

Super-linear Lower Bounds for CSP Non-Redundancy via Shrinking Instances

Joshua Brakensiek* Venkatesan Guruswami† Bart M. P. Jansen‡
Victor Lagerkvist§ Magnus Wahlström¶

Abstract

The non-redundancy (NRD) of a constraint satisfaction problem (CSP) is a combinatorial quantity closely tied to the behavior of CSPs in various computational models including their sparsification, kernelization, and streaming complexity. A primary open question in the study of non-redundancy is the identification of which CSP predicates have near-linear NRD. Recent works by Carbone [CP 2022], Khanna, Putterman and Sudan [STOC 2025], Brakensiek and Guruswami [STOC 2025] and Brakensiek, Guruswami, Jansen, Lagerkvist, and Wahlström [2025] have introduced various forms of *gadget reductions* between CSPs to relate their non-redundancy.

The primary contribution of this work is to recontextualize many of these gadget reductions in a framework which we call *hypergraph projections*. By studying a quantity we call the *shrinking factor* of these hypergraph projections, we can more precisely predict when a gadget reduction between predicates can yield a super-linear NRD lower bound, greatly improving on the analysis of previous works. To illustrate the power of our framework, we identify some concrete CSP predicates whose non-redundancy is at the cusp of our understanding and show how our methods give lower bounds that could not have been achieved with these previous methods. We also demonstrate how these gadget reductions can be automatically deduced using SAT solvers, thereby opening up novel computational avenues for discovering further relationships between the non-redundancy of various CSPs.

1 Introduction

In the study of constraint satisfaction problems (CSPs), the notion of the *non-redundancy* (NRD) of a CSP predicate has recently emerged as a crucial combinatorial benchmark in many questions concerning the structure of CSPs [KK15, FK17, BŽ20, BCK20, CJP20, LW20, Car22, KPS24, KPS25, BG25, BGJ⁺25, BGP26, LGE26, SV26a, SV26b]. The concept of NRD was formally introduced by Bessiere, Carbone, and Katsirelos [BCK20] as a tool for understanding a query complexity problem. Given a predicate¹ $P \subseteq D^r$, where the finite set D is the *domain* of the predicate and r is the *arity* of the predicate, the non-redundancy of P , denoted by $\text{NRD}(P, n)$, measures the largest instance $\mathcal{I}$ of $\text{CSP}(P)$ on n variables such that no constraint $C_i \in \mathcal{I}$ is

*University of California, Berkeley. Contact: josh.brakensiek@berkeley.edu

†Simons Institute for the Theory of Computing and the University of California, Berkeley. Contact: venkatg@berkeley.edu

‡Eindhoven University of Technology, The Netherlands. Contact: b.m.p.jansen@tue.nl

§Linköping University, Linköping. Contact: victor.lagerkvist@liu.se

¶Royal Holloway, University of London. Contact: magnus.wahlstrom@rhul.ac.uk

¹The term predicate is used interchangeably with the term *relation*.

logically implied by the remaining clauses, i.e., there exist assignments to $\mathcal{I}$ which satisfy every clause except C_i .

As a canonical example, consider the *equality* constraint $\text{EQ} \subseteq D^2$ where $(a, b) \in \text{EQ}$ if and only if $a = b$. Since EQ has arity 2, any instance of $\text{CSP}(\text{EQ})$ can be viewed as a graph $G = (V, E)$, where V are the variables and $E \subseteq \binom{V}{2}$ are the constraints. Observe that if G contains a cycle, then G is redundant—if we satisfy EQ for all but one edge of the cycle, then transitivity of equality implies that the last edge must also be satisfied. Therefore, maximal non-redundant instances of $\text{CSP}(\text{EQ})$ are spanning trees; hence, $\text{NRD}(\text{EQ}, n) = n - 1$.

In cases like EQ, if one can establish a predicate P has *near-linear* NRD—that is $\text{NRD}(P, n) = n^{1+o(1)}$ —then one can build many other (often randomized) structures and algorithms concerning $\text{CSP}(P)$. For example, by a result of Brakensiek and Guruswami [BG25] one can use non-uniform random sampling to construct a near-linear *sparsifier* of any instance of $\text{CSP}(P)$, analogous to the near-linear cut sparsifiers due to Benczúr and Karger [Kar93, BK96]. In the setting of kernelization, where one seeks to make an instance of $\text{CSP}(P)$ as small as possible while preserving satisfiability, if the non-redundancy upper bound proof reveals how to constructively detect redundancies, then near-linear kernels can be directly constructed [Car22] (see also discussion in [Hav26]). More recently, a result due to Sharma and Velusamy [SV26a] shows that near-linear NRD directly implies near-linear single-pass streaming algorithms to decide the satisfiability of $\text{CSP}(P)$. Conversely, if the non-redundancy of P is super-linear, that is $\text{NRD}(P, n) = \Omega(n^{1+\varepsilon})$ for some constant $\varepsilon > 0$, then one can rule out both deterministic and randomized constructions of near-linear sparsifiers [BG25] and streaming algorithms [SV26a]. These observations thus motivate a central open question in the study of CSP non-redundancy.

Which CSP predicates have linear or near-linear NRD?

To highlight the complexity and richness of this question, we first discuss additional examples of predicates with near-linear NRD and then discuss some additional predicates which provably have super-linear NRD.

Predicates with linear NRD. The canonical example of a predicate with linear NRD is that of *linear equations*. For example, over the Boolean domain $D = \{0, 1\}$, which we can identify with the field $\mathbb{F}_2$, any predicate $P \subseteq \mathbb{F}_2^r$ which is the set of solutions to a system of linear equations over $\mathbb{F}_2$ (e.g., P is an XOR predicate) yields $\text{NRD}(P, n) \leq n$. This follows from the fact that in this setting non-redundancy is equivalent to *linear independence* of the constraints, and given there are only n variables, the maximum number of linearly independent constraints is n .

A rather subtle consequence of this fact is that, to give a linear upper-bound on NRD, one only needs to show that a predicate P *embeds* into a system of linear equations [LW20, CJP20, BCK20, KPS25]. To illustrate by example, consider the 1-in-3 SAT predicate $P = \{(0, 0, 1), (0, 1, 0), (1, 0, 0)\}$. To bound the non-redundancy of P consider the linear equations $Q = \{(x, y, z) \in \mathbb{F}_3^3 : x + y + z = 1\}$. By the aforementioned logic, we know that $\text{NRD}(Q, n) \leq O(n)$. Further observe that P is precisely the set of Boolean tuples in Q (i.e., $P = Q \cap \{0, 1\}^3$). So if we take a non-redundant instance of $\text{CSP}(P)$ and swap out P -constraints with Q -constraints, it is easily seen that the resulting $\text{CSP}(Q)$ is also non-redundant, witnessed by the same assignments that demonstrate the non-redundancy of each P -constraint. It follows that $\text{NRD}(P, n) \leq \text{NRD}(Q, n) \leq n$. This 1-in-3 SAT example shows that NP-hard predicates can have linear non-redundancy, showing that the study of NRD greatly diverges from the complexity classification of CSPs (e.g., [Bul17, Zhu20]).

More generally, these constructions need not be over a finite field but rather can be done over

any group² or even more general objects known as *Mal'tsev algebras*. That said, in the Boolean setting, [BGJ⁺25] showed that these complex conditions simplify to the much more elementary concept of being *balanced*:

Definition 1.1 ([CJP20]). A Boolean predicate $P \subseteq \{0, 1\}^r$ is *balanced* if for all $k \in \mathbb{N}$ and all tuples $t_1, \dots, t_{2k+1} \in P$ (some possibly equal), if we compute (over $\mathbb{Z}$) the r -tuple

$$t := t_1 - t_2 + t_3 - \dots + t_{2k+1}$$

and $t \in \{0, 1\}^r$, then $t \in P$.

This condition is equivalent to being definable by a system of equations over a ring $\mathbb{Z}/m\mathbb{Z}$ [CJP20, KPS25]. For example, one can easily verify that $P = \{(0, 0, 1), (0, 1, 0), (1, 0, 0)\}$ is balanced. However, the 2-SAT predicate $\text{OR}_2 = \{(0, 1), (1, 0), (1, 1)\}$ is not balanced as

$$(0, 1) - (1, 1) + (1, 0) = (0, 0) \notin \text{OR}_2.$$

Predicates with super-linear NRD. As a first example of a predicate with super-linear NRD, we know that $\text{NRD}(\text{OR}_2, n) = \Theta(n^2)$. The reason is that the graph $G = (V, E = \binom{V}{2})$ is a non-redundant instance of OR_2 (or equivalently³ 2-SAT). Pick any edge $(u, v) \in E$ and consider an assignment $\psi : V \rightarrow \{0, 1\}$ such that $\psi(u) = \psi(v) = 0$ but $\psi(w) = 1$ for all other $w \in V \setminus \{u, v\}$. For this assignment, every OR_2 clause of G is satisfied except for (u, v) , so G is non-redundant. By similar logic, for the predicate $\text{OR}_k = \{0, 1\}^k \setminus \{0^k\}$ (corresponding to k -SAT), we have $\text{NRD}(\text{OR}_k, n) = \Omega_k(n^k)$.

With even just this simple family of lower bounds, we can deduce that many other predicates have super-linear NRD via *gadget reductions* [Car22, KPS25, BGJ⁺25]. To illustrate by example, a recent classification by Brakensiek, Guruswami, and Putterman [BGP26] of Boolean predicates of arity 4, considered as one of their cases the predicate $P = \{(0, 0, 0, 0), (0, 0, 0, 1), (0, 1, 1, 0), (1, 1, 1, 1)\}$. Define a predicate $Q \in \{0, 1\}^2$ to be a *projection* of P as follows

$$(x_1, x_2) \in Q \iff (0, \bar{x}_1, \bar{x}_1, \bar{x}_2) \in P. \quad (1)$$

With this definition, one can inspect that $Q = \text{OR}_2$, from which one can deduce that $\text{NRD}(P, n) \geq \Omega(\text{NRD}(Q, n)) = \Omega(n^2)$. Similar techniques were also used in a related classification by Sharma and Velusamy [SV26b].

A more advanced gadget reduction technique that has recently emerged [BG25, BGJ⁺25] allows for multiple variables per argument in the projection. As a key example, consider from [BG25] the predicate $R = \{(x, y, z) \in \mathbb{F}_3^3 \setminus \{(0, 0, 0)\} : x + y + z = 0\}$. That is, R is a *punctured* set of solutions to a linear equation. A crucial insight by [BG25] is that there exists three functions $g_1, g_2, g_3 : \{0, 1\}^2 \rightarrow \mathbb{F}_3$ such that.

$$\forall x \in \{0, 1\}^3, (x_1, x_2, x_3) \in \text{OR}_3 \iff (g_1(x_1, x_2), g_2(x_1, x_3), g_3(x_2, x_3)) \in R. \quad (2)$$

From this, [BG25] deduce that $\text{NRD}(R, n) = \Omega(n^{1.5})$. Importantly, they also show that $\text{NRD}(R, n) = O(n^{1.6 \log n})$, so no direct reduction from OR_2 could have worked. See Remark 2.7 for a detailed analysis of this example.

More generally, [BGJ⁺25] showed that if one can perform gadget reductions as (2) where k variables are grouped at a time, then one can prove lower bounds of the form $\text{NRD}(P, n) \geq$

²However, for non-Abelian examples, one must consider *cosets* rather than *equations*, see [KPS25, BG25].

³The NRD of a predicate does not change when negations are applied to the variables [Car22].

$\Omega(\text{NRD}(Q, n)^{1/k})$ from which they could construct for every rational $\beta \in \mathbb{Q} \cap [1, \infty)$ a predicate $P \subseteq D^r$ (for some sufficiently large values of $|D|$ and r) such that $\text{NRD}(P, n) = \Theta(n^\beta)$.

Another of class of methods for lower-bounding NRD introduced by [BGJ⁺25] borrows from *extremal combinatorics*.⁴ Such methods could be viewed as analogues of the extremal combinatorics methods used recently to derandomize special cases of the *range avoidance* problem in complexity theory [KS25]. We discuss the relationship between NRD and extremal combinatorics in more detail in our technical overview.

1.1 Our Contributions and Technical Overview

The main technical contribution of this paper is to *revisit* the gadget reduction framework between predicates and improve its analysis. Using this, we show new super-linear lower bounds which were not previously known. For a given predicate $P \subseteq D^r$, we can think of a non-redundant instance of $\text{CSP}(P)$ as an r -uniform hypergraph⁵ $H = (V, E \subseteq V^r)$. By using standard techniques (e.g., [Car22]), we can further assume this instance is r -partite while only changing the non-redundancy by a constant factor. That is, there is a partition $V = V_1 \cup \dots \cup V_r$ of V such that $E \subseteq V_1 \times V_2 \times \dots \times V_r$.

