On the Sparsifiability of Valued Constraint Satisfaction Problems

Silvia Butti

A thesis submitted in partial fulfillment of the requirements for the degree of

MSc in Mathematics and Foundations of Computer Science

Mathematical Institute
University of Oxford
August 2018

To Walter and Armida

Abstract

A Constraint Satisfaction Problems (CSP) is a mathematical framework used to model objects, represented by variables, which are subject to some sort of limitations or constraints. CSPs are deeply connected to hypergraphs and hence both constraint programming and graph theory can benefit from the mutual transfer of knowledge and techniques between the fields. In this dissertation we investigate the problem of extending established results in graph sparsification to valued Contraint Satisfaction Problems, that is, CSPs whose constraints are augmented with a weight. This problem was first posed by Filtser and Krauthgamer [FK17], who focused on binary Boolean valued CSPs. We extend their work to arbitrary binary CSPs on any discrete domain by establishing a dichotomy theorem for sparsifiability of discrete binary valued CSPs. In terms of non-binary CSPs, we identify a class of problems for which we can give lower bounds on the minimum size of a sparsifier, implying a range of negative results for sparsifiability. We further investigate the more general problem of spectral sparsification and make some first steps towards sparsification of continuous valued CSPs.

Acknowledgements

First of all, I would like to thank my supervisor, Professor Stanislav Živný, who has gone out of his way to read this manuscript countless times and gave me precious guidance, careful feedback and constant encouragement. Standa, it was a pleasure working with you.

Secondly, I wouldn't have got to where I am today if it wasn't for all the inspiring teachers and mentors whom I have met so far in my academic journey. Special thanks go to Sergio Mattiolo, Annalisa Tuppini, Robert Bowles, John Talbot, Jorge Lobo and Nava Rubin.

Finally, the greatest gratitude goes to my family. Dodo, Ciano, Ele and Gianna, there are no words to describe how thankful I am for the love, support and inspiration throughout all these years.

Contents

1 Introduction **1**
1.1 Definitions . . . 2
1.2 Related Work . . . 7
1.3 Contributions . . . 9

2 Combinatorial Sparsification **11**
2.1 Graph Covers . . . 11
2.2 Results on Non-Sparsifiability . . . 12
2.3 Singleton Predicates . . . 15
2.4 A Complete Classification of Binary Predicates . . . 18
2.4.1 The Binary Product Predicate . . . 25
2.4.2 The Binary Parity Predicate . . . 26
2.5 Tools and Techniques for Future Work . . . 26
2.5.1 A Note on Multisets . . . 27
2.5.2 Map Reductions . . . 29

3 Spectral Sparsification **37**
3.1 Spectral Sparsifiers and Continuous VCSPs . . . 37
3.1.1 Applications of Spectral Sparsification . . . 42

4 Conclusion **43**

Appendix A **45**
A.1 A Generalised Definition of Valued CSP . . . 45
A.2 Extending Results . . . 46

Appendix B **51**
B.1 Sparsification of Generalised Binary Predicates . . . 51

Appendix C **57**
C.1 The k-ary Parity Predicate . . . 57

Bibliography **61**

Chapter 1

Introduction

A Constraint Satisfaction Problem (CSP) consists of a set of variables and a set of constraints on subsets of the variables. Solving a CSP amounts to assigning a value to each variable so that every constraint is satisfied. In a Valued Constraint Satisfaction Problem (VCSP), each constraint has an associated cost function which attributes a value to each possible assignment of the variables. The cost function defines an optimisation problem on the CSP, where the goal becomes to find an assignment which either maximises or minimises the total weight of the constraints that are satisfied.

CSPs have found applications in a wide range of fields, including but not limited to Artificial Intelligence, Operations Research, Scheduling, Routing, Network Design and Control [RvBW06]. Given that every constraint in a CSP instance can be seen as a relation on a directed hyperedge in the corresponding hypergraph, many well-known problems in Computer Science can be encoded as a CSP - this includes computationally hard decision problems such as Graph Colouring and Boolean Satisfiability, as well as popular mathematical puzzles such as Sudoku and the Eight Queens Puzzle.

In this dissertation we mostly focus on Valued Constraint Satisfaction on discrete domains and we address the topic of sparsification in VCSPs. To sparsify a VCSP means to select a re-weighted sparse subset of the constraints while maintaining the total value of the VCSP under any assignment of the variables up to an approximation factor of ε. We aim to find sparsifiers with a constraint set whose size is linear or quasilinear in the size of the variable set. We concentrate on providing impossibility and constructive results, where in the proof of the latter we take advantage of the efficient algorithms for sparsifier construction that are described in the literature [BK96, BSS12, KK15]. Our main result is a dichotomy theorem for sparsifiability of discrete binary VCSPs. The criterion of sparsifiability that we propose depends on the structure of the constraint support set. We also provide some partial results on non-binary and continuous VCSPs, as well as tools and techniques which may be useful for future research. We conclude by outlining possible directions for further research and questions which remain open.

1.1 Definitions

We begin by presenting a few necessary definitions and setting up the notation that will be used throughout this dissertation.
The set of positive reals (excluding zero) will be denoted by $\mathbb{R}^+$.

Definition 1.1.1. An instance of a *Valued Constraint Satisfaction Problem* (VCSP) is a triple

$$\mathcal{I} = (V, \Pi, w)$$

where

- $V = \{v_1, \ldots, v_n\}$ is a set of variables,
- $\Pi = \{\pi_1, \ldots, \pi_m\}$ is a set of constraints,
- $w : \Pi \to \mathbb{R}^+$ is a strictly positive weight function on the set of constraints.

Each constraint $\pi_i \in \Pi$ consists of a pair $\langle scope, rel \rangle$, where $scope = \mathbf{v} \in V^k$ is an ordered k-tuple of distinct variables and *rel* is a k-ary relation on some domain D, which can be seen as a function $\mathsf{P} : D^k \to \{0, 1\}$ of constant arity $k \geq 2$.

We write VCSP(P) for the class of all VCSP instances $\mathcal{I} = (V, \Pi, w)$ where all constraints in Π are subject to the same predicate P. For a *finite* set of predicates $\Gamma = \{\mathsf{P}_j\}_{j \in J}$, one can naturally define the class VCSP(Γ) of instances whose constraints are all subject to predicates in Γ. In such a context, Γ is called a *language*. We will only consider languages of finite size.
Throughout this dissertation we will mainly deal with discrete domains, whose elements will be referred to as labels. To simplify the notation, we define $[r] = \{0, 1, \ldots, r-1\}$ so that $|[r]| = r$. Moreover, we will denote $|V| = n$ and $|\Pi| = m$.

Example 1.1.1. Some predicates that will be widely used throughout are the following:

$$\begin{aligned}
&\mathsf{Cut} : \{0,1\}^2 \to \{0,1\}, && \mathsf{Cut}(x,y) = 1 && \iff && x \neq y, \\
&\mathsf{UnCut} : \{0,1\}^2 \to \{0,1\}, && \mathsf{UnCut}(x,y) = 1 && \iff && x = y, \\
&r\text{-}\mathsf{Cut} : [r]^2 \to \{0,1\}, && r\text{-}\mathsf{Cut}(x,y) = 1 && \iff && x \neq y, \\
&\mathsf{And} : \{0,1\}^k \to \{0,1\}, && \mathsf{And}(x_1,\ldots,x_k) = 1 && \iff && \min\{x_1,\ldots,x_k\} = 1, \\
&\mathsf{nOr} : \{0,1\}^k \to \{0,1\}, && \mathsf{nOr}(x_1,\ldots,x_k) = 1 && \iff && \max\{x_1,\ldots,x_k\} = 0, \\
&\mathsf{NAE} : \{0,1\}^k \to \{0,1\}, && \mathsf{NAE}(x_1,\ldots,x_k) = 1 && \iff && \min\{x_1,\ldots,x_k\} = 0 \text{ and} \\
& && && && \max\{x_1,\ldots,x_k\} = 1.
\end{aligned}$$

Remark 1.1.1. The literature is not always consistent with regards to the definition of a VCSP. Indeed, what we define as a VCSP is often referred to as simply a CSP or Weighted CSP (WCSP). On the other hand, the term VCSP is sometimes used to define a CSP where each hyperedge e of size k is augmented by a k-ary function $f_e : D^k \to \Omega$ where Ω is a valuation structure[1] often identified with $\overline{\mathbb{Q}^+} = \mathbb{Q}^+ \cup \infty$, the set of extended non-negative rationals [CCJK06, KZ17]. We choose to use the VCSP notation to be consistent with Filtser and Krauthgamer's work on CSP sparsification [FK17].

Remark 1.1.2. The definition of a VCSP instance $\mathcal{I} = (V, \Pi, w)$ is stated with Π being a set. However, in Propositions 2.5.2 and 2.5.3 we show that respectively positive and negative results on sparsifiability are not affected by Π being a multiset.

Definition 1.1.2. Given an instance $\mathcal{I} = (V, \Pi, w) \in \text{VCSP}(\mathsf{P})$, we will call the *underlying hypergraph* of $\mathcal{I}$ the weighted directed k-uniform hypergraph

$$H^{\mathcal{I}} = (V, E, w)$$

where $E = \{\mathbf{v} \in V^k : \langle \mathbf{v}, \mathsf{P} \rangle \in \Pi\}$ and, by a slight abuse of notation, $w(\mathbf{v}) = w(\langle \mathbf{v}, \mathsf{P} \rangle)$. Notice that E is possibly a multiset and that the weight function is always strictly positive.

Conversely, given a weighted directed k-uniform hypergraph $H = (V, E, w)$ and a predicate $\mathsf{P} : D^k \to \{0, 1\}$, the *corresponding* VCSP(P) *instance* is

$$\mathcal{I}^{H,\mathsf{P}} = (V, \Pi, w)$$

where $\Pi = \{\langle e, \mathsf{P} \rangle : e \in E\}$ and $w(\langle e, \mathsf{P} \rangle) = w(e)$.

Remark 1.1.3. Sometimes in the literature a k-uniform hypergraph indicates a hypergraph which only contains hyperedges of size less than or equal to k [KK15]. However, throughout this dissertation by a k-uniform hypergraph we mean a hypergraph whose hyperedges are of size *exactly* k.

Notation. For a vector $\mathbf{v} = (v_1, \ldots, v_k) \in V^k$ and a function $A : V \to D$, which we call an *assignment*, we will denote the vector $(A(v_1), \ldots, A(v_k))$ by $A(\mathbf{v})$. Oftentimes, we will also use the shorthand notation $A(\pi)$ to denote $A(\mathbf{v})$ where $\pi = \langle \mathbf{v}, \mathsf{P} \rangle$ for some predicate P.

Definition 1.1.3. Given a predicate $\mathsf{P} : D^k \to \{0, 1\}$, we say that a constraint $\pi = \langle \mathbf{v}, \mathsf{P} \rangle$ is *satisfied* by an assignment $A : V \to D$ if $\mathsf{P}(A(\mathbf{v})) = 1$.

Definition 1.1.4. The *value* of an instance $\mathcal{I} = (V, \Pi, w) \in \text{VCSP}(\Gamma)$ where $\Gamma = \{P_j : D_j^{k_j} \to \{0, 1\}\}_{j \in J}$ under assignment $A : V \to \cup_{j \in J} D_j$ is given by the sum of the weights of the constraints satisfied by A:

$$Val_{\mathcal{I}}(A) := \sum_{\pi = \langle \mathbf{v}, \mathsf{P}_j \rangle \in \Pi} w(\pi) \mathsf{P}_j(A(\mathbf{v})).$$

For $\mathcal{I} \in \text{VCSP}(\Gamma)$ we say that an assignment $A : V \to \cup_{j \in J} D_j$ is *valid* if $A(\mathbf{v}) \in D_j^{k_j}$ for all $\pi = \langle \mathbf{v}, \mathsf{P}_j \rangle \in \Pi$. In other words, given that each predicate $\mathsf{P}_j \in \Gamma$ may have a distinct

[1] A valuation structure Ω is an ordered set with a minimum element 0 and a maximum element ∞ together with a commutative and associative binary aggregation operator $\oplus$ such that for all $a, b, c \in \Omega$, $a \oplus 0 = a$ and $a \oplus c \geq b \oplus c$ whenever $a \geq b$.

domain, an assignment is said to be valid if each variable $v \in V$ is assigned to a label which belongs to the intersection of the domains of all the predicates whose scope contains v.

