapositive, we have that $Q \cap D^r = P$ implies $I_D(\text{AS}_m) \subseteq \text{pPol}(P)$. $\square$

6.2.2 Catalan Patterns and Group Embeddings

However, we can also generalize the property of being balanced in a non-Abelian manner. In particular, given a set V of variables, we let $\text{Cox}(V)$ denote the group generated by the elements of V with the condition that for each $v \in V$, we have that v^2 is the identity element. We note that $\text{Cox}(V)$ is a special example of *Coxeter* groups [45]. In particular, it is isomorphic to the group generated by $|V|$ reflections in general position in $\mathbb{R}^2$. The group can also be thought of as the *free product* of V copies of $\mathbb{Z}/2\mathbb{Z}$.

The simplest way to think about $\text{Cox}(V)$ is that it consists of *words* that lack immediate repetition. For example, if $V = \{x, y, z\}$, a valid computation is

$$xyz \cdot zxy \cdot yxyxy = y.$$

Furthermore, inverses just involve reversing the symbols in a word, so zyx is the inverse of xyz .

If we let η denote the canonical embedding from V into $\text{Cox}(V)$, then for all $m \in \mathbb{N}$, we can define

$$\text{Cat}_m^V := \text{Grp}_m^{V, \text{Cox}(V), \eta}.$$

If V is countably infinite, we let $\text{Cat}_m := \text{Cat}_m^V$. As the name suggests, these *Catalan pattern polymorphisms* precisely capture the *Catalan identities*.

Proposition 6.11. *For any $P \subseteq D^r$, there exists $\psi_1, \psi_3, \psi_5, \dots \in \text{pPol}(P)$ satisfying the Catalan identities if and only if $I_D(\text{Cat}_m) \subseteq \text{pPol}(P)$ for all odd m .*

Proof. First assume $I_D(\text{Cat}_m) \subseteq \text{pPol}(P)$ for all odd $m \in \mathbb{N}$. Thus, for each odd $m \in \mathbb{N}$, there exists a polymorphism $\psi_m \in I_D(\text{Cat}_m)$ such that $\psi_m(x_1, \dots, x_m) = x_1 \cdots x_m$ if the product in the Coxeter group evaluates to a single generator of $\text{Cox}(D)$. Otherwise, $\psi_m(x_1, \dots, x_m) = \perp$. We claim that $\psi_1, \psi_3, \dots$ satisfy the Catalan identities. First, clearly $\psi_1(x) = x$. Second, for any odd $m \in \mathbb{N}$ and $i \in [m-1]$, we have that $x_i = x_{i+1}$ implies that $x_i x_{i+1}$ in the Coxeter group, so $x_1 \cdots x_m = x_1 \cdots x_{i-1} x_{i+2} \cdots x_m$. In particular, $x_i = x_{i+1}$ implies that $\psi_m(x_1, \dots, x_m) = \psi_{m-2}(x_1, \dots, x_{i-2}, x_{i+1}, \dots, x_m)$ as either both equal the same monomial or both equal $\perp$.

Conversely, assume that there exists $\psi_1, \psi_3, \psi_5, \dots \in \text{pPol}(P)$ satisfying the Catalan identities. For each odd $m \in \mathbb{N}$ and each identity $((x_1, \dots, x_m), y) \in \text{Cat}_m$ for some $x_1, \dots, x_m, y \in V$, we seek

to prove that every interpretation of (t, x) over D is satisfied by ψ_m . We prove this by induction on m . The case $m = 1$ is trivial as $\psi_1(d) = d$ for all $d \in D$.

In order for $((x_1, \dots, x_m), y) \in \text{Cat}_m$, we must have that $x_1 \cdots x_m = y$ in $\text{Cox}(V)$. For this to be the case, we must have that some cancellation occurs. That is, there exists $i \in [m-1]$ such that $x_{i-1} = x_i$. Furthermore, we must have that $((x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m), y) \in \text{Cat}_{m-2}$. Thus, for every interpretation $\iota : V \rightarrow D$, we have by the Catalan identities that

$$\psi_m(\iota(x_1), \dots, \iota(x_m)) = \psi_{m-2}(\iota(x_1), \dots, \iota(x_{i-1}), \iota(x_{i+2}), \dots, \iota(x_m)) = y,$$

where for the last equality we use the induction hypothesis. Thus, $I_D(\text{Cat}_m) \subseteq \text{pPol}(P)$ for all odd m , as desired. $\square$

6.2.3 Corollary: Infinite Mal'tsev to Infinite Group

As an immediate corollary of Theorem 6.4 and Proposition 6.11, we have that infinite Mal'tsev embeddings are equivalent to infinite group embeddings. This sharpens the classification by Lagerkvist and Wahlström [61] who relate the existence of a Mal'tsev embedding to a Mal'tsev embedding over a certain free algebra over a countable infinite domain whose term operations coincide with the so-called *universal partial Mal'tsev* operations. As these “UPM” operations are rather opaque, our characterization in terms of an embedding into $\text{Cox}(D)$ and the resulting partial Catalan operations appears to be a much more useful description.

Corollary 6.12. *A finite predicate $P \subseteq D^r$ has an infinite Mal'tsev embedding if and only if P embeds into $\text{Cox}(D)$.*

Proof. Since a group embedding is a special case of Mal'tsev embedding, the “if” direction is trivial.

For the “only if” direction, by Theorem 6.4, P has partial polymorphisms satisfying the Catalan identities. By Proposition 6.11, this implies that $I_D(\text{Grp}_m^{V, \text{Cox}(V), \eta}) = I_D(\text{Grp}_m^{D, \text{Cox}(D), \eta}) \subseteq \text{pPol}(P)$ for all odd $m \in \mathbb{N}$. Let Q be the subgroup of $\text{Cox}(D)^r$ generated by P . Any $t \in Q \cap D^r$ must be expressed as a product $t^1 \cdots t^m$ of tuples in P . By a simple parity argument, m must be odd. Thus, there exists $p \in I_D(\text{Grp}_m^{D, \text{Cox}(D), \eta})$ such that $p(t_i^1, \dots, t_i^m) = t_i$ for all $i \in [r]$. Thus, $t \in P$, so P has an embedding into $\text{Cox}(D)$. $\square$

Remark 6.13. We can make Corollary 6.12 slightly more precise. Let $\widetilde{\text{Cox}}(D)$ be the index-2 coset of $\text{Cox}(D)$ consisting of words of odd length. Since each $d \in D$ embeds into an element of $\text{Cox}(D)$ of length 1, we in fact constructed an embedding into $\widetilde{\text{Cox}}(D)$. The upcoming Theorem 6.14 is equivalent to the fact that $\widetilde{\text{Cox}}(\{0, 1\}) \cong (\mathbb{Z}, +)$.

6.2.4 Corollary: Boolean Mal'tsev Embeddings are Balanced

As another corollary of our Catalan machinery, we show that for Boolean predicate P , having a Mal'tsev embedding (even over an infinite domain!) is sufficient to prove that P is balanced.

Theorem 6.14. *Let $P \subseteq \{0, 1\}^k$ be a predicate which embeds into a predicate $A \subseteq D^k$ with a Mal'tsev term $\varphi \in \text{Pol}(A)$. Then, P is balanced.*

This resolves an open question of [23] who only proved Theorem 6.14 in the special case that $\varphi(x, y, z) = x \cdot y^{-1} \cdot z$ where D embeds into a (possibly infinite) group. In fact, immediately applying Corollary 6.12 to their group embedding result implies Theorem 6.14.

However, to make the argument a bit more transparent, we give a more structural proof of Theorem 6.14. In particular, we show the identities implying that a predicate is balanced are identical to the identities implying a Boolean predicate has partial polymorphisms satisfying the Catalan identities.

Lemma 6.15. *For all odd $m \in \mathbb{N}$, we have that $I_{\{0,1\}}(\text{Bal}_m) = I_{\{0,1\}}(\text{Cat}_m)$.*

Theorem 6.14 then immediately follows by combining Lemma 6.15 with Proposition 6.9, Proposition 6.11, and Theorem 6.4. In fact, the technical core of Lemma 6.15 was already observed by [23].

Observation 6.16 (Implicit in Claim 7.9 of [23]). *Let $m \geq 3$ be odd and let $x_1, \dots, x_m \in \{0, 1\}$ be such that*

$$x_1 - x_2 + x_3 - \dots + x_m \in \{0, 1\}, \quad (14)$$

then there exists $i \in [m - 1]$ such that $x_i = x_{i+1}$.

The proof of Observation 6.16 is rather elementary, as the only Boolean strings which lack a neighboring pair which are equal are the alternating strings $101 \dots 101$ and $010 \dots 010$, but it is straightforward to verify these alternating strings can never satisfy (14) unless $m = 1$.

Proof of Lemma 6.15. We prove this statement by induction on m . The base case is trivial as both Bal_1 and Cat_1 only have the unary pattern $((x), x)$. Fix $m \in \mathbb{N}$ at least 3 and consider an arbitrary pattern $(t, y) := ((x_1, \dots, x_m), y)$ using only two symbols 0 and 1. We have that $(t, y) \in \text{Bal}_m$ if and only if $x_1 - x_2 + \dots + x_m = y$. Likewise, $(t, y) \in \text{Cat}_m$ if $x_1 x_2 \dots x_m = y$ in $\text{Cox}(\{0, 1\})$ (i.e., there is a series of cancellations resulting in y). It is clear the latter always implies the former, so $I_{\{0,1\}}(\text{Cat}_m) \subseteq I_{\{0,1\}}(\text{Bal}_m)$.

For the other direction, if $(t, y) \in \text{Bal}_m$, by Observation 6.16, we have that there exists $i \in [m - 1]$ such that $x_i = x_{i+1}$. Thus, $x_1 - x_2 + \dots \pm x_{i-1} \mp x_{i+2} \pm \dots + x_m = y$, so $((x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m), y) \in \text{Bal}_{m-2}$. By the induction hypothesis we have that $((x_1, \dots, x_{i-1}, x_{i+2}, \dots, x_m), y) \in \text{Cat}_{m-2}$. Thus, since x_i and x_{i+1} cancel, we have that $((x_1, \dots, x_m), y) \in \text{Cat}_m$, as desired. $\square$

Remark 6.17. As an attempt toward proving Theorem 6.14, [23] considered the following function similar to the arity-5 Catalan polymorphism:

$$f(x_1, x_2, x_3, x_4, x_5) = \varphi(x_1, \varphi(x_2, x_3, \varphi(x_1, x_2, x_3))), \varphi(x_5, x_4, \varphi(x_3, x_2, x_1))).$$

One can verify that f satisfies the Boolean interpretation of Cat_5 , but it fails to interpret Cat_5 over larger domains. In particular assuming $x_3 = x_4$ does not simplify f . This relative lack of structure in f perhaps made it difficult to find a more general polymorphisms.

Of note, this function f was independently found (except for the final x_1 and x_3 being swapped) by [9] to prove that the non-Boolean predicate $P := \{111, 222, 012, 120, 201\}$ lacks a Mal'tsev embedding. We explore such arguments in more detail in Section 6.4.

6.3 A Non-Abelian Group Embedding

Given that Theorem 6.14 (or more precisely Lemma 6.15) shows that the Catalan identities imply Abelian structure for Boolean predicates, one may naturally ask if such an argument extends to

non-Boolean domains. If true, such a result could lead to an elegant boundary between predicates with linear non-redundancy and those with superlinear non-redundancy. Sadly, such a world does not exist.

Theorem 6.18. *There exists an explicit relation $R \subseteq D^6$ with $|D| = 3$ for which R has a group embedding but lacks an Abelian embedding.*

We motivate the construction from the perspective of pattern polymorphisms. If one were to generalize Lemma 6.15 to domain size three, one would hope that for all odd $m \in \mathbb{N}$,

$$I_{\{0,1,2\}}(\text{AS}_m) = I_{\{0,1,2\}}(\text{Cat}_m).$$

It turns out this is not even true for $m = 5$. Up to isomorphism, the identities defining Cat_5 are

$$\text{Cat}_5 = \{ \begin{array}{l} ((x, x, y, y, z), z), \\ ((x, x, z, y, y), z), \\ ((z, x, x, y, y), z), \\ ((x, y, y, x, z), z), \\ ((z, x, y, y, x), z) \end{array} \}$$

However, AS_5 has these five identities along with $((x, y, z, x, y), z)$ —notice the two x 's and the two y 's have opposite signs in an alternating sum. We can immediately lift this mismatch in the identities into an explicit predicate. Consider the domain $D = \{x, y, z\}$ and the relation $\text{PAULI} \subset D^6$ defined by

$$\text{PAULI} := \begin{pmatrix} x & x & y & y & z \\ x & x & z & y & y \\ z & x & x & y & y \\ x & y & y & x & z \\ z & x & y & y & x \\ x & y & z & x & y \end{pmatrix},$$

where the tuples in the relation are the columns of the matrix. Because of the aforementioned list of identities, one can verify that there exists $p \in I_D(\text{AS}_5)$ such that applying p to rows of above matrix gives $(z, z, z, z, z, z) \notin \text{PAULI}$. More transparently, any Abelian group containing R must also contain (z, z, z, z, z, z) . Thus, R cannot be embedded into any Abelian group. However, the same argument fails with $I_D(\text{Cat}_5)$ as $p(x, y, z, x, y)$ cannot be simplified using Catalan identities. In fact, as we shall see PAULI embeds into a non-Abelian group, specifically the Pauli group.

The Pauli Group. Consider the matrices

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -\sqrt{-1} \\ \sqrt{-1} & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The matrices X, Y, Z generate a group known as the *Pauli group* P_1 with 16 elements. In particular, every element of P_1 is of the form $(-1)^a X^b Y^c Z^d$ for $a, b, c, d \in \{0, 1\}$. We also have the following identities

$$X^2 = Y^2 = Z^2 = I, \quad XY = -YX, \quad XZ = -ZX, \quad YZ = -ZY.$$

Now, we claim that the map which send x, y, z to X, Y, Z is a group embedding of PAULI. In particular define $g_1, \dots, g_5 \in P_1^6$ as the embedding of the tuples defining PAULI.

$$\begin{aligned} g_1 &= (X, X, Z, X, Z, X) \\ g_2 &= (X, X, X, Y, X, Y) \\ g_3 &= (Y, Z, X, Y, Y, Z) \\ g_4 &= (Y, Y, Y, X, Y, X) \\ g_5 &= (Z, Y, Y, Z, X, Y). \end{aligned}$$

Let C be the minimal coset of P_1^6 generated by $g_1, \dots, g_5$. That is,

$$\begin{aligned} C &= \{g_{i_1}g_{i_2}^{-1}g_{i_3} \cdots g_{i_{m-1}}^{-1}g_{i_m} : i_1, \dots, i_m \in [5]^m, m \text{ odd}\} \\ &= \{g_{i_1}g_{i_2}g_{i_3} \cdots g_{i_{m-1}}g_{i_m} : i_1, \dots, i_m \in [5]^m, m \text{ odd}\}, \end{aligned}$$

where the latter equality uses the fact that $g_i^2 = (I, I, I, I, I, I)$ for all $I \in [5]$.

We next describe the elements of this coset. Let $Q \subseteq P_1^6$ be the subgroup of P_1^6 whose elements of the form

$$Q := \{((-1)^{a_1}I, \dots, (-1)^{a_6}I) : a_1 + \dots + a_6 \equiv 0 \pmod{2}\}.$$

If we think of each of $g_1, \dots, g_5$ as length 6 vectors over the alphabet $\{X, Y, Z\}$, then each pair of generators has even Hamming distance. Thus, by the (anti-)commutativity rules of X, Y, Z we have that for all $i, j \in [5]$ there exists $q_{i,j} \in Q$ such that

$$g_i g_j = q_{i,j} g_j g_i.$$

Thus, by a suitable induction, we can prove that every element of $c \in C$ has the form

$$c = q g_1^{a_1} \cdots g_5^{a_5},$$

where $a_1, \dots, a_5 \in \{0, 1\}$ and $a_1 + \dots + a_5 \equiv 1 \pmod{2}$. There are 16 such products of the form $g_1^{a_1} \cdots g_5^{a_5}$ whose sum of exponents is odd. If $a_1 + \dots + a_5 = 1$, then we just get $g_1, \dots, g_5$. If $a_1 + \dots + a_5 = 3$, one can verify that there always exists $i \in [6]$ such that $(g^{a_1} g^{a_2} g^{a_3} g^{a_4} g^{a_5})_i = \pm XYZ$. Finally, we can compute that

$$g_1 g_2 g_3 g_4 g_5 = (Z, Z, Z, Z, Z, -Z).$$

Thus, any $c \in C \cap \{X, Y, Z\}^6$ must be of the form $q g_1^{a_1} \cdots g_5^{a_5}$ for some $q \in Q$ and $a_1 + \dots + a_5 \in \{1, 5\}$. If $a_1 + \dots + a_5 = 1$, then these products are just $c \in \{g_1, \dots, g_5\}$. However, if $a_1 + \dots + a_5 = 5$, observe that $g_1 g_2 g_3 g_4 g_5$ has an odd number of negative signs, but every $q \in Q$ has an even number of negative signs. Thus, $(Z, Z, Z, Z, Z, Z) \notin C$. Therefore,

$$C \cap \{X, Y, Z\}^6 = \{g_1, g_2, g_3, g_4, g_5\}.$$

This shows that PAULI embeds into P_1 , so we have proved Theorem 6.18.

Remark 6.19. Analogous to Remark 6.13, we are actually embedding into the index-2 coset of P_1 generated by $\{X, Y, Z\}$. This coset is isomorphic to the dihedral group D_4 (also called D_8) on 8 elements.

Remark 6.20. Recent breakthroughs by Lichter–Pago [69] and Zhuk [82] show analogous results in the context of CSP satisfiability algorithms. In particular, they give examples of CSPs tractable due to non-Abelian group equations but failed to be solved by Affine Integer Programming (AIP) [13, 3] and even stronger variants (e.g., [15, 26]). Interestingly, D_4 is also the group which Zhuk bases his counterexample on, but otherwise we (the authors) are unaware of any direct technical connection between these algorithmic results and Theorem 6.18.

Remark 6.21. Another fgpp-equivalent representation of PAULI is

$$R = \{(1, 1, 1, 1, 1, 1), (2, 2, 2, 2, 2, 2), (0, 1, 2, 1, 0, 2), (2, 0, 1, 2, 1, 0), (1, 2, 0, 0, 2, 1)\}$$

with $(0, 0, 0, 0, 0, 0)$ present in an Abelian embedding of R . Note that R can be viewed as two copies of $\text{BCK} := \{(1, 1, 1), (2, 2, 2), (0, 1, 2), (2, 0, 1), (1, 2, 0)\}$ (see [9, 14]) stuck together except the permutations (such as $(0, 1, 2)$ and $(1, 0, 2)$) have opposite sign. If you project out any coordinate of R , the result fails to have a Mal'tsev embedding (by applying Cat_5 to some permutation of the tuples).

6.4 Application: Excluding Mal'tsev Embeddings

Based on the current understanding of which predicates have (near) linear non-redundancy, an important tool for identifying which predicates may have nontrivial non-redundancy lower bounds is proving there does not exist an embedding into an infinite Mal'tsev domain. There are a few such arguments in the literature [9, 14, 23, 61] but they are rather ad-hoc based on the particular predicate involved. In this section, we give a few examples of how Catalan identities can greatly streamline such arguments.

For instance, the predicate $\text{BCK} := \{111, 222, 012, 120, 201\}$ which first appeared in [9] (see Section 7.1 in [14] for a more extensive discussion) was excluded from having a Mal'tsev embedding by using the operator f from Remark 6.17. However, using the Catalan pattern polymorphism Cat_5 , we can make the same deduction in a more principled manner. In particular,

$$\text{Cat}_5 \begin{pmatrix} 0 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 0 \\ 2 & 2 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \notin \text{BCK},$$

where Cat_5 is applied to the rows of the matrix. Therefore, BCK lacks any Mal'tsev embedding. We now present some more complex examples.

6.4.1 Proof of Theorem 4.17

With the Catalan machinery we have developed, we can now prove Theorem 4.17 with a relatively simple combinatorial argument. First, recall for odd $m \geq 3$, we defined

$$\text{CYC}_m^* := \{(x, y, z) \in (\mathbb{Z}/m\mathbb{Z})^3 : x + y + z = 0 \wedge y - x \in \{0, 1\}\} \setminus \{(0, 0, 0)\}.$$

We seek to prove that by applying Cat_{2m-1} to a suitable permutation of the $2m - 1$ tuples of CYC_m^* we can deduce the tuple $(0, 0, 0) \notin \text{CYC}_m^*$, proving that CYC_m^* lacks a Mal'tsev embedding by Theorem 6.4. In particular, we claim the following.

$$\text{Cat}_{2m-1} \begin{pmatrix} 0 & 1 & m-1 & 2 & m-2 & 3 & m-3 & \cdots & m-1 & 1 \\ 1 & 1 & 0 & 2 & m-1 & 3 & m-2 & \cdots & m-1 & 2 \\ m-1 & m-2 & 1 & m-4 & 3 & m-6 & 5 & \cdots & 2 & m-3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

More precisely, we build a matrix $M \in (\mathbb{Z}/m\mathbb{Z})^{3 \times (2m-1)}$ such that for each $i \in [2m-1]$, the i th column of M is equal to (modulo m)

- $(-j, -j+1, 2j-1)$ if $i = 2j+1$ for $j \in \mathbb{Z}$, or
- $(j, j, -2j)$ if $i = 2j$ for $j \in \mathbb{Z}$.

As a concrete example, when $m = 7$, we can verify by inspection that

$$\text{Cat}_{13} \begin{pmatrix} 0 & 1 & 6 & 2 & 5 & 3 & 4 & 4 & 3 & 5 & 2 & 6 & 1 \\ 1 & 1 & 0 & 2 & 6 & 3 & 5 & 4 & 4 & 5 & 3 & 6 & 2 \\ 6 & 5 & 1 & 3 & 3 & 1 & 5 & 6 & 0 & 4 & 2 & 2 & 4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

To finish the proof, it suffices to show for each row of M a series of cancellations which leave a single 0 remaining.