1.1.1 Reductions for NRD via hypergraph projections

Our goal is to transform H in a way such that it becomes a non-redundant instance of some target predicate $Q \subseteq (D')^{r'}$. Toward this, we consider *projections* of this hypergraph. More precisely, for a set $I \subseteq [r] := \{1, \dots, r\}$ of indices, we can define an $|I|$ -uniform hypergraph $\pi_I H$ whose vertices are $\pi_I V := \bigcup_{i \in I} V_i$ and whose edges are $\pi_I E := \{\pi_I e := (e_i : i \in I) : e \in E\}$. The key idea is that if we “bundle” r' of these projections, we can create an instance of $\text{CSP}(Q)$. More precisely pick r' subsets $I_1, \dots, I_{r'} \subseteq [r]$ (possibly with repetition) and let $\mathcal{I} = \{I_1, \dots, I_{r'}\}$, we define the $\mathcal{I}$ -*projection* of H to be an r' -partite r' -uniform hypergraph $H' = (V', E')$ whose vertex set is the disjoint union of $\{\pi_{I_i} E : i \in [r']\}$ and whose edges correspond to E , except we replace $e \in E$ with $(\pi_{I_1} e, \dots, \pi_{I_{r'}} e) \in E'$.

When is this H' a non-redundant instance of $\text{CSP}(Q)$? It turns out it suffices to check if there is a gadget map $g : D^r \rightarrow (D')^{r'}$ with the following properties:

1. For every $p \in P$, $g(p) \in Q$.
2. For every $p \in D^r \setminus P$, $g(p) \notin Q$.
3. For every $i \in [r']$, the i th coordinate map $g_i : D^r \rightarrow D'$ only depends on the input coordinates indexed by I_i .

If such a map exists, we say that “ P is an $\mathcal{I}$ -substructure of Q ” as it shows that a copy of P in a sense “lives inside” Q . Returning to our previous examples, equation (1) depicts a $(\{\}, \{1\}, \{1\}, \{2\})$ -substructure and equation (2) depicts a $(\{1, 2\}, \{1, 3\}, \{2, 3\})$ -substructure. We remark that for the concrete predicates we consider this paper, whether such a map exists can be found within seconds using a SAT solver—see Appendix A for more details.

⁴We note that *upper bounds* using extremal combinatorics have appeared in multiple works prior to [BGJ⁺25], see [BCK20, Car22].

⁵Technically H is a generalization of a hypergraph, similar to *directed* hypergraphs, in that each tuple has an ordering as implied by V^r , but we omit this distinction for simplicity of notation.

Assuming that P is an $\mathcal{I}$ -substructure of Q , then we can immediately deduce a non-redundancy lower bound on $\text{CSP}(Q)$ from non-redundant instance $H = (V, E)$ of $\text{CSP}(P)$. More precisely, we have that

$$\text{NRD}(Q, \sum_{i=1}^{r'} |\pi_{I_i} E|) \geq |E|,$$

In particular, if we can construct non-redundant instances H of $\text{CSP}(P)$ where $|\pi_{I_i} E| \leq |E|^{1-\varepsilon}$ for all $i \in [r']$, then we can deduce that $\text{NRD}(Q, n)$ has super-linear non-redundancy! We define such instances H to be $(|E|^\varepsilon, \mathcal{I})$ -shrinking instances. This leads to our main technical result.

Theorem 1.2 (Informal version of Theorem 3.5). *Assume that P is an $\mathcal{I}$ -substructure of Q and that $\text{CSP}(P)$ has an infinite family of $(|E|^\varepsilon, \mathcal{I})$ -shrinking instances. Then, $\text{NRD}(Q, n) = \Omega(n^{\frac{1}{1-\varepsilon}})$.*

Comparison to [BGJ⁺25]. Before we dive into various applications of Theorem 1.2, we must compare our lower bound with the *c-fgpp* framework of [BGJ⁺25]. Briefly speaking, this projection framework can be viewed as a tighter combinatorial interpretation of the “*c-fgpp*” reduction framework of [BGJ⁺25]. More precisely, in the language used in this paper, the *c-fgpp* reduction framework assumed that each set I_i had the same size c . Furthermore, the instances constructed by [BGJ⁺25] naively have $|V|^c$ vertices rather than the much sharper bound of $\sum_{i=1}^{r'} |\pi_{I_i} E|$. As a result, [BGJ⁺25] could only prove non-redundancy lower bounds of the form $\text{NRD}(Q, n) \geq \text{NRD}(P, n)^{1/c}$ (see our Corollary 3.6)—meaning their method is useless for proving superlinear lower bounds when $\text{NRD}(P, n) = O(n^c)$. In contrast, we show that in many such situations, super-linear lower bounds are still possible due to the existence of shrinking instances. In other situations where [BGJ⁺25] can also prove super-linear lower bounds, our construction of shrinking instances can produce even stronger lower bounds (see Remark C.6).

1.1.2 Applications

The main application of our reduction framework above is to derive super-linear lower bounds on the NRD of some predicates, which eluded previous approaches. As previously mentioned, the most general criterion we know to determine whether a Boolean predicate $P \subseteq \{0, 1\}^r$ has (near-)linear non-redundancy is to check whether P is balanced (Definition 1.1). Thus being imbalanced, or more generally lacking a Mal'tsev embedding in the case of predicates over larger domains, has been viewed as suggestive of a super-linear NRD. Yet there are several concrete imbalanced predicates for which we do not know if the NRD is linear or not. A particularly important example is the predicate $\text{BoolBCK} \subseteq \{0, 1\}^9$ defined as follows⁶

$$\text{BoolBCK} := \{010100001, 001010100, 100001010, 001100010, 010001100\}.$$

One can think of the tuples in BoolBCK as the sequences obtained from all 3×3 permutation matrices *except* the identity matrix, by listing their entries in row-major order. The reason BoolBCK is imbalanced is an alternating sum of elements of BoolBCK can produce the identity matrix.

$$010100001 - 001100010 + 001010100 - 010001100 + 100001010 = 100010001 \notin \text{BoolBCK}.$$

The strong interest in BoolBCK stems from the following fact. If a Boolean predicate P is imbalanced because an alternating sum of *three* tuples of P gives a 0/1-tuple outside of P , then one

⁶As is common in the literature, we omit commas when depicting long ordered tuples.

can easily show [CJP20, Proposition 2.12] that OR_2 is a substructure of P so that $\text{NRD}(P, n) \geq \text{NRD}(\text{OR}_2, n) \geq \Omega(n^2)$. To understand whether imbalanced Boolean predicates have superlinear NRD, the logical next step is to investigate what happens when P is preserved by all alternating sums of three tuples, but not by a sum of five tuples. This is what BoolBCK captures, as explained in earlier work [CJP20, p. 2240].⁷

Due to its imbalance, proving a non-trivial lower bound on BoolBCK (or close variants) has been posed⁸ as an open problem by numerous authors [BCK20, CJP20, LW20, KPS25, BG25, BGJ⁺25].

These sources also observe that adding the identity matrix to BoolBCK makes it balanced. In other words, if we define $\text{BoolBCK}^+ := \text{BoolBCK} \cup \{100010001\}$ we know that $\text{NRD}(\text{BoolBCK}^+, n) = O(n)$. By using the *conditional non-redundancy* framework of Brakensiek and Guruswami [BG25], one can then show that if BoolBCK has super-linear non-redundancy, then there is a super-linear-sized instance such that the non-redundancy of each constraint can be witnessed by an assignment which maps that constraint to 100010001. That is, it suffices to study the *non-redundancy* of the *conditional predicate* $\text{BoolBCK} \mid \text{BoolBCK}^+$. See Definition 2.2 for an exact definition.

Although we do not know whether a super-linear construction exists, we can still ask about “shadows” of its existence. More precisely, consider a 9-uniform 9-partite hypergraph H which is a non-redundant instance of $\text{BoolBCK} \mid \text{BoolBCK}^+$. Every projection $\pi_I H$ can be viewed as a non-redundant instance of a projected predicate (i.e., $\pi_I \text{BoolBCK} \mid \pi_I \text{BoolBCK}^+$). We show that Theorem 1.2 is strong enough to deduce that *all* of these projections can have super-linear size.

Theorem 1.3 (Informal version of Theorem 4.6). *Every non-trivial projection of $\text{BoolBCK} \mid \text{BoolBCK}^+$ has a super-linear NRD lower bound.*

To get these lower bounds, we construct a *non-Boolean* arity 4 conditional predicate $P \mid Q$ corresponding (essentially) to the product of two 6-cycles. By suitably adapting an extremal combinatorics lower bound of [BGJ⁺25], we can show that $\text{NRD}(P \mid Q, n) = \Omega(n^3)$. We then show that all non-trivial projections of $(\text{BoolBCK} \mid \text{BoolBCK}^+)$ has this conditional predicate $P \mid Q$ as an $\mathcal{I}$ -substructure, where suitable collection $\mathcal{I}$ of sets of size 3 is chosen for each projection. We remark that these substructures were identified via use of a SAT solver (see Appendix A for further details). Due to the parameters of our gadget reduction, the results of [BGJ⁺25] are insufficient for deducing a non-trivial lower bound. However, we can show that there are infinitely many non-redundant instances of $\text{CSP}(P \mid Q)$ which are $(|E|^{1/6}, I)$ -shrinking for every $I \subsetneq [4]$. From this, we can get a super-linear lower bound.

Lower Bounds for Non-Boolean Predicates. Similar to Theorem 1.3, we also proved lower bounds on the non-redundancy of various non-Boolean predicates. The particular ones we study in this paper are inspired by the recently-discovered “Catalan identities” by [BGJ⁺25] which can be viewed as a non-Boolean analogue of Definition 1.1. See Appendix C for these results and further discussion.

The Road Ahead. In this paper, we revisited the gadget reduction framework of previous works studying NRD and related problems. Our use of hypergraph projections and $\mathcal{I}$ -substructures shows that the classification of CSP non-redundancy of different predicates is even more interconnected than previously thought. Furthermore, the observation that the existence of these gadget reductions can be automated using a SAT solver gives further hope that suitable large-scale computations could

⁷The predicate BoolBCK can be obtained by negating the first three coordinates of the predicate R^* described in [CJP20], which does not affect NRD, and then re-ordering columns.

⁸Brakensiek and Guruswami [BG25] coined the name “BoolBCK” due it being the Booleanization of a predicate identified by [BCK20].

be applied to locate further relationships between the non-redundancy of various CSPs. In fact, our identification of interesting non-Boolean substructures in projections of $(\text{BoolBCK} \mid \text{BoolBCK}^+)$ shows that a very rich family of predicates are at our disposal for making such deductions. Carrying out such “computational cartography” of the landscape of NRD is the subject of future work.

1.2 Outline

In Section 2 we formally define all the basic concepts discussed in the introduction, including (conditional) non-redundancy and hypergraph-projections. In Section 3, we formally define a shrinking instance and show how they can be used to construct super-linear non-redundancy lower bounds, establishing Theorem 3.5. In Section 4, we apply Theorem 3.5 to prove Theorem 1.2. In Section 5, we give some concluding thoughts and open questions. In Appendix A, we discuss how SAT solvers can be used to construct gadget reductions. In Appendix B, we discuss how every “conditional” non-redundancy problem is asymptotically equivalent (up to a fixed polynomial factor) with a “traditional” non-redundancy problem. In Appendix C, we discuss non-Boolean extensions of Theorem 1.3.

2 Preliminaries

We define a CSP predicate to be a set $P \subseteq D^r$. We call D the *domain* of P and r the *arity* of P . We say P is *nontrivial* if $P \neq \emptyset$ and $P \neq D^r$. We define an r -uniform *hypergraph* (or *instance*) to be a pair $H = (V, E)$, where V is a set of *vertices* (or *variables*) and $E \subseteq V^r$ is a set of (*hyper*)*edges* (or *clauses*). Note that in this definition, hyperedges are ordered tuples. Given an r -uniform hypergraph $H = (V, E)$ and a domain D , we define an *assignment* to be an arbitrary map $\psi : V \rightarrow D$. Given a predicate $P \subseteq D^r$, we say a hyperedge $e \in E$ is *P -satisfied* (or just *satisfied*) by an assignment $\psi : V \rightarrow D$ if $\psi(e) := (\psi(e_1), \dots, \psi(e_r)) \in P$. We say that H is (P -)satisfied by ψ if every $e \in E$ is satisfied by ψ .