Now, given an instance $\mathcal{I} \in \mathrm{VCSP}(\Gamma)$, we aim to construct a sparse re-weighted subinstance whose value is approximately equal to that of the original instance, up to a factor of $\varepsilon \in (0,1)$, under any possible assignment. Given the equivalence between VCSP instances and hypergraphs, this amounts to finding a suitable reweighted directed subhypergraph of the underlying hypergraph $H^{\mathcal{I}}$ of $\mathcal{I}$.
There are several notions that generalise the idea of subgraph to hypergraphs. We will refer to the notion of partial subhypergraph defined in [Vol09], which matches well with the notion of a VCSP subinstance.

Definition 1.1.5. Given a hypergraph $H = (V, E)$, a *partial subhypergraph* $H' = (V, E' \subseteq E)$ of H is a hypergraph which is obtained from H by removing hyperedges while keeping the vertex set unchanged.

Throughout, we will use the soft-O notation $\tilde{O}(f(n)) = \mathcal{O}(f(n)\log^a(n))$ for some constant $a > 0$ to refer to functions that are quasilinear in n.

Definition 1.1.6 ([KK15])**.** An ε-*sparsifier* of an instance $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\Gamma)$ is a subinstance

$$\mathcal{I}_\varepsilon = (V, \Pi_\varepsilon \subseteq \Pi, w_\varepsilon)$$

such that the size of its constraint set satisfies

$$|\Pi_\varepsilon| = \tilde{\mathcal{O}}\Big(\frac{|V|}{\varepsilon^2}\Big)$$

and its value is such that, for all valid assignments A of the variables in V,

$$Val_{\mathcal{I}_\varepsilon}(A) \in (1 \pm \varepsilon) Val_{\mathcal{I}}(A). \tag{1.1}$$

More precisely, if $\Gamma = \{P_j : D_j^{k_j} \to \{0,1\}\}_{j \in J}$ and $\max_{j \in J} k_j = K$, we take

$$|\Pi_\varepsilon| = \mathcal{O}\big(\frac{|V|(K + \log|V|)}{\varepsilon^2}\big).$$

Equivalently, an ε-P-*sparsifier* of a hypergraph $H = (V, E, w)$ is a partial subhypergraph

$$H_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$$

whose corresponding $\mathrm{VCSP}(\mathsf{P})$ instance $\mathcal{I}^{H_\varepsilon, \mathsf{P}}$ is an ε-sparsifier of the corresponding $\mathrm{VCSP}(\mathsf{P})$ instance $\mathcal{I}^{H,\mathsf{P}}$ of H.

We say that an instance $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\Gamma)$ is sparse if $|\Pi| = \tilde{\mathcal{O}}\big(\frac{|V|}{\varepsilon^2}\big)$. Clearly, every sparse VCSP instance is an ε-sparsifier of itself for all $\varepsilon \in (0,1)$.

Definition 1.1.6 highlights how, for a fixed k-ary predicate P, the class VCSP(P) is equivalent to the class of all positively weighted directed k-uniform hypergraphs. Thus, we will use both notions interchangeably.

Let $\mathbb{1}_S$ denote the indicator function of a set S defined by $\mathbb{1}_S(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{if } x \notin S. \end{cases}$

Example 1.1.2. Consider the binary Cut predicate defined in Example 1.1.1. An instance $\mathcal{I} \in$ VCSP(Cut) essentially corresponds to a positively weighted directed graph H where, for any subset S of the vertices of V, $Val_{\mathcal{I}}(\mathbb{1}_S) = \mathsf{Cut}_H(S)$ is the weighted value of the cut induced by S in H. Therefore, finding an ε-sparsifier of $\mathcal{I}$ corresponds to sparsifying cuts in the underlying graph H.

We will now define several notions of sparsifiability whose relationship is illustrated in Figure 1.1.

Definition 1.1.7. We say that a predicate $\mathsf{P} : D^k \to \{0, 1\}$ is *sparsifiable* if for all $\varepsilon \in (0, 1)$ and for all k-uniform weighted directed hypergraphs $H = (V, E, w)$ there exists an ε-P-sparsifier of H.

Definition 1.1.8. We say that a predicate $\mathsf{P} : D^k \to \{0, 1\}$ is *non-sparsifiable* if there exists a k-uniform weighted directed hypergraph $H = (V, E, w)$ and an $\varepsilon \in (0, 1)$ such that all partial subhypergraphs $H_\varepsilon = (V, E_\varepsilon \subset E, w_\varepsilon)$ with property (1.1) have $|E_\varepsilon| = \Omega(|V|^k)$. We will sometimes refer to non-sparsifiable predicates as *weakly* non-sparsifiable, in contrast to the notion of a strongly non-sparsifiable predicate presented in Definition 1.1.9.

Definition 1.1.9. We say that a predicate $\mathsf{P} : D^k \to \{0, 1\}$ is *strongly non-sparsifiable* if for all k-uniform weighted directed hypergraphs $H = (V, E, w)$ there exists an $\varepsilon \in (0, 1)$ such that all partial subhypergraphs $H_\varepsilon = (V, E_\varepsilon \subset E, w_\varepsilon)$ with property (1.1) have $|E_\varepsilon| = \Omega(|E|)$.

Notice that there is a "gap" between sparsifiable and non-sparsifiable predicates, as the two notions are mutually exclusive but not complementary. For this reason, we define an additional class of predicates that are not sparsifiable.

Definition 1.1.10. We say that a predicate $\mathsf{P} : D^k \to \{0, 1\}$ is *resistant to sparsification* if for all k-uniform weighted directed hypergraphs $H = (V, E, w)$ that are not sparse there exists an $\varepsilon \in (0, 1)$ such that H does not have an ε-P-sparsifier.

Remark 1.1.4. If a predicate is strongly non-sparsifiable, then it is clearly resistant to sparsification. Moreover, considering that a k-uniform hypergraph can have up to $\Theta(n^k)$ hyperedges, a strongly non-sparsifiable predicate is also non-sparsifiable.

Remark 1.1.5. Theorems 1.2.4 and 1.2.5 from [FK17], which we state in Section 1.2, show that a binary Boolean predicate is not sparsifiable if and only if it is non-sparsifiable.

However, for a general k-ary predicate, we do not know whether the classes represented in Figure 1.1 are all distinct, as some of them may possibly coincide. For instance, the existence of a non-sparsifiable predicate that is provably *not* strongly non-sparsifiable is not guaranteed. This classification then remains an interesting open problem.

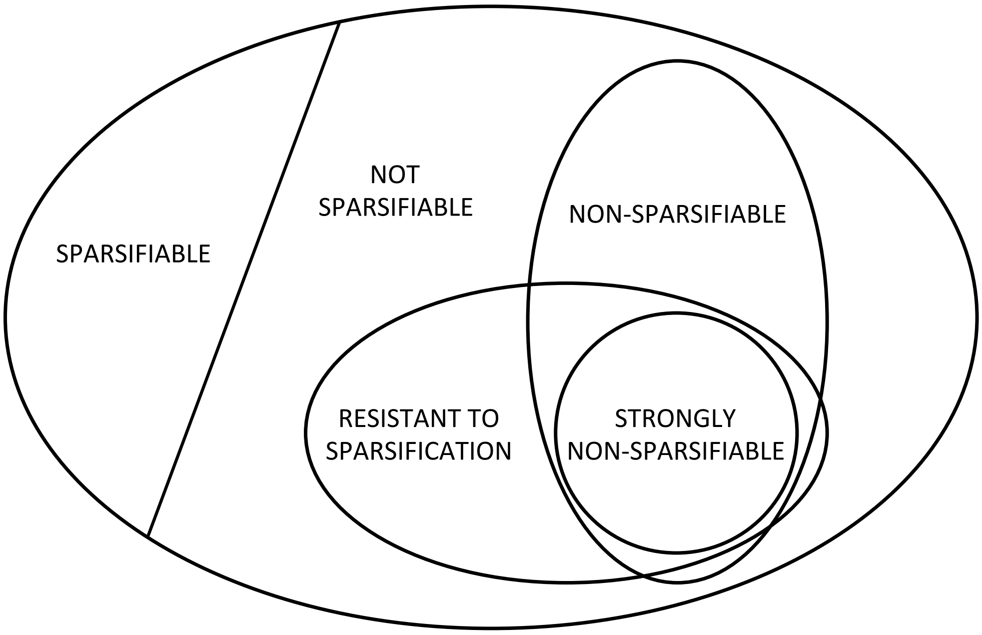


Figure 1.1: A diagram illustrating the relation between different notions of sparsifiable and not sparsifiable predicates.

Throughout, the disjoint union of any two sets S_1, S_2 with $S_1 \cap S_2 = \emptyset$ will be denoted by $S_1 \sqcup S_2$.

Notation. Given an assignment $A : V \to [r]$, we define $S_j := A^{-1}(j) = \{v \in V : A(v) = j\}$. Notice that then $V = \sqcup_{j=0}^{r-1} S_j$, that is, $\mathcal{S} = \{S_0, \dots, S_{r-1}\}$ forms a partition of V. Then, for any predicate $\mathsf{P} : [r]^k \to \{0,1\}$ and any instance $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ we define

$$\mathsf{P}_{\mathcal{I}}(S_0, \dots, S_{r-1}) := Val_{\mathcal{I}}(A) = \sum_{\pi \in \Pi} w(\pi)\mathsf{P}(A(\pi))$$

and similarly, where there is no ambiguity on the predicate, for a weighted directed k-uniform hypergraph $H = (V, E, w)$ we define

$$\mathsf{P}_H(S_0, \dots, S_{r-1}) = Val_H(A) := Val_{\mathcal{I}^{H,\mathsf{P}}}(A).$$

Notice then that $Val_{\mathcal{I}}(A) = Val_{H^{\mathcal{I}}}(A)$.

However, for P of arity 2, we sometimes use the shorthand notation $\mathsf{P}_{\mathcal{I}}(S)$ to define

$$P_{\mathcal{I}}(S) = Val_{\mathcal{I}}(\mathbb{1}_S) = \sum_{\pi \in \Pi} w(\pi)P(\mathbb{1}_S(\pi)).$$

1.2 Related Work

Graph algorithms are widely used in many areas of Computer Science. However, due to increasingly large datasets, even computationally efficient (polynomial time) algorithms can become practically intractable. This is particularly true of algorithms whose runtime depends on the size of the graph's edge set, considering that a k-uniform hypergraph can have up to $\mathcal{O}(n^k)$ hyperedges. If we are prepared to accept an error of order ε in the algorithm's output, graph and hypergraph sparsification allows to run a $T(n, m)$-time algorithm in $T(n, \tilde{\mathcal{O}}(\frac{n}{\varepsilon^2}))$ steps, which can result in a substantial speed-up [EGIN97]. For instance, running the Edmonds-Karp algorithm to find the maximum flow in a dense graph takes $\mathcal{O}(nm^2) = \mathcal{O}(n^5)$ steps. Running the same algorithm on Batson, Spielman and Srivastava's spectral sparsifier [BSS12], which we define in Chapter 3, gives a significantly improved runtime of $\mathcal{O}(\frac{n^3}{\varepsilon^2})$, while the value of the maximum flow in the sparsifier is guaranteed to be within a factor of ε from the value of the maximum flow in the original graph.