First row. Note that $M_{1,m} = M_{1,m+1} = \frac{m+1}{2}$. Likewise, for all $i \in [m-1]$, we have that $M_{1,m+1-i} = M_{1,m+i}$.⁷ After applying these nested cancellations, we are just left with $M_{1,1} = 0$.

Second row. Since $M_{2,1} = M_{2,2} = 1$, those entries cancel. For $i \in [m-2]$, we can verify that $M_{2,m+2-i} = M_{2,m+1+i}$. After applying these nested cancellations, we are left with $M_{2,3} = 0$.

Third row. Note that $M_{3,m+2} = 0$, so we need to demonstrate two series of cancellations⁸ to the left and right of this entry. In particular, for all $i \in [(m+1)/2]$, we have that $M_{3,i} = M_{3,m+2-i}$, so all entries to the left cancel out. Likewise, for all $i \in [(m-3)/2]$, we have that $M_{3,m+2+i} = M_{3,2m-i}$, so all the entries to the right cancel out.

This completes the proof of Theorem 4.17.

7 The Algebraic Structure of Patterns and Pattern Powers

In this section, we delve deeper into the algebraic framework investigations initiated in Section 5 and the structural restrictions associated with polymorphism patterns. As a first step, we give the proof of Theorem 7.9 and its multisorted generalization in Corollary 7.12 (announced already in Section 5), making the connection between fgppp-definitions and (multisorted or regular) polymorphism patterns. However, we also push further, in two directions. First, generalizing the constructions in Section 3 (the *all rational exponents* results), we study a notion of *c-tupled fgppp-definitions* (*c*-fgppp-definitions) where we allow arbitrary *c*-ary maps from tuples of variables in one domain to variables in another domain. We establish a Galois connection for this notion of expressive power to a novel *power* operation on a polymorphism pattern, and prove that *c*-fgppp-definitions behave as expected with respect to conditional non-redundancy; see Section 7.1 and 7.2. We pay special attention to the powers of the *k-cube* (or *k-universal*) partial polymorphism patterns, which regulate precisely when a lower bound of $\text{NRD}(R, n) = \Omega(n^{p/q})$, $p/q > 1$, can be shown by *c*-fgppp-reduction from OR_k ; see Section 7.3. The case when $c = k-1$ is particularly interesting since they, in a certain precise sense, are the weakest patterns that exclude linear NRD. We relate

⁷This can be proved with a case split. If $m+1-i$ is odd, then $M_{1,m+1-i} \equiv -\frac{m-i}{2} \equiv \frac{m+i}{2} \equiv M_{1,m+i} \pmod{m}$ because $m+i$ is even. Likewise, if $m+1-i$ is even, then $M_{1,m+1-i} \equiv \frac{m+1-i}{2} \equiv -\frac{m+i-1}{2} \equiv M_{1,m+i} \pmod{m}$ because $m+i$ is odd. Analogous assertions in this proof are justified in a similar manner.

⁸Note that if m is even, there is an odd number of terms both to the left and right of 0, so the cancellation is not possible in this case. This is why we assume m is odd.

them to Mal'tsev embeddings and give an explicit construction showing that any language with a Mal'tsev embedding is invariant under these cube powers.

Second, generalizing the graph girth connection from Section 4, we study multisorted polymorphism patterns from a more abstract perspective, and show that for any finite set of multisorted polymorphism patterns Q , the maximum conditional non-redundancy of promise relations invariant under Q corresponds to a multipartite hypergraph Turán problem; see Section 7.4. This generalizes results of Carbonnel [21]. This class of problems involves some famously difficult questions. For example, the maximum value of $\text{NRD}(R \mid S, n)$ for ternary promise relations $(R, S) \in \text{Inv}(Q)$ corresponds to the *Erdős box problem*; see Section 7.5 for this and other examples.

As an application, we show that promise relations preserved by the k -edge pattern polymorphism have non-redundancy $O(n^{r - \lfloor r/k \rfloor / 2})$, where r is the arity of the relation. These are generalizations of Mal'tsev polymorphisms, with similar consequences; see Section 7.5.

7.1 FGPP Definitions and Pattern Powers

We first consider the relational counterpart to patterns, which turns out to be a common generalization of *cone-definitions* [23] and *functionally guarded pp-definitions* [21] (fgpp-definitions) in the promise setting, and its generalization to c -tupled fgppp-definitions. First, we generalize the notion of an atom $R(x_1, \dots, x_r)$ in a qfpp-formula to allow terms of the form $f_i(x_{i_1}, \dots, x_{i_m})$ (where f_i is a function symbol of arity i_m) in place of a variable x_i , and thus allow expressions of the form $R(f_1(x_{1_1}, \dots, x_{1_m}), \dots, f_r(x_{r_1}, \dots, x_{r_m}))$. If each f_i is a function $f_i: D_1^{i_m} \rightarrow D_2$ for two sets D_1, D_2 , and $R_2 \subseteq D_2^r$, then $R(f_1(x_{1_1}, \dots, x_{1_m}), \dots, f_r(x_{r_1}, \dots, x_{r_m}))$ is interpreted in the obvious way, i.e., an assignment h from the variables of the atom to D_1 satisfies the atom if

$$(f_1(h(x_{1_1}), \dots, h(x_{1_m})), \dots, f_r(h(x_{r_1}), \dots, h(x_{r_m}))) \in R.$$

Definition 7.1. Let (S, T) be an r -ary promise relation over a domain D_1 and let (Γ_1, Γ_2) be a promise constraint language over D_2 . We say that (Γ_1, Γ_2) *c-tuple fgpp-defines* (*c-fgpp-defines*) (S, T) if there are functions $g_1, \dots, g_m: D_1^c \rightarrow D_2$ where

$$S(x_1, \dots, x_r) \equiv \varphi^{\Gamma_1}(\ell_1, \dots, \ell_k)$$

and

$$T(x_1, \dots, x_r) \equiv \varphi^{\Gamma_2}(\ell_1, \dots, \ell_k)$$

where each ℓ_i is a term of the form $g_i(x_{i_1}, \dots, x_{i_c})$ for $g_i \in \{g_1, \dots, g_m\}$, and where $\varphi(\ell_1, \dots, \ell_k)^{\Gamma_1}$ is a qfpp-formula over Γ_1 , and $\varphi(\ell_1, \dots, \ell_k)^{\Gamma_2}$ the corresponding qfpp-formula over Γ_2 .

If $c = 1$ then we say fgpp-definition rather than 1-fgpp-definition. We remark that if $D_1 = D_2$ then by letting $g_i = \pi_1^1$ be the identity function we may effectively also allow variables from $x_1, \dots, x_r$ to directly appear in an fgpp-definition. Note that nothing in Definition 7.1 prevents multisorted relations: if (P_1, Q_1) is a multisorted promise relation over $D_1 \times \dots \times D_r$ and (P_2, Q_2) a multisorted promise relation over $E_1 \times \dots \times E_s$, then a c -tupled fgpp-definition of (P_1, Q_1) over (P_2, Q_2) are allowed to use functions from D_i^c to E_j for $i \in [r], j \in [s]$. However, in the multisorted setting we are primarily interested in the case when $c = 1$ due to the strong connection to hypergraph Turán bounds in Section 7.4.

Similar to qfppp-definitions we introduce the stronger notion of an *c-fgppp-definition* as a sequence of strict relaxations and *c-fgppp-definitions*. Let us now continue by relating fgppp-definitions to polymorphism patterns. To accomplish this we from P define a new pattern P^c which under

certain assumptions is guaranteed to commute with P . Hence, let P be a pattern and for $c \geq 1$ we define a new pattern P^c via c -tuples of patterns from P . More precisely, assume P has arity n and consists of m patterns $P_i = (X_i, y_i)$ where P_i is over a set of variables V_i . Let $S \in \binom{[m]}{c}$ with $S = \{i_1, \dots, i_c\}$ where $i_1 < \dots < i_c$. For each such S , P^c contains a pattern $P_S = (X_S, y_S)$ over variables $V_S = V_{i_1} \times \dots \times V_{i_c}$, where for each $p \in [n]$ we have $X_S[p] = (X_{i_1}[p], \dots, X_{i_c}[p])$ and $y_S = (y_{i_1}, \dots, y_{i_c})$. Here, we treat each distinct tuple of variables as a distinct variable for notational convenience (e.g., instead of a tuple $(x_{a_1}, \dots, x_{a_c})$ we could use a single variable name $x_{a_1, \dots, a_c}$ but that introduces unreasonable overhead in the proofs to “extract” parts of the variable index). We call the pattern P^c the c th power of P and the basic question is now when a given promise language is invariant under P^c for some $c \geq 1$.

Definition 7.2. Let (S, T) be a promise language over D and let P be a polymorphism pattern. For $c \geq 1$ we say that P c -preserves (S, T) if (1) P^c is consistent, (2) $|I_D(P^c)| = 1$, and (3) and $P^c \in \text{Pattern}(S, T)$.

We remark that while $|I_D(P)| > 1$ can sometimes be useful for patterns (for the multisorted application in Section 7.4) it does not seem to be useful for powers of patterns since the power is then trivial. We illustrate these complications with an example.

Example 7.3. A *majority* polymorphism is a ternary polymorphism m that satisfies $m(x, x, y) = m(x, y, x) = m(y, x, x) = x$ for all x . As a total polymorphism, it guarantees that a language is *2-decomposable*, i.e., all relations in the language can be written as intersections of binary constraints (and hence the non-redundancy is at most quadratic). Let $P = \{((x, x, y), x), ((x, y, x), x), ((y, x, x), x)\}$ be the corresponding polymorphism pattern. Since $I_D(P)$ is partial (if D is not Boolean) the presence of P does not imply that a relation is 2-decomposable. However, the absence of P guarantees that a relation is not 2-decomposable. The square of P is the pattern

$$P^2 = \{(((x, x), (x, y), (y, x)), (x, x)), (((x, y), (x, x), (y, x)), (x, x)), (((x, y), (y, x), (x, x)), (x, x))\}$$

and if we rename variables we obtain

$$\{((x, y, z), x), ((x, y, z), y), ((x, y, z), z)\}.$$

Hence, P^2 is inconsistent. This agrees well with the fact that the relation $a + b + c + d = 1$ is 2-fgpp-definable over the equality relation, e.g., using maps $g_1(a, b) = a + b$ and $g_2(c, d) = 1 - c - d$. Note also that P^3 (after renaming variables) yields only the identity $((x, y, z), v)$. Thus, $I_D(P^3)$ contains every total, ternary function over D , and P^3 is thus trivial.

On the other hand, let

$$P = \{((x, x, x, x, y), x), \dots, ((y, x, x, x, x), x)\}$$

be the *near-unanimity* pattern of arity 5 (5-NU). Similarly as above, the absence of P as a pattern polymorphism guarantees that a relation is not 4-decomposable. Then P^2 is the polymorphism pattern that defines the 5-ary partial threshold operation $p(x_1, \dots, x_5)$ whose output is the majority value if at least three of its arguments take the same value, otherwise p is undefined. Thus by Lemma 7.13, any relation not preserved by p can 2-fgpp-define a relation R that is not 4-decomposable.

There is a strong similarity between pattern powers and the standard power operation in universal algebra even though the two operations are not equivalent.

Remark 7.4. Recall that the well-known algebraic c -th *power* operation of an n -ary function f over a domain D produces the function $\text{pow}(f, c)$ on D^c which for any sequence $t^{(1)}, \dots, t^{(n)} \in D^c$ is defined as $f(t^{(1)}, \dots, t^{(n)}) = (f(t_1^{(1)}, \dots, t_1^{(n)}), \dots, f(t_c^{(1)}, \dots, t_c^{(n)}))$. For a partial function p we can similarly define its power $\text{pow}(p, c)$ to be defined if and only if each application of p is defined. Then, if $p \in I_D(P)$ for a pattern P , one can verify that for any $c \geq 1$ $\text{pow}(p, c)$ is a *superfunction* of some function in $I_{D^c}(P)$ in the sense that it satisfies P over D^c but might be defined for some additional tuples. However, perhaps contrary to intuition, $\text{pow}(p, c) \in I_{D^c}(P^c)$ does in general *not* hold. To see this, consider e.g. the identities defining the ternary *cube* (also called *3-universal* in Lagerkvist and Wahlström [62])

$$U_3 = \left\{ \begin{array}{l} ((x, x, x, y, y, y), x), \\ ((x, y, y, x, x, y), x), \\ ((y, x, y, x, y, x), x) \end{array} \right\}.$$

It is easy to verify that $I_{\{0,1\}}(U_3) = \{u_3\}$ for a uniquely defined partial operation u_3 and that $\text{pow}(u_3, 2)$ satisfies U_3 over $\{0, 1\}^2$. However, if we e.g. consider U_3^2 then we obtain a system of identities equivalent to

$$\left\{ \begin{array}{l} (((x, x), (x, y), (x, y), (y, x), (y, x), (y, y), (y, y)), (x, x)), \\ (((x, y), (x, x), (x, y), (y, x), (y, y), (y, x), (y, y)), (x, x)), \\ (((x, y), (y, x), (y, y), (x, x), (x, y), (y, x), (y, y)), (x, x)) \end{array} \right\}$$

over variables $\{(x, x), (x, y), (y, x), (y, y)\}$, and we see that $\text{pow}(u_3, 2)$ does not satisfy these identities over $\{0, 1\}^2$ since it e.g. is undefined for the instantiation $\alpha((x, x)) = (0, 1), \alpha((x, y)) = (0, 0), \alpha((y, x)) = (1, 0), \alpha((y, y)) = (1, 1)$ since

$$\begin{aligned} & (\alpha((x, x)), \alpha((x, y)), \alpha((x, y)), \alpha((y, x)), \alpha((y, x)), \alpha((y, y)), \alpha((y, y))) \\ &= ((0, 1), (0, 0), (0, 0), (1, 0), (1, 0), (1, 1), (1, 1)) \end{aligned}$$

but $u_3^2((0, 1), (0, 0), (0, 0), (1, 0), (1, 0), (1, 1), (1, 1))$ is undefined.

We now prove that P^c and P commute in the following sense.

Definition 7.5. Let f_1 and f_2 be n -ary partial functions over D_1 and D_2 , respectively. We say that f_1, f_2 *commute* with a c -ary $g: D_1^c \rightarrow D_2$ if

$$f_2(g(x_1^1, \dots, x_1^c), \dots, g(x_n^1, \dots, x_n^c)) = g(f_1(x_1^1, \dots, x_n^1), \dots, f_1(x_1^c, \dots, x_n^c))$$

and is defined for all $(x_1^1, \dots, x_n^1), \dots, (x_1^c, \dots, x_n^c) \in \text{dom}(f_1)$.

Proposition 7.6. Let P be an n -ary polymorphism pattern, let $c \geq 1$ such that P^c is consistent, let D_1 and D_2 be finite sets. For every $f_1 \in I_{D_1}(P)$ there exists $f_2 \in I_{D_2}(P^c)$ such that f_1, f_2 commute with every c -ary $g: D_1^c \rightarrow D_2$.

Proof. Let $P_1, \dots, P_m$ be an enumeration of the elements in P and let $f_1 \in I_{D_1}(P)$. Assume that $(x_1^1, \dots, x_n^1), \dots, (x_1^c, \dots, x_n^c) \in \text{dom}(f_1)$, and let $P_{i_1}, \dots, P_{i_c}$ be witnessing patterns (where there may be multiple matching patterns for each P_{i_j} , but the value of $f_1(x_1^j, \dots, x_n^j)$ is invariant under the choice of pattern, since otherwise P would be inconsistent and we could not choose $f_1 \in I_{D_1}(P)$). This corresponds to the pattern $P' = P_{\{i_1, \dots, i_c\}} \in P^c$; note that for every function $g: D_1^c \rightarrow D_2$ the tuple $(g(x_1^1, \dots, x_n^1), \dots, g(x_n^1, \dots, x_n^c)) \in D_2^n$ matches the pattern P' . Now, either $P' = (X, y)$ is a pattern where y does not occur in X , in which case we can choose $f_2 \in I_{D_2}(P^c)$ so

that the criterion is satisfied, or otherwise $f_2(g(x_1^1, \dots, x_c^1), \dots, g(x_1^n, \dots, x_c^n))$ is an instantiation of $P_{\{i_1, \dots, i_c\}}$ and

$$g(f_1(x_1^1, \dots, x_n^1), \dots, f_1(x_1^n, \dots, x_n^n)) = f_2(g(x_1^1, \dots, x_c^1), \dots, g(x_1^n, \dots, x_c^n))$$

by a similar reasoning to Proposition 5.22. $\square$

Proposition 7.7. *Let P be an n -ary polymorphism pattern and let $c \geq 1$. Let (S_1, T_1) and (S_2, T_2) be promise languages over D_1 and D_2 . If (S_1, T_1) is c -fgppp-definable over (S_2, T_2) , and (S_2, T_2) is c -preserved by P , then (S_1, T_1) is preserved by P .*

Proof. We only consider the case when (S_1, T_1) is c -fgpp-definable since relaxations follow via arguments completely analogous to those in Lemma 5.6. Thus, let

$$S_1(x_1, \dots, x_r) \equiv \varphi^{S_2}(g_1(x_{1_1}, \dots, x_{1_c}), \dots, g_k(x_{k_1}, \dots, x_{k_c}))$$

and

$$T_1(x_1, \dots, x_r) \equiv \varphi^{T_2}(g_1(x_{1_1}, \dots, x_{1_c}), \dots, g_k(x_{k_1}, \dots, x_{k_c}))$$

for guarding function maps $g_i: D_1^c \rightarrow D_2$. Let

$$S'_1(y_1, \dots, y_k) \equiv \varphi^{S_2}(y_1, \dots, y_k)$$

and

$$T'_1(y_1, \dots, y_k) \equiv \varphi^{T_2}(y_1, \dots, y_k).$$

Then (S'_1, T'_1) is qfpp-definable over (S_2, T_2) and is by Theorem 5.13 thus invariant under $I_{D_2}(P)$.

Assume that there exists $P_1^{D_1} \in I_{D_1}(P)$ and $t^{(1)}, \dots, t^{(n)} \in S_1$ such that $P^{D_1}(t^{(1)}, \dots, t^{(n)}) = t \notin T_1$. We observe that for each $t^{(i)}$, $1 \leq i \leq n$, we must have

$$(g_1(t_{1_1}^{(i)}, \dots, t_{1_c}^{(i)}), \dots, g_k(t_{k_1}^{(i)}, \dots, t_{k_c}^{(i)})) \in S'_1.$$

Proposition 7.6 then implies that there exists $P^{D_2} \in I_{D_2}(P)$ such that P^{D_1} and P^{D_2} commute with all c -ary guarding functions from D_1 to D_2 . This, in turn, implies that

$$\begin{aligned} & (P^{D_2}(g_1(t_{1_1}^{(1)}, \dots, t_{1_c}^{(1)}), \dots, g_1(t_{1_1}^{(n)}, \dots, t_{1_c}^{(n)})), \dots, P^{D_2}(g_k(t_{k_1}^{(1)}, \dots, t_{k_c}^{(1)}), \dots, g_k(t_{k_1}^{(n)}, \dots, t_{k_c}^{(n)}))) \\ &= (g_1(P^{D_1}(t_{1_1}^{(1)}, \dots, t_{1_c}^{(1)}), \dots, P^{D_1}(t_{1_1}^{(n)}, \dots, t_{1_c}^{(n)})), \dots, g_k(P^{D_1}(t_{k_1}^{(1)}, \dots, t_{k_c}^{(1)}), \dots, P^{D_1}(t_{k_1}^{(n)}, \dots, t_{k_c}^{(n)}))) \\ &= (g_1(t_{1_1}, \dots, t_{1_c}), \dots, g_k(t_{k_1}, \dots, t_{k_c})) \notin T'_1. \end{aligned} \tag{15}$$

This contradicts the assumption that P^{D_2} preserves (S'_1, T'_1) . $\square$

Next, we prove a converse result which shows that we can c -fgppp-define (S_1, T_1) if it is invariant under P whenever (S_2, T_2) is invariant under P^c . For simplicity we present it for individual relations but the generalization to promise constraint languages is effortless.

Lemma 7.8. *Let (S_1, T_1) , (S_2, T_2) be promise relations over D_1 , D_2 . If $P \in \text{Pattern}(S_1, T_1)$ whenever P c -preserves (S_2, T_2) then (S_2, T_2) c -fgppp-defines (S_1, T_1) .*

Proof. Assume that $P \in \text{Pattern}(S_1, T_1)$ whenever $P^c \in \text{Pattern}(S_2, T_2)$, for a consistent pattern P^c . We begin by introducing some notation.

1. Let $|S_1| = m$ and let $t_1, \dots, t_m \in S_1$ be an enumeration of the tuples in S_1 .
2. Let $r_1 \geq 1$ be the arity of S_1 and $r_2 \geq 1$ the arity of S_2 . For each $1 \leq i \leq r_1$ we let $c^i = (t_i^{(1)}, \dots, t_i^{(m)})$ be the i th column of S_1 under the enumeration $t_1, \dots, t_m$. We assume without loss of generality that $|\{c^1, \dots, c^{r_1}\}| = r_1$ (all columns are distinct) since we can otherwise qfppp-define relations of lower arity by identifying arguments. These relations can, in turn, qfppp-define the original relation by equality constraints.