Definition 2.1. Given a predicate $P \subseteq D^r$, we say that an r -uniform hypergraph $H = (V, E)$ is a *non-redundant* (NRD) instance of $\text{CSP}(P)$ if for every edge $e \in E$, there exists an assignment $\psi_e : V \rightarrow D$ such that ψ_e satisfies e' for every $e' \in E \setminus \{e\}$, but ψ_e does *not* satisfy e . We define $\text{NRD}(P, n)$ to be the size of the largest non-redundant instance of $\text{CSP}(P)$ on n vertices.

In logical terms, the condition of NRD is equivalent to none of the clauses of the instance being implied by any of the others. If P is nontrivial, then $\text{NRD}(P, n) \geq \lfloor n/r \rfloor$ by considering a collection of vertex-disjoint hyperedges. As such, linear non-redundancy is the ‘baseline’ for every non-redundancy problem.

Conditional non-redundancy. We also make use of the notion of *conditional* non-redundancy coined⁹ by Brakensiek and Guruswami [BG25].

Definition 2.2. Given predicates $P \subsetneq Q \subseteq D^r$, we say that an r -uniform hypergraph $H = (V, E)$ is a (*conditionally*) *non-redundant* instance of $\text{CSP}(P \mid Q)$ if for every edge $e \in E$, there exists an assignment $\psi_e : V \rightarrow D$ such that ψ_e P -satisfies e' for every $e' \in E \setminus \{e\}$, but e is $Q \setminus P$ -satisfied by ψ_e . We define $\text{NRD}(P \mid Q, n)$ to be the size of the largest non-redundant instance of $\text{CSP}(P \mid Q)$ on n vertices.

⁹A notion similar to conditional non-redundancy was used as a proof technique in an earlier paper by Bessiere, Carbonnel, and Katsirelos [BCK20].

We note that $\text{NRD}(P \mid D^r, n) = \text{NRD}(P, n)$ as being $D^r \setminus P$ -satisfying is equivalent to not being P -satisfying. We make use of the following triangle inequality due to [BG25].

Lemma 2.3 (Triangle inequality [BG25]). *For any predicates $P \subsetneq Q \subsetneq R \subseteq D^r$ and $n \in \mathbb{N}$, we have that*

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

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

$$\text{NRD}(P, n) \leq \text{NRD}(P \mid Q, n) + \text{NRD}(Q, n).$$

Remark 2.4. Using techniques similar to [BGJ⁺25, Lemma 4.9] and [BGP26, Proposition 2.4], one can show that *any* non-trivial conditional predicate $\text{CSP}(P \mid Q)$ has a corresponding “traditional” predicate $\text{CSP}(R)$ such that $\text{NRD}(R, n) = \text{NRD}(P \mid Q, n) \cdot \Theta(n^c)$, where c is a fixed constant depending only on P, Q, R —we prove this in Appendix B. Thus, studying conditional non-redundancy, is *necessary* toward understanding the broader NRD landscape.

r -partite instances. We say that an r -uniform hypergraph $H = (V, E)$ is *r -partite*, if there exists a partition $V_1 \sqcup \dots \sqcup V_r = V$ of the vertices such that $E \subseteq V_1 \times V_2 \times \dots \times V_r$. We use the notation $H = (V_1, \dots, V_r, E)$ to denote an r -partite r -uniform hypergraph. By results of [Car22, BG25, BGJ⁺25], we know that any hypergraph $H = (V, E)$ has a subhypergraph $H' = (V_1, \dots, V_r, E')$ with $E' \subseteq E$ such that $|E'| = \Omega_r(|E|)$.¹⁰ Due to this fact (and for convenience) all non-redundant instances we construct in this paper are r -partite.

Projections. Given a sets I and X , we define an I -tuple over X to be a sequence $(x_i \in X : i \in I)$. We denote the set of all such I -tuples by X^I . Given a subset $J \subseteq I$, we define $\pi_J : X^I \rightarrow X^J$ which “forgets” all coordinates not in J . This function may be applied to individual I -tuples or to a set of I -tuples. For example, if we consider $R = \{(0, 1, 2), (1, 2, 0), (2, 0, 0)\}$, then

$$\pi_{\{1,3\}}R = \{\pi_{\{1,3\}}(0, 1, 2), \pi_{\{1,3\}}(1, 2, 0), \pi_{\{1,3\}}(2, 0, 0)\} = \{(0, 2), (1, 0), (2, 0)\}.$$

Given an r -partite hypergraph $H = (V_1, \dots, V_r, E)$ as well as a subset $I \subseteq [r] := \{1, 2, \dots, r\}$, we define $\pi_I H$ to be the I -partite hypergraph $\pi_I H = ((V_i : i \in I), \pi_I E)$. In other words, we “forget” the parts of H not indexed by I .

Gadget Reductions. Previous papers [Car22, BG25, BGJ⁺25] have made use of *functionally guarded primitive promise (fgpp)* definitions (and generalizations) as a type of gadget reduction that can be used to relate the NRD of various CSP predicates. For the lower bounds we prove in this paper, we do not need the full suite of techniques for fgpp, but rather focus on a simpler class of definitions called *substructures* (compare with the less general “projections” of [KPS25, BGP26]).

Definition 2.5. Consider predicates $P \subsetneq Q \subseteq D_1^{r_1}$ and $R \subsetneq S \subseteq D_2^{r_2}$. Given a sequence $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$, we say that $P \mid Q$ is an $\mathcal{I}$ -*substructure* of $R \mid S$ if there exist maps $\{g_j : D_1^{I_j} \rightarrow D_2 \mid j \in [r_2]\}$ (called the *witnessing maps*) such that the following two conditions are true.

$$\forall x \in P, (g_1(\pi_{I_1} x), g_2(\pi_{I_2} x), \dots, g_{r_2}(\pi_{I_{r_2}} x)) \in R, \quad (3)$$

$$\forall x \in Q \setminus P, (g_1(\pi_{I_1} x), g_2(\pi_{I_2} x), \dots, g_{r_2}(\pi_{I_{r_2}} x)) \in S \setminus R. \quad (4)$$

¹⁰We use Ω_r , O_r , Θ_r to denote asymptotics where the multiplicative dependence on r is hidden. For example $2^r \cdot n = O_r(n)$, but $n^r \neq O_r(n)$.

In practice, it is convenient to think about an $\mathcal{I}$ -substructure directly as a map $\Sigma : Q \rightarrow S$ subject to some restrictions on Σ based on the nature of $\mathcal{I}$. We encode this more precisely as follows.

Proposition 2.6. *Consider predicates $P \subsetneq Q \subseteq D_1^{r_1}$ and $R \subsetneq S \subseteq D_2^{r_2}$. Given a sequence $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$, we have that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$ if and only if there exists a map $\Sigma : Q \rightarrow S$ subject to the following conditions*

- (1) $\Sigma(P) \subseteq R$ and $\Sigma(Q \setminus P) \subseteq S \setminus R$.
- (2) For all $i \in [r_1]$ and $j \in [r_2]$, if $i \notin I_j$ and $x, y \in Q$ differ only in coordinate i , then $\Sigma(x)_j = \Sigma(y)_j$.

Proof. First assume that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$ with witnessing maps $g_1, \dots, g_{r_2}$. Define

$$\Sigma(x) = (g_1(\pi_{I_1}x), g_2(\pi_{I_2}x), \dots, g_{r_2}(\pi_{I_{r_2}}x)).$$

Then (3) and (4) ensure that (1) holds. Further, since the j th coordinate (for $j \in [r_2]$) of $\Sigma(x)$ depends only on $\pi_{I_j}x$, we have that (2) holds as well.

Conversely, given $\Sigma : Q \rightarrow S$ subject to (1) and (2), for all $j \in [r_2]$ let g_j be defined such that $g_j(\pi_{I_j}x) = \Sigma_j(x)$. (If some inputs of g_j are undefined, pick them arbitrarily). Condition (2) on Σ ensures this choice of g_j is well-defined. Further condition (1) ensure that (3) and (4) are satisfied, as desired. $\square$

If each $|I_j| = 1$, then we recover the projections framework of Khanna, Putterman, and Sudan [KPS25] and a fragment¹¹ of the fgpp-definition framework of Carbonnel [Car22]. If each $|I_j| = c$ for some $c \geq 1$, then we recover a fragment of the c -fgpp-definition framework of [BGJ⁺25]. In Section 3, we show that if $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$, then a non-redundant instance of $\text{CSP}(P \mid Q)$ can be converted into a non-redundant instance of $\text{CSP}(R \mid S)$.

Remark 2.7. As a non-trivial example of Definition 2.5, consider from [BG25] the example

$$\begin{aligned} P &= \{0, 1\}^3 \setminus \{(0, 0, 0)\} \\ Q &= \{0, 1\}^3 \\ R &= \{(0, 1, 2), (0, 2, 1), (1, 0, 2), (1, 1, 1), (1, 2, 0), (2, 0, 1), (2, 1, 0), (2, 2, 2)\} \\ S &= R \cup \{(0, 0, 0)\}. \end{aligned}$$

Let $\mathcal{I} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$. We have that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$ via the maps g_1, g_2, g_3 such that

$$\begin{aligned} g_1(0, 0) &= 0, g_1(0, 1) = 1, g_1(1, 0) = 2, g_1(1, 1) = 0 \\ g_2(0, 0) &= 0, g_2(0, 1) = 1, g_2(1, 0) = 1, g_2(1, 1) = 2 \\ g_3(0, 0) &= 0, g_3(0, 1) = 2, g_3(1, 0) = 2, g_3(1, 1) = 1. \end{aligned}$$

To confirm that (3) and (4) hold, consult the following table.

¹¹The fgpp reduction framework of [Car22] and the c -fgpp framework of [BGJ⁺25] also allow for conjugation of predicates. For example, if $(x_1, x_2, x_3) \in R \iff [(x_1, x_2) \in P] \wedge [(x_2, x_3) \in Q]$ then $\text{NRD}(R, n) \leq O(\text{NRD}(P, n) + \text{NRD}(Q, n))$. The hypergraph projections framework in this paper could also be augmented with such operations, but we omit such details for clarity and succinctness of presentation.

x	$(g_1(\pi_{\{1,2\}}x), g_2(\pi_{\{1,3\}}x), g_3(\pi_{\{2,3\}}x))$
$(0, 0, 0)$	$(0, 0, 0)$
$(0, 0, 1)$	$(0, 1, 2)$
$(0, 1, 0)$	$(1, 0, 2)$
$(0, 1, 1)$	$(1, 1, 1)$
$(1, 0, 0)$	$(2, 1, 0)$
$(1, 0, 1)$	$(2, 2, 2)$
$(1, 1, 0)$	$(0, 1, 2)$
$(1, 1, 1)$	$(0, 2, 1)$

The map $x \mapsto (g_1(\pi_{\{1,2\}}x), g_2(\pi_{\{1,3\}}x), g_3(\pi_{\{2,3\}}x))$ is precisely the map Σ predicated by Proposition 2.6. In Section 4, we construct much more elaborate reductions.

3 NRD Lower Bounds from Shrinking Instances

In this section, we prove the main result of this paper, which is a general technique for proving superlinear lower-bounds for CSP non-redundancy. A crucial element of our lower bound technique is that they do not apply to general non-redundant instances but only that of *shrinking instances*.

Definition 3.1. Given an r -partite hypergraph $H = (V_1, \dots, V_r, E)$, a family of sets $\mathcal{I}$ of subsets of $[r]$, and a parameter $\lambda \geq 1$, we say that H is $(\lambda, \mathcal{I})$ -*shrinking* if for all $I \in \mathcal{I}$, we have that $|\pi_I E| \leq |E|/\lambda$. We say that H is λ -*shrinking* if H is $(\lambda, \mathcal{I})$ -shrinking for all $\mathcal{I} \subseteq [r]$.

In this work we are interested in shrinkage $\lambda = |E|^{\Omega(1)}$. If a non-redundant instance $H = (V, E \subseteq V^r)$ of $\text{CSP}(P \mid Q)$ is $(|E|^{\Omega(1)}, \mathcal{I})$ -shrinking where $\mathcal{I} = \{\{i\} : i \in [r]\}$ is the set of all singleton sets, then we can immediately deduce that the non-redundancy of $\text{CSP}(P \mid Q)$ is superlinear. However, shrinkage for larger subsets is useful in reductions, which can be realized through the notion of a *projection hypergraph*.

Definition 3.2. Given an r -partite hypergraph $H = (V_1, \dots, V_r, E)$ and a family of sets $\mathcal{I} = (I_1, \dots, I_\ell)$ of subsets of $[r]$ (possibly with repetition), define the ℓ -partite *projection hypergraph* $\text{pr}_{\mathcal{I}} H = (\pi_{I_1} E, \dots, \pi_{I_\ell} E, \text{pr}_{\mathcal{I}}(E))$, where

$$\text{pr}_{\mathcal{I}}(E) = \{\text{pr}_{\mathcal{I}} e := (\pi_{I_1} e, \dots, \pi_{I_\ell} e) \mid e \in E\}.$$

We now make the following observation which can be viewed as a subtle but substantial improvement on the techniques of [BGJ⁺25].