A graph r-cut is a partition of its vertices into r non-empty sets. Cuts play a fundamental role in many graph problems such as connectivity, maximum flow, minimum s-t cut, balanced cut, and more. For that matter, early work on sparsification has focused on finding cut sparsifier, that is, reweighted subgraphs and partial subhypergraphs which approximately preserve the value of each cut in the original graph. In 1994 Karger first introduced the notion of a graph skeleton [Kar94b], a subgraph constructed by sampling the edges independently and randomly with small constant probability, and subsequently proved that graph skeletons accurately approximate all cut values in the original graph [Kar94a]. However, the size of Karger's skeleton's edge set is inversely proportional to that of the graph's minimum cut, meaning that graphs with small cuts cannot effectively be sparsified. In 1996, Benczúr and Karger [BK96] provided a new pioneering sampling scheme to construct an ε-Cut-sparsifiers of size $\mathcal{O}(\frac{n \log n}{\varepsilon^2})$ for any graph, irrespective of the size of its minimum cut. The key to such a scheme is to sample edges with nonuniform probabilities, each probability being inversely proportional to the connectivity of any subgraph containing that edge.

All sparsification results found at this point relied on combinatorial and probabilistic techniques. In 2011, Spielman and Teng [ST11] introduced the new notion of Spectral Sparsification, a highly algebraic technique which does not refer explicitly to cuts: indeed, the quantity that the sparsifier is required to approximately preserve is the graph's Laplacian quadratic form. Following on this algebraic approach, Batson et al. [BSS12] gave a deterministic polynomial time algorithm to construct a graph spectral sparsifier (which is also a cut sparsifier) of size $\mathcal{O}(\frac{n}{\varepsilon^2})$, and Andoni et al. [ACK+16] showed that the $\mathcal{O}(\frac{n}{\varepsilon^2})$ lower bound is tight for any cut sparsifier (not necessarily spectral). Recently, Kogan and Krauthgamer [KK15] extended Benczúr and Karger's nonuniform sampling scheme to obtain a combinatorial hypergraph cut sparsifier[2] of size $\mathcal{O}\big(\frac{n(k+\log n)}{\varepsilon^2}\big)$ for any weighted k-uniform hypergraph.[3]

The first attempt at general CSP sparsification was made by Filtser and Krauthgamer [FK17], who proved that a necessary and sufficient condition for the existence of an ε-P-sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ for any binary Boolean predicate $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ is that $|\mathsf{P}^{-1}(1)| \neq 1$. They also proved that for all but a few somewhat trivial binary Boolean predicates the $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ bound is tight. Finally, they provided a relatively few results on non-Boolean and non-binary sparsification.

In this dissertation we extend Filtser and Krauthgamer's work to arbitrary binary VCSPs on a discrete domain, and provide some insight into sparsification of non-binary VCSPs. Finally, we explore the notion of spectral sparsification and prove the sparsifiability of two continuous binary predicates that can be seen as the continuous analogue of Cut and UnCut.

In what follows we list some of the results that will be used throughout this dissertation, as they were presented by their original authors.

Theorem 1.2.1 ([BK96, Theorem 1.2])**.** Given a weighted graph G and an error parameter ε, there is a re-weighted graph G_ε such that G_ε has $\mathcal{O}(\frac{n \log n}{\varepsilon^2})$ edges and the value of every cut in G_ε is within $(1 \pm \varepsilon)$ times the value of the corresponding cut in G. Furthermore, G_ε can be constructed from G in $\mathcal{O}(m \log^3 n)$ time.

Theorem 1.2.2 ([KK15, Theorem 3.1])**.** Given a weighted k-uniform hypergraph H and an error parameter $\varepsilon \in (0,1)$, there is a re-weighted partial subhypergraph H_ε such that H_ε has $\mathcal{O}(\frac{n(k+\log n)}{\varepsilon^2})$ hyperedges and the value of every cut in H_ε is within $(1 \pm \varepsilon)$ times the value of the corresponding cut in H. Furthermore, H_ε can be constructed from H in $\mathcal{O}(mn^2)$ time.

Theorem 1.2.3 ([FK17, Theorem 3.2])**.** For every digraph $G = (V, E, w)$ with strictly positive weights and for every $\varepsilon \in (0,1)$, every ε-And-sparsifier $G_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ must satisfy $|E_\varepsilon| \geq \frac{1}{2}|E|$.

Theorem 1.2.4 ([FK17, Theorem 3.5])**.** For every digraph $G = (V, E, w)$ and for every $\varepsilon \in (0,1)$ there exists a sub-digraph G_ε with $\mathcal{O}\big(\frac{|V|}{\varepsilon^2}\big)$ edges, such that for every predicate $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ with $|\mathsf{P}^{-1}(1)| \neq 1$, the digraph G_ε is an ε-P-sparsifier of G. Note that G_ε does not depend on P.

Theorem 1.2.5 ([FK17, Theorem 3.6])**.** Given parameters n and $m \leq \binom{n}{2}$, there exists a digraph $G = (V, E, w)$ with $2n$ vertices and m edges such that for every $\varepsilon \in (0,1)$ and every predicate $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ with $|\mathsf{P}^{-1}(1)| = 1$, every ε-P-sparsifier $G_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of G has $E_\varepsilon = E$.

[2] A hypergraph cut corresponds to the predicate $\mathsf{Not\text{-}All\text{-}Equal}$ (NAE) defined in Example 1.1.1.

[3] The sparsifier proposed in [KK15] works for all hypergraphs with edges of size less than or equal to k. However, as noted in Remark 1.1.3, throughout this dissertation we restrict our interest to hypergraphs with edges of size exactly k.

Let $\mathcal{P}(S)$ denote the power set of a set S. We define the restriction of predicate $\mathsf{P} : D^2 \to \{0,1\}$ to $B \times C \subseteq D^2$ in the natural way:

Definition 1.2.1. For $B, C \in \mathcal{P}(D)$, we have

$$\mathsf{P}_{|B\times C} : B \times C \to \{0,1\}, \quad \mathsf{P}_{|B\times C}(b,c) := \mathsf{P}(b,c) \quad \forall (b,c) \in B \times C$$

and for all assignments $A : V \to B \cup C$

$$(\mathsf{P}_{|B\times C})_{\mathcal{I}}(A) := \sum_{\substack{\pi\in\Pi \\ A(\pi)\in B\times C}} w(\pi)\mathsf{P}_{|B\times C}(A(\pi)).$$

Notice that B and C do not necessarily need to be disjoint.

The following is a generalisation of Theorems 1.2.4 and 1.2.5 to arbitrary binary 2×2 domains. A detailed proof is provided in Appendix A.

Theorem 1.2.6. For any $\mathsf{P} : D^2 \to \{0,1\}$ and for any subset $B \times C \subseteq D^2$ with $|B| = |C| = 2$, $\mathsf{P}_{|B\times C}$ is sparsifiable if and only if $|\mathsf{P}^{-1}_{|B\times C}(1)| \neq 1$.

Proof. The restriction of P to $B \times C$ is just a predicate in the generalised VCSP framework described in Appendix A. □

Proposition 1.2.1 ([FK17, Section 6.2])**.** Let r-$\mathsf{Cut} : [r]^2 \to \{0,1\}$ be defined as in Example 1.1.1. For every weighted directed graph G, if G_ε is an ε-Cut-sparsifier, then G_ε is also an ε-r-Cut-sparsifier. It follows that r-Cut is sparsifiable.

1.3 Contributions

Our main contribution is the discovery of a new criterion to determine whether a binary predicate is sparsifiable. We begin in Chapter 2 by showing that there is a class of predicates of any arity, which we will call singleton predicates, that are both resistant to sparsification (Proposition 2.3.3) and non-sparsifiable (Proposition 2.3.4).

Given that the conclusion of Filtser and Krauthgamer's work (Theorems 1.2.4 and 1.2.5) is that a binary Boolean predicate is sparsifiable if and only if it is not a singleton, one would be tempted to generalise this classification to predicates on larger domains and higher arities. However, in Section 2.4 we show that this is not the case. Theorem 2.4.1 states that the class of sparsifiable binary predicates is exactly the class of binary predicates that do not *contain* a singleton subpredicate. Indeed, there exist binary non-Boolean and non-binary Boolean predicates that are not sparsifiable but are also not singletons.

We conclude with an overview of spectral sparsification in Chapter 3 and we present two examples of continuous predicates for which we construct a spectral sparsifier.

In Appendix A, we prove that Filtser and Krauthgamer's results on binary Boolean sparsification remain valid if the predicate's domain is an arbitrary 2×2 set $\{b_1, b_2\} \times \{c_1, c_2\}$.

This is of independent theoretical interest, but it is also one of the key steps needed for the proof of Theorem 2.4.1. In Appendix B, we extend Theorem 2.4.1 to arbitrary binary predicates where each of the two variables may take values in different domains, not necessarily of the same size. In Appendix C, we prove two results about the k-ary Parity predicate which demonstrate the complexity of classifying such predicate according to sparsifiability.

Chapter 2

Combinatorial Sparsification

2.1 Graph Covers

The main tool used in the proof of Theorem 1.2.4 is a graph reduction technique known as the bipartite double cover, which we describe below. We generalise this notion to that of a k-partite k-fold cover for hypergraphs.

Definition 2.1.1. Given a weighted directed graph $G = (V, E, w)$, the *bipartite double cover* of G is the directed weighted bipartite graph $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$ where

- $V^\gamma = \{v, -v : v \in V\}$,
- $E^\gamma = \{(u, -v) : (u, v) \in E\}$,
- $w^\gamma(u, -v) = w(u, v)$.

Below is a generalisation of the bipartite double cover to k-uniform hypergraphs, which will be useful in some proofs related to k-ary predicates.

Definition 2.1.2. Given a weighted directed k-uniform hypergraph $H = (V, E, w)$, the *k-partite k-fold cover* of G is the directed weighted hypergraph $\gamma(H) = (V^\gamma, E^\gamma, w^\gamma)$ where

- $V^\gamma = \{v^{(0)}, v^{(1)}, \ldots, v^{(k-1)} : v \in V\}$
- $E^\gamma = \{(v_1^{(0)}, \ldots, v_k^{(k-1)}) : (v_1, \ldots, v_k) \in E\}$
- $w^\gamma((v_1^{(0)}, \ldots, v_k^{(k-1)})) = w((v_1, \ldots, v_k))$.

For any r-partition $P = \{S_0, \ldots, S_{r-1}\}$ of the vertices of V, define $S_i^{(j)} = \{v^{(j)} : v \in S_i\}$. As a result, $P^\gamma := \{S_0^{(0)}, \ldots, S_0^{(k-1)}, \ldots, S_{r-1}^{(0)}, \ldots, S_{r-1}^{(k-1)}\}$ is a kr-partition of the vertices of V^γ.

The following proposition is a generalisation to non-binary, non-Boolean predicates of an argument used in the proof of Theorem 1.2.4. We denote the set of all r-partitions of a set S by $Part_r(S)$.

Proposition 2.1.1. Let $\mathsf{P} : [r]^k \to \{0,1\}$ and $\mathsf{P}' : [r']^k \to \{0,1\}$ be k-ary predicates with $r, r' \in \mathbb{N}$. Suppose that there is a function $f_{\mathsf{P}} : Part_r(V) \to Part_{r'}(V^\gamma)$ such that for any weighted directed k-uniform hypergraph H on V and for any r-partition $\mathcal{S} \in Part_r(V)$ it holds that

$$\mathsf{P}_H(\mathcal{S}) = \mathsf{P}'_{\gamma(H)}(f_{\mathsf{P}}(\mathcal{S})),$$

$\gamma(H) = (V^\gamma, E^\gamma, w^\gamma)$ being the k-partite k-fold cover of H. Then,

1. If P' is sparsifiable, P is also sparsifiable. Moreover, if ε-P'-sparsifiers have size $g(n)$, then there also exist ε-P-sparsifiers of size $\mathcal{O}(g(n))$.