Consider the r_1 -ary relations S'_1 and T'_1 defined via c -fgpp-formulas $\varphi^{S_2}(x_1, \dots, x_{r_1})$ and $\varphi^{T_2}(x_1, \dots, x_{r_1})$ over variables $x_1, \dots, x_{r_1}$ where we for every sequence $s^{(1)}, \dots, s^{(m)} \in S_2$, with columns

$$d^1 = (s_1^{(1)}, \dots, s_1^{(m)}), \dots, d^{r_2} = (s_{r_2}^{(1)}, \dots, s_{r_2}^{(m)})$$

add

1. $S_2(g_1(x_{h(1,1)}, \dots, x_{h(1,c)}), \dots, g_{r_2}(x_{h(r_2,1)}, \dots, x_{h(r_2,c)}))$ to φ^{S_2} , and
2. $T_2(g_1(x_{h(1,1)}, \dots, x_{h(1,c)}), \dots, g_{r_2}(x_{h(r_2,1)}, \dots, x_{h(r_2,c)}))$ to φ^{T_2} ,

if there exist $g_1, \dots, g_{r_2}: D_1^c \rightarrow D_2$ and $h: [r_2] \times [c] \rightarrow [r_1]$ such that $g_1(c^{h(1,1)}, \dots, c^{h(1,c)}) = d^1, \dots, g_{r_2}(c^{h(r_2,1)}, \dots, c^{h(r_2,c)}) = d^{r_2}$.

Similarly, we add the equality constraint

$$g_1(x_{h(1,1)}, \dots, x_{h(1,c)}) = g_2(x_{h(2,1)}, \dots, x_{h(2,c)})$$

to φ^{S_2} and φ^{T_2} if there are $g_1, g_2: D_1^c \rightarrow D_2$ and $h: [2] \times [c] \rightarrow [r_1]$ such that $g_1(c^{h(1,1)}, \dots, c^{h(1,c)}) = g_2(c^{h(2,1)}, \dots, c^{h(2,c)})$. The equality constraints are necessary to make sure that the resulting pattern can be defined in a consistent way.

We then view $c^1, \dots, c^{r_1}$ as an incomplete pattern P' by viewing D_1 as a set of variables V and each c^i as a tuple of variables. Say that a pattern P over V is an *instantiation* of P' if $((x_1, \dots, x_m), y) \in P$ for some $y \in V$ if and only if $(x_1, \dots, x_m) \in P'$. Then the following statements are true.

1. If $t \in S'_1$ then there is an instantiation P of P' such that P^c is consistent and $P^c \in \text{Pattern}(S_2, S_2)$ and such that t (viewed as an m -ary function) is a subfunction of a partial function in $I_{D_1}(P)$. To see this, we construct the pattern P from P' by letting it contain $(c^i, t[i])$ for each $1 \leq i \leq r_1$. We first claim that P does not contain any pattern (c^i, c) , for $c \notin \{c^1, \dots, c^{r_1}\}$. To see this, consider maps h_c, h'_c defined such that $h_c(c^i) = h'_c(c^i) = c^i$ but such that $h_c(c) \neq h'_c(c)$. Then the above construction will contain $h_c(x_i) = h'_c(x_i)$ (or, to be precise, c -ary maps defined such that $h_c(x_i, \dots, x_i) = h'_c(x_i, \dots, x_i)$) and will not be satisfied by t . Similarly, assume that P is inconsistent, i.e., that there are $(c^i, a), (c^j, b) \in P$ and maps $h, h': D_1 \rightarrow D_1$ such that $h(c^i) = h'(c^j)$ but such that $h(a) \neq h(b)$. It again follows that the above construction contains an equality constraint that forbids this possibility. Entirely analogous arguments can be used for P^c to show that it must be consistent.
2. If $t \in T'_1$ then there is an instantiation P of P' where P^c is consistent and $P^c \in \text{Pattern}(S_2, T_2)$ such that t (viewed as an m -ary function) is a subfunction of a partial function in $I_{D_1}(P)$. The arguments are entirely analogous to the case above.

Hence, if there exists $f \in T'_1$ where $f \notin T_1$ then there exists m -ary $P^c \in \text{Pattern}(S_2, T_2)$ such that f is a subfunction of a partial function in $I_{D_1}(P)$ but where $f \notin \text{pPol}(S_1, T_1)$.

We conclude that $T'_1 = T_1, S_1 \subseteq S'_1$, and that (S_1, T_1) is fgppp-definable via a strict relaxation of (S'_1, T'_1) . $\square$

For $c = 1$ we obtain the following precise characterization which generalizes Carbonnel [21] in the promise setting.

Theorem 7.9. *Let (S_1, T_1) be a promise relation over D_1 and (S_2, T_2) a promise relation over D_2 . Then $\text{Pattern}(S_2, T_2) \subseteq \text{Pattern}(S_1, T_1)$ if and only if (S_2, T_2) fgppp-defines (S_1, T_1) .*

For $c \geq 1$ we for a promise relation (S, T) let

$$\text{Pattern}^c(S, T) = \{P \mid P \text{ is a pattern, } P^c \text{ preserves } (S, T)\}.$$

Hence, $\text{Pattern}^c(S, T)$ is the set of patterns P where P^c is consistent and preserves (S, T) . We obtain the following.

Theorem 7.10. *Let (S_1, T_1) be a promise relation over D_1 and (S_2, T_2) a promise relation over D_2 , and let $c \geq 1$. Then $\text{Pattern}^c(S_2, T_2) \subseteq \text{Pattern}(S_1, T_1)$ if and only if (S_2, T_2) c-fgppp-defines (S_1, T_1) .*

For multisorted patterns we concentrate on $c = 1$ and obtain the following characterization.

Corollary 7.11. *Let Q be a set of multisorted polymorphism patterns. If (S_1, T_1) over disjoint $D_1, \dots, D_r$ is fgppp-definable over (S_2, T_2) over disjoint $E_1, \dots, E_s$ which is preserved by Q , then (S_1, T_1) is preserved by Q ($(S_1, T_1) \in \text{Inv}(Q)$).*

Note that this implies that $\text{Inv}(Q)$, when Q is a set of multisorted patterns, is closed under fgppp-definitions.

Corollary 7.12. *Let (S_1, T_1) be a multisorted promise relation over disjoint $D_1, \dots, D_r$ and (S_2, T_2) a multisorted promise relation over disjoint $E_1, \dots, E_s$. Then $\text{mPattern}(S_2, T_2) \subseteq \text{mPattern}(S_1, T_1)$ if and only if (S_2, T_2) fgppp-defines (S_1, T_1) .*

Finally, the following statement is useful for constructing a canonical relation not invariant under a given pattern (power) (cf. Lemma 5.9). We state it in the non-multisorted setting but it naturally extends to this setting, too.

Lemma 7.13. *Let P be a polymorphism pattern and $c \in \mathbb{N}$ such that P^c is consistent and $|I_D(P^c)| = 1$ for every domain D . Let (S_1, T_1) be a promise relation such that $P \notin \text{Pattern}^c(S_1, T_1)$. Then (S_1, T_1) c-fgppp-defines a promise relation (S_2, T_2) such that $P \notin \text{Pattern}(S_2, T_2)$.*

Proof. Let P be an n -ary pattern and $c \in \mathbb{N}$ such that P^c is consistent and $P \notin \text{Pattern}^c(S_1, T_1)$ for a promise relation (S_1, T_1) of arity r over some domain D . That is, $P^c \notin \text{Pattern}(S_1, T_1)$. Let $p \in I_D(P^c)$ and let $t^1, \dots, t^n \in S_1$ be such that $p(t^1, \dots, t^n) = u$ is defined and $u \notin T_1$. For every $i \in [r]$, let $(X^i, y^i) \in P^c$ be a pair that matches $((t^1_i, \dots, t^n_i), u_i)$. Let $y^i = X^n_i$; this is defined since $|I_D(P^c)| = 1$. Furthermore, let (X^i, y^i) be the combination of patterns $(X^{i,1}, y^{i,1}), \dots, (X^{i,c}, y^{i,c}) \in P$. Let D_2 be the domain consisting of all variables occurring in P and for $i \in [r]$ let $g_i: D_2^c \rightarrow D$ be any function such that $g_i(X^{i,1}_j, \dots, X^{i,c}_j) = t^i_j$ for each $j \in [n]$. Consider the c -fgpp-definition $\varphi(v_{1,1}, \dots, v_{r,c}) = R(\ell_1, \dots, \ell_r)$ where $\ell_i = g_i(v_{i,1}, \dots, v_{i,c})$ for every $i \in [r]$. Let (S_2, T_2) be the

promise relation defined by φ using (S_1, T_1) . We claim that $P \notin \text{Pattern}(S_2, T_2)$. Consider the tuples $a^1, \dots, a^n \in D_2^{rc}$ where $a_{i,j}^d = X_d^{i,j}$ for every $i \in [r]$, $j \in [c]$, $d \in [n]$. Let $q \in I_{D_2}(P)$ (and note that q exists and is unique). We claim that $a^1, \dots, a^n \in S_2$ and that $q(a^1, \dots, a^n) = b$ is defined with $b \notin T_2$. Indeed, by design, for every $i \in [r]$, $j \in [c]$ the tuple $(a_{i,j}^1, \dots, a_{i,j}^n)$ matches the pair $(X^{i,j}, y^{i,j}) \in P$, so q produces an output b where $b_{i,j} = a_{i,j}^{n_i}$ for every $i \in [r]$, $j \in [c]$. Thus b under the maps g_i maps to the tuple $u \notin T_1$, so $b \notin T_2$. $\square$

7.2 Gadget Reductions for Conditional Non-redundancy

We continue by relating fgppp-definitions, and thus also patterns, to conditional non-redundancy. Our goal is to prove the following (compare with Proposition 2 in Carbonnel [21]) statement. The theorem naturally extends to finite sets of promise relations but we present and prove the results in the singleton setting to simplify the formal details.

Theorem 7.14. *Let (S_1, T_1) and (S_2, T_2) be promise relations over D_1 , respectively, D_2 . If $(S_2, \tilde{T}_2)$ c -fgppp-defines $(S_1, \tilde{T}_1)$ for some $c \geq 1$ then $\text{NRD}(S_1 \mid T_1, n) = O_{D_1, D_2, r_1, r_2}(n + \text{NRD}(S_2 \mid T_2, n^c))$.*

As an immediate corollary of Theorem 7.9, we have the following algebraic test for bounding non-redundancy.

Corollary 7.15. *Let (S_1, T_1) and (S_2, T_2) be promise relations over D_1 , respectively, D_2 . If $\text{Pattern}(S_2, \tilde{T}_2) \subseteq \text{Pattern}(S_1, \tilde{T}_1)$ then $\text{NRD}(S_1 \mid T_1, n) = O_{D_1, D_2, r_1, r_2}(n + \text{NRD}(S_2 \mid T_2, n))$.*

For pattern powers we obtain the more general statement.

Corollary 7.16. *Let (S_1, T_1) and (S_2, T_2) be promise relations over D_1 , respectively, D_2 , and let $c \geq 1$. If $\text{Pattern}^c(S_2, \tilde{T}_2) \subseteq \text{Pattern}(S_1, \tilde{T}_1)$ then $\text{NRD}(S_1 \mid T_1, n) = O_{D_1, D_2, r_1, r_2}(n + \text{NRD}(S_2 \mid T_2, n^c))$.*

Our overall proof strategy is similar to that of Carbonnel [21] but more involved due to the richer machinery. We write our proof in a modular fashion by proving Theorem 7.14 for each of the “atomic” operations involved in an fgppp-definition. In certain cases the resulting bound is thus sharper than the general bound in Theorem 7.14. We start with strict relaxations.

Proposition 7.17. *Assume $D := D_1 = D_2$ and $r := r_1 = r_2$. If $S_1 \subseteq S_2$ and $\tilde{T}_2 \subseteq \tilde{T}_1$, then $\text{NRD}(S_1 \mid T_1, n) \leq \text{NRD}(S_2 \mid T_2, n)$.*

Proof. Consider any conditionally non-redundant instance $(X, Y \subseteq X^r)$ of $\text{CSP}(S_1 \mid T_1)$ with witnessing assignments $\psi_y : X \rightarrow D$ for all $y \in Y$. If $\tilde{T}_2 \subseteq \tilde{T}_1$, then $T_1 \setminus S_1 \subseteq T_2 \setminus S_2$. Since we also have $S_1 \subseteq S_2$, the assignments $\{\psi_y : y \in Y\}$ also witness that (X, Y) is a conditionally non-redundant instance of $\text{CSP}(S_2 \mid T_2)$. $\square$

Second, we handle equality constraints.

Proposition 7.18. *Assume $D := D_1 = D_2$ and $r := r_1 = r_2$. Assume that there exists distinct $i, j \in [r]$ such that*

$$\begin{aligned} S_1(x_1, \dots, x_r) &= S_2(x_1, \dots, x_r) \wedge x_i = x_j, \\ \tilde{T}_1(x_1, \dots, x_r) &= \tilde{T}_2(x_1, \dots, x_r) \wedge x_i = x_j. \end{aligned} \tag{16}$$

Then, $\text{NRD}(S_1 \mid T_1, n) \leq \text{NRD}(S_2 \mid T_2, n) + n - 1$.

Proof. Let (X, Y) be a conditionally non-redundant instance of $\text{CSP}(S_1 \mid T_1)$, and let $\{\psi_y : X \rightarrow D \mid y \in Y\}$ be the witnessing assignments. Let $Y_{\text{EQ}} \subseteq Y$ be the set of clauses $y \in Y$ for which $\psi_y(y_i) \neq \psi_y(y_j)$. It is clear that the graph $\{(y_i, y_j) \in X^2 \mid y \in Y_{\text{EQ}}\}$ is a acyclic, so $|Y_{\text{EQ}}| \leq |X| - 1$.

By negating both sides of (16), we have that

$$(T_1 \setminus S_1)(x_1, \dots, x_r) = (T_2 \setminus S_2)(x_1, \dots, x_r) \vee x_i \neq x_j.$$

For each $y \in Y$, we have that $\psi_y(y) \in T_1 \setminus S_1$. Further, by definition of Y_{EQ} , for each $y \in Y \setminus Y_{\text{EQ}}$, we have that $\psi_y(y_i) = \psi_y(y_j)$, so $\psi_y(y) \in T_2 \setminus S_2$. Therefore, $(X, Y \setminus Y_{\text{EQ}})$ is a conditionally non-redundant instance of $\text{CSP}(S_2 \mid T_2)$. $\square$

Third, we handle the conjunction of two predicates, which immediately extends to an arbitrary number of predicates.

Proposition 7.19. *Assume $D := D_1 = D_2$. Let $f, g : [r_2] \rightarrow [r_1]$ be functions such that*

$$\begin{aligned} S_1(x_1, \dots, x_{r_1}) &= S_2(x_{g(1)}, \dots, x_{g(r_2)}) \wedge S_2(x_{f(1)}, \dots, x_{f(r_2)}), \\ \tilde{T}_1(x_1, \dots, x_{r_1}) &= \tilde{T}_2(x_{g(1)}, \dots, x_{g(r_2)}) \wedge \tilde{T}_2(x_{f(1)}, \dots, x_{f(r_2)}). \end{aligned} \quad (17)$$

Then, $\text{NRD}(S_1 \mid T_1, n) \leq \text{NRD}(S_2 \mid T_2, n)$.

Proof. Let $(X, Y \subseteq X^{r_1})$ be a conditionally non-redundant instance of $\text{CSP}(S_1 \mid T_1)$, and let $\{\psi_y : X \rightarrow D \mid y \in Y\}$ be the witnessing assignments. For each $y \in Y$, let $y_f, y_g \in X^{r_2}$ be

$$\begin{aligned} y_f &:= (y_{g(1)}, \dots, y_{g(r_2)}), \\ y_g &:= (y_{f(1)}, \dots, y_{f(r_2)}). \end{aligned}$$

For each $y, y' \in Y$, we have that $\psi_y(y') \in S_1$ if and only if $\psi_y(y'_f) \in S_2$ and $\psi_y(y'_g) \in S_2$. Furthermore, by negating both sides of (17), we can deduce that

$$(T_1 \setminus S_1)(x_1, \dots, x_{r_1}) = (T_2 \setminus S_2)(x_{g(1)}, \dots, x_{g(r_2)}) \vee (T_2 \setminus S_1)(x_{f(1)}, \dots, x_{f(r_2)}).$$

In particular, since $\psi_y(y) \in T_1 \setminus S_1$, we have that $\psi_y(y_f) \in T_2 \setminus S_2$ or $\psi_y(y_g) \in T_2 \setminus S_2$. Let $y_{\text{rep}} \in \{y_f, y_g\}$ be an arbitrary representative for which $\psi_y(y_{\text{rep}}) \in T_2 \setminus S_2$. Then define $Y_{\text{rep}} := \{y_{\text{rep}} : y \in Y\} \subseteq X^{r_2}$. From our discussion, we have that $\{\psi_y : y \in Y\}$ witness that (X, Y_{rep}) is a conditionally non-redundant instance of $\text{CSP}(S_2 \mid T_2)$. In particular, $|Y_{\text{rep}}| = |Y|$ as each ψ_y fails to S_2 -satisfy a unique clause y_{rep} of (X, Y_{rep}) . $\square$

As an immediate corollary of our progress so far, we have that conditional non-redundancy behaves well with respect to qfppp-reductions. However, we now conclude by showing that functional guarding also behaves well with respect to conditional non-redundancy.

Proposition 7.20. *For $c \geq 1$ let $h : [r_2] \times [c] \rightarrow [r_1]$ and $g_1, \dots, g_{r_2} : D_1^c \rightarrow D_2$ be functions such that*

$$S_1(x_1, \dots, x_{r_1}) = S_2(g_1(x_{h(1,1)}, \dots, x_{h(1,c)}), \dots, g_{r_2}(x_{h(r_2,1)}, \dots, x_{h(r_2,c)})) \quad (18)$$

$$\tilde{T}_1(x_1, \dots, x_{r_1}) = \tilde{T}_2(g_1(x_{h(1,1)}, \dots, x_{h(1,c)}), \dots, g_{r_2}(x_{h(r_2,1)}, \dots, x_{h(r_2,c)})). \quad (19)$$

Then, $\text{NRD}(S_1 \mid T_1, n) \leq O_{r_2}(\text{NRD}(S_2 \mid T_2, n^c))$.

Proof. Let $(X, Y \subseteq X^{r_1})$ be a conditionally non-redundant instance of $\text{CSP}(S_1 \mid T_1)$ with witnessing assignments $\{\psi_y : X \rightarrow D_2 \mid y \in Y\}$. Let $X_+ := X^c \times [r_2]$ and for each $y \in Y$, let

$$y_+ := ((y_{h(1,1)}, \dots, y_{h(1,c)}, 1), \dots, (y_{h(r_2,1)}, \dots, y_{h(r_2,c)}, r_2)) \in X_+^{r_2}.$$

Let $Y_+ := \{y_+ \mid y \in Y\}$ and for each $y_+ \in Y_+$ let $\psi_{y_+} : X_+ \rightarrow D_1$ be such that for all $(x, i) \in X_+$, we have that $\psi_{y_+}(x, i) = g_i(\psi_y(x))$, where ψ_y is applied componentwise to x and therefore yields a c -ary tuple matching the arity of g_i . From (18) and (19), we can see that $\{\psi_{y_+} \mid y_+ \in Y_+\}$ witness that (X_+, Y_+) is a conditionally non-redundant instance of $\text{CSP}(S_2 \mid T_2)$ and thus $|Y_+| = |Y|$. Since $|X_+| = r_2|X|^c$ we therefore have $\text{NRD}(S_1 \mid T_1, n) \leq \text{NRD}(S_2 \mid T_2, n^c \cdot r_2)$, and since $\text{NRD}(S_2 \mid T_2, n^c \cdot r_2) \in O_{r_2}(\text{NRD}(S_2 \mid T_2, n^c))$ (Lemma 2.4) the desired bound follows. $\square$

Note that this bound, in contrast to Carbonnel [21] who obtain a dependency on $|D|^{|D|}$, does not depend on the size of the domains.

Last, we conclude the proof of Theorem 7.14.

Proof of Theorem 7.14. If (S_2, T_2) fgppp-defines (S_1, T_1) , then this definition can be constructed using at most $O_{D_1, D_2, r_1, r_2}(1)$ iterations of the operations in Proposition 7.17, Proposition 7.18, Proposition 7.19, and Proposition 7.20. Furthermore, the arity of any intermediate predicate pair is at most $O_{D_1, D_2, r_1, r_2}(1)$, so the constant factor bound of Proposition 7.20 still applies. Last, as a special case, consider the case when (S_1, T_1) is fgppp-definable with qfgppp-definition over (S_2, T_2) which involves one or more $f = \emptyset$ constraints. Then, $S_1 = T_1 = \emptyset$ and the claim trivially follows. $\square$

7.3 The Cube-power Polymorphism Patterns

We now review a particularly important class of polymorphism patterns, corresponding to the powers of the *cube* identities mentioned in Remark 7.4, studied by Berman et al. [8]. The k -cube pattern polymorphisms were referred to as *k-universal* partial polymorphisms in previous work [21, 62].

Definition 7.21. Let $k \geq 2$. The k -cube polymorphism pattern U_k is the pattern with arity $2^k - 1$ on variable set $V = \{x, y\}$, with patterns $P_i = (X_i, y)$ for $i \in [k]$ where for each $j \in [2^k - 1]$, position j of X_i equals x if the i th bit of the binary expansion of j is 1, and y otherwise.

Since $|I_D(U_k)| = 1$ for every domain D , we let u_k refer to the unique member of $I_D(U_k)$, where the domain D will frequently be understood from context. The significance of U_k is that every relation $R \subseteq D^r$ for some domain D and arity r can fgpp-define the k -ary OR relation if and only if $u_k \notin \text{pPol}(R)$; see [21, 62]. Then Lemma 7.13 gives the following.

Lemma 7.22. Let $R \subseteq D^r$ be a relation over some domain D , and assume $u_k^c \notin \text{pPol}(R)$ for some $k > c \geq 1$. Then $\text{NRD}(R, n) = \Omega(n^{k/c})$.