Lemma 3.3. Consider predicates $P \subsetneq Q \subseteq D_1^{r_1}$ and $R \subsetneq S \subseteq D_2^{r_2}$. Given a sequence $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$, assume that that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$ with witnessing maps $g_1, \dots, g_{r_2}$. If $H = (V_1, \dots, V_{r_1}, E)$ is a non-redundant instance of $\text{CSP}(P \mid Q)$, then $\text{pr}_{\mathcal{I}} H = (\pi_{I_1} E, \dots, \pi_{I_{r_2}} E, \text{pr}_{\mathcal{I}}(E))$ is a non-redundant instance of $\text{CSP}(R \mid S)$.

Proof. By that fact that H is a non-redundant instance of $\text{CSP}(P \mid Q)$, for every $e \in E$, there exists $\psi_e : V_1 \sqcup \dots \sqcup V_{r_1} \rightarrow D_1$ such that $\psi_e(e') \in P$ for all $e' \in E \setminus \{e\}$ but $\psi_e(e) \in Q \setminus P$.

Let $\hat{V} := \bigsqcup_{i=1}^{r_2} \pi_{I_i} E$ be the vertices of $\text{pr}_{\mathcal{I}} H$. For each $e \in E$, define $\phi_e : \hat{V} \rightarrow D_2$ as follows.

$$\forall i \in [r_2], \forall t \in \pi_{I_i}(E), \quad \phi_e(t) := g_i(\psi_e(t)).$$

In words, we apply ψ_e to the edges of $\pi_{I_i}(E)$, then we convert each assigned edge to a single value with g_i . Now, consider an arbitrary $e' \in E$, note that

$$\begin{aligned}\phi_e(\text{pr}_{\mathcal{I}} e') &= (\phi_e(\pi_{I_1} e'), \dots, \phi_e(\pi_{I_{r_2}} e')) \\ &= (g_1(\psi_e(\pi_{I_1} e')), \dots, g_{r_2}(\psi_e(\pi_{I_{r_2}} e'))) \\ &= (g_1(\pi_{I_1} \psi_e(e')), \dots, g_{r_2}(\pi_{I_{r_2}} \psi_e(e'))).\end{aligned}$$

Now, if $e' \in E \setminus \{e\}$, then $\psi_e(e') \in P$. Thus, by (3), we have that $\phi_e(\text{pr}_{\mathcal{I}} e') \in R$. Likewise, if $e' = e$, then $\psi_e(e) \in Q \setminus P$. Thus, by (4), we have that $\phi_e(\text{pr}_{\mathcal{I}} e) \in S \setminus R$. This completes the proof that $\text{pr}_{\mathcal{I}} H$ is a non-redundant instance of $\text{CSP}(R \mid S)$. $\square$

Remark 3.4. The primary difference between Lemma 3.3 and comparable techniques in [BGJ⁺25] is that [BGJ⁺25] was inefficient in naively putting $|V|^{|I_i|}$ vertices appear in the i th part of $\text{pr}_{\mathcal{I}}(E)$, whereas Lemma 3.3 shows that one can be more careful in the reduction. As shall be seen in Theorem 3.5 and Section 4, this observation can be the fundamental difference between a trivial lower bound and a superlinear lower bound.

We can finally state our main result on deducing superlinear non-redundancy lower bounds.

Theorem 3.5. *Consider predicates $P \subsetneq Q \subseteq D_1^{r_1}$ and $R \subsetneq S \subseteq D_2^{r_2}$. Given a sequence $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$, assume that that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$. Further let $H_1 = (V_1, E_1), H_2 = (V_2, E_2), \dots$ be an infinite sequence of non-redundant instances of $\text{CSP}(P \mid Q)$ with the following properties. Assume there exists absolute constants $C_1, C_2, \varepsilon > 0$ (with $\varepsilon < 1$) such that the following conditions hold for all $i \geq 1$:*

1. H_i is $(C_1 \cdot |E_i|^{-\varepsilon}, \mathcal{I})$ -shrinking, and
2. $1 \leq |E_{i+1}|/|E_i| \leq C_2$.

Then, $\text{NRD}(R \mid S, n) = \Omega(n^{\frac{1}{1-\varepsilon}})$, where Ω hides constant factors dependent on $|E_1|, C_1, C_2, r_1, r_2, \varepsilon$.

Proof. By Lemma 3.3, we have that $\text{pr}_{\mathcal{I}} H_i$ is a non-redundant instance of $\text{CSP}(R \mid S)$ for all $i \geq 1$. For all $i \geq 1$, define

$$\begin{aligned}m_i &= |E_i|, \\ n_i &= \sum_{j=1}^{r_2} |\pi_{I_j} E_i| \leq r_2 C_1 |E_i|^{1-\varepsilon} = r_2 C_1 m_i^{1-\varepsilon}.\end{aligned}$$

In particular, $\text{NRD}(R \mid S, r_2 C_1 m_i^{1-\varepsilon}) \geq \text{NRD}(R \mid S, n_i) \geq m_i$. By assumption, we have that $1 \leq m_{i+1}/m_i \leq C_2$ for all $i \geq 1$. Therefore, for any $n \geq r_2 C_1 m_1^{1-\varepsilon} = O(1)$, we have that there exists $i \geq 1$ such that

$$n \in [r_2 C_1 m_i^{1-\varepsilon}, r_2 C_1 C_2 m_i^{1-\varepsilon}]$$

Hence,

$$\text{NRD}(R \mid S, n) \geq \text{NRD}(R \mid S, r_2 C_1 m_i^{1-\varepsilon}) \geq m_i \geq \Omega(n^{\frac{1}{1-\varepsilon}}),$$

as desired. $\square$

As an immediate corollary, we can recover the primary content of [BGJ⁺25, Theorem 7.14].

Corollary 3.6. *Consider predicates $P \subsetneq Q \subseteq D_1^{r_1}$ and $R \subsetneq S \subseteq D_2^{r_2}$ with $P, R \neq \emptyset$. Given a sequence $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$ of size at most c , assume that $P \mid Q$ is an $\mathcal{I}$ -substructure of $R \mid S$. Assume that $\text{NRD}(P \mid Q, n) = \Omega(n^\alpha)$, then*

$$\text{NRD}(R \mid S, n) = \Omega(n^{\alpha/c}).$$

Proof. Since R is non-trivial, a lower bound of $\text{NRD}(R \mid S, n) = \Omega(n)$ is immediate by considering an instance with n/r_2 disjoint hyperedges. Thus, we may assume that $\alpha > c$.

For a fixed $n \in \mathbb{N}$, let $H_n = (V_n, E_n)$ be a non-redundant instance of $\text{CSP}(P \mid Q)$ with $|V_n| = n$ vertices and $|E_n| = \Theta(n^\alpha)$ edges (and also assume the number of edges is non-decreasing as n increases). For any $I \subseteq [r_1]$ of size at most c , we have $|\pi_I E_n| \leq n^c$. Thus,

$$\frac{|\pi_I E_n|}{|E_n|} = O\left(\frac{n^c}{n^\alpha}\right) = O(|E_n|^{1-c/\alpha}).$$

That is, H_n is $(O(|E_n|^{1-c/\alpha}), \mathcal{I})$ -shrinking. Furthermore, we can see that $\frac{|E_{n+1}|}{|E_n|} = \Theta(\frac{(n+1)^\alpha}{n^\alpha}) = \Theta(1)$. Therefore, the hypothesis of Theorem 3.5 for $\varepsilon = 1 - c/\alpha$ hold. Thus,

$$\text{NRD}(R \mid S, n) = \Omega(n^{\frac{1}{1-\varepsilon}}) = \Omega(n^{\alpha/c}),$$

as desired. □

4 Applications

In this section, we repeatedly apply Theorem 3.5 to prove new lower bounds for a variety of (conditional) CSP predicates.

4.1 Shrinking Instances via Extremal Combinatorics

We now construct λ -shrinking non-redundant instances for two conditional predicates. Adopting notation from [BGJ⁺25], define C_6 to be the 6-cycle written as a binary predicate of domain size 3:

$$\begin{aligned} C_6 &:= \{(0, 0), (0, 1), (1, 0), (1, 2), (2, 1), (2, 2)\} \\ C_6^* &:= C_6 \setminus \{(0, 0)\}. \end{aligned}$$

We recall the following fact about $\text{CSP}(C_6^* \mid C_6)$.

Proposition 4.1 (Lemma 4.3 [BGJ⁺25]). *A bipartite graph $G = (V_1, V_2, E)$ is a non-redundant instance of $\text{CSP}(C_6^* \mid C_6)$ if and only if G has girth at least 6.*

It is well-known that there are bipartite graphs of girth 6 on n vertices and $\Omega(n^{1.5})$ edges [EHM64], so $\text{NRD}(C_6^* \mid C_6, n) \geq \Omega(n^{1.5})$. We use this fact in the construction of more elaborate predicates.

Definition 4.2. Given predicates $P_1 \subsetneq Q_1 \subseteq D^{r_1}$ and $P_2 \subsetneq Q_2 \subseteq D^{r_2}$, we define their *box product* $(P_1 \mid Q_1) \boxtimes (P_2 \mid Q_2) = (P_3 \mid Q_3)$ with $P_3 \subsetneq Q_3 \subseteq D^{r_1+r_2}$ as follows.

$$\begin{aligned} Q_3 &:= Q_1 \times Q_2 = \{(t_1, \dots, t_{r_1}, t'_1, \dots, t'_{r_2}) : t \in Q_1, t' \in Q_2\} \\ P_3 &:= (P_1 \times Q_2) \cup (Q_1 \times P_2). \end{aligned}$$

Using these relations, we build two conditional predicates as follows.

$$\begin{aligned} R_1 \mid S_1 &:= (C_6^* \mid C_6) \boxtimes (\{1, 2\} \mid \{0, 1, 2\}) \\ R_2 \mid S_2 &:= (C_6^* \mid C_6) \boxtimes (C_6^* \mid C_6) \end{aligned}$$

Remark 4.3. The conditional predicate $R_1 \mid S_1$ is only needed for lower bounds of non-Boolean instances considered in Appendix C. However, the construction of non-redundant shrinking instances of $\text{CSP}(R_1 \mid S_1)$ is a good warm-up to the construction of non-redundant shrinking instances of $\text{CSP}(R_2 \mid S_2)$, so we present both in this section.

In the next two lemmas, we construct shrinking instances of each.

Lemma 4.4. *For all but finitely many m , $\text{CSP}(R_1 \mid S_1)$ has non-redundant instances with m edges which are $O(m^{1/4})$ -shrinking.*

Proof. Pick an integer $n \geq 1$ and pick sets V_1, V_2, V_3 of size n^2, n^2 , and n , respectively. Pick $E_{12} \subseteq V_1 \times V_2$ of size $\Theta(n^3)$ such that (V_1, V_2, E_{12}) has girth 6. Then, let $E = E_{12} \times V_3$. By adapting the proof of [BGJ⁺25, Lemma 4.9], we can show that $H_n := (V_1, V_2, V_3, E)$ is a non-redundant instance of $\text{CSP}(R_1 \mid S_1)$. More precisely, given $e \in E$, we construct $\psi_e : V_1 \sqcup V_2 \sqcup V_3 \rightarrow \{0, 1, 2\}$ as follows. For $v \in V_1 \sqcup V_2$, assign $\psi_e(v)$ the assignment which v is given for the edge (e_1, e_2) as promised by Proposition 4.1 for the instance (V_1, V_2, E_{12}) of $\text{CSP}(C_6^* \mid C_6)$. Finally, for $v \in V_3$, let $\psi_e(v)$ be 0 if $v = e_3$ and 1 otherwise. These assignments show that H_n is a non-redundant instance of $\text{CSP}(R_1 \mid S_1)$.

Now, observe that

$$\begin{aligned} |E| &= |E_{12}| \cdot |V_3| = \Theta(n^4) \\ |\pi_{\{1,2\}} H_n| &= |E_{12}| = \Theta(n^3) \\ |\pi_{\{1,3\}} H_n| &= |\pi_{\{2,3\}} H_n| \leq n^2 \cdot n = O(n^3). \end{aligned}$$

Thus, H_n is $O(n^{-1}) = O(|E|^{-1/4})$ -shrinking, as desired. To get the hypothesis of “all but finitely many m ”, we pick a least n such that H_n has at least m edges and then delete $o(m)$ edges to get an instance with exactly m edges. Since we are only deleting edges, the shrinking factor will change by at most a constant factor. $\square$

Lemma 4.5. *For infinitely many m , $\text{CSP}(R_2 \mid S_2)$ has non-redundant instances with m edges which are $O(m^{1/6})$ -shrinking.*

Proof. Pick an integer $n \geq 1$ and pick sets V_1, V_2, V_3, V_4 each of size n . Pick $E_{12} \subseteq V_1 \times V_2$ of size $\Theta(n^{1.5})$ such that (V_1, V_2, E_{12}) has girth 6. Likewise, pick $E_{34} \subseteq V_3 \times V_4$ of size $\Theta(n^{1.5})$ such that (V_3, V_4, E_{34}) has girth 6.