2. If P is resistant to sparsification, then P' is resistant to sparsification.

Proof. Given $H = (V, E, w)$, let $\gamma(H)_\varepsilon = (V, E_\varepsilon^\gamma, w_\varepsilon^\gamma)$ be an ε-P'-sparsifier of the k-partite k-fold cover $\gamma(H)$. Next, construct a partial subhypergraph $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of H such that $E_\varepsilon = \{(v_1, \ldots, v_k) : (v_1^{(0)}, \ldots, v_k^{(k-1)}) \in E_\varepsilon^\gamma\}$ and $w_\varepsilon(v_1, \ldots, v_k) = w_\varepsilon^\gamma(v_1^{(0)}, \ldots, v_k^{(k-1)})$. Notice that $\gamma(H_\varepsilon) = \gamma(H)_\varepsilon$, $E_\varepsilon \subset E$, and H is sparse if and only if $\gamma(H)$ is sparse.
Then, we have

$$\mathsf{P}_{H_\varepsilon}(\mathcal{S}) = \mathsf{P}'_{\gamma(H_\varepsilon)}(f_P(\mathcal{S})) = \mathsf{P}'_{\gamma(H)_\varepsilon}(f_{\mathsf{P}}(\mathcal{S})) \in (1 \pm \varepsilon)\mathsf{P}'_{\gamma(H)}(f_{\mathsf{P}}(\mathcal{S})) = (1 \pm \varepsilon)\mathsf{P}_H(\mathcal{S})$$

and

$$|E_\varepsilon| \leq |E_\varepsilon^\gamma| = \mathcal{O}\Big(\frac{|V^\gamma|(k + \log|V^\gamma|)}{\varepsilon^2}\Big) = \mathcal{O}\Big(\frac{k|V|(k + \log k|V|)}{\varepsilon^2}\Big) = \tilde{\mathcal{O}}\Big(\frac{|V|}{\varepsilon^2}\Big)$$

for constant k, implying that H_ε is also an ε-P-sparsifier of H.
Moreover, $|E_\varepsilon| \leq |E_\varepsilon^\gamma| = g(n) \implies |E_\varepsilon| = \mathcal{O}(g(n))$. $\square$

2.2 Results on Non-Sparsifiability

In this section we provide some negative results on sparsifiability, that is to say, sufficient conditions for a predicate not to be sparsifiable.

Definition 2.2.1. A predicate $\mathsf{P} : D^k \to \{0,1\}$ is a *singleton* if $|\mathsf{P}^{-1}(1)| = 1$.

For any set S and any constant $a \leq |S|$, we denote denote by $\binom{S}{a}$ the set $\{T \subseteq S : |T| = a\}$.

Definition 2.2.2. We say that a binary predicate $\mathsf{P} : D^2 \to \{0,1\}$ *contains a singleton square* if there exist $D_1 = \{d_1^1, d_2^1\}, D_2 = \{d_1^2, d_2^2\} \in \binom{D}{2}$ and $n(1), n(2) \in \{1,2\}$ such that, for $i_1, i_2 \in \{1,2\}$,

$$\mathsf{P}(d_{i_1}^1, d_{i_2}^2) = \begin{cases} 1 & \text{if } i_1 = n(1) \text{ and } i_2 = n(2), \\ 0 & \text{otherwise.} \end{cases}$$

This is equivalent to saying that $\mathsf{P}_{|_{D_1 \times D_2}}$ is a singleton predicate.

Definition 2.2.3. Let S_k denote the set of all permutations of k elements. We say that a k-ary predicate $\mathsf{P} : D^k \to \{0,1\}$ *contains a singleton ℓ-cube* for some $2 \leq \ell \leq k$ if there exist subdomains $\{D_j := \{d_1^j, d_2^j\}\}_{j=1}^{\ell} \in \binom{D}{2}$, integers $\{n(j)\}_{j=1}^{\ell} \in \{1,2\}$, and a permutation $\sigma \in S_k$ such that there exist $x_{\ell+1}, \dots, x_k \in D$ which satisfy

$$\mathsf{P}(\sigma(d_{n(1)}^1, \dots, d_{n(\ell)}^{\ell}, x_{\ell+1}, \dots, x_k)) = 1$$

and for all $y_{\ell+1}, \dots, y_k \in D$, for all $i_j \in \{1,2\}$,

$$\mathsf{P}(\sigma(d_{i_1}^1, \dots, d_{i_\ell}^{\ell}, y_{\ell+1}, \dots, y_k)) = 1 \quad \Longrightarrow \quad i_j = n(j) \text{ for all } j = 1, \dots, \ell.$$

Remark 2.2.1. A singleton square is exactly a singleton 2-cube.

Proposition 2.2.1. Let $\mathsf{P} : D^k \to \{0,1\}$ be a k-ary predicate which contains a singleton ℓ-cube. Then, there exists a k-uniform weighted directed hypergraph $H = (V, E, w)$ such that, for all partial subhypergraphs $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ which satisfy property (1.1), we have $|E_\varepsilon| = \Omega(n^\ell)$.

Proof. Let $\{D_j := \{d_1^j, d_2^j\}\}_{j=1}^{\ell}$ and $\{n(j)\}_{j=1}^{\ell}$ be as in Definition 2.2.3, and for simplicity assume that $\sigma = id$. Define

$$\neg n(j) := i \in \{1,2\} : i \neq n(j).$$

Consider the positively weighted directed hypergraph $H = (V, E, w)$ on $n = kn'$ vertices with $V = V_1 \sqcup \ldots \sqcup V_k$, $|V_i| = \frac{n}{k} = n'$ for $i = 1, \dots, k$, and $E = \{(u_1, \dots, u_k) : u_i \in V_i\}$. Notice that $|E| = \frac{n^k}{k^k}$. Consider a hyperedge $f = (v_1, \dots, v_k) \in E$, and notice that $v_j \in V_j$ for all j. Furthermore, pick some $x_{\ell+1}, \dots, x_k$ such that $\mathsf{P}(d_{n(1)}^1, \dots, d_{n(\ell)}^{\ell}, x_{\ell+1}, \dots, x_k) = 1$.
Define the assignment

$$A^f : V \to D, \qquad \begin{cases} A^f(v_j) = d_{n(j)}^j & \text{for } j \leq \ell, \\ A^f(v) = d_{\neg n(j)}^j \quad \forall v \in V_j \setminus \{v_j\} & \text{for } j \leq \ell, \\ A^f(v) = x_j \quad \forall v \in V_j & \text{for } \ell+1 \leq j \leq k. \end{cases}$$

Notice that $\mathsf{P}(A^f(u_1, \dots, u_k)) = 1 \iff u_j = v_j$ for all $j \leq \ell$, therefore at least one of the $\left(\frac{n}{k}\right)^{k-\ell}$ edges whose first ℓ variables are $v_1, \dots, v_\ell$ must belong to E_ε for property (1.1) to be satisfied. We repeat the same procedure for all $\left(\frac{n}{k}\right)^{\ell}$ combinations of vertices $(u_1, \dots, u_\ell) \in V_1 \times \ldots \times V_\ell$. Therefore we must have $|E_\varepsilon| \geq \frac{n^\ell}{k^\ell} = \Theta(n^\ell)$ and hence $|E_\varepsilon| = \Omega(n^\ell)$ as required. $\square$

Corollary 2.2.1. If a predicate P contain a singleton ℓ-cube for some $\ell \geq 2$, then P is not sparsifiable.

Corollary 2.2.2 (Singleton Square). Let $\mathsf{P} : D^2 \to \{0,1\}$ be a binary predicate which contains a singleton square. Then, P is non-sparsifiable.

Proof. By Proposition 2.2.1, there exists a weighted directed graph $H = (V, E, w)$ such that, for all subgraphs $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ which satisfy property (1.1), it holds that $|E_\varepsilon| = \Omega(n^2)$. Hence, P is non-sparsifiable. □

For a predicate on a domain D, an unused label $z \in D$ is an element of the domain which never appears in the tuples that belong to the predicate's support set. For instance, in the table representation of a binary predicate, z is an unused label if the row and column corresponding to z are all zeros.

Proposition 2.2.2 (Unused Label). Let $\mathsf{P} : D^k \to \{0, 1\}$ be a predicate with $\mathsf{P}^{-1}(1) \neq \emptyset$ and suppose that there exists $z \in D$ such that, for all $x_1, \ldots, x_{k-1} \in D$ and for all permutations $\sigma \in S_k$, $\mathsf{P}(\sigma(x_1, \ldots, x_{k-1}, z)) = 0$. Then, P is strongly non-sparsifiable.

Proof. We proceed by contradiction. Suppose there exists a weighted directed k-uniform hypergraph $H = (V, E, w)$ such that, for all $\varepsilon \in (0, 1)$, there is a partial subhypergraph $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of H which satisfies property 1.1. Consider some tuple $(a_1, \ldots, a_k) \in \mathsf{P}^{-1}(1)$, and notice that we must have $a_j \neq z$ for all j. Pick a hyperedge $e = (u_1, \ldots, u_k) \in E$ and define $U = \{u_1, \ldots, u_k\}$. Define the assignment $A : V \to D$, $A(u_j) = a_j$ for $j = 1, \ldots, k$, $A(v) = z$ for all $v \in V \setminus U$. Notice that the a_j may not be all distinct.
Let $D = [r]$. For $i \in D$, define δ_i to be the number of times i appears in $(a_1, \ldots, a_k)$. Further define

$$M := \prod_{i \in D, \delta_i \neq 0} \delta_i! \quad .$$

Then, there are $M_E \leq M$ hyperedges e in E (including $(u_1, \ldots, u_k)$) such that $\mathsf{P}(A(e)) = 1$. Call these $e_1, \ldots, e_{M_E}$. Then

$$Val_H(A) = \sum_{e \in E} w(e)\mathsf{P}(A(e)) = \sum_{i=1}^{M_E} w(e_i) > 0.$$

Let $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ be an ε-P-sparsifier of H. Then, at least one of $e_1, \ldots, e_{M_E}$ must be in E_ε, since otherwise we would have

$$Val_{H_\varepsilon}(A) = \sum_{e \in E_\varepsilon} w_\varepsilon(e)\mathsf{P}(A(e)) = 0 \notin (1 \pm \varepsilon)Val_H(A).$$

Then, noticing that this argument holds for all hyperedges $e \in E$ and that $M \leq k!$, we have

$$|E_\varepsilon| \geq \frac{1}{M_E}|E| \geq \frac{|E|}{M} \geq \frac{|E|}{k!}.$$

Therefore, we have $|E_\varepsilon| \geq \frac{1}{k!}|E| = \Omega(|E|)$ implying that P is strongly non-sparsifiable. □

Remark 2.2.2. Notice that for $k = 2$, Corollary 2.2.2 implies (weak) non-sparsifiability if the conditions of Proposition 2.2.2 are satisfied. If there is an unused label $z \in D$ and $(b, c) \in P^{-1}(1)$ with $z \neq b, c$, then we can set $B = \{b, z\}$, $C = \{c, z\}$ and apply Corollary 2.2.2 to show that P is non-sparsifiable.

2.3 Singleton Predicates

In this section we generalise Theorem 1.2.5 to show a range of negative results for sparsifiability of singleton predicates of any arity. The main result is that, for any predicate P with $|\mathsf{P}^{-1}(1)| = 1$, the class VCSP(P) is non-sparsifiable and resistant to sparsification. We start by considering binary predicates.