Example 7.23. Consider the following relation from [14]. Define $R = 3\text{LIN}_{\mathbb{Z}/3\mathbb{Z}} = \{(x, y, z) \in \{0, 1, 2\}^3 \mid x + y + z \equiv 0 \pmod{3}\}$ and $R^* = 3\text{LIN}_{\mathbb{Z}/3\mathbb{Z}}^* = R \setminus \{(0, 0, 0)\}$. It was shown in [14] that $\text{NRD}(R^*, n) = \Omega(n^{1.5})$ and $\text{NRD}(R^*, n) = \mathcal{O}(n^{1.6} \log n)$. The lower bound now follows from Lemma 7.22 as $u_3^2((0, 1, 2), (1, 0, 2), (1, 1, 1), (1, 2, 0), (1, 0, 2), (2, 2, 2), (2, 0, 1)) = (0, 0, 0)$.

Lemma 7.22 is a powerful approach towards lower bounds on non-redundancy; e.g., beyond this example, it also covers all lower bounds of Section 3. However, it is also not a “universal” method of showing tight lower bounds on the non-redundancy, as shown at the end of this section.

In our study of relations with near-linear NRD, we will be particularly interested in the power patterns U_k^{k-1} , since these are the weakest patterns that exclude near-linear NRD. Let us make some quick observations about these patterns. The first is a direct observation that follows from the definition (we include the case $U_2^1 = U_2$ for uniformity).

Proposition 7.24. *Let $k \in \mathbb{N}$, $k \geq 2$. The polymorphism pattern U_k^{k-1} has arity $2^k - 1$, and its patterns are described as follows: For every $i \in [k]$, there is a pattern $P_i = (X_i, y_i)$ where $X_i[p] = X_i[q]$ for $p, q \in [2^k - 1]$ if and only if $p \oplus q = e_i$ when p and q are viewed as bitstrings and e_i is the i th unit vector, and where y_i is the unique variable that occurs only once in X_i .*

We continue with the following observations.

Lemma 7.25. *The following hold.*

1. *For any $k \geq 2$, U_k^{k-1} is symmetric in the following sense: Let $P_1, \dots, P_k$ be the patterns of U_k^{k-1} as in Proposition 7.24. For every permutation π on $[k]$ there is a permutation σ on $[2^k - 1]$ such that for every $i \in [k]$, the values of pattern P_i read in order σ match the values of pattern $P_{\pi(i)}$ read in the original order.*
2. *Over any domain D , u_p^{p-1} implies u_q^c for every $p \geq q \geq 2$ and every $c < q$.*
3. *Let $R \subset D^r$ be a relation for some domain D and arity r . Then R is preserved by u_k^c for every $k \geq 2$ and $c < k$ if and only if R is preserved by u_r^{r-1} .*

Proof. For the first, if we view the arguments of U_k^{k-1} as corresponding to non-zero bitstrings $b \in \{0, 1\}^k$, then applying π directly to the bitstring yields the desired permutation σ . For the second, first let $R \subseteq D^r$ be an r -ary relation preserved by u_p^{p-1} for some $p \geq 2$, and let $t_1, \dots, t_m \in R$ be such that $u_q^{q-1}(t_1, \dots, t_m) = t$ is defined, $q < p$. Select $t' \in R$ arbitrarily. Then $u_p^{p-1}(t_1, \dots, t_m, t', \dots, t') = t$ is defined, by the description in Proposition 7.24. Thus u_p^{p-1} implies u_q^{q-1} . To complete the property, we note that for any k , and any $a \leq b < k$, u_k^b implies u_k^a : in line with Proposition 7.24, the patterns of u_k^c for any c correspond to sets $S \subset [k]$ containing c indexes from $[k]$, and such a pattern has the same variable in two positions i, j if and only if i and j match on all bits present in S . Thus, for every pattern P_S of u_k^a , indexing a bits $S \subset [k]$, let $S' \subset [k]$ be any subset of S with b bits. Then the pattern $P_{S'}$ is strictly less restrictive than P_S , so any tuple that matches a pattern of u_k^a also matches a pattern of u_k^b . Similarly, the output of pattern $P_{S'}$ matches an argument that also matches the output of P_S .

For the last point, we note by the second property that R is preserved by all u_k^c , $k \geq 2$, $c < k$ if and only if R is preserved by u_k^{k-1} for all $k \geq 2$. Thus, assume that R is violated by u_k^{k-1} for some $k \geq 2$. If $k < r$, then R is violated by u_r^{r-1} by the second property. Otherwise, assume $k > r$, and let $t_1, \dots, t_m \in R$ be such that $u_k^{k-1}(t_1, \dots, t_m) = t$ is defined where $t \notin R$. By the first property, we may assume that this application of u_k^{k-1} only uses patterns P_i corresponding to the r least significant bits according to Proposition 7.24. But then, all “repeats” in the pattern have distance at most 2^{r-1} and $u_r^{r-1}(t_1, \dots, t_{2^{r-1}}) = t$ is defined, showing $u_r^{r-1} \notin \text{pPol}(R)$. $\square$

The last property allows us to verify with a finite computation that a relation cannot be shown to have superlinear NRD using c -fgppp-definitions of OR (i.e., via Lemma 7.22). This may be interesting to compare with Malt’s embeddings and the Catalan terms from Section 6: to verify that a language admits a Malt’s embedding it needs to be invariant under all (infinitely many) Catalan terms.

Unfortunately, the second property of Lemma 7.25 does not generalize: in general, if $p/q \leq k/c$, it does not necessarily follow that u_p^q implies u_k^c . Let $P_1 = (X_1, y_1), \dots, P_6 = (X_6, y_6)$ enumerate the patterns of U_4^2 , and let V be the set of variables of U_4^2 . Define a relation $R \subseteq V^6$ by $R = \{t_1, \dots, t_{15}\}$ where $t_i = (X_1[i], \dots, X_6[i])$. Then by design R is not preserved by u_4^2 , but one can verify that R is preserved by $u_2^1 = u_2$.

Finally, we note that the final property of Lemma 7.25 represents a weakness in the method, in that the cube-power patterns U_p^q cannot be used to prove a lower bound of $\Omega(n^{1+\varepsilon})$ for the non-redundancy of an r -ary relation R for any $0 < \varepsilon < 1/(r-1)$, despite the results of Section 4 showing that infinitely many such bounds exist, even for $r \leq 3$ (at least conditionally, in Section 4.1, and conjecturally, in Section 4.3). We show a complementary upper bound, showing that U_p^q cannot show a lower bound $\Omega(n^{r-\varepsilon})$ for the non-redundancy of R for any $0 < \varepsilon < 1$ unless $\text{NRD}(R, n) = \Theta(n^r)$.

Lemma 7.26. *Let $R \subseteq D^r$ for $r \geq 2$ be a relation such that $U_r \in \text{Pattern}(R)$. Then $U_p^q \in \text{Pattern}(R)$ for every $p/q > r-1$.*

Proof. Assume $u_p^q \notin \text{pPol}(R)$ for some $p/q > r-1$. Then R q -fgpp-defines OR_p , and this definition can be assumed to consist of a single constraint, i.e.,

$$\text{OR}_p(x_1, \dots, x_p) \equiv R(g_1(X_1), \dots, g_r(X_r))$$

for some maps $g_i: \{0, 1\}^q \rightarrow D$ and some scopes $X_i \subseteq X$ of q variables each, where $X = \{x_1, \dots, x_p\}$. First assume that there is a scope X_i , say X_r , such that every variable in X_r also occurs in X_j for some $j < r$. Then $|\bigcup_{i=1}^{r-1} X_i| = |X| = p \leq (r-1)q$. Otherwise, let x_{i_j} , $j \in [r]$ be variables such that x_{i_j} occurs only in X_j for $j \in [r]$. Consider the set of assignments to X where $x_t = 0$ for every $t \in [r] \setminus \{i_1, \dots, i_r\}$. Each map g_i , $i \in [r]$, takes two distinct values under these assignments, and this defines a subset $F = \prod_{i=1}^r \{a_i, b_i\} \subseteq D^r$ such that $|F \setminus R| = 1$. Thus we have a witness that R violates u_r . $\square$

Last, it is interesting to relate cube powers to Mal'tsev embeddings. First, Lemma 7.22 implies that if $\text{pPol}(P)$ lacks u_k^{k-1} , then $\text{NRD}(P, n) = \Omega(n^{k/(k-1)})$, and P lacks a finite Mal'tsev embedding. However, we can give a more direct argument as follows.

Lemma 7.27. *Let R be a finite relation with a Mal'tsev embedding (possibly into an infinite domain). Then R is preserved by u_k^{k-1} for every $k \geq 2$.*

Proof. Let $R \subseteq D^r$ for some domain D and arity r . Fix $k \geq 2$ and let $u_k^{k-1}(t_1, \dots, t_m) = t$ be defined, where $t_1, \dots, t_m \in R$ and $m = 2^k - 1$. Assume that the arguments of u_k^{k-1} are ordered as in Prop. 7.24. Let D' be the domain of the Mal'tsev embedding of R , let $\varphi: (D')^3 \rightarrow D'$ be a Mal'tsev term, and let $f_1, f_3, \dots, f_{2^k-1} \in [\varphi]$ be Catalan terms, which exist by Theorem 6.4. Let σ be an ordering on $\{0, \dots, 2^k-1\}$ that corresponds to the reflected binary Gray code, where $\sigma(i)$ and $\sigma(i+1)$ differ in precisely one digit. The Gray code has a recursively defined structure as follows: $\sigma(0) = 0$, $\sigma(1) = 1$, and the $(n+1)$ -bit Gray code is defined by letting $\sigma(0)$ through $\sigma(2^n-1)$ be the n -bit Gray code, then letting the second half be the first half reflected with the top bit set. We note that due to this symmetry, the terms in any tuple in the domain of u_k^{k-1} cancel exhaustively if read in the order of σ .

Claim 7.28. *Let $u_k^{k-1}(x_1, \dots, x_{2^k-1}) = y$ be defined for some tuple of values $x_1, \dots, x_{2^k-1}, y \in D$. Then $f_{2^k-1}(x_{\sigma(1)}, \dots, x_{\sigma(2^k-1)}) = y$.*

Proof. Let $(x_1, \dots, x_{2^{k-1}})$ match the pattern $P_i = (X_i, y_i)$ for $i \in [k]$ in U_k^{k-1} , i.e., $X_i[p]$ corresponds to the binary expansion of p with bit i removed, and consider the sequence $S = (y_i, x_{\sigma(1)}, \dots, x_{\sigma(2^{k-1})})$, where y_i in a sense corresponds to $\sigma(0) = 0$. Let p be the first position such that bit i of $\sigma(p)$ is set; then $p = 2^q$ for some $q \in \{0, \dots, k-1\}$. Now, for every $d \in \mathbb{N}$, the subsequence $S[d \cdot 2p : (d+1) \cdot 2p]$ of S (exclusive of the final point) is symmetric around the pair $(p-1, p)$ and will vanish if identical adjacent terms are cancelled exhaustively. Applying this to all pairs except the unique pair involving y_i shows that cancellations inside the X_i -part of S will leave a copy of y_i as the unique uncanceled term. Since every tuple in the domain of u_k^{k-1} comes from some such pattern P_i , the claim follows. $\square$

Thus, for every $i \in [r]$ we have that $f_{2^{k-1}}(t_1[i], \dots, t_{2^{k-1}}[i]) = t[i]$, so $t \in R$ and $u_k^{k-1} \in \text{pPol}(R)$. $\square$

7.4 Connections to Hypergraph Turán

Given a set of patterns Q we are now interested in studying the conditional non-redundancy of families of promise relations invariant under Q . We study this systematically in this section and exemplify our main result in Section 7.5

Before we can state the main result we need to extend the algebraic vocabulary so that it is applicable to CSP instances and not only to relations. First, given $R \subseteq S \subseteq D^r$ and an instance (X, Y) of $\text{CSP}(R \mid S)$ recall that we (by Lemma 2.3) without loss of generality may assume that it is r -partite and that each $i \in [r]$ is associated with exactly one sort X_i . Given a multisorted pattern $P = (P_1, \dots, P_r)$ of arity n each P_i then induces a set of partial functions $I_{X_i}(P_i)$, and given $p_i \in I_{X_i}(P_i)$ for every $1 \leq i \leq r$ we can then apply $p_1, \dots, p_r$ to $y^1, \dots, y^n$ as $(p_1, \dots, p_r)(y^1, \dots, y^n)$ which for each $1 \leq i \leq r$ yields $p_i(y_i^1, \dots, y_i^n)$.

Lemma 7.29. *Let $R \subseteq S \subseteq D_1 \times \dots \times D_r$ be multisorted relations with r sorts $D_1 \cup \dots \cup D_r = D$ and let $P = (P_1, \dots, P_r)$ be an n -ary multisorted pattern such that $(R, \tilde{S}) \in \text{Inv}(P)$. Let (X, Y) be an r -partite instance of $\text{CSP}(R \mid S)$ and let $y^{(1)}, \dots, y^{(n)}, y \in Y$ such that $(p_1, \dots, p_r)(y^{(1)}, \dots, y^{(n)}) = y$ is defined and $y \notin \{y^{(1)}, \dots, y^{(n)}\}$ for some $p_1 \in I_{X_1}(P_1), \dots, p_r \in I_{X_r}(P_r)$. Then y is conditionally redundant in (X, Y) .*

Proof. Assume that there is an assignment $\sigma: X \rightarrow D$ such that $\sigma(y) \in S \setminus R$ and $\sigma(y') \in R$ for every $y' \in Y - \{y\}$. It follows that there exists $q_1 \in I_{D_1}(P_1), \dots, q_r \in I_{D_r}(P_r)$ such that $\sigma(y) = (q_1, \dots, q_r)(\sigma(y^{(1)}), \dots, \sigma(y^{(n)}))$: for every position $i \in [r]$, we have $y_i = p_i(y_i^{(1)}, \dots, y_i^{(n)})$, hence there exists a pair $((x_1, \dots, x_n), x_{j_i}) \in P_i$, so that $y_i = y_i^{(j_i)}$ and:

1. if $x_{j_i} \in \{x_1, \dots, x_n\}$ then $q_i(\sigma(y^{(1)})_i, \dots, \sigma(y^{(n)})_i) = \sigma(y^{(j_i)})_i$ for every possible $q_i \in I_{D_i}(P_i)$, and
2. if $x_{j_i} \notin \{x_1, \dots, x_n\}$ then $I_{D_i}(P_i)$ for every $d \in |D_i|$ contains a partial function q^d such that $q^d(\sigma(x_1), \dots, \sigma(x_n)) = d$, and we choose $d = \sigma(y_i)$.

But now $\sigma(y^{(1)}), \dots, \sigma(y^{(n)}) \in R$ and $\sigma(y) \in S$, which implies that $\sigma(y) \in R$, contrary to our assumption. $\square$

Lemma 7.29 requires a fair amount of notation to state but as the following example shows is actually a reasonable and straightforward property on conditionally non-redundant constraints.

Example 7.30. We recall the relations $R = 3\text{LIN}_{\mathbb{Z}/3\mathbb{Z}} = \{(x, y, z) \in \{0, 1, 2\}^3 \mid x + y + z \equiv 0 \pmod{3}\}$ and $R^* = 3\text{LIN}_{\mathbb{Z}/3\mathbb{Z}}^* = R \setminus \{(0, 0, 0)\}$ from Brakensiek & Guruswami [14], introduced in Section 7.3. Consider the multisorted polymorphism pattern $P = (P_1, P_2, P_3)$ where (1) $P_1 = P_2 = \{((x), x)\}$ is the pattern satisfied only by the unary projection and (2) $P_3 = \{((x), y)\}$ is an unrestricted pattern, i.e., one satisfied by any unary function. We claim that $(R^*, \tilde{R}) \in \text{Inv}(P)$. To see this, note first that $I_{\{0,1,2\}}(P_1) = I_{\{0,1,2\}}(P_2) = \{\pi_1^1\}$, while $I_{\{0,1,2\}}(P_3) = \{0, 1, 2\}^{\{0,1,2\}}$. Thus, the preservation condition boils to the following: for any tuple $t \in R^*$ we must have that $(t_1, t_2, d) \in \tilde{R}$ for any $d \in \{0, 1, 2\}$, i.e., that (t_1, t_2, d) is not included in $R \setminus R^* = \{(0, 0, 0)\}$. This is clearly true since if at least only entry in t is non-zero and the sum is 0, t_1 and t_2 cannot both be 0, and the claim follows.

Now, consider an instance $(\{x, y, z, v\}, \{(x, y, z), (x, y, v)\})$ of $\text{CSP}(R^* \mid R)$. We introduce disjoint copies X_1, X_2, X_3 of X and then use functions

1. $\pi_1^1 \in I_{X_1}(P_1)$,
2. $\pi_1^1 \in I_{X_2}(P_2)$,
3. $f \in I_{X_3}(P_3)$ where $f(d) = v$ for every $d \in X_3$.

It then follows that $(p_1, p_2, p_3)((x, y, z)) = (x, y, v) \in Y$ and that (x, y, v) is conditionally redundant in $\text{CSP}(R^* \mid R)$.

To connect to the hypergraph structure, we need the following assumptions. Let $P = (P_1, \dots, P_r)$ be an n -ary multisorted polymorphism pattern. Say that P is *unit* if $|P_i| = 1$ for every $i \in [r]$, i.e., each P_i consists of a single pattern $((x_{i,1}, \dots, x_{i,n}), y_i)$. For a set of patterns Q with r sorts we let $\text{Inv}^r(Q) \subseteq \text{Inv}(Q)$ be the set of promise relations of arity r (with exactly r sorts).

Proposition 7.31. *Let $P = (P_1, \dots, P_r)$ be an n -ary multisorted polymorphism pattern. Then there exists a set of unit patterns $U \subseteq [P]_{\min}$ such that $\text{Inv}^r(U) = \text{Inv}^r(P)$.*

Proof. First, observe that $U = \{(U_1, \dots, U_r) \mid 1 \leq i \leq r, U_i \subseteq P_i, |U_i| = 1\} \subseteq [P]_{\min}$ since we allow subsets in the minor definition. Due to Proposition 5.28 this implies that $\text{Inv}(P) \subseteq \text{Inv}(U)$ and hence also that $\text{Inv}^r(P) \subseteq \text{Inv}^r(U)$. For the other direction, assume there exists $(S, T) \in \text{Inv}^r(U)$ where $(S, T) \notin \text{Inv}_D^r(P)$ for any finite set $D = D_1 \cup \dots \cup D_r$, and let $p_1 \in I_{D_1}(P_1), \dots, p_r \in I_{D_r}(P_r)$ witness this, i.e., $(p_1, \dots, p_r)(s^{(1)}, \dots, s^{(n)}) = t \notin T$ for $s^{(1)}, \dots, s^{(n)} \in S$. Each such application of a p_i is witnessed by a unit pattern, and it follows that there exist subfunctions $u_1 \in I_{D_1}(U_1), \dots, u_r \in I_{D_r}(U_r)$ of $p_1, \dots, p_r$ such that $(u_1, \dots, u_r)(s^{(1)}, \dots, s^{(n)}) = t \notin T$, contradicting the original assumption. $\square$

Let $P = (P_1, \dots, P_r)$ be a unit multisorted pattern with patterns $P_i = \{((x_{i,1}, \dots, x_{i,n}), y_i)\}$ (where the $x_{i,j}$ are not necessarily distinct). Say that P is a *partial projection* if there is $j \in [n]$ such that for every $i \in [r]$, $y_i = x_{i,j}$, and *non-trivial* otherwise.

We define a hypergraph $\mathcal{H}(P)$ from P as follows. For $i \in [r]$, let X_i be the variable set that P_i is defined over, and assume (syntactically) that $X_i \cap X_j = \emptyset$ for $i \neq j$. Then $\mathcal{H}(P)$ is the hypergraph on vertex set $X = X_1 \cup \dots \cup X_r$ and with $n + 1$ hyperedges given by $Y = \{(x_{1,j}, \dots, x_{r,j}) \mid j \in [n]\} \cup \{(y_1, \dots, y_r)\}$. Refer to $(y_1, \dots, y_r)$ as the *output edge* of $\mathcal{H}(P)$.

We can now observe the following.

Lemma 7.32. *Let $R \subseteq S \subseteq D^r$ be a pair of multisorted relations and let P be a unit multisorted pattern that is not a partial projection, such that $P \in \text{mPattern}(R, \tilde{S})$. Let (X, Y) be a conditionally non-redundant instance of $\text{CSP}(R \mid S)$. Then (X, Y) is $\mathcal{H}(P)$ -free (when interpreted as a hypergraph).*

Proof. Let $\{y^{(1)}, \dots, y^{(n)}, y\} \subseteq Y$ form a hypergraph isomorphic to $\mathcal{H}(P)$ such that each $y^{(i)}$ maps to the i th edge of $\mathcal{H}(P)$ in the order of construction and where y maps to the output edge. Then, by construction, $(p_1, \dots, p_r)(y^{(1)}, \dots, y^{(n)})$ is defined for all $p_1 \in I_{X_1}(P_1), \dots, p_r \in I_{X_r}(P_r)$, and it is clear that there exists such functions which yield y . Since P is not a partial projection it follows that $y \notin \{y^{(1)}, \dots, y^{(n)}\}$ and that (X, Y) is conditionally redundant by Lemma 7.29. $\square$

Recall by Proposition 7.31 that for r -ary relations a general multisorted pattern $P = (P_1, \dots, P_r)$ is equivalent to the collection of unit patterns produced by selecting one pair from P_i for every $i \in [r]$. Hence, working with only unit patterns is not restrictive, and Lemma 7.32 then implies that every multisorted polymorphism pattern implies a finite collection of excluded multipartite hypergraphs.