Then, let $E = E_{12} \times E_{34}$. We claim that $H_n := (V_1, V_2, V_3, V_4, E)$ is a non-redundant instance of $\text{CSP}(R_2 \mid S_2)$. For any $e \in E$, we construct $\psi_e : V_1 \sqcup V_2 \sqcup V_3 \sqcup V_4 \rightarrow \{0, 1, 2\}$ as follows. For $v \in V_1 \sqcup V_2$, assign $\psi_e(v)$ the assignment which v is given for the edge (e_1, e_2) as promised by Proposition 4.1 for the instance (V_1, V_2, E_{12}) of $\text{CSP}(C_6^* \mid C_6)$. Likewise, for $v \in V_3 \sqcup V_4$, assign $\psi_e(v)$ the assignment which v is given for the edge (e_3, e_4) as promised by Proposition 4.1 for the instance (V_3, V_4, E_{34}) of $\text{CSP}(C_6^* \mid C_6)$. To see why ψ_e has the desired property, note that $e' \in E$ is not satisfied by R_2 if and only if both $\psi_e(e'_1, e'_2) = (0, 0)$ and $\psi_e(e'_3, e'_4) = (0, 0)$, which happens only when $e = e'$. We have thus shown that H_n is a non-redundant instance of $\text{CSP}(R_1 \mid S_1)$.

Now, observe that $|E| = |E_{12}| \cdot |E_{34}| = \Theta(n^3)$. Furthermore, for any $I \subseteq [4]$ of size 3, we have that $|\pi_I E| = n \cdot \Theta(n^{1.5}) = \Theta(n^{2.5})$. Thus, H_n is $O(n^{-1/2}) = O(|E|^{-1/6})$ -shrinking, as desired. $\square$

4.2 Boolean Lower Bounds

Adapting notation from [BG25], we let $\text{BoolBCK}^+ \subset \{0, 1\}^9$ denote the set of all 3×3 permutation matrices. More precisely, $\text{BoolBCK}^+ = \{t^{(e)}, t^{(12)}, t^{(13)}, t^{(23)}, t^{(123)}, t^{(321)}\}$ where

$$\begin{aligned} t^{(e)} &= 100010001 \\ t^{(12)} &= 010100001 \\ t^{(13)} &= 001010100 \\ t^{(23)} &= 100001010 \\ t^{(123)} &= 001100010 \\ t^{(321)} &= 010001100 \end{aligned}$$

We then let $\text{BoolBCK} := \{t^{(12)}, t^{(13)}, t^{(23)}, t^{(123)}, t^{(321)}\}$. One can verify that BoolBCK^+ is a *balanced* predicate: for every n and every sequence $t^1, t^2, \dots, t^{2n}, t^{2n+1} \in \text{BoolBCK}^+$ the alternating sum (applied component-wise) $t^1 - t^2 + t^3 - \dots - t^{2n} + t^{2n+1} \in \text{BoolBCK}^+$, and thus $\text{NRD}(\text{BoolBCK}^+, n) = O(n)$ [CJP20]. Therefore, by the triangle inequality, we have that

$$\begin{aligned} \text{NRD}(\text{BoolBCK}, n) &\leq \text{NRD}(\text{BoolBCK} \mid \text{BoolBCK}^+, n) + \text{NRD}(\text{BoolBCK}^+, n) \\ &\leq \text{NRD}(\text{BoolBCK} \mid \text{BoolBCK}^+, n) + O(n). \end{aligned}$$

It is straightforward to verify that $\text{NRD}(\text{BoolBCK}, n) \geq \text{NRD}(\text{BoolBCK} \mid \text{BoolBCK}^+, n) = \Omega(n)$ (e.g., consider $n/3$ disjoint constraints), so we have that

$$\text{NRD}(\text{BoolBCK}, n) = \Theta(\text{NRD}(\text{BoolBCK} \mid \text{BoolBCK}^+, n)).$$

Numerous papers have asked about the non-redundancy of BoolBCK [CJP20, LW20, KPS25, BG25, BGJ⁺25] or closely related predicates. In particular, due to the lack of being balanced, it is very plausible that BoolBCK (and thus $\text{BoolBCK} \mid \text{BoolBCK}^+$) has a truly superlinear non-redundancy lower bound. We do not know how to prove such a bound using Theorem 3.5. However, we can use Theorem 3.5 to prove a superlinear lower bound for every *projection* of $\text{BoolBCK} \mid \text{BoolBCK}^+$.

Theorem 4.6. *For every $J \subsetneq [9]$ such that $\pi_J \text{BoolBCK}^+ \neq \pi_J \text{BoolBCK}$, we have that*

$$\text{NRD}(\pi_J \text{BoolBCK} \mid \pi_J \text{BoolBCK}^+, n) \geq \Omega(n^{6/5}). \quad (5)$$

Proof. For any $I \subseteq J \subseteq [9]$, assuming $\pi_I \text{BoolBCK}^+ \neq \pi_I \text{BoolBCK}$, we can see that $\text{NRD}(\pi_I \text{BoolBCK} \mid \pi_I \text{BoolBCK}^+, n) \geq \Omega(\text{NRD}(\pi_J \text{BoolBCK} \mid \pi_J \text{BoolBCK}^+, n))$ by a simple gadget reduction: if H is an r -partite non-redundant instance of $\text{CSP}(\pi_J \text{BoolBCK} \mid \pi_J \text{BoolBCK}^+)$, then $\pi_I H$ is a non-redundant instance of $\text{CSP}(\pi_I \text{BoolBCK} \mid \pi_I \text{BoolBCK}^+)$.

Thus, it suffices to prove (5) in the case that $|J| = 8$. By applying a suitable (common) permutation to the coordinates of BoolBCK and BoolBCK^+ , it suffices to actually prove (5) in two cases: $J_1 := [9] \setminus \{1\}$ and $J_2 := [9] \setminus \{2\}$, see Lemma 4.7.

By Lemma 4.8 and Lemma 4.9, we have that $(C_6^* \mid C_6) \boxtimes (C_6^* \mid C_6)$ is a suitable substructure of both $(\pi_{J_1} \text{BoolBCK}^+ \mid \pi_{J_1} \text{BoolBCK})$ and $(\pi_{J_2} \text{BoolBCK}^+ \mid \pi_{J_2} \text{BoolBCK})$. By applying Lemma 4.5 to Theorem 3.5 with the parameter choice of $\varepsilon = 1/6$, we thus get our $\Omega(n^{6/5})$ lower bound for both predicates, as desired. $\square$

We now turn to proving the symmetry-breaking Lemma 4.7.

Lemma 4.7. *For every $J \subseteq [9]$ of size 8, either*

$$\begin{aligned} \text{NRD}(\pi_J \text{BoolBCK} \mid \pi_J \text{BoolBCK}^+, n) &= \text{NRD}(\pi_{J_1} \text{BoolBCK} \mid \pi_{J_1} \text{BoolBCK}^+, n), \text{ or} \\ \text{NRD}(\pi_J \text{BoolBCK} \mid \pi_J \text{BoolBCK}^+, n) &= \text{NRD}(\pi_{J_2} \text{BoolBCK} \mid \pi_{J_2} \text{BoolBCK}^+, n), \end{aligned}$$

where $J_1 = [9] \setminus \{1\}$ and $J_2 = [9] \setminus \{2\}$.

Proof. Given a conditional predicate $P \subsetneq Q \subseteq D^r$ and a permutation $\sigma : [r] \rightarrow [r]$, we can define a permuted instance $P^\sigma \subsetneq Q^\sigma \subseteq D^r$ by

$$\begin{aligned} P^\sigma &= \{(x_{\sigma(1)}, \dots, x_{\sigma(r)}) : x \in P\}, \\ Q^\sigma &= \{(x_{\sigma(1)}, \dots, x_{\sigma(r)}) : x \in Q\}. \end{aligned}$$

It is straightforward to verify that $\text{NRD}(P^\sigma \mid Q^\sigma, n) = \text{NRD}(P \mid Q, n)$ as any non-redundant instance $H = (V, E)$ of $P \mid Q$ can be transformed into a non-redundant instance of $P^\sigma \mid Q^\sigma$ by apply the permutation σ to each constraint $e \in E$. Furthermore, since σ is invertible, this is a bijection between non-redundant instances.

To prove the lemma, we apply this fact to the various projections of $\text{BoolBCK} \mid \text{BoolBCK}^+$. More precisely, for $i \in [9]$, let $J_i := [9] \setminus \{i\}$. For $i \in \{5, 9\}$, we construct bijections $\sigma_i : J_i \rightarrow J_1$ such that

$$(\pi_{J_1} \text{BoolBCK})^{\sigma_i} \mid (\pi_{J_1} \text{BoolBCK}^+)^{\sigma_i} = \pi_{J_i} \text{BoolBCK} \mid \pi_{J_i} \text{BoolBCK}^+.$$

Likewise, for $i \in \{3, 4, 6, 7, 8\}$, we construct bijections $\sigma_i : J_i \rightarrow J_2$ such that

$$(\pi_{J_2} \text{BoolBCK})^{\sigma_i} \mid (\pi_{J_2} \text{BoolBCK}^+)^{\sigma_i} = \pi_{J_i} \text{BoolBCK} \mid \pi_{J_i} \text{BoolBCK}^+.$$

We summarize these maps in the following table, with $\perp$ indicating the given input is not in the domain.

j	1	2	3	4	5	6	7	8	9
$\sigma_3(j)$	1	3	$\perp$	7	9	8	4	6	5
$\sigma_4(j)$	1	4	7	$\perp$	5	8	3	6	9
$\sigma_5(j)$	5	2	8	4	$\perp$	7	6	3	9
$\sigma_6(j)$	9	6	3	8	5	$\perp$	8	4	1
$\sigma_7(j)$	1	7	4	3	9	6	$\perp$	8	5
$\sigma_8(j)$	9	3	6	7	1	4	8	$\perp$	5
$\sigma_9(j)$	5	6	4	8	9	7	2	3	$\perp$

For example, using σ_3 , one can see that

$$\begin{aligned} (\pi_{J_2} t^{(e)})^{\sigma_3} &= \pi_{J_1} t^{(e)} & (\pi_{J_2} t^{(12)})^{\sigma_3} &= \pi_{J_1} t^{(13)} \\ (\pi_{J_2} t^{(13)})^{\sigma_3} &= \pi_{J_1} t^{(12)} & (\pi_{J_2} t^{(23)})^{\sigma_3} &= \pi_{J_1} t^{(23)} \\ (\pi_{J_2} t^{(123)})^{\sigma_3} &= \pi_{J_1} t^{(321)} & (\pi_{J_2} t^{(321)})^{\sigma_3} &= \pi_{J_1} t^{(123)}, \end{aligned}$$

which certifies that $(\pi_{J_2} \text{BoolBCK})^{\sigma_3} \mid (\pi_{J_2} \text{BoolBCK}^+)^{\sigma_3} = \pi_{J_3} \text{BoolBCK} \mid \pi_{J_3} \text{BoolBCK}^+$. The validity of these remaining bijections can be checked by a similar argument. $\square$

With the symmetry breaking established, it now suffices to prove Lemma 4.8 and Lemma 4.9.

Lemma 4.8. *There exists $\mathcal{I} = \{I_2, I_3, \dots, I_9\}$ each of size 3 such that $(C_6^* \mid C_6) \boxtimes (C_6^* \mid C_6)$ is an $\mathcal{I}$ -substructure of $(\pi_{J_1} \text{BoolBCK} \mid \pi_{J_1} \text{BoolBCK}^+)$.*

Proof. We use the notation $R_2 \mid S_2$ for $(C_6^* \mid C_6) \boxtimes (C_6^* \mid C_6)$. By Proposition 2.6, it suffices to construct a map $\Sigma : S_2 \rightarrow \pi_{J_1} \text{BoolBCK}^+$ such that the following conditions hold:

- $\Sigma(R_2) \subseteq \pi_{J_1} \text{BoolBCK}$
- $\Sigma(S_2 \setminus R_2) \subseteq \pi_{J_1} \text{BoolBCK} \setminus \pi_{J_1} \text{BoolBCK}$
- Every output coordinate of Σ depends only on three input coordinates.