Proposition 2.3.1. Let $\mathsf{P} : D^2 \to \{0,1\}$ with $|D| > 2$. If $|\mathsf{P}^{-1}(1)| = 1$, then P is strongly non-sparsifiable.

Proof. Let $\mathsf{P} : D^2 \to \{0,1\}$ and suppose that $\mathsf{P}^{-1}(1) = \{(x,y)\}$ for some $x, y \in D$. Pick some $z \in D \setminus \{x,y\}$. This is possible since $|D \setminus \{x,y\}| \geq 1$. Then, for all $d \in D$, we have

$$\mathsf{P}(z,d) = \mathsf{P}(d,z) = 0$$

and hence z is an unused label. Hence we can apply Proposition 2.2.2 to deduce that P is strongly non-sparsifiable. $\square$

In the following propositions we extend this result to non-binary predicates. First we consider singleton k-ary predicates with a very specific support set.

Proposition 2.3.2. Let $\mathsf{P} : D^k \to \{0,1\}$ be a singleton predicate with $|D| \geq 2$ such that $\mathsf{P}^{-1}(1) = \{(a,a,\ldots,a)\}$ for some $a \in D$. Then, P is strongly non-sparsifiable.

Proof. By assumption, $\mathsf{P}^{-1}(1) \neq \emptyset$ and $|D \setminus \{a\}| \geq 1$. Notice that, for any $z \in D \setminus \{a\}$, any $x_1, \ldots, x_{k-1} \in D$ and any permutation $\sigma \in S_k$, we have that

$$\mathsf{P}(\sigma(x_1,\ldots,x_{k-1},z)) \neq \mathsf{P}(a,a,\ldots,a) \quad \Longrightarrow \quad \mathsf{P}(\sigma(x_1,\ldots,x_{k-1},z)) = 0.$$

Then, z is an unused label and so by Proposition 2.2.2, P is strongly non-sparsifiable. $\square$

Finally, we consider general k-ary singleton predicates. First we establish resistance to sparsification.

Proposition 2.3.3. If $\mathsf{P} : D^k \to \{0,1\}$ is a singleton predicate, then P is resistant to sparsification.

Proof. Let $H = (V,E,w)$ be a k-uniform weighted directed hypergraph. Suppose that $D = [r]$ and $\mathsf{P}^{-1}(1) = \{(a_1,\ldots,a_k)\}$.
For any assignment $A : V \to D$, denote by $\mathcal{S} := \{S_0, S_1, \ldots, S_{r-1}\} \in Part_r(V)$ the induced r-partition of V. Let nOr:$D^k \to \{0,1\}$ be such that $\mathsf{nOr}^{-1}(1) = \{(0,0,\ldots,0)\}$. Notice that, by Proposition 2.3.2, nOr is strongly non-sparsifiable and hence it is resistant to sparsification. Suppose there is a function $f_{\mathsf{P}} : Part_r(V) \to Part_r(V^\gamma)$ such that for any such $\mathcal{S} \in Part_r(V)$ it holds that

$$\mathsf{nOr}_H(\mathcal{S}) = \mathsf{P}_{\gamma(H)}(f_{\mathsf{P}}(\mathcal{S})).$$

Then, we could apply Proposition 2.1.1 to show that P is resistant to sparsification. It remains to prove that such f_P exists for all singletons $\mathsf{P} : D^k \to \{0, 1\}$.

Claim 2.3.1. Let $S_j^{\mathsf{P}} := \bigcup_{i=1}^{k} S^{(i-1)}_{(j-a_i) \pmod r}$. Then, $f_{\mathsf{P}} : Part_r(V) \to Part_r(V^\gamma)$ is defined by

$$f_{\mathsf{P}}(\{S_0, \dots, S_{r-1}\}) = \{S_0^{\mathsf{P}}, \dots, S_{r-1}^{\mathsf{P}}\}.$$

Proof. Let $\mathsf{P}^{-1}(1) = \{(a_1, \dots, a_k)\} \subset D^k$. We need to prove that

$$\mathsf{nOr}_H(S_0, \dots, S_{r-1}) = \mathsf{P}_{\gamma(H)}(f_{\mathsf{P}}(\{S_0, \dots, S_{r-1}\})).$$

Now define an assignment $A : V \to D$ such that $A(v) = j \iff v \in S_j$. Notice that

$$\mathsf{nOr}_H(A) = \mathsf{nOr}_H(S_0, \dots, S_{r-1}).$$

We let the assignment $A^\gamma : V^\gamma \to D$ be such that $A^\gamma(v^{(i)}) = j \iff v^{(i)} \in S_j^{\mathsf{P}}$. Notice that

$$\mathsf{P}_{\gamma(H)}(A^\gamma) = \mathsf{P}_{\gamma(H)}(\{S_0^{\mathsf{P}}, \dots, S_{r-1}^{\mathsf{P}}\}) =: \mathsf{P}_{\gamma(H)}(f_{\mathsf{P}}(\{S_0, \dots, S_{r-1}\})).$$

Now for any edge $e = (v_1, \dots, v_k) \in E$, by a slight abuse of notation, we define $\gamma(e) = e^\gamma := (v_1^{(0)}, \dots, v_k^{(k-1)})$. We have

$$\begin{aligned}
\mathsf{nOr}(A(e)) = 1 &\iff A(v_1) = \ldots = A(v_k) = 0 \\
&\iff (v_1, \dots, v_k) \in S_0 \times \ldots \times S_0 \\
&\iff \gamma((v_1, \dots, v_k)) = (v_1^{(0)}, \dots, v_k^{(k-1)}) \in S_0^{(0)} \times \ldots \times S_0^{(k-1)}.
\end{aligned}$$

Now, for $i = 1, \dots, k$,

$$\begin{aligned}
S_0^{(i-1)} \subset S_j^{\mathsf{P}} &\iff S_0^{(i-1)} = S^{(i-1)}_{j-a_i \pmod r} && \text{for some } j \in \{0, \dots, r-1\} \\
&\iff j = a_i \pmod r \iff j = a_i && (\text{since } 0 \le a_i, j < r).
\end{aligned}$$

Therefore

$$\begin{aligned}
S_0^{(i-1)} \subset S_j^{\mathsf{P}} &\implies (v_1^{(0)}, \dots, v_k^{(k-1)}) \in S_{a_1}^{\mathsf{P}} \times \ldots \times S_{a_k}^{\mathsf{P}} \\
&\iff A^\gamma(v_1^{(0)}, \dots, v_k^{(k-1)}) = (a_1, \dots, a_k) \\
&\iff \mathsf{P}(A^\gamma(v_1^{(0)}, \dots, v_k^{(k-1)})) = \mathsf{P}(A^\gamma(\gamma(v_1, \dots, v_k))) = \mathsf{P}(A^\gamma(\gamma(e))) = 1.
\end{aligned}$$

Therefore, for any $e \in E$,

$$\mathsf{nOr}(A(e)) = 1 \iff \mathsf{P}(A^\gamma(\gamma(e))) = 1$$

which implies

$$\begin{aligned}\mathsf{nOr}_H(S_0,\dots,S_{r-1}) = \mathsf{nOr}_H(A) &= \sum_{e\in E} w(e)\mathsf{nOr}(A(e)) = \sum_{e\in E} w(e)\mathsf{P}(A^\gamma(\gamma(e)))\\ &= \sum_{e\in E} w^\gamma(\gamma(e))\mathsf{P}(A^\gamma(\gamma(e))) = \sum_{e^\gamma\in E^\gamma} w^\gamma(e^\gamma)\mathsf{P}(A^\gamma(e^\gamma))\\ &= \mathsf{P}_{\gamma(H)}(A^\gamma) = \mathsf{P}_{\gamma(H)}(f_{\mathsf{P}}(\{S_0,\dots,S_{r-1}\}))\end{aligned}$$

and hence

$$\mathsf{nOr}_H(S_0,\dots,S_{r-1}) = \mathsf{P}_{\gamma(H)}(f_{\mathsf{P}}(\{S_0,\dots,S_{r-1}\}))$$

as required. □

□

Second, we establish non-sparsifiability.

Proposition 2.3.4. If $\mathsf{P} : D^k \to \{0,1\}$ is a singleton predicate, then P is non-sparsifiable.

Proof. Suppose that $D = [r]$ and $\mathsf{P}^{-1}(1) = \{(a_1,\dots,a_k)\}$. Consider the weighted directed hypergraph $H = (V, E, w)$ on $n = kn'$ vertices with $V = V_1 \sqcup \ldots \sqcup V_k$, $|V_i| = \frac{n}{k} = n'$ for $i = 1,\dots,k$, and $E = \{(u_1,\dots,u_k) : u_i \in V_i\}$. Notice that $|E| = \frac{n^k}{k^k}$.
Consider a hyperedge $f = (v_1,\dots,v_k) \in E$, and notice that $v_j \in V_j$ for all j. Define the assignment

$$A^f : V \to [r], \qquad \begin{cases} A^f(v_j) = a_j, \\ A^f(v) = 0 \quad \forall v \in V_j \setminus \{v_j\} \quad \text{if } a_j \neq 0, \\ A^f(v) = 1 \quad \forall v \in V_j \setminus \{v_j\} \quad \text{if } a_j = 0. \end{cases}$$

Notice that $\mathsf{P}(A^f(u_1,\dots,u_k)) = 1 \iff u_i = v_j$, therefore

$$Val_H(A) = \sum_{e\in E} w(e)\mathsf{P}(A(e)) = w(f) > 0$$

so we must have $f \in E_\varepsilon$, otherwise it would follow that

$$Val_{H_\varepsilon}(A) = \sum_{e\in E_\varepsilon} w_\varepsilon(e)\mathsf{P}(A(e)) = 0 \notin (1\pm\varepsilon)Val_H(A).$$

Therefore we must have $|E_\varepsilon| = |E| = \frac{n^k}{k^k} = \Omega(n^k)$ and hence P is non-sparsifiable.

□

2.4 A Complete Classification of Binary Predicates

In this section we prove that if a binary predicate P with domain of size $r \geq 2$ contains no singleton squares, then the value of a VCSP(P) instance under any assignment can be expressed as the value of a corresponding VCSP(ℓ-Cut) instance (for some $\ell \leq 2r$) under the same assignment. This will imply that all such predicates are sparsifiable. We begin by defining some useful notation and by proving a technical lemma which will be crucial to the proof of Proposition 2.4.1.

For any graph $G = (V, E)$ and any subset $U \subseteq V$ of the vertices of G, we denote the vertex-induced subgraph with vertex set U by $G(U)$ and its edge set by $E(U)$. For a possibly disconnected graph G, we denote the connected component containing a vertex v by $G_v = (V(G_v), E(G_v))$. Finally, we denote the degree of vertex v in graph G by $d_G(v)$.

Definition 2.4.1. Let $G = (V, E)$ be an undirected bipartite graph with $V = V_1 \sqcup V_2$. We define the *bipartite complement* $\overline{G} = (V, \overline{E})$ of G to be the graph on the same vertex set as G and with edge set

$$\overline{E} := \{\{u, v\} : u \in V_1, v \in V_2, \{u, v\} \notin E\}.$$

Remark 2.4.1. Notice that the operation $\overline{\cdot}$ is the inverse of itself, that is, for any bipartite graph G, $\overline{\overline{G}} = G$.