To show the result in the other direction, consider the following. Let an instance (X, Y) of $\text{CSP}(R \mid S)$ be *minimal redundant* if (X, Y) is conditionally redundant but for every $y \in Y$, $(X, Y - \{y\})$ is a conditionally non-redundant instance of $\text{CSP}(R \mid S)$. Note that this is a well-behaved notion since conditionally non-redundant instances (X, Y) are closed under taking subsets of Y . We now show the following.

Lemma 7.33. *Let (X, Y) be a minimal redundant instance of $\text{CSP}(R \mid S)$ for some $R \subseteq S \subseteq D^r$. Let $y \in Y$ be such that there is no assignment ψ that satisfies every $y' \in Y - \{y\}$ (i.e., $\psi(y') \in R$) but has $\psi(y) \in S \setminus R$. Then there is a unit multisorted polymorphism pattern P and a permutation $y_1, \dots, y_m$ of $Y - \{y\}$ such that $P \in \text{mPattern}(R, \tilde{S})$ and $p_1 \in I_{X_1}(P_1), \dots, p_r \in I_{X_r}(P_r)$ such that $(p_1, \dots, p_r)(y_1, \dots, y_m) = y$ and $\mathcal{H}(P) = (X, Y)$.*

Proof. Let $Y = \{y_1, \dots, y_m, y\}$ and let y be such that there is no assignment ψ such that $\psi(y') \in R$ for every $y' \in Y - \{y\}$ and $\psi(y) \in S \setminus R$. This exists, as otherwise (X, Y) is conditionally non-redundant. Define a multisorted pattern $P = (P_1, \dots, P_r)$ on variable set X such that for $i \in [r]$ we have $P_i = \{((y_1(i), \dots, y_m(i)), y(i))\}$. Observe that P is a unit and non-trivial. Next, we show that $P \in \text{mPattern}(R, \tilde{S})$.

Assume for the sake of contradiction that $P \notin \text{mPattern}(R, \tilde{S})$. Then, there exist tuples $t_1, \dots, t_m \in R$ and $p_1 \in I_{X_1}(P_1), \dots, p_r \in I_{X_r}(P_r)$ such that $t = (p_1, \dots, p_r)(t_1, \dots, t_m)$ is defined and $t \in S \setminus R$. Define $\psi: X \rightarrow D$ by letting $\psi(y_i(j)) = t_i(j)$ for every $i \in [m], j \in [r]$, and likewise $\psi(y(j)) = t(j)$. Note that this is consistently defined by construction of P . Then $\psi(y_i) \in R$ for every $i \in [m]$, and $\psi(y) \in S \setminus R$, which is contrary to our assumptions. Thus $P \in \text{mPattern}(R, \tilde{S})$. The remaining statements hold by construction. $\square$

To finish our characterization we first define the following tensor product which allows us to combine two promise relations (S_1, T_1) and (S_2, T_2) into a promise relation of the same arity but over a larger domain. First, given R and S we let

$$R \otimes S := (\{((a_1, b_1), \dots, (a_r, b_r)) : a \in R, b \in S\})$$

and then

$$(S_1, T_1) \otimes (S_2, T_2) := (S_1 \otimes S_2, T_1 \otimes T_2).$$

Proposition 7.34. *Let (S_1, T_1) and (S_2, T_2) be two r -ary multisorted relations with r sorts. Then $(S_1, T_1) \otimes (S_2, T_2)$ is fgppp-equivalent to $\{(S_1, T_1), (S_2, T_2)\}$.*

Proof. Assume that $S_1 \subseteq T_1 \subseteq D_1 \times \dots \times D_r$ and that $S_2 \subseteq T_2 \subseteq E_1 \times \dots \times E_r$. For $1 \leq i \leq r$ let $g_1^i: D_i \times E_i \rightarrow D_i$ be the first projection and $g_2^i: D_i \times E_i \rightarrow E_i$ the second projection. Then,

$$(S_1 \otimes S_2)(x_1, \dots, x_r) \equiv S_1(g_1^1(x_1), \dots, g_1^r(x_r)) \wedge S_2(g_2^1(x_1), \dots, g_2^r(x_r)),$$

and

$$(T_1 \otimes T_2)(x_1, \dots, x_r) \equiv T_1(g_1^1(x_1), \dots, g_1^r(x_r)) \wedge T_2(g_2^1(x_1), \dots, g_2^r(x_r)).$$

Thus, $\{(S_1, T_1), (S_2, T_2)\}$ fgppp-defines $(S_1, T_1) \otimes (S_2, T_2)$. For the other direction, fix $s \in S_2 \subseteq T_2$, and for $i \in [r]$, let $h_i: D_i \rightarrow D_i \times E_i$ be the map $h_i(x) = (x, s_i)$. Then,

$$S_1(x_1, \dots, x_r) \equiv (S_1 \otimes S_2)(h_1(x_1), \dots, h_r(x_r)),$$

and

$$T_1(x_1, \dots, x_r) \equiv (T_1 \otimes T_2)(h_1(x_1), \dots, h_r(x_r)).$$

By an analogous formula, $(S_1, T_1) \otimes (S_2, T_2)$ fgppp-defines (S_2, T_2) as well. $\square$

Via Corollary 7.12 we obtain the following general description.

Corollary 7.35. *Let (S_1, T_1) and (S_2, T_2) be two multisorted promise relations with r sorts. Then*

1. $(S_1, T_1) \otimes (S_2, T_2)$ is fgppp-equivalent to $\{(S_1, T_1), (S_2, T_2)\}$, and
2. $\text{mPattern}(\{(S_1, T_1), (S_2, T_2)\}) = \text{mPattern}((S_1, T_1) \otimes (S_2, T_2))$.

Given a set of multisorted patterns Q (all with exactly r sorts), we define $\text{Inv}^{=r}(Q)$ to be the set of multisorted pairs $(R, \tilde{S})$ of arity r over any domain with exactly r sorts, one for each argument, such that $(R, \tilde{S}) \in \text{Inv}(Q)$. We then define

$$\text{NRD}(\text{Inv}^{=r}(Q), n) := \max_{(R, \tilde{S}) \in \text{Inv}^{=r}(Q)} \text{NRD}(R \mid \tilde{S}, n).$$

We now prove that $\text{NRD}(\text{Inv}^{=r}(Q), n)$ is precisely captured by a hypergraph Turán problem, generalizing observations of [9, 21].

Theorem 7.36. *Let Q be a finite set of multisorted polymorphism patterns with exactly r sorts. Then there is a finite set $\mathcal{H}(Q)$ of r -partite hypergraphs with the following property: for all $n \in \mathbb{N}$,*

$$\text{NRD}(\text{Inv}^{=r}(Q), n) = \text{ex}_r(n, \mathcal{H}(Q)), \quad (20)$$

where $\text{ex}_r(n, \mathcal{H})$ is the size of the largest r -partite r -uniform hypergraph on n vertices excluding $\mathcal{H}$.

Proof. Let $[Q]_{\text{surj}}$ be the closure of Q with respect to *surjective* minors: in other words, if $P \in Q$ of arity m , then P' of arity n is in $[Q]_{\text{surj}}$ if there exists a surjective function $h: [m] \rightarrow [n]$ (so $n \leq m$) such that for $i \in [r]$ and every $((x_1, \dots, x_n), x) \in P'_i$ we have that $((x_{h(1)}, \dots, x_{h(m)}), x) \in P_i$.

Recall that if P is a unit pattern polymorphism, then there is a corresponding hypergraph $\mathcal{H}(P)$. Slightly overloading the notation we then define

$$\mathcal{H}(Q) := \{\mathcal{H}(P) : P \in [Q]_{\text{surj}}, P \text{ unit}\}.$$

We prove (20) as a pair of inequalities. Since $\text{Inv}^{=r}(Q) = \text{Inv}^{=r}([Q]_{\text{surj}})$ (this is a special case of Proposition 5.28), we have that the $\leq$ direction of (20) is an immediate consequence of Lemma 7.32.

Let (X, Y) be the largest r -partite hypergraph on n vertices which is $\mathcal{H}(Q)$ -free. For every $Y' \subseteq Y$ and $y' \in Y'$, let $P(Y', y')$ be the r -sorted unit pattern polymorphism of arity $\text{ar}(P(Y', y')) = |Y'| - 1$ which maps $Y' \setminus \{y'\}$ to y' .

The key observation is that $P(Y', y') \notin [Q]_{\min}$. If not, there exists $P' \in Q$ and $P'' \subseteq P'$ unit such that $P(Y', y') = (P'')_{/h}$ for some *non*-surjective $h: [\text{ar}(P'')] \rightarrow [\text{ar}(P(Y', y'))]$. Note then there must exist $Y'' \subseteq Y'$ and a surjective map $h': [\text{ar}(P'')] \rightarrow [\text{ar}(P(Y'', y'))]$ such that h and h' are equivalent in the sense that $P(Y'', y') = (P'')_{/h'}$. In that case, $P(Y'', y') \in [Q]_{\min}$, which contradicts the fact that (X, Y) and thus (X, Y'') is $\mathcal{H}(Q)$ -free.

Thus, $P(Y', y') \notin [Q]_{\min}$, so $\text{Inv}^{\text{=r}}(Q) \supsetneq \text{Inv}^{\text{=r}}(Q \cup \{P(Y', y')\})$. Thus, it is possible to find $(R(Y', y'), S(Y', y')) \in \text{Inv}^{\text{=r}}(Q)$ such that $P(Y', y') \notin \text{mPattern}(R(Y', y'), S(Y', y'))$. Now define

$$(R, S) := \bigotimes_{Y' \subseteq Y, y' \in Y'} (R(Y', y'), S(Y', y')).$$

By Corollary 7.35, we know that $Q \subseteq \text{mPattern}(R, S)$, but $P(Y', y') \notin \text{mPattern}(R, S)$ for all Y' and y' . To finish, it suffices to prove that (X, Y) is a conditionally non-redundant instance of $\text{CSP}(R \mid S)$. If not, there exists a minimally redundant subinstance (X, Y') . By Lemma 7.33, we then have that $P(Y', y') \in \text{mPattern}(R, \tilde{S})$, a contradiction. Thus, (X, Y) is non-redundant. $\square$

7.5 Hypergraph Turán Examples and Applications

Let us now consider some prominent examples of this connection.

Example 7.37. One of the most basic polymorphism patterns is the k -NU pattern

$$P = \{((y, x, \dots, x), x), ((x, y, \dots, x), x), \dots, ((x, x, \dots, y), x)\}$$

(introduced in Example 7.3). If we turn this into a k -partite, k -uniform hypergraph by placing each pattern in a different vertex class, then we get the hypergraph $\mathcal{H}$ with edges

$$\{(x, x, \dots, x), (x, x, \dots, y), \dots, (y, x, \dots, x)\}.$$

It is easy to see that a k -partite, k -uniform hypergraph (X, Y) is $\mathcal{H}$ -free if and only if the following holds: for each edge $E \in Y$ there is a strict subedge $E' \subset E$ such that E' only occurs in E . Therefore, $|Y| = \mathcal{O}(|X|^{k-1})$ and $\text{ex}_r(n, \mathcal{H}) = \Theta(n^{k-1})$. In this case, the bound is tight, in that OR_{k-1} (e.g., with a fictitious added argument) is preserved by the k -NU pattern.

Example 7.38. Another class of tight examples comes from Section 4. First consider the promise relation (C_{2k}^*, C_{2k}) for $k \geq 2$. In this case, the conditional polymorphisms are the patterns with $P_1 = \{((0, 1, 1, 2, 2, \dots, t-1, t-1), 0)\}$ and $P_2 = \{((1, 0, 2, 1, \dots, t-2, t-1), 0)\}$ for $t < k$, and the corresponding forbidden hypergraphs (which are graphs, since $r = 2$) are the even cycles of length less than $2k$. By the results in Section 4.1, we have $\text{NRD}(C_{2k}^* \mid C_{2k}, n) = \Theta(\text{ex}_2(n, \{C_2, C_4, \dots, C_{2k-2}\}))$. Similarly, for the relation R_{2k} in Section 4.2, we get a multisorted pattern with $P_1 = \{((x, \dots, x), x)\}$ and where P_2 and P_3 enumerate a cycle C_{2t} , $t < k$ as above. Let $\mathcal{H}(Q)$ be the corresponding set of hypergraphs. Then a 3-uniform, 3-partite hypergraph (X, Y) is $\mathcal{H}(Q)$ -free if and only if the neighbourhood of every vertex in part 1 forms a graph with girth at least $2k$, and again, the results of Section 4.2 imply that $\text{NRD}(R_{2k}, N) = \Theta(\text{ex}_3(n, \mathcal{H}(Q)))$.

Example 7.39. For a complementary example where a similar tight result does not hold (for finite $\mathcal{H}(Q)$), consider the case of binary relations preserved by the partial Mal'tsev operation u_2 . That

is, let $P = \{((x, x, y), y), ((x, y, y), x)\}$ and let $R \subseteq D^2$ with $P \in \text{Pattern}(R)$. The only non-trivial multisorted unit pattern is $P' = (P_1, P_2)$ where $P_1 = \{((x, x, y), y)\}$ and $P_2 = \{((x, y, y), x)\}$, and the corresponding forbidden graph is $\mathcal{H}(P') = C_4$. Then $\text{ex}_2(n, \mathcal{H}(P)) = \Theta(n^{1.5})$. However, if R is preserved by u_2 as a (non-conditional) partial polymorphism, then $\text{NRD}(R, n) = \mathcal{O}(n)$, as shown by previous work [20, 9]. This can be viewed as follows: For any relation $R \subseteq D^2$, view R as a bipartite graph G . Applying u_2 exhaustively to R (i.e., adding $u_2(t_1, t_2, t_3)$ to R whenever this is defined and not already present in R) yields a rectangular relation, where every connected component of G is completed into a biclique. The same argument applies to any instance (X, Y) of $\text{CSP}(R)$. Thus a non-redundant instance (X, Y) of $\text{CSP}(R)$ must be acyclic. This “closure” aspect of repeated application of u_2 is missed by the Turán connection. One could include further patterns that capture the consequences of such propagation, which would then correspond to adding the graphs $C_6, C_8, \dots$ to the collection of forbidden subgraphs, but for any finite collection of such subgraphs we get a Turán exponent of $\text{ex}_2(n, \mathcal{H}(Q)) = \Theta(n^{1+\varepsilon})$ for some $\varepsilon > 0$ as above. Thus no finite collection of forbidden hypergraphs captures the consequences of the partial polymorphism u_2 .

Example 7.40. For a much more drastic example, let $r \geq 3$ and let R be an r -ary linear equation, e.g., $R \subset \{0, 1\}^r$ defined as $R(x_1, \dots, x_r) \equiv (x_1 \oplus \dots \oplus x_r = 0)$. Then a hypergraph $\mathcal{H}$ is a forbidden configuration for a non-redundant instance (X, Y) of $\text{CSP}(R)$ if and only if the edges of $\mathcal{H}$ are linearly dependent (as a set of vectors over $\text{GF}(2)$). Unfortunately, large systems of r -ary equations are possible where no constant-sized subsystem is linearly dependent. For example, let $\varepsilon > 0$ and consider a random instance (X, Y) of $\text{CSP}(R)$ with $|X| = n$ and edges chosen uniformly at random so that $E[|Y|] = n^{r/2-\varepsilon}$. Then it can be shown that for any $t \in \mathbb{N}$, with high probability (asymptotically in n) every set $F \subseteq Y$ with $|F| \leq t$ has $|\bigcup F| > r|F|/2$, and consequently has a variable that occurs only once in F . Since every linearly dependent set contains a minimal linearly dependent set, and no such set can contain a variable that occurs only once, it follows that every set of $\leq t$ edges from Y is independent. Thus, any finite collection $\mathcal{H}(Q)$ of forbidden hypergraphs for $\text{CSP}(R)$ has $\text{ex}_r(n, \mathcal{H}(Q)) = \Omega(n^{r/2-o(1)})$ (at least), despite $\text{NRD}(R, n) = \mathcal{O}(n)$. Note that $n^{r/2}$ is also the threshold at which constant degrees of the Sum-of-Squares hierarchy fails to refute r -ary linear equations (e.g., [58]). Similar insights can also be applied to the *range avoidance* problem in computational complexity (e.g., [60]).

Given these examples, one of the more interesting questions corresponds to the k -cube patterns U_k . For these patterns, the corresponding hypergraph $\mathcal{H}(U_k)$ is the complete k -partite k -uniform hypergraph $K_{2,2,\dots,2}$ with two vertices per part, and the value of $\text{ex}(n, \mathcal{H}(U_k))$ is a notorious open question in extremal combinatorics, even for $k = 3$, known as the *Erdős box problem* [35, 53, 40, 28, 41]. This connection was first noticed by Carbonnel [21], who used it to show that every r -ary relation R has either $\text{NRD}(R, n) = \Theta(n^r)$ or $\text{NRD}(R, n) = \mathcal{O}(n^{r-\varepsilon_r})$, where $\varepsilon_r \geq 2^{1-r}$ comes from $\text{ex}(\mathcal{H}(U_r), n) = \text{ex}(K_{2,\dots,2}, n) = \mathcal{O}(n^{r-\varepsilon_r})$ [35], and is based on the assumption that R is preserved by the cube pattern U_r . Focusing on $r = 3$, our results in Section 7.4 imply that there is an infinite family of ternary promise relations whose conditional non-redundancy matches $\text{ex}(K_{2,2,2}, n)$, but what about a single such promise relation? At the moment, our largest known bound for $r = 3$ below $\text{NRD}(\text{OR}_3, n) = \Theta(n^3)$ is $\text{NRD}(R, n) = \Theta(n^{2.5})$ from Section 4.2.

Based on these connections it is natural to investigate if stronger $\mathcal{O}(n^{r-\varepsilon})$ bounds can be obtained for stronger patterns than U_r . Promising candidates are k -edge operations, that are known to generalize k -NU and Mal'tsev operations [8]. It is a $(k+1)$ -ary operation that satisfies

the following identities ($k \geq 2$) for all x, y .

$$\begin{aligned}
p(x, x, y, y, y, \dots, y) &= y \\
p(x, y, x, y, y, \dots, y) &= y \\
p(y, y, y, x, y, \dots, y) &= y \\
p(y, y, y, y, x, \dots, y) &= y \\
&\dots \\
p(y, y, y, y, y, \dots, x) &= y
\end{aligned}$$

In particular, a 2-edge operation is a Mal'tsev operation. For example, the forbidden hypergraph for the 3-edge polymorphism pattern has edge set $\{(x, x, x), (x, x, y), (x, y, x), (y, x, x), (y, y, x)\}$, i.e., the same as 3-NU plus the edge (y, y, x) . A language has a (total) k -edge polymorphism if and only if it has *few subpowers*, i.e., only $2^{n^{O(1)}}$ distinct n -ary relations can be pp-defined in the language; see Berman et al. [8] or the survey by Barto, Krokhin and Willard [4]. Lagerkvist and Wahlström [61] showed that a language with a k -edge embedding, generalizing Mal'tsev embeddings, has non-redundancy $O(n^{k-1})$. Here, we show that a partial k -edge polymorphism (i.e., the partial pattern polymorphisms defined from the above identities) also implies a non-trivial non-redundancy result.

Lemma 7.41. *Let $R \subseteq S \subseteq D^r$ be a pair of relations such that $(R, \tilde{S})$ is preserved by the k -edge polymorphism pattern, where $r \geq k$. Then $\text{NRD}(R \mid S, n) = O(n^{r - \lfloor r/k \rfloor / 2})$.*

Proof. We first show the result for the case $r = k$. Let $I = (X, Y)$ be a k -partite instance of $\text{CSP}(R \mid S)$ and consider a constraint $(v_1, \dots, v_k) \in Y$. Via Lemma 7.32, we find the following. Select $k - 2$ positions, w.l.o.g. $1, \dots, k - 2$, and consider the bipartite graph G where $uw \in E(G)$ if and only if there is a constraint $(v_1, \dots, v_{k-2}, u, w)$ in I . Then the hypergraph representation of the k -edge pattern for k -ary relations allows us to discard R if two things apply: (1) $v_{k-1}v_k$ participates in a C_4 in G and (2) for every $i \in [k - 2]$ there is a constraint $(v_1, \dots, v'_i, \dots, v_k)$ in I where $v'_i \neq v_i$. To argue the density we proceed as follows. Given $I = (X, Y)$, we first mark one constraint from Y for every $(k - 1)$ -ary projection; this makes for $O(n^{k-1})$ constraints. Put these constraints aside, then form the bipartite graph G as above using only the remaining constraints Y' . Let $C = (v_1, \dots, v_k)$ be any constraint in Y' . Then the second condition is automatically satisfied: since $C \in Y'$ there is for every $i \in [k]$ a constraint that we already set aside that only differs from C in coordinate i . Therefore it suffices to find $k - 2$ variables in C , say $v_1, \dots, v_{k-2}$ (again w.l.o.g. in terms of the identity of the parts $1, \dots, k - 2$) such that their neighborhood induces a graph G with a C_4 . If no such selection exists, then for every selection $v_1, \dots, v_{k-2}$ the graph G has $O(n^{1.5})$ edges, and $|Y| = O(|X|^{k-1/2})$.