Using the Satisfiability Modulo Theories (SMT) solver Z3 [dMB08], we construct such a Σ as follows, which can be checked by inspection.

$\Sigma(0, 0, 0, 0) = (0, 0, 0, 1, 0, 0, 0, 1)$	$\Sigma(0, 1, 0, 0) = (1, 0, 1, 0, 0, 0, 0, 1)$
$\Sigma(1, 0, 0, 0) = (0, 1, 0, 1, 0, 1, 0, 0)$	$\Sigma(1, 2, 0, 0) = (0, 1, 1, 0, 0, 0, 1, 0)$
$\Sigma(2, 1, 0, 0) = (1, 0, 0, 0, 1, 1, 0, 0)$	$\Sigma(2, 2, 0, 0) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(0, 0, 0, 1) = (1, 0, 1, 0, 0, 0, 0, 1)$	$\Sigma(0, 1, 0, 1) = (1, 0, 1, 0, 0, 0, 0, 1)$
$\Sigma(1, 0, 0, 1) = (1, 0, 0, 0, 1, 1, 0, 0)$	$\Sigma(1, 2, 0, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(2, 1, 0, 1) = (1, 0, 0, 0, 1, 1, 0, 0)$	$\Sigma(2, 2, 0, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(0, 0, 1, 0) = (0, 1, 0, 1, 0, 1, 0, 0)$	$\Sigma(0, 1, 1, 0) = (0, 1, 1, 0, 0, 0, 1, 0)$
$\Sigma(1, 0, 1, 0) = (0, 1, 0, 1, 0, 1, 0, 0)$	$\Sigma(1, 2, 1, 0) = (0, 1, 1, 0, 0, 0, 1, 0)$
$\Sigma(2, 1, 1, 0) = (0, 0, 0, 0, 1, 0, 1, 0)$	$\Sigma(2, 2, 1, 0) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(0, 0, 1, 2) = (1, 0, 0, 0, 1, 1, 0, 0)$	$\Sigma(0, 1, 1, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(1, 0, 1, 2) = (1, 0, 0, 0, 1, 1, 0, 0)$	$\Sigma(1, 2, 1, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(2, 1, 1, 2) = (0, 1, 1, 0, 0, 0, 1, 0)$	$\Sigma(2, 2, 1, 2) = (0, 1, 1, 0, 0, 0, 1, 0)$
$\Sigma(0, 0, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0)$	$\Sigma(0, 1, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0)$
$\Sigma(1, 0, 2, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$	$\Sigma(1, 2, 2, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(2, 1, 2, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$	$\Sigma(2, 2, 2, 1) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(0, 0, 2, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$	$\Sigma(0, 1, 2, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(1, 0, 2, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$	$\Sigma(1, 2, 2, 2) = (0, 0, 0, 0, 1, 0, 1, 0)$
$\Sigma(2, 1, 2, 2) = (0, 1, 1, 0, 0, 0, 1, 0)$	$\Sigma(2, 2, 2, 2) = (0, 1, 1, 0, 0, 0, 1, 0)$

In particular, if we let $(i_2, \dots, i_9) = (1, 2, 3, 1, 2, 4, 1, 2)$, then the j th coordinate of the output does not depend on the i_j th coordinate of the input (that is, $I_j = [4] \setminus \{i_j\}$). $\square$

Lemma 4.9. *There exists $\mathcal{I} = \{I_1, I_3, I_4, \dots, I_9\}$ each of size 3 such that $(C_6^* \mid C_6) \boxtimes (C_6^* \mid C_6)$ is an $\mathcal{I}$ -substructure of $(\pi_{J_2} \text{BoolBCK} \mid \pi_{J_2} \text{BoolBCK}^+)$.*

Proof. By Proposition 2.6, it suffices to construct a map $\Sigma : S_2 \rightarrow \pi_{J_2} \text{BoolBCK}^+$ such that the following conditions hold:

- $\Sigma(R_2) \subseteq \pi_{J_2} \text{BoolBCK}$
- $\Sigma(S_2 \setminus R_2) \subseteq \pi_{J_2} \text{BoolBCK} \setminus \pi_{J_2} \text{BoolBCK}$

- Every output coordinate of Σ depends only on three input coordinates.

Using the Z3 SMT solver [dMB08], we construct such a Σ as follows, which can be checked by inspection.

$$\begin{array}{ll}
\Sigma(0, 0, 0, 0) = (1, 0, 0, 1, 0, 0, 0, 1) & \Sigma(0, 1, 0, 0) = (0, 0, 1, 0, 0, 0, 0, 1) \\
\Sigma(1, 0, 0, 0) = (1, 0, 0, 0, 1, 0, 1, 0) & \Sigma(1, 2, 0, 0) = (0, 0, 0, 0, 1, 1, 0, 0) \\
\Sigma(2, 1, 0, 0) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(2, 2, 0, 0) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(0, 0, 0, 1) = (0, 0, 1, 0, 0, 0, 0, 1) & \Sigma(0, 1, 0, 1) = (0, 0, 1, 0, 0, 0, 0, 1) \\
\Sigma(1, 0, 0, 1) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(1, 2, 0, 1) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(2, 1, 0, 1) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(2, 2, 0, 1) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(0, 0, 1, 0) = (0, 1, 0, 1, 0, 1, 0, 0) & \Sigma(0, 1, 1, 0) = (0, 1, 1, 0, 0, 0, 1, 0) \\
\Sigma(1, 0, 1, 0) = (0, 0, 0, 0, 1, 1, 0, 0) & \Sigma(1, 2, 1, 0) = (0, 0, 0, 0, 1, 1, 0, 0) \\
\Sigma(2, 1, 1, 0) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(2, 2, 1, 0) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(0, 0, 1, 2) = (0, 0, 0, 0, 1, 1, 0, 0) & \Sigma(0, 1, 1, 2) = (1, 0, 0, 0, 1, 0, 1, 0) \\
\Sigma(1, 0, 1, 2) = (0, 0, 0, 0, 1, 1, 0, 0) & \Sigma(1, 2, 1, 2) = (0, 0, 0, 0, 1, 1, 0, 0) \\
\Sigma(2, 1, 1, 2) = (1, 0, 0, 0, 1, 0, 1, 0) & \Sigma(2, 2, 1, 2) = (0, 0, 0, 0, 1, 1, 0, 0) \\
\Sigma(0, 0, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(0, 1, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0) \\
\Sigma(1, 0, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(1, 2, 2, 1) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(2, 1, 2, 1) = (0, 1, 1, 0, 0, 0, 1, 0) & \Sigma(2, 2, 2, 1) = (0, 1, 0, 1, 0, 1, 0, 0) \\
\Sigma(0, 0, 2, 2) = (1, 0, 0, 0, 1, 0, 1, 0) & \Sigma(0, 1, 2, 2) = (1, 0, 0, 0, 1, 0, 1, 0) \\
\Sigma(1, 0, 2, 2) = (1, 0, 0, 0, 1, 0, 1, 0) & \Sigma(1, 2, 2, 2) = (0, 0, 0, 0, 1, 1, 0, 0) \\
\Sigma(2, 1, 2, 2) = (1, 0, 0, 0, 1, 0, 1, 0) & \Sigma(2, 2, 2, 2) = (0, 0, 0, 0, 1, 1, 0, 0)
\end{array}$$

In particular, if we let $(i_1, i_3, i_4, \dots, i_9) = (1, 2, 1, 3, 2, 1, 4, 2)$, then the j th coordinate of the output does not depend on the i_j th coordinate of the input (that is, $I_j = [4] \setminus \{i_j\}$). $\square$

5 Conclusion and Open Questions

In this paper, we use a hypergraph projection framework to study the relationship between the non-redundancy of various CSP predicates. By studying what we call *shrinking instances*, we can obtain sharper lower bounds on the non-redundancy of many predicates. The progress we made in this paper serves as the next step toward conquering the biggest open questions concerning the nature of non-redundancy, especially the classification of which predicates have (near-)linear non-redundancy. The use of automated reasoning highlights the sophistication of the gadget reductions we consider as well the possibility of even more elaborate reductions in the future.

The primary open question we leave is whether super-linear lower bounds exist for BoolBCK. Given how our methods can yield such lower bounds for every nontrivial projection of BoolBCK | BoolBCK⁺, the primary barrier at this point may be to identify a suitable (conditional) predicate with well-behaved shrinking instances. Given that we found that every nontrivial projection of BoolBCK | BoolBCK⁺ has $C_6 \boxtimes C_6$ as a substructure, we can infer that the construction of high-girth graphs can yield versatile shrinking instances which can be beneficial in complex gadget reductions. However, even when we tried longer cycles, we were not able to construct a suitable substructure of BoolBCK | BoolBCK⁺ itself. One potential route to such a substructure would be to

consider high-girth *hypergraph* problems rather than graph problems. We leave such investigations as the subject of future work.

Acknowledgments

JB and VG are supported in part by a Simons Investigator award and NSF grant CCF-2211972. JB is also supported by NSF grant DMS-2503280. BJ is supported by the Dutch Research Council (NWO) through Gravitation-grant NETWORKS-024.002.003.

References

- [BCK20] Christian Bessiere, Clément Carbonnel, and George Katsirelos. Chain Length and CSPs Learnable with Few Queries. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(02):1420–1427, April 2020. doi:10.1609/aaai.v34i02.5499.
- [BG25] Joshua Brakensiek and Venkatesan Guruswami. Redundancy is all you need. In Michal Koucký and Nikhil 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. doi:10.1145/3717823.3718212.
- [BGJ⁺25] Joshua Brakensiek, Venkatesan Guruswami, Bart M. P. Jansen, Victor Lagerkvist, and Magnus Wahlström. The Richness of CSP Non-redundancy. (arXiv:2507.07942), July 2025. arXiv:2507.07942, doi:10.48550/arXiv.2507.07942.
- [BGP26] Joshua Brakensiek, Venkatesan Guruswami, and Aaron Putterman. Classification of non-redundancy of boolean predicates of arity 4. *arXiv preprint arXiv:2603.21353*, 2026.
- [BK96] András A. Benczúr and David R. Karger. Approximating s - t minimum cuts in $\tilde{O}(n^2)$ time. In Gary 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. doi:10.1145/237814.237827.
- [Bul17] Andrei A. Bulatov. A dichotomy theorem for nonuniform csps. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 319–330. IEEE Computer Society, 2017. doi:10.1109/FOCS.2017.37.
- [BŽ20] Silvia Butti and Stanislav Živný. Sparsification of Binary CSPs. *SIAM Journal on Discrete Mathematics*, 34(1):825–842, January 2020. doi:10.1137/19M1242446.
- [Car22] Clément 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. doi:10.4230/LIPIcs.CP.2022.11.
- [CJP20] Hubie Chen, Bart M. P. Jansen, and Astrid Pieterse. Best-Case and Worst-Case Sparsifiability of Boolean CSPs. *Algorithmica*, 82(8):2200–2242, August 2020. doi:10.1007/s00453-019-00660-y.

- [dMB08] Leonardo Mendonça de Moura and Nikolaj S. Bjørner. Z3: an efficient SMT solver. In C. R. Ramakrishnan and Jakob Rehof, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29–April 6, 2008. Proceedings*, Lecture Notes in Computer Science, pages 337–340. Springer, 2008. doi:10.1007/978-3-540-78800-3_24.
- [EHM64] Paul Erdős, András Hajnal, and John W. Moon. A problem in graph theory. *The American Mathematical Monthly*, 71(10):1107–1110, 1964.
- [FK17] Arnold Filtser and Robert Krauthgamer. Sparsification of Two-Variable Valued Constraint Satisfaction Problems. *SIAM Journal on Discrete Mathematics*, 31(2):1263–1276, January 2017. doi:10.1137/15M1046186.
- [Hav26] Ishay Haviv. Kernelization bounds for constrained coloring, 2026. URL: <https://arxiv.org/abs/2604.21531>, arXiv:2604.21531.
- [Kar93] David R. Karger. Global min-cuts in RNC, and other ramifications of a simple min-cut algorithm. In Vijaya 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.
- [KK15] Dmitry Kogan and Robert 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, January 2015. Association for Computing Machinery. doi:10.1145/2688073.2688093.
- [KPS24] Sanjeev Khanna, Aaron L. Putterman, and Madhu 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, January 2024. doi:10.1137/1.9781611977912.185.
- [KPS25] Sanjeev Khanna, Aaron Putterman, and Madhu Sudan. Efficient algorithms and new characterizations for CSP sparsification. In Michal Koucký and Nikhil 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. doi:10.1145/3717823.3718205.
- [KS25] Neha Kuntewar and Jayalal Sarma. Avoiding range via Turan-type bounds. In Alina Ene and Eshan Chattopadhyay, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2025, Berkeley, CA, USA, August 11-13, 2025*, LIPIcs, pages 62:1–62:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025. URL: <https://doi.org/10.4230/LIPICS.APPROX/RANDOM.2025.62>, doi:10.4230/LIPICS.APPROX/RANDOM.2025.62.
- [LGE26] Victor Lagerkvist, Johanna Groven, and Leif Eriksson. Towards single exponential time for temporal and spatial reasoning: A study via redundancy and dynamic programming. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances*

in *Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 14287–14294. AAAI Press, 2026. URL: <https://doi.org/10.1609/aaai.v40i17.38443>, doi:10.1609/AAAI.V40I17.38443.