Lemma 2.4.1. Let $G = (V, E)$ be a bipartite graph on $2r$ vertices ($r \geq 2$), where $V = V_1 \sqcup V_2$ and $|V_1| = |V_2| = r$. If for any $u_1, v_1 \in V_1$ and for any $u_2, v_2 \in V_2$ we have $|E(\{u_1, v_1, u_2, v_2\})| \neq 1$, then for any $v \in V$, $d_{\overline{G}}(v) > 0$ implies that $\overline{G}_v$ is a complete bipartite graph with partition classes $\{V_1 \cap V(\overline{G}_v)\}$, $\{V_2 \cap V(\overline{G}_v)\}$.

Proof. By induction on r.
r=2. Let $G = (V, E)$ with $V = \{u_1, v_1\} \sqcup \{u_2, v_2\}$. There are five non-isomorphic bipartite graphs on $2 \cdot 2$ vertices such that $|E| \neq 1$. Table 2.1 shows that all five bipartite complements on the right satisfy the lemma.

Induction step. Suppose that the lemma holds for $4 \leq |V| \leq 2(r-1)$. We need to prove that it holds for $|V| = 2r$. Let $G = (V, E)$ be a bipartite graph with $V = V_1 \sqcup V_2$ and let $V_j = \{v_0^{(j)}, \dots, v_{r-1}^{(j)}\}$ for $j = 1, 2$. Consider the vertex-induced subgraph $G(U) = (U, E(U))$ with $U = U_1 \sqcup U_2$ and $U_j = \{v_0^{(j)}, \dots, v_{r-2}^{(j)}\}$ for $j = 1, 2$. Suppose that G satisfies the conditions of the Lemma. Then $G(U)$ satisfies such conditions too, and so by the induction hypothesis we have that for any $v \in U$, $d_{\overline{G(U)}}(v) > 0$ implies that $\overline{G(U)}_v$ is a complete bipartite graph with partition classes $\{U_1 \cap V(\overline{G(U)}_v)\}$, $\{U_2 \cap V(\overline{G(U)}_v)\}$. Now consider $\overline{G}$.
Denote each $\{v_i^{(1)}, v_j^{(2)}\}$ by e_{ij}. If for all $i = 0, \dots, r-1$ we have $e_{i,r-1} \notin \overline{E}$ and $e_{r-1,i} \notin \overline{E}$, then there is nothing to prove and the lemma holds.

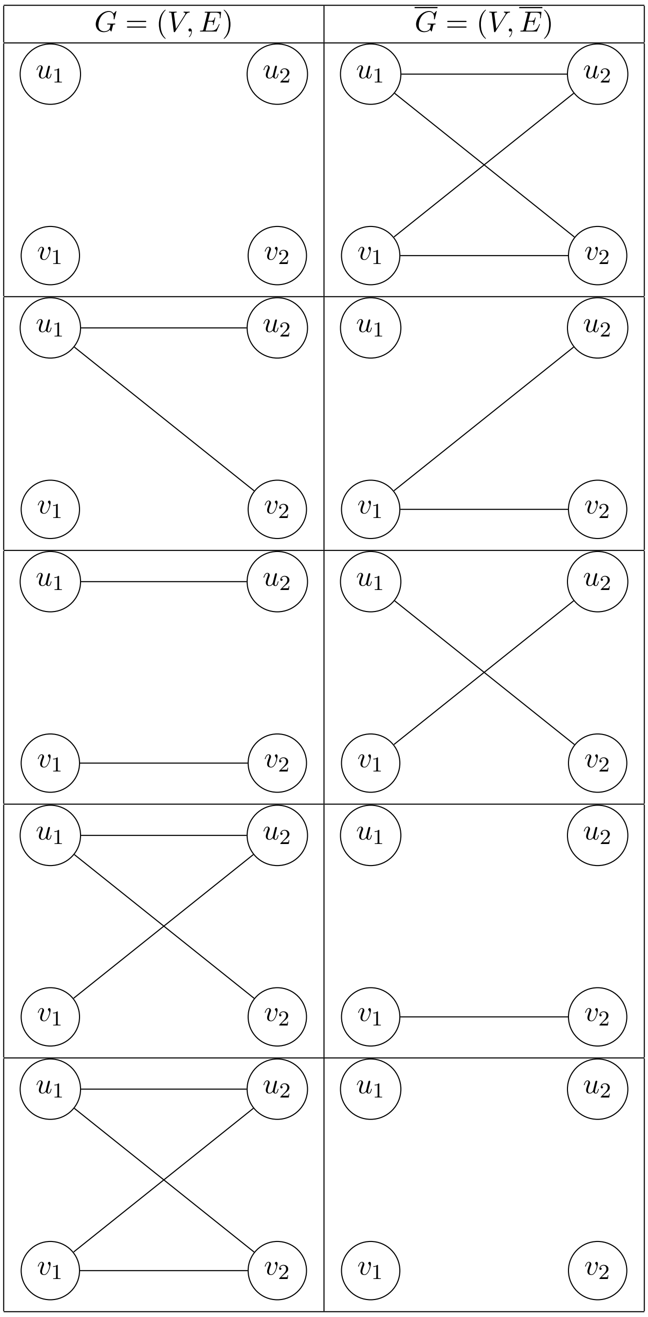
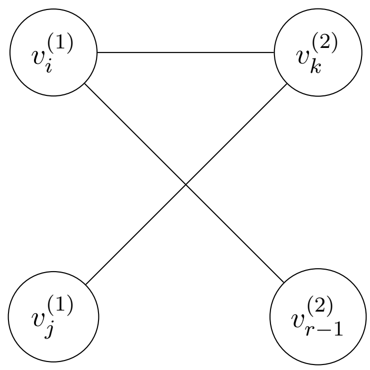
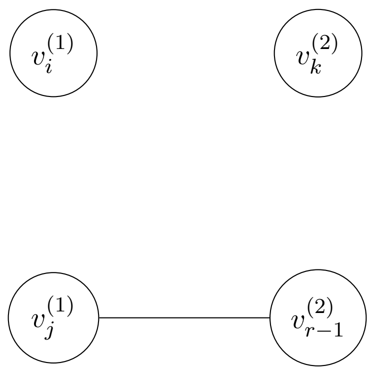


Table 2.1: On the left hand side, the five non-isomorphic bipartite graphs on $2 \cdot 2$ vertices with $|E| \neq 1$. On the right hand side, each of the connected components of the corresponding bipartite complements is a complete bipartite graph.

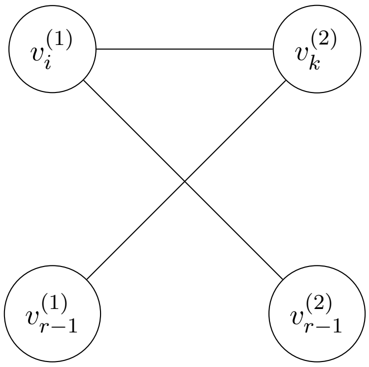
So suppose without loss of generality that $e_{i,r-1} \in \overline{E}$ for some $i \in [r]$. We need to prove that $e_{j,r-1} \in \overline{E}$ for all j such that $v_j^{(1)} \in V(\overline{G}_{v_i^{(1)}})$.
Suppose not. Then there exists $j \in [r]$ such that $v_j^{(1)} \in V(\overline{G}_{v_i^{(1)}})$ and $e_{j,r-1} \notin \overline{E}$. If $j \neq r-1$, then by the induction hypothesis $v_i^{(1)}, v_j^{(1)}$ belong to the complete bipartite graph $\overline{G(U)}_{v_i^{(1)}}$, so for all $v \in U_2 \cap V(\overline{G(U)}_{v_i^{(1)}})$ we have $\{v_i^{(1)}, v\}, \{v_j^{(1)}, v\} \in \overline{E}$. Notice that $U_2 \cap V(\overline{G(U)}_{v_i^{(1)}}) \neq \emptyset$ since $v_i^{(1)}, v_j^{(1)}$ are path-connected. So choose any $k \in \{0, \dots, r-2\}$ such that $v_k^{(2)} \in U_2 \cap V(\overline{G(U)}_{v_i^{(1)}})$. Then, $e_{ik}, e_{jk} \in \overline{E}$ and we have the following picture for $\overline{G}(\{v_i^{(1)}, v_j^{(1)}, v_k^{(2)}, v_{r-1}^{(2)}\})$



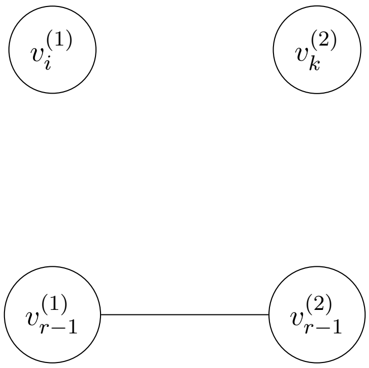
which means that in $\overline{\overline{G}} = G$ we have the following picture for $G(\{v_i^{(1)}, v_j^{(1)}, v_k^{(2)}, v_{r-1}^{(2)}\})$



contradicting the initial assumption that for any $u_1, v_1 \in V_1$ and for any $u_2, v_2 \in V_2$ we have $|E(\{u_1, v_1, u_2, v_2\})| \neq 1$.
On the other hand, if $j = r-1$, there must exist some $k \in \{0, \dots, r-2\}$ such that $e_{r-1,k} \in \overline{E}$. Now by the induction hypothesis, $v_i^{(1)}, v_k^{(2)}$ belong to the complete bipartite graph $\overline{G(U)}_{v_i^{(1)}}$, implying that $e_{ik} \in \overline{E}$. Then, we have the following picture for $\overline{G}(\{v_i^{(1)}, v_{r-1}^{(1)}, v_k^{(2)}, v_{r-1}^{(2)}\})$



which means that in $\overline{\overline{G}} = G$ we have the following picture for $G(\{v_i^{(1)}, v_{r-1}^{(1)}, v_k^{(2)}, v_{r-1}^{(2)}\})$



contradicting the initial assumption that for any $u_1, v_1 \in V_1$ and for any $u_2, v_2 \in V_2$ we have $|E(\{u_1, v_1, u_2, v_2\})| \neq 1$.

Then, we deduce that $v_{r-1}^{(2)}$ is connected to all the vertices in $V_1 \cap \overline{G}_{v_{r-1}^{(2)}}$ and hence $\overline{G}_{v_{r-1}^{(2)}}$ is complete bipartite with the required vertex set. By symmetry, the same argument holds for $\overline{G}_{v_{r-1}^{(1)}}$. □

Proposition 2.4.1. Let $\mathsf{P} : [r]^2 \to \{0, 1\}$ be a binary predicate which contains no singleton squares. Then, P is sparsifiable and there exists an ε-P-sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$.

Proof. Let $G = (V, E, w)$ be an arbitrary weighted directed graph and $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$ be its bipartite double cover. For an assignment $A : V \to [r]$, recall that $S_i := A^{-1}(i)$ and hence $\mathcal{S} := \{S_0, \ldots, S_{r-1}\}$ forms an r-partition of V. Suppose that for a predicate $\mathsf{P} : [r]^2 \to \{0, 1\}$ there is a function $f_\mathsf{P} : Part_r(V) \to Part_\ell(V^\gamma)$ (for some $\ell \leq 2r$) such that for any weighted directed graph G on V it holds that

$$\mathsf{P}_G(\mathcal{S}) = \ell\text{-}\mathsf{Cut}_{\gamma(G)}(f_\mathsf{P}(\mathcal{S})).$$

Recall that the binary ℓ-Cut predicate is sparsifiable (Proposition 1.2.1). Then, by Proposition 2.1.1, P is also sparsifiable. Furthermore, given that there exists an ε-Cut-sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ (Theorem 1.2.4) and that every ε-Cut-sparsifier is also an ε-ℓ-Cut-sparsifier (Proposition 1.2.1), it follows that there exists an ε-P-sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$. It remains to prove that such a function f_{P} exists fort all binary predicates P which contain no singleton squares.