For the case $r > k$, we first partition the arguments of the constraint into k approximately equal-sized groups, with $\lfloor r/k \rfloor$ or $\lceil r/k \rceil$ arguments per group, then we reinterpret (R, S) as a k -ary promise constraint (R', S') over the product domain $D' = D^{\lceil r/k \rceil}$. It is easy to check that this operation preserves polymorphism patterns (cf. [62] and Example 11 of [21] for the non-promise case). Furthermore, performing the corresponding product operation on $I = (X, Y)$, we get a k -partite instance with $O(n^{\lceil r/k \rceil})$ variables and $|Y|$ constraints, and this operation clearly does not decrease non-redundancy. Following the analysis above, if $X = X_1 \cup \dots \cup X_k$, the non-redundancy is upper-bounded (up to ordering of arguments) by $O(\prod_{i=1}^{k-2} |X_i| \cdot (|X_{k-1}| + |X_k|)^{1.5})$, which is less than $|X|^k$ by a factor of $\min_i |X_i|^{0.5}$. $\square$

8 Conclusion

In this paper, we systematically studied the (conditional) non-redundancy of arbitrary finite-domain constraints. Earlier techniques for bounding non-redundancy have typically yielded bounds of the form $\Theta(n^k)$ for integer exponents k , but as we have proven in this paper such bounds are the exception rather than the norm, and the full non-redundancy landscape is far richer and allows all possible rational exponents. To systematically study these questions we developed a novel algebraic approach for conditional non-redundancy and proved strong connections to hypergraph Turán problems. We took a special interest in near linear non-redundancy and solved an open question in the literature [23], namely, whether the set of Boolean predicates with Mal'tsev embeddings coincides with the set of Boolean predicates which are balanced. To prove this, we introduced the Catalan terms which provide a novel and useful reinterpretation of a Mal'tsev embedding that more generally also allowed us to prove that a Mal'tsev embedding always implies a group embedding. For non-Boolean domains we showed that Abelian and Mal'tsev embeddings differ and gave the first example of a non-Abelian embedding via the Pauli group. Currently, we do not know whether Mal'tsev embeddings capture all predicates with linear non-redundancy, but we now know that if this is the case then it is barely true. In particular, we constructed an explicit family of ternary predicates provably lacking Mal'tsev embeddings whose non-redundancy approaches linear. We now conclude with a few exciting directions for future investigation.

Mal'tsev embeddings and linear redundancy. The central open question in the field of non-redundancy currently is a classification of which predicates have (near) linear non-redundancy. One potential next step toward this question is to unify concepts of infinite and finite Mal'tsev embeddings.

Conjecture 8.1. *Any predicate over a finite domain with an infinite Mal'tsev embedding also has a finite Mal'tsev embedding. In particular, any predicate with an infinite Mal'tsev embedding has linear non-redundancy.*

In particular, we hope that the techniques involving Catalan terms in Section 6 could lead to solid progress toward this conjecture. Conversely, one can hope that (in)finite Mal'tsev embeddings capture all predicates with linear non-redundancy. However, to establish such a fact, one needs to first be able to analyze predicates such as $\text{BCK} := \{111, 222, 012, 120, 201\}$ from [9] and independently discovered in a “Booleanized” form by many others [61, 23, 54], and the family of CYC_m^* predicates constructed in Section 4.3. See Section 7.1 of [14] for a more thorough discussion of such predicates. Note that $u_k^{k-1} \in \text{pPol}(\text{BCK})$ for all $k \geq 2$, so any non-redundancy lower bound for BCK will necessarily require techniques beyond those studied in this paper.

We also leave as an open question whether there is a polynomial-time algorithm (where the input is the truth table of the predicate) for determining whether a predicate satisfies the Catalan identities. We note that tools such as SMT solvers like Z3 [30] can assist in identifying partial polymorphisms, but the worst-case behavior of such techniques is likely exponential time.

Non-redundancy of Boolean languages. Is there a Boolean relation with fractional NRD exponent?

As we have seen, for relations over arbitrary finite domains the NRD exponent can be any fractional number $p/q \geq 1$, with an infinite number of exponents occurring even for arity 3. However, no such relation is known in the Boolean domain, where up to arity 3 all relations have integral NRD exponents [23, 54]. Boolean relations are also more well-behaved than general languages in other ways, e.g., [23, 54] showed that for every $R \subseteq \{0, 1\}^r$, either R fgpp-defines OR_r and

$\text{NRD}(R, n) = \Theta(n^r)$, or $\text{NRD}(R, n) = \mathcal{O}(n^{r-1})$ (in fact, R is captured as the roots of a polynomial of degree $r - 1$). Again, this is false for general languages even with $r = 3$ and domain $|D| = 3$, cf. R_6 of Section 4.2. There is also the aforementioned fact that all Boolean relations with Mal'tsev embeddings have affine embeddings, which is again false over general domains. Does the structural simplicity of Boolean languages stretch so far that all Boolean languages have integral NRD exponents? It seems unlikely, but we cannot exclude it at the moment. A similar consequence was shown to hold for Boolean MaxCSP [48].

One particularly interesting case is the Booleanization $\text{BCK}_{\mathbb{B}} \subset \{0, 1\}^9$ of the BCK predicate (discussed in [23, 61, 54, 14]). Using a SAT solver, an SMT solver [30], and Lemma 7.25 we have verified that $\text{BCK}_{\mathbb{B}}$ is preserved by u_k^c for every $k/c > 1$, hence there are no simple lower bounds on its non-redundancy. However, since it fgpp-defines BCK it also does not satisfy the Catalan conditions and does not have a Mal'tsev embedding.

Algorithmic aspects of non-redundancy. An important open question raised in the work of Carbonnel [21] is whether for every predicate P , every instance of $\text{CSP}(P)$ on n variables can be efficiently kernelized to $\mathcal{O}(\text{NRD}(P, n))$ clauses. As observed by Brakensiek and Guruswami [14], answering such a question is a necessary step toward efficiently constructing optimal-sized sparsifiers for all CSPs. Note that Appendix A.1 resolves Carbonnel's conjecture in a special case.

An important observation to keep in mind is that *local consistency* methods such as arc consistency [22, 4], the Sherali-Adams hierarchy, and sum of squares (i.e., the Lasserre hierarchy) are *not* sufficient to resolve Carbonnel's open question. In particular, one cannot use a constant number of rounds of sum of squares to kernelize *random* instances of many CSPs due to strong gaps [42, 75, 78, 58] (see also Example 7.40). As such, any "universal" kernelization algorithm must combine techniques from both Mal'tsev and local consistency algorithms. In the study of the satisfiability of (promise) CSPs, understanding such a combination is a very active topic of research [17, 80, 15, 26, 81, 69, 82].

Algebra and non-redundancy. Our algebraic approach is based on a generalized form of conjunctive formulas that allows functional atoms of fixed arity c . We obtained a Galois correspondence to sets of polymorphism patterns which, curiously, turned out to be interesting algebraic objects in their own right since they can also be seen as a type of partial minion. Can strong partial minions $\text{pPol}(S, T)$ and $\text{Pattern}(S, T)$ both be defined as a form of abstract partial minion, or are partially defined objects inherently too brittle to admit such a unifying treatment?

It is worth remarking that c -fgppp-definitions are *orthogonal* to algebraic reductions from the CSP literature in the sense that they (1) allow for arbitrary guarding functions regardless of whether these are definable from the language itself but (2) are weaker in the sense that they do not allow arbitrary existential quantification. Nevertheless, this framework is algebraically well-behaved and fundamentally builds on the principle that patterns always produce functions which commute with each other. Thus, algebraically, is commutativity all you need, or are there algebraic constructions compatible with (conditional) non-redundancy which break commutativity?

Ternary predicates and Erdős box problem. As mentioned, previous works [9, 20] gave a dichotomy theorem for the non-redundancy of binary predicates. In this paper, we extended their classification to all binary conditional non-redundancy problems. A natural next step is to classify the non-redundancy of all ternary predicates. As we demonstrate in this paper, even for non-conditional non-redundancy, infinitely many exponents are possible. As such, a complete classification seems unlikely in the near future, but we highlight a few interesting directions in the study of ternary predicates.

- Is there any subset of $[1, 3]$ for which the (conditional) non-redundancy exponents of ternary predicates are dense?
- What is the non-redundancy of the CYC_m^* predicates studied in this paper?
- What is the non-redundancy of the 3LIN_G^* predicates studied in [14]? Could the non-redundancy exponents even be irrational? Intriguingly, we currently know that $\text{NRD}(3\text{LIN}_{\mathbb{F}_3}^*, n) \in [\Omega(n^{1.5}), \tilde{O}(n^{1.6})]$, which is consistent with $\text{NRD}(3\text{LIN}_{\mathbb{F}_3}^*, n) = \Theta(n^{\log_2 3})$.
- A result of Carbonnel [21] states that if P is a ternary predicate and $\text{NRD}(P, n) = o(n^3)$, then $\text{NRD}(P, n) = \mathcal{O}(\text{ex}(n, K_{2,2,2}))$ which is notoriously known as the Erdős box problem [35, 53, 41, 40, 28]. We complement Carbonnel’s result in Section 7.4 by showing that an infinite-domain ternary predicate has conditional non-redundancy asymptotic to $\text{ex}(n, K_{2,2,2})$. Does an analogous finite-domain predicate also have this property? Even more ambitiously, can we leverage the algebraic tools developed in this paper to improve the longstanding lower and upper bounds of $\Omega(n^{8/3})$ and $\mathcal{O}(n^{11/4})$, respectively for $\text{ex}(n, K_{2,2,2})$?

With all these different directions of exploration, we believe the study of non-redundancy and its related questions will be the topic of active research for years to come.

References

- [1] M. Arenas and L. Libkin. An information-theoretic approach to normal forms for relational and XML data. *Journal of the ACM*, 52(2):246–283, 2005.
- [2] P. Austrin, V. Guruswami, and J. Håstad. $(2+\varepsilon)$ -Sat Is NP-hard. *SIAM Journal on Computing*, 46(5):1554–1573, Jan. 2017.
- [3] L. Barto, J. Bulín, A. Krokhin, and J. Opršal. Algebraic Approach to Promise Constraint Satisfaction. *J. ACM*, 68(4):28:1–28:66, July 2021.
- [4] L. Barto, A. Krokhin, and R. Willard. Polymorphisms, and How to Use Them. In A. Krokhin and S. Zivny, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 1–44. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2017.
- [5] A. Basu, P. K. Kothari, Y. P. Liu, and R. Meka. Sparsifying sums of positive semidefinite matrices. In K. G. Larsen and B. Saha, editors, *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2026, Vancouver, BC, Canada, January 11-14, 2026*, pages 6042–6064. SIAM, 2026.
- [6] J. Batson, D. A. Spielman, and N. Srivastava. Twice-ramanujan sparsifiers. *SIAM Journal on Computing*, 41(6):1704–1721, 2012.
- [7] A. A. Benczúr and D. R. Karger. Approximating s - t minimum cuts in $\tilde{O}(n^2)$ time. In G. L. Miller, editor, *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 47–55. ACM, 1996.
- [8] J. Berman, P. Idziak, P. Marković, R. McKenzie, M. Valeriote, and R. Willard. Varieties with few subalgebras of powers. *Transactions of the American Mathematical Society*, 362(3):1445–1473, Oct. 2009.

- [9] C. Bessiere, C. Carbonnel, and G. Katsirelos. Chain Length and CSPs Learnable with Few Queries. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(02):1420–1427, Apr. 2020.
- [10] C. Bessiere, R. Coletta, E. Hebrard, G. Katsirelos, N. Lazaar, N. Narodytska, C.-G. Quimper, and T. Walsh. Constraint acquisition via partial queries. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
- [11] C. Bessiere and F. Koriche. Non learnability of constraint networks with membership queries. Technical report, Technical report, Coconut, Montpellier, France, 2012.
- [12] M. Bodirsky. *Complexity of Infinite-Domain Constraint Satisfaction*. Cambridge University Press, 2021.
- [13] J. Brakensiek and V. Guruswami. Promise Constraint Satisfaction: Algebraic Structure and a Symmetric Boolean Dichotomy. *SIAM Journal on Computing*, 50(6):1663–1700, Jan. 2021.
- [14] J. Brakensiek and V. Guruswami. Redundancy is all you need. In M. Koucký and N. Bansal, editors, *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, Prague, Czechia, June 23–27, 2025*, pages 1614–1625. ACM, 2025.
- [15] J. Brakensiek, V. Guruswami, M. Wrochna, and S. Živný. The Power of the Combined Basic Linear Programming and Affine Relaxation for Promise Constraint Satisfaction Problems. *SIAM Journal on Computing*, 49(6):1232–1248, Jan. 2020.
- [16] A. Bulatov and V. Dalmau. A Simple Algorithm for Mal’tsev Constraints. *SIAM Journal on Computing*, 36(1):16–27, Jan. 2006.
- [17] A. A. Bulatov. A Dichotomy Theorem for Nonuniform CSPs. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 319–330, Oct. 2017.
- [18] A. A. Bulatov and P. Jeavons. An algebraic approach to multi-sorted constraints. In *Proceedings of the 9th International Conference on Principles and Practice of Constraint Programming (CP-2003)*, pages 183–198, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
- [19] S. Burris and H. P. Sankappanavar. *A course in universal algebra*, volume 78 of *Graduate texts in mathematics*. Springer, 1981.
- [20] S. Butti and S. Živný. Sparsification of Binary CSPs. *SIAM Journal on Discrete Mathematics*, 34(1):825–842, Jan. 2020.
- [21] C. Carbonnel. On Redundancy in Constraint Satisfaction Problems. In *28th International Conference on Principles and Practice of Constraint Programming (CP 2022)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.
- [22] H. Chen. A rendezvous of logic, complexity, and algebra. *ACM SIGACT News*, 37(4):85–114, 2006.
- [23] H. Chen, B. M. P. Jansen, and A. Pieterse. Best-Case and Worst-Case Sparsifiability of Boolean CSPs. *Algorithmica*, 82(8):2200–2242, Aug. 2020.
- [24] Y. Chen, S. Khanna, and A. Nagda. Near-linear Size Hypergraph Cut Sparsifiers. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 61–72, Nov. 2020.

- [25] F. R. K. Chung, R. L. Graham, P. Frankl, and J. B. Shearer. Some intersection theorems for ordered sets and graphs. *Journal of Combinatorial Theory, Series A*, 43(1):23–37, Sept. 1986.
- [26] L. Ciardo and S. Živný. CLAP: A New Algorithm for Promise CSPs. *SIAM Journal on Computing*, 52(1):1–37, Feb. 2023.
- [27] E. F. Codd. Recent investigations in relational data base systems. In *Proceedings of the IFIP Congress*, pages 1017–1021. North-Holland, 1974.
- [28] D. Conlon, C. Pohoata, and D. Zakharov. Random multilinear maps and the Erdős box problem. *Discrete Analysis*, Sept. 2021.
- [29] H. Darwen, C. J. Date, and R. Fagin. A normal form for preventing redundant tuples in relational databases. In *Proceedings of the 15th International Conference on Database Theory (ICDT-2012)*, pages 114–126. ACM, 2012.
- [30] L. de Moura and N. Bjørner. Z3: An Efficient SMT Solver. In C. R. Ramakrishnan and J. Rehof, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340, Berlin, Heidelberg, 2008. Springer.
- [31] A. Dechter and R. Dechter. Removing redundancies in constraint networks. In *Proc. of the 6th National Conference on Artificial Intelligence (AAAI-1987)*, pages 105–109, 1987.
- [32] H. Dell and D. Marx. Kernelization of packing problems. In Y. Rabani, editor, *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 68–81. SIAM, 2012.
- [33] H. Dell and D. van Melkebeek. Satisfiability Allows No Nontrivial Sparsification unless the Polynomial-Time Hierarchy Collapses. *J. ACM*, 61(4):23:1–23:27, July 2014.
- [34] D. Djukić, V. Janković, I. Matić, and N. Petrović. *The IMO compendium: a collection of problems suggested for the International Mathematical Olympiads: 1959-2004*, volume 119. Springer, 2006.
- [35] P. Erdős. On extremal problems of graphs and generalized graphs. *Israel Journal of Mathematics*, 2(3):183–190, Sept. 1964.
- [36] L. Eriksson, J. Groven, and V. Lagerkvist. Towards single exponential time for temporal and spatial reasoning: A study via redundancy and dynamic programming. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI-2026)*, volume 40, pages 14287–14294. AAAI Press, 2026.
- [37] A. Filtser and R. Krauthgamer. Sparsification of Two-Variable Valued Constraint Satisfaction Problems. *SIAM Journal on Discrete Mathematics*, 31(2):1263–1276, Jan. 2017.
- [38] F. Fomin, D. Lokshtanov, S. Saurabh, and M. Zehavi. *Kernelization: theory of parameterized preprocessing*. Cambridge University Press, 2019.
- [39] P. Frankl. All rationals occur as exponents. *Journal of Combinatorial Theory, Series A*, 42(2):200–206, July 1986.
- [40] Z. Füredi and M. Simonovits. *The History of Degenerate (Bipartite) Extremal Graph Problems*, pages 169–264. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

- [41] A. Gordeev. Combinatorial Nullstellensatz and Turán numbers of complete r -partite r -uniform hypergraphs. *Discrete Mathematics*, 347(7):114037, July 2024.
- [42] D. Grigoriev. Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity. *Theoretical Computer Science*, 259(1):613–622, May 2001.
- [43] A. Gupta and S. Mahajan. Using amplification to compute majority with small majority gates. *computational complexity*, 6(1):46–63, Mar. 1996.
- [44] J. Hsieh, D. Z. Lee, S. Mohanty, A. Putterman, and R. Y. Zhang. Sparsifying cayley graphs on every group. In K. G. Larsen and B. Saha, editors, *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2026, Vancouver, BC, Canada, January 11-14, 2026*, pages 6029–6041. SIAM, 2026.
- [45] J. E. Humphreys. *Reflection Groups and Coxeter Groups*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1990.
- [46] A. Jambulapati, Y. P. Liu, and A. Sidford. Chaining, group leverage score overestimates, and fast spectral hypergraph sparsification. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 196–206, 2023.
- [47] B. M. P. Jansen and A. Pieterse. Optimal Sparsification for Some Binary CSPs Using Low-Degree Polynomials. *ACM Trans. Comput. Theory*, 11(4):28:1–28:26, Aug. 2019.
- [48] B. M. P. Jansen and M. Włodarczyk. Optimal polynomial-time compression for boolean max CSP. *ACM Trans. Comput. Theory*, 16(1):4:1–4:20, 2024.
- [49] P. Jonsson, V. Lagerkvist, and G. Osipov. Csps with few alien constraints. In *Proceedings of the 30th International Conference on Principles and Practice of Constraint Programming (CP-2024)*, volume 307 of *LIPIcs*, pages 15:1–15:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.
- [50] M. Kapralov, R. Krauthgamer, J. Tardos, and Y. Yoshida. Towards tight bounds for spectral sparsification of hypergraphs. In S. Khuller and V. V. Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 598–611. ACM, 2021.
- [51] D. R. Karger. Global min-cuts in RNC, and other ramifications of a simple min-cut algorithm. In V. Ramachandran, editor, *Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, 25-27 January 1993, Austin, Texas, USA*, pages 21–30. ACM/SIAM, 1993.
- [52] G. Katona. A theorem of finite sets. In *Theory of Graphs*, pages 187–207. Academic Press, New York, 1968.
- [53] N. H. Katz, E. Krop, and M. Maggioni. Remarks on the box problem. *Mathematical Research Letters*, 9(4):515–519, 2002.
- [54] S. Khanna, A. Putterman, and M. Sudan. Efficient algorithms and new characterizations for CSP sparsification. In M. Koucký and N. Bansal, editors, *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, Prague, Czechia, June 23-27, 2025*, pages 407–416. ACM, 2025.

- [55] S. Khanna, A. L. Putterman, and M. Sudan. Code Sparsification and its Applications. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Proceedings, pages 5145–5168. Society for Industrial and Applied Mathematics, Jan. 2024.
- [56] S. Khanna, A. L. Putterman, and M. Sudan. Near-optimal Size Linear Sketches for Hypergraph Cut Sparsifiers. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, Oct. 2024.
- [57] D. Kogan and R. Krauthgamer. Sketching Cuts in Graphs and Hypergraphs. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, ITCS '15, pages 367–376, New York, NY, USA, Jan. 2015. Association for Computing Machinery.
- [58] P. K. Kothari, R. Mori, R. O’Donnell, and D. Witmer. Sum of squares lower bounds for refuting any CSP. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 132–145, New York, NY, USA, June 2017. Association for Computing Machinery.
- [59] J. B. Kruskal. The number of simplices in a complex. In *Mathematical Optimization Techniques*, pages 251–278. University of California Press, Berkeley, CA, 1963.
- [60] N. Kuntewar and J. Sarma. Range avoidance in boolean circuits via turan-type bounds. *CoRR*, abs/2503.17114, 2025.
- [61] V. Lagerkvist and M. Wahlström. Sparsification of SAT and CSP Problems via Tractable Extensions. *ACM Transactions on Computation Theory*, 12(2):1–29, June 2020.
- [62] V. Lagerkvist and M. Wahlström. The (Coarse) Fine-Grained Structure of NP-Hard SAT and CSP Problems. *ACM Trans. Comput. Theory*, 14(1):2:1–2:54, Dec. 2021.
- [63] J. H. Lee, S. Li, Z. Long, and M. Sioutis. On redundancy in simple temporal networks. In *Proc. 22nd European Conference on Artificial Intelligence (ECAI-2016)*, pages 828–836, 2016.
- [64] J. R. Lee. Spectral hypergraph sparsification via chaining. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 207–218, 2023.
- [65] A. Levy and Y. Sagiv. Constraints and redundancy in datalog. In *Proceedings of the Eleventh ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, PODS ’92, page 67–80, New York, NY, USA, 1992. Association for Computing Machinery.
- [66] S. Li, Z. Long, W. Liu, M. Duckham, and A. Both. On redundant topological constraints. *Artificial Intelligence*, 225:51–76, 2015.
- [67] P. Liberatore. Redundancy in logic I: CNF propositional formulae. *Artificial Intelligence*, 163(2):203–232, 2005.
- [68] P. Liberatore. Redundancy in logic II: 2CNF and Horn propositional formulae. *Artificial Intelligence*, 172(2-3):265–299, 2008.
- [69] M. Lichter and B. Pago. Limitations of affine integer relaxations for solving constraint satisfaction problems. In K. Censor-Hillel, F. Grandoni, J. Ouaknine, and G. Puppis, editors, *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, Aarhus, Denmark, July 8-11, 2025*, LIPIcs, pages 166:1–166:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025.