- [LW20] Victor Lagerkvist and Magnus Wahlström. Sparsification of SAT and CSP Problems via Tractable Extensions. *ACM Transactions on Computation Theory*, 12(2):1–29, June 2020. doi:10.1145/3389411.
- [SV26a] Amatya Sharma and Santhoshini Velusamy. Characterizing streaming decidability of csps via non-redundancy, 2026. URL: <https://arxiv.org/abs/2604.21922>, arXiv:2604.21922.
- [SV26b] Amatya Sharma and Santhoshini Velusamy. Non-redundancy of low-arity symmetric boolean csps, 2026. URL: <https://arxiv.org/abs/2605.14007>, arXiv:2605.14007.
- [Zhu20] Dmitriy Zhuk. A Proof of the CSP Dichotomy Conjecture. *J. ACM*, 67(5):30:1–30:78, August 2020. doi:10.1145/3402029.

A Finding $\mathcal{I}$ -substructures using a SAT-solver

Given predicates $P_1 \subsetneq Q_1 \subseteq D_1^{r_1}$ and $P_2 \subsetneq Q_2 \subseteq D_2^{r_2}$ as well as a family $\mathcal{I} = (I_1, \dots, I_{r_2})$ of subsets of $[r_1]$ a natural question is how one can verify whether $P_1 \mid Q_1$ is a $\mathcal{I}$ -substructure of $P_2 \mid Q_2$. In this appendix, we briefly discuss how this can be encoded into SAT.

From Proposition 2.6, it suffices to construct a map $\Sigma : Q_1 \rightarrow Q_2$ satisfying the following conditions.

- (1) $\Sigma(P_1) \subseteq P_2$ and $\Sigma(Q_1 \setminus P_1) \subseteq Q_2 \setminus P_2$.
- (2) For all $i \in [r_1]$ and $j \in [r_2]$, if $i \notin I_j$ and $q, q' \in Q_1$ differ only in coordinate i , then $\Sigma(q)_j = \Sigma(q')_j$

For each $q \in Q_1$, $i \in [r_1]$, and $d \in D_2$, we have a variable $x_{q,i,d}$ which represents whether $\Sigma_i(q) = d$. From this, condition (2) can be quickly encoded as equality conditions between some of the variables. More precisely $x_{q,j,d} = x_{q',j,d}$ (aka $(x_{q,j,d} \vee \bar{x}_{q',j,d}) \wedge (\bar{x}_{q,j,d} \vee x_{q',j,d})$) whether q and q' differ only in the i th coordinate and $i \notin I_j$.

We enforce (1) using auxiliary variables. For every $q \in Q_1$ and $q' \in Q_2$, we have a variable $y_{q,q'}$ representing whether $\sigma(q) = q'$. For each such $y_{q,q'}$ we construct a series of implications connecting them to the $x_{q,i,d}$'s.

$$\forall i \in [r_1], d \in D_2, y_{q,q'} \implies x_{q,i,d} \text{ if } d = q'_i \text{ otherwise } y_{q,q'} \implies \bar{x}_{q,i,d}.$$

To enforce that $\Sigma(P_1) \subseteq P_2$ and $\Sigma(Q_1 \setminus P_1) \subseteq Q_2 \setminus P_2$, we force $y_{q,q'}$ if $q \in P_1$ but $q' \notin P_1$ or vice-versa. Finally, we need for each $q \in Q_1$ that $y_{q,q'}$ is true for some $q' \in Q_2$ which can be done by a suitable CNF: $\bigvee_{q' \in Q_2} y_{q,q'}$. This completes a valid SAT-encoding of the $\mathcal{I}$ -substructure condition.

Remark A.1. This approach is not used directly in this paper. Rather, using the Satisfiability Modulo Theories (SMT) solver Z3 [dMB08], we write out the conditions of being a $\mathcal{I}$ -substructure in a higher level of abstraction (i.e., Σ_1 -sentences involving integer variables) which Z3 then can find a solution for using a variety of techniques including SAT solving.

B Conditional Non-redundancy Can Be Made Non-conditional

In [BGJ⁺25, Lemma 4.9] and [BGP26, Proposition 2.4], the authors show for many conditional predicates $P \subsetneq Q \subseteq D^r$, there is a corresponding (non-conditional) predicate $R \subseteq D^{r+1}$ such that $\text{NRD}(R, n) = \Theta(\text{NRD}(P \mid Q, n) \cdot n)$, showing that conditional non-redundancy is not merely a useful abstraction but also a necessary quantity to analyze the non-redundancy of all predicates. In this appendix, we show that their arguments can be adapted to prove a similar phenomenon holds for *all* nontrivial conditional predicates.

Recall from Definition 4.2 the notion of a box product of two predicates.

Theorem B.1. *Let D be a domain of size at least 2 with $\{0, 1\} \subseteq D$. Let $P \subsetneq Q \subseteq D^r$ be a conditional predicate with P non-empty. Further define $R \subsetneq S \subseteq D^{2r}$ such that*

$$R \mid S = (P \mid Q) \boxtimes (\text{OR}_r \mid \{0, 1\}^r)$$

Then, $\text{NRD}(R, n) = \Theta_r(\text{NRD}(P \mid Q, n) \cdot n^r)$, where Θ_r hides a factor depending only on r .

Proof. First, we prove that $\text{NRD}(R, n) = \Omega r(\text{NRD}(P \mid Q, n) \cdot n^r)$. To see why, let $H = (V, E)$ be a non-redundant instance of $\text{CSP}(P \mid Q)$ on $n/2$ vertices with $|E| = \text{NRD}(P \mid Q, n/2) = \Omega(P \mid Q, n)$. Let V' be a disjoint set of $n/2$ vertices. Define $H' = (V \cup V', E' := E \times \binom{V'}{r})$ which has $\Omega r(\text{NRD}(P \mid Q, n) \cdot n^r)$ constraints. We claim that H' is a non-redundant instance of $\text{CSP}(R)$. To see why, fix an edge $e' \in E'$, we need to find an assignment $\psi : V \cup V' \rightarrow D$ which satisfies every constraint of H' except e' . Observe that $\pi_{[r]}e'$ is an edge of H , thus by the non-redundancy of H as instance of $\text{CSP}(P \mid Q)$ there is an assignment $\phi : V \rightarrow D$ which satisfies every edge of H except $\pi_{[r]}e'$. We can thus define ψ as follows

$$\psi(v) := \begin{cases} \phi(v) & v \in V \\ 0 & v = e'_i \text{ for some } i \in \{r+1, \dots, 2r\} \\ 1 & \text{otherwise} \end{cases}$$

To see why this works, recall that $R = (P \times \{0, 1\}^r) \cup (Q \times \text{OR}_r)$. Thus, for any $e'' \in E' \setminus \{e'\}$ with $\pi_{[r]}e'' \neq \pi_{[r]}e'$, we have that $\psi(e'') \in P \times \{0, 1\}^r$. Likewise, if $e'' \in E' \setminus \{e'\}$ with $\pi_{\{r+1, \dots, 2r\}}e'' \neq \pi_{\{r+1, \dots, 2r\}}e'$, then $\psi(e'') \in Q \times \text{OR}_r$. However, $\psi(e') = (\phi(\pi_{[r]}e'), 0, \dots, 0) \notin R$. This shows that H' is a non-redundant instance of $\text{CSP}(R)$.

We now turn to showing that $\text{NRD}(R, n) = O_r(\text{NRD}(P \mid Q, n) \cdot n^r)$. By Lemma 2.3, we have that

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

Thus, it suffices to show that $\text{NRD}(R \mid S, n) = O_r(\text{NRD}(P \mid Q, n) \cdot n^r)$ and $\text{NRD}(S, n) = O_r(\text{NRD}(P \mid Q, n) \cdot n^r)$.

For the former upper bound, Let $H = (V, E \subseteq V^{2r})$ be a non-redundant instance of $\text{CSP}(R \mid S)$ with $|V| = n$ and $|E| = \text{NRD}(R \mid S, n)$. By the pigeonhole principle, there must be some string $s \in V^r$ such that $\pi_{\{r+1, \dots, 2r\}}e = s$ for at least $|E|/n^r$ choices of $e \in E$. Define $H' = (V, E' \subseteq V^r)$ where

$$E' := \{\pi_{[r]}e : e \in E, \pi_{\{r+1, \dots, 2r\}}e = s\}.$$

We claim that H' is a non-redundant instance of $\text{CSP}(P \mid Q)$. Pick distinct $e'_1, e'_2 \in E'$ and let $e_1, e_2 \in E$ be the corresponding edge in H . Since H is non-redundant, there is an assignment $\psi : V \rightarrow D$ which satisfies every edge of H except e_1 . In particular, $\psi(e_2) \in R = (P \times \{0, 1\}^r) \cup$

$(Q \times \text{OR}_r)$ but $\psi(e_1) \in S \setminus R = (Q \setminus P) \times \{0^r\}$. Since e_1 and e_2 share the same last r coordinates, we must have that $\psi(e_2) \in P \times \{0^r\}$. In other words (returning to H'), $\psi(e'_1) \in Q \setminus P$ but $\psi(e'_2) \in P$ for every $e'_2 \in E' \setminus \{e'_1\}$. This proves that H' is a non-redundant instance of $\text{CSP}(P \mid Q)$. Therefore, $\text{NRD}(P \mid Q, n) = \Omega(\text{NRD}(R \mid S, n)/n^r)$, as desired.

As a last step, we must bound $\text{NRD}(S, n)$. Recall that $S = Q \times \{0, 1\}^r$. Consider a non-redundant instance $H = (V, E \subseteq V^{2r})$ of $\text{CSP}(S)$. Given a string $s \in V^r$, let $H_s = (V, E_s)$ be the subinstance where

$$E_s := \{e \in E : \pi_{[r]}e = s\}.$$

Then, we claim that $|E_s| \leq n$. To see why, fix any $e \in E_s$ and let $\psi : V \rightarrow D$ be a map which fails to satisfy e but satisfies all other $e' \in E_s$. Since $\pi_{[r]}e' = \pi_{[r]}e = s$, we must have that $\psi(s) \in Q$. Therefore, for $\psi(e) \notin S$, we must have that $\psi(\pi_{\{r+1, \dots, 2r\}}e) \notin \{0, 1\}^r$. That is, $\psi(e_i) \notin \{0, 1\}$ for some $i \in \{r+1, \dots, 2r\}$. Crucially, this e_i cannot appear as a vertex in $\pi_{\{r+1, \dots, 2r\}}e'$ for any $e' \in E_s \setminus \{e\}$ (or else $\psi(e') \in S$ would be false). Therefore, there exists an injection from E_s to V by picking this vertex which maps outside of $\{0, 1\}$. This proves that $|E_s| \leq n$.