Now for $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ and $|\mathsf{P}^{-1}(1)| \in \{0,2,4\}$, a suitable function $f_{\mathsf{P}} : Part_2(V) \to Part_2(V^\gamma)$ is provided in the proof of Theorem 1.2.4. For $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ and $|\mathsf{P}^{-1}(1)| = 3$, we have $\ell = 3$ and

- $\mathsf{P}_1^{-1}(1) = \{(0,0),(0,1),(1,0)\} \quad \Longrightarrow \quad f_{\mathsf{P}_1}(\{S_0,S_1\}) = \{S_0, S_0', S_1 \cup S_1'\}$
- $\mathsf{P}_2^{-1}(1) = \{(0,0),(0,1),(1,1)\} \quad \Longrightarrow \quad f_{\mathsf{P}_2}(\{S_0,S_1\}) = \{S_0, S_1', S_1 \cup S_0'\}$
- $\mathsf{P}_3^{-1}(1) = \{(0,0),(1,0),(1,1)\} \quad \Longrightarrow \quad f_{\mathsf{P}_3}(\{S_0,S_1\}) = \{S_0', S_1, S_0 \cup S_1'\}$
- $\mathsf{P}_4^{-1}(1) = \{(0,1),(1,0),(1,1)\} \quad \Longrightarrow \quad f_{\mathsf{P}_4}(\{S_0,S_1\}) = \{S_1, S_1', S_0 \cup S_0'\}.$

More generally, for predicates $\mathsf{P} : [r]^2 \to \{0,1\}$ with $r \geq 2$ which contain no singleton squares, we show that a suitable function f_{P} always exists.

Consider a weighted directed graph $G = (V, E, w)$ and its undirected[1] bipartite double cover $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$. Define an auxiliary graph $G^{\mathsf{P},r} := (V^{\mathsf{P},r}, E^{\mathsf{P},r})$, where

$$V^{\mathsf{P},r} := \{v_0, v_0', \ldots, v_{r-1}, v_{r-1}'\},$$

$$E^{\mathsf{P},r} := \{\{v_i, v_j'\} : \mathsf{P}(i,j) = 1\}.$$

Let ℓ be the number of connected components of $\overline{G^{\mathsf{P},r}}$. Clearly $\ell \leq |V^{P,r}| = 2r$.

Now, a function $f_{\mathsf{P}} : Part_r(V) \to Part_\ell(V^\gamma)$ which satisfies $\mathsf{P}_G(\mathcal{S}) = \ell\text{-}\mathsf{Cut}_{\gamma(G)}(f_{\mathsf{P}}(\mathcal{S}))$ for all $\mathcal{S} \in Part_r(V)$ corresponds to a colouring $c : V^{\mathsf{P},r} \to \{0, \ldots, \ell-1\}$ of the vertices of $G^{\mathsf{P},r}$ with the following property:

$$\forall i,j \in [r] \quad \begin{cases} \{v_i, v_j'\} \in E^{\mathsf{P},r} \implies c(v_i) \neq c(v_j') \\ \{v_i, v_j'\} \notin E^{\mathsf{P},r} \implies c(v_i) = c(v_j'). \end{cases}$$

We call colour maps which satisfy the above property *proper efficient* colourings. Indeed, the colouring c induces an assignment $A^\gamma : V^\gamma \to \{0, \ldots, \ell-1\}$ of the vertices of $\gamma(G)$ which satisfies

$$A^\gamma(u) = c(v_{A(u)}), \qquad A^\gamma(u') = c(v'_{A(u)})$$

and which in turn induces a partition $\{U_i\}_{i=0}^{\ell-1}$ of V^γ with $U_i := (A^\gamma)^{-1}(i)$. We define $f_{\mathsf{P}}(\mathcal{S}) := \{U_0, \ldots, U_{\ell-1}\}$.

[1] Originally, we had defined the bipartite double cover as a directed graph. However, here it is easier to deal with undirected graphs, as since ℓ-Cut is a symmetric predicate, the direction of the edges makes no difference. Furthermore, notice that by the way the bipartite double cover is constructed, removing the direction does *not* turn the graph into a multigraph.

Now for any $u, t \in V$ and for any assignment $A : V \to [r]$, we have

$$\begin{aligned}\mathsf{P}(A(u), A(t)) = 1 &\iff \{v_{A(u)}, v'_{A(t)}\} \in E^{\mathsf{P},r} \\ &\iff c(v_{A(u)}) \neq c(v'_{A(t)}) \\ &\iff A^\gamma(u) \neq A^\gamma(t') \\ &\iff \ell\text{-}\mathsf{Cut}(A^\gamma(u), A^\gamma(t')) = 1\end{aligned}$$

Moreover, by the definition of the graph bipartite double cover, we have $w(u,t) = w^\gamma(u,t')$ for all $u, t \in V$, implying that

$$\begin{aligned}\mathsf{P}_G(S_0, \ldots, S_{r-1}) = \mathsf{P}_G(A) &= \sum_{(u,t)\in E} w(u,t)\mathsf{P}(A(u), A(t)) \\ &= \sum_{(u,t')\in E^\gamma} w^\gamma(u,t')\ell\text{-}\mathsf{Cut}(A^\gamma(u), A^\gamma(t')) = \ell\text{-}\mathsf{Cut}_{\gamma(G)}(A^\gamma) \\ &= \ell\text{-}\mathsf{Cut}_{\gamma(G)}(U_0, \ldots, U_{\ell-1}) = \ell\text{-}\mathsf{Cut}_{\gamma(G)}(f_{\mathsf{P}}(\mathcal{S}))\end{aligned}$$

as required.
Now, a proper efficient colouring does not exist for any arbitrary bipartite graph. In the remainder of this proof, we show that a proper efficient colouring exists if the auxiliary graph $G^{\mathsf{P},r}$ arises from a predicate P which contains no singleton squares.
Consider the graph $G^{\mathsf{P},r}$ for some predicate P which contains no singleton squares. Given that for any $B, C \in \binom{[r]}{2}$ we have $|\mathsf{P}^{-1}_{|B\times C}(1)| \neq 1$, for any $b_1, b_2 \in V_1$, $c_1, c_2 \in V_2$ we must have $|E(\{b_1, b_2, c_1, c_2\})| \neq 1$, implying that $G^{\mathsf{P},r}$ satisfies the conditions of Lemma 2.4.1. Therefore, the ℓ separate connected components which form its bipartite complement $\overline{G^{\mathsf{P},r}}$ are complete bipartite graphs. Then, we can assign one of ℓ colours to each connected component to get a proper efficient colouring for the graph $G^{\mathsf{P},r}$. Now, $G^{\mathsf{P},r}$ has a proper efficient colouring and hence we can construct a suitable map f_{P} as specified above, implying that P is indeed sparsifiable and that its sparsifiers have size $\mathcal{O}\left(\frac{n}{\varepsilon^2}\right)$. □

Theorem 2.4.1 (Main Classification Theorem)**.** Let $\mathsf{P} : D^2 \to \{0,1\}$ be a predicate. Then P is sparsifiable if and only if for any subset $B \times C \subseteq D^2$ with $B, C \in \binom{D}{2}$, $\mathsf{P}_{|B\times C}$ is sparsifiable. Moreover, if P is sparsifiable, it has a sparsifier of size $\mathcal{O}\left(\frac{n}{\varepsilon^2}\right)$.

Proof. ($\Leftarrow$). Assume that for any $B, C \in \binom{D}{2}$, $\mathsf{P}_{|B\times C}$ is sparsifiable. Then, by Theorem 1.2.6, $|\mathsf{P}^{-1}_{|B\times C}(1)| \neq 1$ for all $B, C \in \binom{D}{2}$. This means that P contains no singleton squares, and therefore by Proposition 2.4.1, P is sparsifiable and its sparsifiers have size $\mathcal{O}\left(\frac{n}{\varepsilon^2}\right)$.
($\Rightarrow$). Assume by contradiction that there exist $B, C \in \binom{D}{2}$ such that $\mathsf{P}_{|B\times C}$ is not sparsifiable. By Theorem 1.2.6, we must have $|\mathsf{P}^{-1}_{|B\times C}(1)| = 1$. Therefore, $\mathsf{P}_{|B\times C}$ is a singleton predicate and hence P contains a singleton square. Then, by Corollary 2.2.2, P is non-sparsifiable. □

For some predicates of specific support size, we can draw a conclusion on whether the predicate is sparsifiable simply by inspecting the number of tuples in the support.

Theorem 2.4.2. For any predicate $\mathsf{P} : [r]^2 \to \{0, 1\}$,

1. $|\mathsf{P}^{-1}(1)| = 0 \implies \mathsf{P}$ is sparsifiable
2. $0 < |\mathsf{P}^{-1}(1)| < r \implies \mathsf{P}$ is non-sparsifiable
3. $|\mathsf{P}^{-1}(1)| \geq r^2 - 2 \implies \mathsf{P}$ is sparsifiable
4. For all $1 \leq j < r^2 - 2$, there exists a non-sparsifiable predicate P with $|\mathsf{P}^{-1}(1)| = j$.

Proof of 1: Every sparse subgraph is an ε-P-sparsifier .

Proof of 2: Suppose $0 < |\mathsf{P}^{-1}(1)| < r$ and assume by contradiction that P is sparsifiable. Then, by Corollary 2.2.2, for all subdomains $B \times C \in \binom{[r]}{2}^2$ we have $|\mathsf{P}^{-1}_{|B\times C}(1)| \neq 1$. There are two possibilities:

1. There exist $B, C \in \binom{[r]}{2}$ such that $|\mathsf{P}^{-1}_{|B\times C}(1)| = 2$. In this case, let $B = \{b_1, b_2\}$, $C = \{c_1, c_2\}$, and suppose without loss of generality that $\mathsf{P}(b_1, c_1) = \mathsf{P}(b_1, c_2) = 1$, $\mathsf{P}(b_2, c_1) = \mathsf{P}(b_2, c_2) = 0$ [2]. Now consider the subdomains $B \times C_x := \{b_1, b_2\} \times \{c_1, x\}$ for all $x \in [r] \setminus \{c_1, c_2\}$. Since we are assuming that $|\mathsf{P}^{-1}_{|B\times C_x}(1)| \neq 1$, it must hold that for all $x \in [r] \setminus \{c_1, c_2\}$ there exists some $y \in B$ such that $\mathsf{P}_{|B\times C_x}(y, x) = 1$. But this implies that $|\mathsf{P}^{-1}(1)| \geq (r - 2) + 2 = r$ which contradicts $|\mathsf{P}^{-1}(1)| < r$. Then, P is not sparsifiable.