- [70] A. I. Mal'tsev. Constructive Algebras I. *Russian Mathematical Surveys*, 16(3):77, June 1961.
- [71] U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Sciences*, 7:95–132, 1974.
- [72] M. Paulin, C. Bessiere, and J. Sallantin. Automatic Design of Robot Behaviors through Constraint Network Acquisition. In *2008 20th IEEE International Conference on Tools with Artificial Intelligence*, volume 1, pages 275–282, Nov. 2008.
- [73] N. Pippenger. Galois theory for minors of finite functions. *Discret. Math.*, 254(1-3):405–419, 2002.
- [74] H. Schnoor and I. Schnoor. Partial polymorphisms and constraint satisfaction problems. In *Complexity of Constraints - An Overview of Current Research Themes [Result of a Dagstuhl Seminar]*, pages 229–254. Springer, 2008.
- [75] G. Schoenebeck. Linear Level Lasserre Lower Bounds for Certain k-CSPs. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 593–602, Oct. 2008.
- [76] T. Soma and Y. Yoshida. Spectral sparsification of hypergraphs. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2570–2581. SIAM, 2019.
- [77] D. A. Spielman and S.-H. Teng. Spectral Sparsification of Graphs. *SIAM Journal on Computing*, 40(4):981–1025, Jan. 2011.
- [78] M. Tulsiani. CSP gaps and reductions in the Lasserre hierarchy. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pages 303–312, New York, NY, USA, May 2009. Association for Computing Machinery.
- [79] J. H. Van Lint and R. M. Wilson. *A course in combinatorics*. Cambridge university press, 2001.
- [80] D. Zhuk. A Proof of the CSP Dichotomy Conjecture. *J. ACM*, 67(5):30:1–30:78, Aug. 2020.
- [81] D. Zhuk. A simplified proof of the CSP Dichotomy Conjecture and XY-symmetric operations. Apr. 2024.
- [82] D. Zhuk. Singleton algorithms for the constraint satisfaction problem. *CoRR*, abs/2509.18434, 2025.

A Omitted Details from Section 3

A.1 Connections to Kernelization

Recall that for $p \geq q$, we define $\text{SAT-DP}_{p,q}$ to be $\{\text{OR-DP}_{p,q}, \text{CUT}\}$ where $\text{OR-DP}_{p,q}$ is defined in Definition 3.4 and $\text{CUT} = \{(0, 1), (1, 0)\}$. We now work toward showing that the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ is p/q .

Lemma A.1. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$, there is a polynomial-time reduction that transforms any n -variable instance ϕ of CNF- p -SAT into an equivalent instance (V, C) of $\text{CSP}(\text{SAT-DP}_{p,q})$ with $|V| = 2n + (2n)^q$ variables.*

Proof. Consider an instance ϕ of CNF- p -SAT. It consists of a set of clauses $A_1, \dots, A_m$ over Boolean variables $x_1, \dots, x_n$, with each clause A_i being a disjunction of literals. Each literal is either a variable x_i or its negation $\neg x_i$.

We construct an equivalent instance of $\text{CSP}(\text{SAT-DP}_{p,q})$ in the following way. Let $X := \{x_i, \bar{x}_i \mid i \in [n]\}$ and $\tilde{X} := \{\tilde{x}_{\mathbf{x}} \mid \mathbf{x} \in X^q\}$. So X contains one element for each possible literal of ϕ , while $\tilde{X}$ contains one element for each q -tuple $\mathbf{x}$ of literals. We use $V = X \cup \tilde{X}$ as the set of variables for the CSP under construction. We will refer to the variables in X as the *Boolean* variables and to those in $\tilde{X}$ as the *padding* variables. Observe that $|V| = 2n + (2n)^q$, as claimed. In the remainder, it will be convenient to let $\text{var}(\ell) \in X$ denote the variable in X corresponding to a literal ℓ . So if $\ell = x_i$ is a positive literal, we have $\text{var}(\ell) = x_i$, while for negative literals $\neg x_i$ we have $\text{var}(\ell) = \bar{x}_i$.

To ensure that the variables x_i and $\bar{x}_i$ act as each others negations, we add the constraint $\text{CUT}(x_i, \bar{x}_i)$ to C for each $i \in [n]$.

We transform clauses of ϕ into constraints using the relation $\text{OR-DP}_{p,q}$, as follows. For each clause $A_j = \ell_1 \vee \dots \vee \ell_p$, where each ℓ_k is a literal of the form x_i or $\neg x_i$, we add a corresponding constraint $\text{OR-DP}_{p,q}(\mathbf{v}_j)$ to C , where $\mathbf{v}_j$ is a tuple of $p + p^q$ elements of V . The i th value of $\mathbf{v}_j$, denoted $\mathbf{v}_j[i]$, is defined as follows:

- if $i \in [p]$, then $\mathbf{v}_j[i] = \text{var}(\ell_i)$.
- if $i > p$, then $\mathbf{v}_j[i] = \tilde{x}_{\mathbf{x}}$, where $\mathbf{x} \in X^q$ is the q -tuple of Boolean variables defined as follows. Let $I := (i_1, \dots, i_q) = \text{id}_{X,p,q}(i - p)$ be the tuple of q indices in the range $[p]$ that is labeled $i - p$ by the chosen bijection $\text{id}_{X,p,q}$. Then we define:

$$\mathbf{x} = (\text{var}(\ell_{i_1}), \dots, \text{var}(\ell_{i_q})). \quad (21)$$

So the tuple of variables to which the constraint associated to clause A_j is applied is computed as follows: the first p variables are Boolean variables corresponding to the literals occurring in A_j ; the remaining variables are padding variables. The padding variable $\tilde{x}_{\mathbf{x}}$ used at position $p + i$ is the one that was created for the q -tuple of literals that appear in clause A_j at the positions encoded by $\text{id}_{X,p,q}(i)$.

This concludes the description of the $\text{CSP}(\text{SAT-DP}_{p,q})$ -instance (V, C) , which can easily be done in polynomial time for fixed p and q . It remains to argue that the Boolean formula ϕ is satisfiable if and only if (C, V) is satisfiable. We first argue the converse. If $\sigma: V \rightarrow D_{p,q}$ is an assignment that satisfies all constraints of C , then its restriction to the original variables $\{x_1, \dots, x_n\}$ satisfies all clauses of ϕ . To see this, consider an arbitrary clause A_j of ϕ . Since σ satisfies all constraints of C , in particular it satisfies the constraint $\text{OR-DP}_{p,q}(\mathbf{v}_j)$ that was inserted into C on account of clause A_j . By construction, the first p variables of $\mathbf{v}_j$ correspond to the literals in A_j . By definition

of A_j , the constraint $\text{OR-DP}_{p,q}(\mathbf{v}_j)$ can only be satisfied if the values of the first p variables of $\mathbf{v}_j$ are Boolean, and at least one of them is 1. Hence there is a literal ℓ_i in clause A_j such that the corresponding variable (either x_i or $\bar{x}_i$) in X has value 1 under σ . If ℓ_i is positive, this directly shows that the restriction of σ to $\{x_1, \dots, x_n\}$ satisfies clause A_j . If ℓ_i is a negative literal of the form $\neg x_k$, then we know $\sigma(\bar{x}_k) = 1$. Since σ also satisfies the constraint $(\text{CUT}, (x_k, \bar{x}_k))$ that was inserted into C , we find $\sigma(x_k) = 0$. This again guarantees that the restriction of σ to $\{x_1, \dots, x_n\}$ satisfies A_j . As this argument applies for an arbitrary clause C_j , the entire formula ϕ is indeed satisfied by the stated assignment.

To conclude the proof, we establish correctness in the converse direction: if ϕ has a satisfying assignment $f: \{x_1, \dots, x_n\} \rightarrow \{0, 1\}$, then (C, V) has a satisfying assignment $\sigma: V \rightarrow D_{i,j}$. The assignment σ is derived from f as follows.

- For each positive Boolean variable $x_i \in V \cap X$ we set $\sigma(x_i) = f(x_i)$.
- For each negative Boolean variable $\bar{x}_i \in V \cap X$ we set $\sigma(\bar{x}_i) = 1 - f(x_i)$.
- For each padding variable $\tilde{x}_{\mathbf{x}}$ where $\mathbf{x} = (x_{i_1}, \dots, x_{i_q}) \in X^q$, the value $\sigma(\tilde{x}_{\mathbf{x}})$ is the bitstring $(f(x_{i_1}) \dots f(x_{i_q}))$.

Hence the value of a padding variable $\tilde{x}_{\mathbf{x}}$ that was created on account of a tuple of q Boolean variables is simply the bitstring obtained by concatenating the f -values of these Boolean values. From these definitions, it is straightforward to verify that σ satisfies (V, C) . Since each clause of ϕ is satisfied by f , in each constraint of the form $\text{OR-DP}_{p,q}(\mathbf{v}_j)$ created for a clause A_j , one of the first p Boolean variables will be set to 1 since f satisfies a literal of A_j . Since the values of the padding variables appearing in the rest of the constraint are defined precisely as the bitstrings of the values whose positions they control, the remaining conditions of Definition 3.4 are also satisfied. This concludes the proof of Lemma A.1. $\square$

By combining the reduction from in the previous lemma with known kernelization lower bounds for CNF- p -SAT due to Dell and van Melkebeek [33], we obtain kernelization lower bounds for $\text{CSP}(\text{SAT-DP}_{p,q})$. We introduce some terminology that allows us to draw this connection. In the following, a *parameterized problem* Q is a subset of $\Sigma^* \times \mathbb{N}_+$, where Σ is a finite alphabet. When applying tools of parameterized complexity to our study of CSPs, we use the number of variables n in the CSP as the parameter k of the parameterized problem.

Definition A.2. Let $Q, Q' \subseteq \Sigma^* \times \mathbb{N}_+$ be parameterized problems and let $h: \mathbb{N}_+ \rightarrow \mathbb{N}_+$ be a computable function. A *generalized kernelization for Q into Q' of size $h(k)$* is an algorithm that, on input $(x, k) \in \Sigma^* \times \mathbb{N}_+$, takes time polynomial in $|x| + k$ and outputs an instance (x', k') such that:

1. $|x'|$ and k' are bounded by $h(k)$, and
2. $(x', k') \in Q'$ if and only if $(x, k) \in Q$.

The algorithm is a *polynomial generalized kernelization* if $h(k)$ is a polynomial.

The adjective *generalized* corresponds to the fact that the algorithm is allowed to map instances from the source problem Q into equivalent instances of a different (but fixed) decision problem Q' . Using this terminology, the lower bound by Dell and van Melkebeek can be stated as follows.

Theorem A.3 ([33]). *For each integer $p \geq 3$ and $\varepsilon > 0$, the CNF- p -SAT problem parameterized by the number of variables n does not admit a generalized kernelization of size $\mathcal{O}(n^{p-\varepsilon})$ unless $\text{NP} \subseteq \text{coNP/poly}$.*

In this theorem, and in Definition A.2, the size of the instances is measured in terms of the number of symbols over alphabet Σ , i.e., in terms of the number of bits needed to encode the instance. Our definition of the size of a kernelization for $\text{CSP}(\Gamma)$ in the introduction was measuring the number of *constraints* in the reduced instance. Notice that in our setting of CSPs over finite constraint languages consisting of finite relations, the difference between these two types of measurements is small: an instance that is encoded in $f(n)$ bits has at most $f(n)$ constraints, while an instance of $f(n)$ constraints can be encoded in $\mathcal{O}_\Gamma(f(n) \log n)$ bits by encoding, for each constraint, which relation $R \in \Gamma$ it uses ($\mathcal{O}_\Gamma(1)$ bits) and to which variables the constraint is applied ($\mathcal{O}_R(1)$ variables, each encoded in $\mathcal{O}(\log n)$ bits). Since our notion of kernelization exponent is oblivious to $\log n$ factors, when considering the kernelization exponent of nonuniform CSPs it does not matter whether we measure kernelization size in bits or constraints. With this knowledge, we present the lower bound in the following lemma.

Lemma A.4. *Assuming $\text{NP} \not\subseteq \text{coNP/poly}$, for each fixed $p, q \in \mathbb{N}$ with $p \geq q$ and $p \geq 3$, the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ is at least $\frac{p}{q}$.*

Proof. Assume for a contradiction that the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ is less than $\frac{p}{q}$. It follows that there exists a positive real ε such that $\text{CSP}(\text{SAT-DP}_{p,q})$ has a kernelization that reduces to equisatisfiable instances with $\mathcal{O}(n^{\frac{p}{q}-\varepsilon})$ constraints.

By Theorem A.3, to contradict the assumption $\text{NP} \not\subseteq \text{coNP/poly}$ it suffices to leverage Lemma A.1 together with the assumed kernelization for $\text{CSP}(\text{SAT-DP}_{p,q})$ into a generalized kernelization for CNF- p -SAT of bitsize $\mathcal{O}(n^{p-\delta})$ for some $\delta > 0$. This can be done as follows.

Given an n -variable instance ϕ of CNF- p -SAT, the generalized kernelization transforms it into an equivalent instance (V, C) of $\text{CSP}(\text{SAT-DP}_{p,q})$ on $|V| = 2n + (2n)^q$ variables by invoking Lemma A.1. It then applies the assumed kernelization to (V, C) , resulting in an equivalent instance (V', C') on at most $\mathcal{O}(|V|^{\frac{p}{q}-\varepsilon}) \leq \mathcal{O}(n^q)^{\frac{p}{q}-\varepsilon} \leq \mathcal{O}(n^{p-q\varepsilon})$ constraints. Since variables not occurring in any constraint can be removed without changing the answer to the instance, while the arity of the constraints is $\mathcal{O}(1)$ for fixed p and q , we may assume without loss of generality that $|V'| \leq \mathcal{O}(|C'|) \leq \mathcal{O}(n^{p-q\varepsilon})$. Hence the identity of a single variable can be encoded in $\log_2 \mathcal{O}(n^{p-q\varepsilon}) \leq \mathcal{O}(\log_2 n)$ bits; here we use the fact that $\log_2(n^{p-q\varepsilon}) = (p - q\varepsilon) \cdot \log_2(n)$ while $p \in \mathcal{O}(1)$. As the arity of each constraint is constant, each constraint of the instance (V', C') can be encoded in $\mathcal{O}(\log n)$ bits by listing the binary representations of the indices of all the variables occurring in the constraint. Hence instance (V', C') can be encoded in $\mathcal{O}(n^{p-q\varepsilon} \log n)$ bits and is equivalent to the CNF- p -SAT instance ϕ . Since both transformation steps can be carried out in polynomial time, this results in a generalized kernelization for CNF- p -SAT of bitsize $\mathcal{O}(n^{p-q\varepsilon} \log n)$. Since $\log n \in \mathcal{O}(n^\beta)$ for each $\beta > 0$, this shows that CNF- p -SAT has a generalized kernelization of bitsize $\mathcal{O}(n^{p-q\varepsilon+\frac{q\varepsilon}{2}})$, so of size $\mathcal{O}(n^{p-\delta})$ for $\delta = \frac{q\varepsilon}{2} > 0$. But this implies $\text{NP} \subseteq \text{coNP/poly}$ via the results of Dell and van Melkebeek [33]. $\square$

Note that while the statement of Lemma A.4 requires $p \geq 3$, for any rational number $\frac{p}{q} \geq 1$ we can write it as a ratio $\frac{3p}{3q}$ of numbers that are at least three. We now prove the corresponding upper bounds. The manner in which we use the Kruskal-Katona theorem to bound the size of the instance after preprocessing is very similar to the proof of Lemma 3.10, but additional effort is needed to make the proof algorithmic and avoid the assumption of having multipartite instances.

Lemma A.5. *For each fixed $p, q \in \mathbb{N}$ with $p \geq q$, there is a polynomial-time kernelization algorithm that transforms any n -variable instance (V, C) of $\text{CSP}(\text{SAT-DP}_{p,q})$ into an equivalent instance (V', C') with $|C'| \leq \mathcal{O}_{p,q}(n^{\frac{p}{q}})$. Hence the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ is at most $\frac{p}{q}$.*

Proof. Given an instance (V, C) of $\text{CSP}(\text{SAT-DP}_{p,q})$ on $n := |V|$ variables, the kernelization consists of two phases aimed at reducing the number of constraints over CUT and over OR-DP $_{p,q}$, respectively. For the first phase, it is easy to reduce the number of constraints over CUT to $\mathcal{O}(n)$. Consider the graph G on vertex set V which has an edge for each constraint over CUT. If this graph is not bipartite, then since each edge corresponds to a CUT constraint that requires the variables represented by its endpoints to have distinct Boolean values, the instance is unsatisfiable. It then suffices to output a constant-size unsatisfiable instance as the result of the kernelization. If G is bipartite, then we compute a spanning forest F of G , which contains a spanning tree of each connected component. Remove all CUT constraints that do not correspond to an edge in F . We argue that this does not change the satisfiability of the instance, as follows. For each removed constraint $\text{CUT}(x, x')$, there is a path P between x and x' in the spanning forest F . The path P has an odd number of edges, as otherwise it would make an odd cycle together with the edge $\{x, x'\}$ which would violate the assumption that G is bipartite. As each edge on P corresponds to a CUT constraint that we preserve in the instance, any satisfying assignment to the remaining constraints must alternate the assigned Boolean true/false value for each variable on path P . As the number of edges on P is odd, to satisfy all cut constraints from P the value assigned to x must differ to that of x' . Hence any omitted constraint $\text{CUT}(x, x')$ must be satisfied by any assignment that satisfies the remaining clauses. Since a spanning forest for a graph on $|V|$ vertices has less than $|V|$ edges and can be computed efficiently, the first phase of the kernelization can be executed in polynomial time and reduces the number of CUT constraints to $\mathcal{O}(n)$.

For the second phase of the algorithm, which is the most difficult, the goal is to reduce the number of OR-DP $_{p,q}$ constraints. As an opening step for phase two, we partition the variables of the instance into types: *Boolean* variables (that occur in a constraint over CUT, or that occur in one of the first p positions of a constraint over OR-DP $_{p,q}$) and *padding* variables (that occur in a constraint over OR-DP $_{p,q}$ at a position after p). If these two sets are not disjoint, then the choice of OR-DP $_{p,q}$ ensures that even the subinstance consisting of two clauses that have this conflicting use of the variable, is unsatisfiable. Hence we may output such a subinstance as the result of the kernelization and terminate. In the remainder, we may assume the variables are partitioned into Boolean and padding variables.

To reduce the number of constraints of type OR-DP $_{p,q}$ we apply two simple reduction rules based on the semantics of the OR-DP $_{p,q}$ predicate, exploiting the fact that the values of padding variables in such constraints uniquely determine the values of the Boolean variables on the positions they govern. We exhaustively apply the following two reduction rules.

1. Suppose there are indices $i, i' \in [p^q]$ for which the instance contains two (not necessarily distinct) constraints

$$\begin{aligned} y &= \text{OR-DP}_{p,q}(x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q}), \\ y' &= \text{OR-DP}_{p,q}(x'_1, \dots, x'_p, \tilde{x}'_1, \dots, \tilde{x}'_{p^q}), \end{aligned}$$

such that $\tilde{x}_i = \tilde{x}'_{i'}$ but for the associated tuples $(t_1, \dots, t_q) = \text{id}_{\mathbf{x}_{p,q}}(i)$ and $(t'_1, \dots, t'_q) = \text{id}_{\mathbf{x}_{p,q}}(i')$ we have $(x_{t_1}, \dots, x_{t_j}) \neq (x'_{t'_1}, \dots, x'_{t'_j})$. In other words, there are two constraints that use the same padding variable $\tilde{x}_i = \tilde{x}'_{i'}$, but the tuples of Boolean variables they control

differ. Let $j \in [q]$ such that $x_{t_j} \neq x'_{t'_j}$. We now simplify the instance as follows: replace all occurrences of variable $x'_{t'_j}$ in the CSP by x_{t_j} and remove variable $x'_{t'_j}$. It is clear that any satisfying assignment of the reduced instance yields a satisfying assignment to the original, by simply assigning the eliminated variable $x'_{t'_j}$ the same value as x_{t_j} . Due to the semantics of $\text{OR-DP}_{p,q}$ (Property 1 in Definition 3.4) the other direction holds as well: any satisfying assignment to the original instance must assign x_{t_j} and $x'_{t'_j}$ the same value, since the value of both variables must match the j -th bit in the bitstring assigned as the value for $\tilde{x}_i = \tilde{x}'_i$. Hence this replacement preserves the satisfiability of the instance while strictly reducing the number of variables. Intuitively, this reduction rule ensures that each q -tuple of Boolean variables that occurs in some constraint has a *unique* padding variable associated to it, which we will exploit in the analysis below.

2. Suppose there is an index $i \in [p^q]$ for which the instance contains two constraints

$$\begin{aligned} y &= \text{OR-DP}_{p,q}(x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q}), \\ y' &= \text{OR-DP}_{p,q}(x'_1, \dots, x'_p, \tilde{x}'_1, \dots, \tilde{x}'_{p^q}), \end{aligned}$$

in which the padding variables $\tilde{x}_i, \tilde{x}'_i$ used on the i -th padding position are distinct, but in which the tuples of Boolean variables appearing at positions $(t_1, \dots, t_q) := \text{id}_{\mathbf{x}_{p,q}}(i)$ are identical, so that $\tilde{x}_i \neq \tilde{x}'_i$ while $(x_{t_1}, \dots, x_{t_q}) = (x'_{t_1}, \dots, x'_{t_q})$. Then we do the following: replace all occurrences of padding variable $\tilde{x}'_i$ by padding variable $\tilde{x}_i$ and then remove $\tilde{x}'$ from the instance. Due to the semantics of $\text{OR-DP}_{p,q}$ (Property 1 in Definition 3.4), any satisfying assignment to the original instance must give $\tilde{x}_i$ and $\tilde{x}'_i$ the same value since the value of both variables is the bitstring of values assigned to the q Boolean variables $(x_{t_1}, \dots, x_{t_q}) = (x'_{t_1}, \dots, x'_{t_q})$. Hence this replacement preserves the satisfiability of the instance.