Since there are n^r choices for $s \in V^r$, we have proved that $\text{NRD}(S, n) \leq n^{r+1}$. Recall that P is non-empty. Thus, $\text{NRD}(P \mid Q, n) \geq n/r = \Omega_r(n)$ by considering an instance with n/r disjoint clauses. Thus,

$$\text{NRD}(S, n) \leq n^{r+1} = O_r(\text{NRD}(P \mid Q, n) \cdot n^r).$$

This fact completes the proof. $\square$

C Lower Bounds for Non-Boolean Predicates

Going beyond Theorem 4.6, we also use Theorem 3.5 to deduce novel super-linear lower bounds for *non-Boolean* predicates. Very recently, [BGJ⁺25] developed an analogue of the Boolean notion of balanced which applies to any domain by a notion called the *Catalan identities*. To illustrate this notion, consider a finite domain D and a word $w \in D^\ell$, where ℓ is an odd integer. We can play the following “cancellation game” on w . If w has two consecutive symbols which are equal, delete both, and recurse until no more progress can be made. For example, if $D = \{0, 1, 2\}$ then playing the game with 0221221 yields

$$0221221 \rightarrow 01221 \rightarrow 011 \rightarrow 0.$$

It turns out the outcome of this game is always the same no matter the order the cancellations are done. Furthermore, this cancellation game can be used to determine whether a predicate has a Mal'tsev embedding. As an example, consider the predicate $\text{Cat}_5 \subseteq \{0, 1, 2\}^5$ defined by

$$\text{Cat}_5 = \{01012, 11111, 12201, 22222, 20120\}.$$

Now build the following matrix whose *columns* are tuples in Cat_5 .

$$\begin{pmatrix} 0 & 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 2 & 0 \\ 0 & 1 & 2 & 2 & 1 \\ 1 & 1 & 0 & 2 & 2 \\ 2 & 1 & 1 & 2 & 0 \end{pmatrix} \tag{6}$$

If one plays the cancellation game on each *row* of this matrix, one is left with the tuple 00000 which is *not* in Cat_5 . By a deep result of [BGJ⁺25], this implies that Cat_5 lacks a Mal'tsev embedding

over any domain, so we have no evidence that Cat_5 has (near-)linear NRD. In fact, similar to how BoolBCK is the ‘canonical’ non-balanced predicate with five tuples, Cat_5 is the canonical predicate with five tuples which lacks a Mal’tsev embedding. More precisely, the rows of (6) correspond to every possible cancellation game on 5-tuples. As such, one can show that any non-redundancy lower bound of $\text{CSP}(\text{Cat}_5)$ implies a non-redundancy lower bound for $\text{CSP}(\text{BoolBCK})$. This is witnessed by the map $\Sigma : \text{Cat}_5^+ \rightarrow \text{BoolBCK}^+$ defined by

$$\begin{aligned}\Sigma(00000) &= 100010001 \\ \Sigma(01012) &= 010100001 \\ \Sigma(11111) &= 001100010 \\ \Sigma(12201) &= 001010100 \\ \Sigma(22222) &= 010001100 \\ \Sigma(20120) &= 100001010\end{aligned}$$

Therefore, resolving the non-redundancy of Cat_5 is an essential question toward determining which predicates have linear non-redundancy.

Similar to BoolBCK, if we let $\text{Cat}_5^+ = \text{Cat}_5 \cup \{00000\}$, then Cat_5^+ can be shown to have linear non-redundancy. Thus, we can always assume every non-assignment to a non-redundant instance of $\text{CSP}(\text{Cat}_5^+)$ is 00000, reducing our problem to the non-redundancy of the conditional predicate $\text{Cat}_5 \mid \text{Cat}_5^+$. Although proving super-linear lower bounds for Cat_5 is harder than for BoolBCK, we still give a super-linear lower bound for one of the maximal non-trivial projections of $\text{Cat}_5 \mid \text{Cat}_5^+$.

Theorem C.1 (See Theorem C.3). $\text{NRD}(\pi_{[4]} \text{Cat}_5 \mid \pi_{[4]} \text{Cat}_5^+, n) \geq n^{6/5}$.

This result can be viewed as one step closer toward showing that Mal’tsev embeddings are equivalent to having (near-)linear non-redundancy.

C.1 Lower Bounds

In addition to $\pi_{[4]} \text{Cat}_5 \mid \pi_{[4]} \text{Cat}_5^+$, we also give stronger lower bounds for some arity 3 projections. More precisely, consider the following predicates. For ease of discussion, we give each succinct names.

$$\begin{aligned}P_1 &:= \pi_{\{1,3,4\}} \text{Cat}_5 = \{001, 111, 120, 222, 212\} \\ Q_1 &:= \pi_{\{1,3,4\}} \text{Cat}_5^+ = P_1 \cup \{000\} \\ P_2 &:= \pi_{\{1,2,4\}} \text{Cat}_5 = \{011, 111, 120, 222, 202\} \\ Q_2 &:= \pi_{\{1,2,4\}} \text{Cat}_5^+ = P_2 \cup \{000\} \\ P_3 &:= \pi_{\{1,2,3,4\}} \text{Cat}_5 = \{0101, 1111, 1220, 2222, 2012\} \\ Q_3 &:= \pi_{\{1,2,3,4\}} \text{Cat}_5^+ = P_3 \cup \{0000\}\end{aligned}$$

Remark C.2. A projection we do *not* consider in this paper is

$$\pi_{\{3,4,5\}} \text{Cat}_5 = \{012, 111, 201, 222, 120\},$$

which is equivalent to a ternary predicate identified by [BCK20] which has lower non-redundancy than BoolBCK. See Section 7 of [BG25] for further discussion.

The goal of this section is to prove the following lower bounds.

Theorem C.3. *We following lower bounds hold.*

$$\text{NRD}(P_1 \mid Q_1, n) \geq \Omega(n^{4/3}) \quad (7)$$

$$\text{NRD}(P_2 \mid Q_2, n) \geq \Omega(n^{4/3}) \quad (8)$$

$$\text{NRD}(P_3 \mid Q_3, n) \geq \Omega(n^{6/5}) \quad (9)$$

Proof. By applying Lemma 4.4 and Lemma 4.5 to Theorem 3.5 with $\varepsilon = 1/4$ and $\varepsilon = 1/6$, respectively, it suffices to prove Lemma C.4, Lemma C.5, and Lemma C.7 to establish (7), (8), and (9), respectively. $\square$

Lemma C.4. *There exists $\mathcal{I} = \{I_1, I_2, I_3\}$ each of size 3 such that $R_1 \mid S_1$ is an $\mathcal{I}$ -substructure of $P_1 \mid Q_1$.*

Proof. By Proposition 2.6, it suffices to construct a map $\Sigma : S_1 \rightarrow Q_1$ such that the following conditions hold:

- $\Sigma(R_1) \subseteq P_1$
- $\Sigma(S_1 \setminus R_1) \subseteq Q_1 \setminus P_1$
- Every output coordinate of Σ depends only on two input coordinates.

Using the Z3 SMT solver [dMB08], we constructed such a Σ as follows, which can be checked by inspection.

$\Sigma(0, 0, 0) = (0, 0, 0)$	$\Sigma(0, 0, 1) = (0, 0, 1)$
$\Sigma(0, 0, 2) = (0, 0, 1)$	$\Sigma(0, 1, 0) = (1, 2, 0)$
$\Sigma(0, 1, 1) = (1, 1, 1)$	$\Sigma(0, 1, 2) = (1, 1, 1)$
$\Sigma(1, 0, 0) = (0, 0, 1)$	$\Sigma(1, 0, 1) = (0, 0, 1)$
$\Sigma(1, 0, 2) = (0, 0, 1)$	$\Sigma(1, 2, 0) = (1, 1, 1)$
$\Sigma(1, 2, 1) = (1, 1, 1)$	$\Sigma(1, 2, 2) = (1, 1, 1)$
$\Sigma(2, 1, 0) = (2, 2, 2)$	$\Sigma(2, 1, 1) = (2, 1, 2)$
$\Sigma(2, 1, 2) = (2, 1, 2)$	$\Sigma(2, 2, 0) = (2, 1, 2)$
$\Sigma(2, 2, 1) = (2, 1, 2)$	$\Sigma(2, 2, 2) = (2, 1, 2)$

Here $(I_1, I_2, I_3) = (\{1, 2\}, \{2, 3\}, \{1, 3\})$. $\square$

Lemma C.5. *There exists $\mathcal{I} = \{I_1, I_2, I_3\}$ each of size 3 such that $R_1 \mid S_1$ is an $\mathcal{I}$ -substructure of $P_2 \mid Q_2$.*

Proof. By Proposition 2.6, it suffices to construct a map $\Sigma : S_1 \rightarrow Q_2$ such that the following conditions hold:

- $\Sigma(R_1) \subseteq P_2$
- $\Sigma(S_1 \setminus R_1) \subseteq Q_2 \setminus P_2$
- Every output coordinate of Σ depends only on two input coordinates.

Using the Z3 SMT solver [dMB08], we constructed such a Σ as follows, which can be checked by inspection.

$$\begin{array}{ll}
\Sigma(0,0,0) = (0,0,0) & \Sigma(0,0,1) = (1,0,2) \\
\Sigma(0,0,2) = (1,0,2) & \Sigma(0,1,0) = (2,2,0) \\
\Sigma(0,1,1) = (2,2,2) & \Sigma(0,1,2) = (2,2,2) \\
\Sigma(1,0,0) = (0,1,1) & \Sigma(1,0,1) = (1,1,1) \\
\Sigma(1,0,2) = (1,1,1) & \Sigma(1,2,0) = (1,1,1) \\
\Sigma(1,2,1) = (1,1,1) & \Sigma(1,2,2) = (1,1,1) \\
\Sigma(2,1,0) = (2,2,2) & \Sigma(2,1,1) = (2,2,2) \\
\Sigma(2,1,2) = (2,2,2) & \Sigma(2,2,0) = (1,0,2) \\
\Sigma(2,2,1) = (1,0,2) & \Sigma(2,2,2) = (1,0,2)
\end{array}$$

Here $(I_1, I_2, I_3) = (\{1, 2\}, \{2, 3\}, \{1, 3\})$. □

Remark C.6. For $\text{CSP}(P_1 \mid Q_1)$ and $\text{CSP}(P_2 \mid Q_2)$, since the projections sets have size at most 2 and $\text{NRD}(R_1 \mid S_1, n) = \Omega(n^{5/2})$, the methods of [BGJ⁺25] can be used to immediately prove a lower bound of $\Omega(n^{5/4})$. However, using our shrinking instances, we get a sharper lower bound of $\Omega(n^{4/3})$.

Lemma C.7. *There exists $\mathcal{I} = \{I_1, I_2, I_3, I_4\}$ each of size 3 such that $R_2 \mid S_2$ is an $\mathcal{I}$ -substructure of $P_3 \mid Q_3$.*

Proof. By Proposition 2.6, it suffices to construct a map $\Sigma : S_2 \rightarrow Q_3$ such that the following conditions hold:

- $\Sigma(R_2) \subseteq P_3$
- $\Sigma(S_2 \setminus R_2) \subseteq Q_3 \setminus P_3$
- Every output coordinate of Σ depends only on three input coordinates.

Using the Z3 SMT solver [dMB08], we constructed such a Σ as follows, which can be checked by inspection.

$$\begin{array}{ll}
\Sigma(0,0,0,0) = (0,0,0,0) & \Sigma(0,0,0,1) = (1,2,2,0) \\
\Sigma(0,0,1,0) = (0,1,0,1) & \Sigma(0,0,1,2) = (1,1,1,1) \\
\Sigma(0,0,2,1) = (2,2,2,2) & \Sigma(0,0,2,2) = (2,0,1,2) \\
\Sigma(0,1,0,0) = (2,0,1,2) & \Sigma(0,1,0,1) = (2,2,2,2) \\
\Sigma(0,1,1,0) = (1,1,1,1) & \Sigma(0,1,1,2) = (1,1,1,1) \\
\Sigma(0,1,2,1) = (2,2,2,2) & \Sigma(0,1,2,2) = (2,0,1,2) \\
\Sigma(1,0,0,0) = (0,1,0,1) & \Sigma(1,0,0,1) = (1,1,1,1) \\
\Sigma(1,0,1,0) = (0,1,0,1) & \Sigma(1,0,1,2) = (1,1,1,1) \\
\Sigma(1,0,2,1) = (2,0,1,2) & \Sigma(1,0,2,2) = (2,0,1,2) \\
\Sigma(1,2,0,0) = (1,1,1,1) & \Sigma(1,2,0,1) = (1,1,1,1) \\
\Sigma(1,2,1,0) = (1,1,1,1) & \Sigma(1,2,1,2) = (1,1,1,1)
\end{array}$$

$\Sigma(1, 2, 2, 1) = (2, 0, 1, 2)$	$\Sigma(1, 2, 2, 2) = (2, 0, 1, 2)$
$\Sigma(2, 1, 0, 0) = (2, 2, 2, 2)$	$\Sigma(2, 1, 0, 1) = (2, 2, 2, 2)$
$\Sigma(2, 1, 1, 0) = (1, 2, 2, 0)$	$\Sigma(2, 1, 1, 2) = (1, 2, 2, 0)$
$\Sigma(2, 1, 2, 1) = (2, 2, 2, 2)$	$\Sigma(2, 1, 2, 2) = (2, 2, 2, 2)$
$\Sigma(2, 2, 0, 0) = (1, 2, 2, 0)$	$\Sigma(2, 2, 0, 1) = (1, 2, 2, 0)$
$\Sigma(2, 2, 1, 0) = (1, 2, 2, 0)$	$\Sigma(2, 2, 1, 2) = (1, 2, 2, 0)$
$\Sigma(2, 2, 2, 1) = (2, 2, 2, 2)$	$\Sigma(2, 2, 2, 2) = (2, 2, 2, 2)$

Here $(I_1, I_2, I_3, I_4) = (\{2, 3, 4\}, \{1, 3, 4\}, \{1, 2, 4\}, \{1, 2, 3\})$.

□