Exhaustive application of this reduction rule ensures that for each p -tuple of Boolean variables, there is at most one constraint over $\text{OR-DP}_{p,q}$ in which that exact p -tuple appears on the first p positions.

The instance (V', C') that is obtained from exhaustively applying these reduction rules is returned as the output of the kernelization algorithm. Since all preprocessing steps preserve the satisfiability of the instance, it is equisatisfiable to (V, C) . As the reduction rules can easily be implemented to run in polynomial time, it remains to argue that the number of constraints $|C'|$ in the reduced instance is bounded by $\mathcal{O}_{p,q}(n^{\frac{p}{q}})$. Since the number of constraints over CUT has been reduced to $\mathcal{O}(n)$ by the first phase, it suffices to bound the constraints $C^* \subseteq C'$ over $\text{OR-DP}_{p,q}$. Note that $|V'| \leq |V| = n$ since we never introduce new variables. Now we set up an application of the Kruskal-Katona theorem to bound the number of constraints $|C^*|$ in terms of the number of variables $|V'|$.

The set family $\mathcal{A}$ to which we apply the Kruskal-Katona theorem is defined over the universe $V' \times [p]$. For each constraint $y = \text{OR-DP}_{p,q}(x_1, \dots, x_p, \tilde{x}_1, \dots, \tilde{x}_{p^q})$ in C^* , we add a set $A_y = \{(x_1, 1), (x_2, 2), \dots, (x_p, p)\}$ to $\mathcal{A}$. Since all pairs in the set have different second coordinates, the resulting set family $\mathcal{A}$ is p -uniform. Let $\mathcal{B}$ be the q -shadow of $\mathcal{A}$, i.e., it contains each size- q subset $B \subseteq V' \times [p]$ for which B is contained in some set A_y of $\mathcal{A}$. Next, we relate $|\mathcal{A}|$ to $|C^*|$ and $|\mathcal{B}|$ to $|V'|$.

Observe first that $|C^*| = |\mathcal{A}|$. This follows from the fact that each constraint $y \in C^*$ contributes a set A_y to $\mathcal{A}$, while no two distinct constraints can contribute the same set: that would mean there are two distinct constraints over $\text{OR-DP}_{p,q}$ in which the *same* p -tuple of variables appears on the first p positions, which triggers the second reduction rule.

Next, we will argue that $|\mathcal{B}| \leq |V'| \cdot p^q$ by relating sets in the q -shadow to padding variables. To establish this we provide an injection $f: \mathcal{B} \rightarrow V' \times [p^q]$, where the Cartesian product with $[p^q]$ plays the role of the reduction to a multipartite instance in Lemma 3.10. The injection f is defined as follows. Consider a q -set $B = \{(x_{t_1}, t_1), \dots, (x_{t_q}, t_q)\} \in \mathcal{B}$ and label its elements such that $t_1 < \dots < t_q$. Let $S_y \in \mathcal{A}$ be a set that contains B and consider the constraint $y = \text{OR-DP}_{p,q}(x'_1, \dots, x'_p, \tilde{x}'_1, \dots, \tilde{x}'_{p^q}) \in C^*$ for which S_y was added to $\mathcal{A}$. Since B is a subset of S_y , while the pairs over $V' \times [p]$ encode the positions at which Boolean variables appear in constraints of C^* , we have $x_{t_j} = x'_{t_j}$ for each $j \in [q]$: for each pair $(x, j) \in B$ consisting of a Boolean variable x and a position $j \in [p]$, the j -th Boolean variable in constraint y is x . Let $i := \text{id}_{x_{p,q}}^{-1}(t_1, \dots, t_q) \in [p^q]$ be the index of the q -tuple of positions represented by B . Then we define $f(B) := (\tilde{x}_i, i)$, that is, we map set B to the pair consisting of i and the padding variable that occurs in constraint y at the position i that controls the q -tuple of variables in B .

Next, we argue that f is indeed an injection. Suppose for a contradiction that two distinct sets $B, B' \in \mathcal{B}$ map to the same pair $(\tilde{x}, i)$; let these be $B = \{(x_{t_1}, t_1), \dots, (x_{t_q}, t_q)\}$ and $B' = \{(x'_{t'_1}, t'_1), \dots, (x'_{t'_q}, t'_q)\}$ labeled with $t_1 < \dots < t_q$ and $t'_1 < \dots < t'_q$. Since $f(B) = f(B') = (\tilde{x}, i)$, they agree on their second coordinate which implies that $i = \text{id}_{x_{p,q}}(t_1, \dots, t_q) = \text{id}_{x_{p,q}}(t'_1, \dots, t'_q)$. Hence the q -sets B, B' are obtained from p -sets $A_y, A_{y'}$ by selecting the Boolean variables from the *same* positions $(t_1, \dots, t_q) = (t'_1, \dots, t'_q)$ of the respective constraints y, y' . Since the sets B, B' are distinct while their projection onto the second element of the pairs yields the same set $\{t_1, \dots, t_q\} = \{t'_1, \dots, t'_q\}$ of indices, there is an index $j \in [q]$ for which $(x_{t_j}, t_j) \neq (x'_{t'_j}, t'_j)$ with $t_j = t'_j$. Consequently, $x_{t_j} \neq x'_{t'_j}$ are *distinct* Boolean variables in clauses y, y' , that appear on a position t_j controlled by a padding variable with index i , and the two constraints y, y' use the *same* padding variable $\tilde{x}$ at position i . But this contradicts the assumption that the first reduction rule is exhaustively applied. We conclude that f is indeed an injection.

We now use these bounds to finish the proof. As in the proof of Lemma 3.10, let $c_{p,q} \geq 1$ be any constant such that $\binom{m}{p} \leq c_{p,q} \cdot \binom{m}{q}^{\frac{p}{q}}$. Since f is an injection from $\mathcal{B}$ to $V' \times [p^q]$ we infer $|\mathcal{B}| \leq |V'| \cdot p^q$. Let $m \geq p$ be the largest integer such that $|\mathcal{A}| \geq \binom{m}{p}$, so that $|\mathcal{A}| < \binom{m+1}{p} \leq 2p \binom{m}{p}$. By the Kruskal-Katona theorem (Theorem 3.8) we have $|V'| \cdot p^q \geq |\mathcal{B}| \geq \binom{m}{q}$. Hence we can now derive:

$$|C^*| = |\mathcal{A}| \leq 2p \cdot \binom{m}{p} \leq 2p \cdot c_{p,q} \cdot \binom{m}{q}^{\frac{p}{q}} \leq 2p \cdot c_{p,q} \cdot (|V'| \cdot p^q)^{\frac{p}{q}} \leq \mathcal{O}_{p,q}(n^{\frac{p}{q}}).$$

As this directly bounds the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$, it concludes the proof of lemma A.5. $\square$

By combining the upper- and lower bounds, we now derive the main result of this section.

Theorem A.6. *Let $p \geq q \in \mathbb{N}$ such that $p \geq 3$. Assuming $\text{NP} \not\subseteq \text{coNP}/\text{poly}$, the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ is $\frac{p}{q}$. Hence every rational number of value at least one is the kernelization exponent of some CSP.*

Proof. For $p \geq q \in \mathbb{N}$ with $p \geq 3$, Lemma A.4 shows that under the assumption $\text{NP} \not\subseteq \text{coNP}/\text{poly}$ the kernelization exponent of $\text{CSP}(\text{SAT-DP}_{p,q})$ must be at least $\frac{p}{q}$, while Lemma A.5 gives a matching upper bound. Hence the kernelization exponent is $\frac{p}{q}$. As every fraction $\frac{p'}{q'} \geq 1$ of integers can be written as $\frac{3p'}{3q'}$ satisfying this premise, every rational at least one is the kernelization exponent of some CSP. $\square$

A.2 A Construction of Lower Arity

In this portion of the appendix, we prove Theorem 3.13. To help with understanding the main proof technique, we first prove a special case.

A.2.1 Warm-up Predicate

As a warmup, we consider $p = 3$, $q = 2$ and $\mathcal{F} = \{\{1, 2\}, \{2, 3\}, \{3, 1\}\}$. It is clear that $\mathcal{F}$ is 2/3-regular. In that case, we have that the domain $D := E_{3,2}$ is $\{00, 01, 10, 11\}$ and the relation $\text{DP}_{\mathcal{F}}$ being

$$\text{DP}_{\mathcal{F}} := \{(00, 00, 00), (00, 01, 10), (01, 10, 00), (01, 11, 10), \\ (10, 00, 01), (10, 01, 11), (11, 10, 01), (11, 11, 11)\}.$$

With $\text{OR-DP}_{\mathcal{F}}$ being the same relation except we omit $(00, 00, 00)$.

Remark A.7. An fgpp-equivalent formulation of $\text{OR-DP}_{\mathcal{F}}$ is $D := \{0, 1, 2, 3\}$ and

$$R := \{012, 120, 201, 321, 213, 132, 333\}.$$

Using an argument similar to that of Lemma 3.11, we can prove that $\text{NRD}(\text{DP}_{\mathcal{F}}, n) = \mathcal{O}(n)$. As such, it suffices understand the asymptotics of $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n)$. We begin with a lower bound.

Lemma A.8. $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n) \geq \Omega(n^{1.5})$.

Proof. Assume that $n = 3t^2$. Let $X := [3] \times [t]^2$. Define $Y \subseteq X^3$ as follows

$$Y := \{e_{a,b,c} := ((1, a, b), (2, b, c), (3, c, a)) \in X^3 : a, b, c \in [t]\}.$$

Obviously, $|Y| = t^3$, so it suffices to prove that (X, Y) is conditionally non-redundant. Fix $(a, b, c) \in [t]^3$ and define $\sigma_{a,b,c} : X \rightarrow D$ as follows

$$\begin{aligned} \sigma_{a,b,c}(1, a', b') &:= \begin{cases} 00 & a = a' \wedge b = b' \\ 01 & a = a' \wedge b \neq b' \\ 10 & a \neq a' \wedge b = b' \\ 11 & a \neq a' \wedge b \neq b'. \end{cases} \\ \sigma_{a,b,c}(2, b', c') &:= \begin{cases} 00 & b = b' \wedge c = c' \\ 01 & b = b' \wedge c \neq c' \\ 10 & b \neq b' \wedge c = c' \\ 11 & b \neq b' \wedge c \neq c'. \end{cases} \\ \sigma_{a,b,c}(3, c', a') &:= \begin{cases} 00 & c = c' \wedge a = a' \\ 01 & c = c' \wedge a \neq a' \\ 10 & c \neq c' \wedge a = a' \\ 11 & c \neq c' \wedge a \neq a'. \end{cases} \end{aligned}$$

One can check that for every $(a', b', c') \in [t]^3$ we have that $\sigma_{a,b,c}(e_{a',b',c'}) \in \text{DP}_{\mathcal{F}}$ and $\sigma_{a,b,c}(e_{a',b',c'}) = (00, 00, 00)$ if and only if $(a, b, c) = (a', b', c')$. $\square$

Now we prove an essentially matching upper bound.

Lemma A.9. $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n) = \mathcal{O}(n^{1.5})$.

Proof. Let (X, Y) be a conditionally non-redundant instance. By Lemma 2.3, we may assume that the instance is tripartite. That is, $X := X_1 \sqcup X_2 \sqcup X_3$ with $Y \subseteq X_1 \times X_2 \times X_3$. For each $y \in Y$, let $\sigma_y : X \rightarrow D$ be an assignment such that $\sigma_y(y) = (00, 00, 00)$ and $\sigma_y(y') \in \text{OR-DP}_{\mathcal{F}}$ for all $y' \in Y \setminus \{y\}$. Since no $t \in \text{OR-DP}_{\mathcal{F}}$ has two coordinates equal to 00, we must have that Y is linear (i.e., any two clauses have at most one variable in common).

For distinct $i, j \in [3]$, let $Y_{ij} := \{(y_i, y_j) : y \in Y\} \subseteq X_i \times X_j$. Since Y is linear, we have that each Y_{ij} has the same cardinality as Y . We say that $x, x' \in X_i$ are j -equivalent, denoted by $x \sim_j x'$ if x and x' are in the same connected component of Y_{ij} . A crucial observation is as follows.

Claim A.10. *Let $x, x' \in X_i$ be such that $x \sim_{i+1} x'$ and $x \sim_{i+2} x'$, then for every $\text{DP}_{\mathcal{F}}$ -satisfying assignment $\sigma : X \rightarrow D$ for Y , we have that $\sigma(x) = \sigma(x')$.*

Proof. Note that for any $i \in [3]$, we have that $\text{DP}_{\mathcal{F}}|_{i,i+1} = \{(00, 00), (00, 01), (10, 00), (10, 01)\} \cup \{(01, 10), (01, 11), (11, 10), (11, 11)\}$, where indices are taken modulo 3. Thus, if $x, x' \in X_i$ satisfy $x \sim_{i+1} x'$, then $(\sigma(x), \sigma(x')) \in \{00, 10\}^2 \cup \{01, 11\}^2$ for any $\text{DP}_{\mathcal{F}}$ -satisfying assignment σ .

Likewise, since $\text{DP}_{\mathcal{F}}|_{i,i+2} := \{(00, 00), (00, 10), (01, 00), (01, 10)\} \cup \{(10, 01), (10, 11), (11, 01), (11, 11)\}$, so any $x, x' \in X_i$ satisfying $x \sim_{i+2} x'$ must have $(\sigma(x), \sigma(x')) \in \{00, 01\}^2 \cup \{10, 11\}^2$. Therefore, if $x \sim_{i+1} x'$ and $x \sim_{i+2} x'$, then

$$(\sigma(x), \sigma(x')) \in (\{00, 10\}^2 \cup \{01, 11\}^2) \cap (\{00, 01\}^2 \cup \{10, 11\}^2) = \{(00, 00), (01, 01), (10, 10), (11, 11)\},$$

as desired. $\square$

As such, if for some $i \in [3]$ and distinct $x, x' \in X_i$, we have that $x \sim_{i+1} x'$ and $x \sim_{i+2} x'$, then we can replace X with $X \setminus \{x'\}$ and replace every $y \in Y$ with $y_i = x'$ with a corresponding y' with $y'_i = x$ while preserving non-redundancy. Thus, assume without loss of generality that for every distinct $x, x' \in X_i$, we have that either $x \not\sim_{i+1} x'$ or $x \not\sim_{i+2} x'$.

For each distinct $i, j \in [3]$, let ℓ_{ij} be the number of connected components of Y_{ij} . Let $\alpha_{ij} : X_i \sqcup X_j \rightarrow [\ell_{ij}]$ be an indicator map of these connected components. By our WLOG assumption, we have that each $x \in X_i$ is uniquely determined by $(\alpha_{i,i+1}(x), \alpha_{i,i+2}(x))$. In particular, this means that each edge $y \in Y$ is uniquely determined by

$$(\alpha_{1,2}(y_1) = \alpha_{1,2}(y_2), \alpha_{1,3}(y_1) = \alpha_{1,3}(y_3), \alpha_{2,3}(y_2) = \alpha_{2,3}(y_3)).$$

In other words, we can identify Y with $\tilde{Y} \subseteq [\ell_{12}] \times [\ell_{13}] \times [\ell_{23}]$ and each X_i with $\tilde{X}_i \subseteq [\ell_{i,i+1}] \times [\ell_{i,i+2}]$ corresponding to the projection of $\tilde{Y}$ onto $[\ell_{i,i+1}] \times [\ell_{i,i+2}]$. By Shearer's inequality⁹ [25], we then have that

$$|Y| = |\tilde{Y}| \leq \sqrt{|\tilde{X}_1| \cdot |\tilde{X}_2| \cdot |\tilde{X}_3|} = \sqrt{|X_1| \cdot |X_2| \cdot |X_3|} \leq n^{1.5}. \quad \square$$

Remark A.11. This proof in fact shows that the construction in Lemma A.8 is the optimal-sized tripartite instance for infinitely many choice of n .

As an immediate corollary of Lemma A.8 and Lemma A.9.

Theorem A.12. $\text{NRD}(\text{OR-DP}_{\mathcal{F}}, n) = \Theta(n^{1.5})$.

⁹This precise version of Shearer's inequality appeared as Problem 5 at the 1992 International Mathematical Olympiad [34].

A.2.2 Every Rational Number

We now work toward proving Theorem 3.13 in general. By adapting the proof of Lemma 3.11, we have that $\text{NRD}(\text{DP}_{\mathcal{F}}, n) = \mathcal{O}(n)$, so it suffices to bound $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n)$. We begin with a lower bound.

Lemma A.13. $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n) = \Omega_{\mathcal{F}}(n^{p/q})$.

Remark A.14. Lemma A.13 does not need $\mathcal{F}$ to be regular. In fact, the only assumption needed is that every $i \in [p]$ is contained in some $S \in \mathcal{F}$.

Proof. Assume that $n = |\mathcal{F}|t^q$. Let $X := \mathcal{F} \times [t]^q$, and define $Y \subseteq X^{\mathcal{F}}$ as

$$Y := \{e_z := ((S, z|_S) : S \in \mathcal{F}) \in X^{\mathcal{F}} : z \in [t]^p\}$$

Now fix, $z \in [t]^p$, we define an assignment $\sigma_z : X \rightarrow D$ as follows. For each $S \in \mathcal{F}$, $w \in [t]^S$ and $i \in S$, we have that

$$\sigma_z((S, w))_i := \mathbf{1}[w_i \neq z_i].$$

First, note that for all $z' \in [t]^p$, we have that $\sigma_z(e_{z'}) \in \text{DP}_{\mathcal{F}}$. Further, $\sigma_z(e_{z'}) = \text{proj}_{\mathcal{F}}(0^p)$ if and only if $z = z'$. Thus, $\{\sigma_z : z \in [t]^p\}$ witness that (X, Y) is a non-redundant instance with $O_{\mathcal{F}}(t^q)$ variables and t^p clauses. $\square$

We now prove a corresponding upper bound for $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n)$.

Lemma A.15. *If $\mathcal{F}$ is p/q -regular then $\text{NRD}(\text{OR-DP}_{\mathcal{F}} \mid \text{DP}_{\mathcal{F}}, n) = \mathcal{O}(n^{p/q})$.*

Proof. By Lemma 2.3, we may assume without loss of generality that our conditionally non-redundant instance $(X, Y \subseteq X^{\mathcal{F}}, \{\sigma_y : X \rightarrow D \mid y \in Y\})$ is $\mathcal{F}$ -partite. That is, X is the disjoint union of variables sets X_S for each $S \in \mathcal{F}$.

For each $i \in [p]$ define $\mathcal{F}_i := \{S \in \mathcal{F} : i \in S\}$ and $X_{\mathcal{F}_i} := \bigcup_{S \in \mathcal{F}_i} X_S$. Further define $Y_i := \{y|_{\mathcal{F}_i} : y \in Y\} \subseteq X_{\mathcal{F}_i}^{\mathcal{F}_i}$. It may be the case that $|Y_i| < |Y|$. Let $\mathcal{C}_i$ be the set of connected components of Y_i (viewed as a hypergraph on vertex set $X_{\mathcal{F}_i}$). A key observation is that for every satisfying assignment $\sigma : X \rightarrow D$ to $\text{CSP}(\text{DP}_{\mathcal{F}})$, and any $y, y' \in Y$ for which $y|_{\mathcal{F}_i}$ and $y'|_{\mathcal{F}_i}$ are in the same connected component of Y_i , we have that

$$\text{proj}_{\mathcal{F}}^{-1}(\sigma(y))_i = \text{proj}_{\mathcal{F}}^{-1}(\sigma(y'))_i.$$

The reason why is that for any $y \in Y$ and any $S \in \mathcal{F}_i$, $\text{proj}_{\mathcal{F}}^{-1}(\sigma(y))_i = \sigma(y_S)_i$, so $\text{proj}_{\mathcal{F}}^{-1} \circ \sigma$ must be constant on the hyperedges of a connected component of Y_i . Thus, for any $y \in Y$ that the list of connected components it resides in $\mathcal{C}_1, \dots, \mathcal{C}_p$ uniquely identifies y .

Let $c_i : X_{\mathcal{F}_i} \rightarrow \mathcal{C}_i$ be the connected component identification maps. Now fix $S \in \mathcal{F}$. Consider any $x, x' \in X_S$ such that for every $i \in S$, x and x' are in the same connected component of Y_i . By the previous logic, we must have for every satisfying assignment $\sigma : X \rightarrow D$ to $\text{CSP}(\text{DP}_{\mathcal{F}})$, we have that $\sigma(x) = \sigma(x')$. Thus, without loss of generality, we may replace every instance of x' in some $y \in Y$ with x without changing the non-redundancy of the instance. In other words, we may assume that $x \mapsto (c_i(x) : x \in S)$ is an injection from X_S to $\prod_{i \in S} \mathcal{C}_i$.

Now, let Z be the random variable on $\mathcal{C}_1 \times \dots \times \mathcal{C}_p$ corresponding to the list of connected components of a uniformly random sample of $y \in Y$. Let $\mathcal{P}$ be the q/p -regular probability distribution over $\mathcal{F}$. By Shearer's inequality, we have that

$$\log_2 |Y| = H(Z) \leq \frac{p}{q} \cdot \mathbb{E}_{S \sim \mathcal{P}}[H(Z|_S)] \leq \frac{p}{q} \cdot \mathbb{E}_{S \sim \mathcal{P}}[\log_2 |X_S|] \leq \frac{p}{q} \log_2 |X|.$$

Thus, $|Y| \leq |X|^{p/q}$, as desired. $\square$

Combining the lemmas of this section with the triangle inequality, we have proved Theorem 3.13.