2. For all $B, C \in \binom{[r]}{2}$, $|\mathsf{P}^{-1}_{|B\times C}(1)| = 0$ or $|\mathsf{P}^{-1}_{|B\times C}(1)| \geq 3$. Now a simple double counting argument gives

$$\sum_{B,C\in\binom{[r]}{2}} |\mathsf{P}^{-1}_{|B\times C}(1)| = \frac{4\binom{r}{2}^2}{r^2} |\mathsf{P}^{-1}(1)| = (r-1)^2 |\mathsf{P}^{-1}(1)|$$

and so, if it were the case that for all $B, C \in \binom{[r]}{2}$, $|\mathsf{P}^{-1}_{|B\times C}(1)| \geq 3$, we would have

$$|\mathsf{P}^{-1}(1)| = \frac{1}{(r-1)^2} \sum_{B,C\in\binom{[r]}{2}} |\mathsf{P}^{-1}_{|B\times C}(1)| \geq \frac{1}{(r-1)^2} \binom{r}{2}^2 \cdot 3 = \frac{3}{4} r^2 \geq r \quad \forall r \geq 2$$

which again contradicts $|\mathsf{P}^{-1}(1)| < r$, so we can assume that there exists some $B, C \in \binom{[r]}{2}$ such that $|\mathsf{P}^{-1}_{|B\times C}(1)| = 0$. Let $\gamma \in C$. Then, for all $C' \in \binom{[r]}{2}$ with $\gamma \in C'$, we have $|\mathsf{P}^{-1}_{|B\times C'}(1)| \leq 2$ since $\mathsf{P}_{|B\times C'}(b, \gamma) = 0$ for all $b \in B$. But we assumed that $|\mathsf{P}^{-1}_{|B\times C'}(1)| \neq 1, 2$ hence we must have $|\mathsf{P}^{-1}_{|B\times C'}(1)| = 0$ for all C' as above. Then, $\mathsf{P}(b, d) = 0$ for all $b \in B$ and all $d \in D$. Similarly, let $\beta \in B$. Then, for all $B' \in \binom{[r]}{2}$

[2] The proof goes in the same way for $\mathsf{P}^{-1}(1) = \{(b_i, c_j), (b_k, c_\ell)\}$ for any other combination of $i, j, k, \ell \in \{1, 2\}$ with $(i, j) \neq (k, \ell)$.

with $\beta \in B'$, we have $|\mathsf{P}^{-1}_{|B'\times C}(1)| \leq 2$ since $\mathsf{P}_{|B'\times C}(\beta, c) = 0$ for all $c \in C$. But we assumed that $|\mathsf{P}^{-1}_{|B'\times C}(1)| \neq 1, 2$ hence we must have $|\mathsf{P}^{-1}_{|B'\times C}(1)| = 0$ for all B' as above. Then, $\mathsf{P}(d, c) = 0$ for all $c \in C$ and all $d \in D$. Given that $|\mathsf{P}^{-1}(1)| > 0$, let $(x, y) \in \mathsf{P}^{-1}(1)$. Then, for $b \in B$ and $c \in C$, $\mathsf{P}_{|\{b,x\}\times\{c,y\}}$ is a singleton predicate and therefore by Corollary 2.2.2, P is non-sparsifiable, a contradiction.

Proof of 3: If $|\mathsf{P}^{-1}(1)| \geq r^2 - 2$, then for all $B, C \in \binom{[r]}{2}$ we have $|\mathsf{P}^{-1}_{|B\times C}(1)| \geq 2$. Then, $\mathsf{P}_{|B\times C}$ is sparsifiable for all $B, C \in \binom{[r]}{2}$ and therefore by Theorem 2.4.1, P is sparsifiable.

Proof of 4: It is enough to choose $\mathsf{P}(0, 0) = 1$, $\mathsf{P}(0, 1) = \mathsf{P}(1, 0) = \mathsf{P}(1, 1) = 0$: any such predicate must have $|\mathsf{P}^{-1}(1)| < r^2 - 2$ and is non-sparsifiable due to containing the singleton predicate $\mathsf{P}_{|\{0,1\}\times\{0,1\}}$. □

2.4.1 The Binary Product Predicate

Motivated by Filtser and Krauthgamer's efforts to classify the binary Sum predicate [FK17, Section 6.2], we consider the binary Product predicate

$$\mathsf{Prod}^r_a : [r]^2 \to \{0, 1\}, \quad \mathsf{Prod}^r_a(x, y) = 1 \quad \iff \quad xy = a \pmod r$$

for $r \geq 3$.

Corollary 2.4.1 (of Proposition 2.2.2, Corollary 2.2.2, and Theorem 2.4.1)**.** Consider the binary Product predicate $\mathsf{Prod}^r_a : [r]^2 \to \{0, 1\}$.

1. If $a \neq 0 \pmod r$, then Prod^r_a is strongly non-sparsifiable;
2. If $a = 0 \pmod r$ and k is not a prime, then Prod^r_a is non-sparsifiable;
3. If $a = 0 \pmod r$ and k is a prime, then Prod^r_a is sparsifiable.

Proof of 1: Since $a \neq 0 \pmod r$, it holds that for all $x \in [r]$, $\mathsf{Prod}^r_a(0, x) = \mathsf{Prod}^r_a(x, 0) = 0$. Moreover, $\mathsf{Prod}^r_a(a, 1) = 1$ and so $(\mathsf{Prod}^r_a)^{-1}(1) \neq \emptyset$. Then, 0 is an unused label and so by Proposition 2.2.2, Prod^r_a is strongly non-sparsifiable.

Proof of 2: Since r is not a prime, there exist $x, y \in [r] \setminus \{0, 1\}$ such that $xy = r$. Therefore, $xy = 0 = a \pmod r$ and so $\mathsf{Prod}^r_a(x, y) = 1$. Conversely, $\mathsf{Prod}^r_a(x, 1) = \mathsf{Prod}^r_a(1, y) = 0$ since $x \neq 0 \pmod r$ and $y \neq 0 \pmod r$. It follows that $\mathsf{Prod}^r_{a|\{1,x\}\times\{1,y\}}$ is a singleton predicate and therefore by Corollary 2.2.2, Prod^r_a is non-sparsifiable.

Proof of 3: Notice that, since $a = 0 \pmod r$ and r is a prime, $\mathsf{Prod}^r_a(x, y) = a$ if and only if $x = 0 \pmod r$ or $y = 0 \pmod r$. Pick any 2×2 subdomain $B \times C \in \binom{[r]}{2}^2$ and let $B = \{b_1, b_2\}$, $C = \{c_1, c_2\}$. Now, if $0 \in \{b_1, b_2, c_1, c_2\}$, we have $|\mathsf{Prod}^r_a|^{-1}_{B\times C}(1)| \geq 2$ and so $\mathsf{Prod}^r_{a|B\times C}$ is sparsifiable. Else, if $0 \notin \{b_1, b_2, c_1, c_2\}$, we have that $\mathsf{Prod}^r_a(b_i, c_j) \neq 0 \pmod r$ for all $i, j \in \{1, 2\}$, hence $|\mathsf{Prod}^r_a|^{-1}_{B\times C}(1)| = 0$ and so $\mathsf{Prod}^r_{a|B\times C}$ is sparsifiable. Then, by Theorem 2.4.1, Prod^r_a is sparsifiable. □

2.4.2 The Binary Parity Predicate

In [FK17, Section 6.1], Filtser and Krauthgamer introduce the problem of sparsifiability of the Parity predicate, which we define below. While we do not yet have results for the Parity predicate of arbitrary arity, we can classify the binary Parity predicate on any domain.

Definition 2.4.2. The Parity predicate $\mathsf{Par}^0 : [r]^k \to \{0,1\}$ is the k-ary predicate which satisfies

$$\mathsf{Par}^0(x_1,\ldots,x_k) = 1 \quad\iff\quad \sum_{i=1}^{k} x_i = 0 \pmod 2.$$

The Imparity predicate $\mathsf{Par}^1 : [r]^k \to \{0,1\}$ is the k-ary predicate which satisfies

$$\mathsf{Par}^1(x_1,\ldots,x_k) = 1 \quad\iff\quad \sum_{i=1}^{k} x_i = 1 \pmod 2.$$

Corollary 2.4.2 (of Theorem 2.4.1)**.** The binary predicates $\mathsf{Par}^0 : [r]^2 \to \{0,1\}$ and $\mathsf{Par}^1 : [r]^2 \to \{0,1\}$ are sparsifiable.

Proof. Define the parity function $h : [r] \to \{0,1\}$, $h(x) = x \pmod 2$. Notice that

$$\mathsf{Par}^0(x,y) = 1 \iff h(x) = h(y),$$

$$\mathsf{Par}^1(x,y) = 1 \iff h(x) \neq h(y).$$

Pick any 2×2 subdomain $B \times C \in \binom{[r]}{2}^2$ and let $B = \{b_1, b_2\}$, $C = \{c_1, c_2\}$. Assume by contradiction that $\mathsf{Par}^0_{|B\times C}$ is a singleton predicate. Without loss of generality, take $\mathsf{Par}^0(b_1,c_1) = 1$ and $\mathsf{Par}^0(b_1,c_2) = \mathsf{Par}^0(b_2,c_1) = \mathsf{Par}^0(b_2,c_2) = 0$. Then we must have $h(b_1) = h(c_1)$, $h(b_1) \neq h(c_2)$ and $h(c_1) \neq h(b_2)$, which by the pigeonhole principle implies that $h(b_2) = h(c_2)$, a contradiction. Therefore, $\mathsf{Par}^0_{|B\times C}$ is sparsifiable for all $B, C \in \binom{[r]}{2}$ and so by Theorem 2.4.1, Par^0 is sparsifiable.
Similarly, assume by contradiction that $\mathsf{Par}^1_{|B\times C}$ is a singleton predicate. Without loss of generality, take $\mathsf{Par}^1(b_1,c_1) = 1$ and $\mathsf{Par}^1(b_1,c_2) = \mathsf{Par}^1(b_2,c_1) = \mathsf{Par}^1(b_2,c_2) = 0$. Then we must have $h(b_1) \neq h(c_1)$, $h(b_1) = h(c_2)$ and $h(c_1) = h(b_2)$, which by transitivity implies that $h(b_2) \neq h(c_2)$, a contradiction. Therefore, $\mathsf{Par}^1_{|B\times C}$ is sparsifiable for all $B, C \in \binom{[r]}{2}$ and so by Theorem 2.4.1, Par^1 is sparsifiable. $\square$

2.5 Tools and Techniques for Future Work

In this section we present a number of results which may be used as a tool for future research to tackle the sparsifiability problem of non-binary VCSP classes. We first establish that sparsifiability of multigraphs is equivalent to the sparsifiability of graphs with no multiple edges. Subsequently we use graphs and multigraphs to prove some reduction techniques between different VCSP classes.

2.5.1 A Note on Multisets

To begin, we extend [FK17, Theorem 5.1] to non-Boolean, non-binary predicates. Our proof is similar in spirit, but we add a necessary observation on the size of the sparsifier.

Proposition 2.5.1. Let $\mathcal{I} \in \mathrm{VCSP}(\Gamma)$ be an instance with $\ell = |\Gamma|$ constant. If all the predicates occurring in the constraints of $\mathcal{I}$ are sparsifiable, then for every $\varepsilon \in (0,1)$ there exists an ε-sparsifier of $\mathcal{I}$.

Proof. Let $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\Gamma)$ where $\Gamma = \{\mathsf{P}_j\}_{j=1}^{\ell}$ for some constant ℓ and suppose that $\mathsf{P}_1, \ldots, \mathsf{P}_\ell$ are sparsifiable. For each $\mathsf{P}_j : D^{k_j} \to \{0,1\}$ define

$$\Pi^{\mathsf{P}_j} := \{\pi \in \Pi : \pi = \langle e, \mathsf{P}_j \rangle\}$$

for some $e \in V^{k_j}$. Notice that $\{\Pi^{\mathsf{P}_j}\}_{j=1}^{\ell}$ forms a partition of Π. For each predicate P_j we further define

$$\mathcal{I}^{\mathsf{P}_j} := (V, \Pi^{\mathsf{P}_j}, w^{\mathsf{P}_j})$$

where $w^{\mathsf{P}_j} = w|_{\Pi^{\mathsf{P}_j}}$ is the restriction of w to Π^{P_j}. Now each P_j is sparsifiable, so let $\mathcal{I}_\varepsilon^{\mathsf{P}_j} = (V, \Pi_\varepsilon^{\mathsf{P}_j}, w_\varepsilon^{\mathsf{P}_j})$ be an ε-P_j-sparsifier of $\mathcal{I}^{\mathsf{P}_j}$. Recall that $\Pi_\varepsilon^{\mathsf{P}_j} \subset \Pi^{\mathsf{P}_j}$ and $|\Pi_\varepsilon^{\mathsf{P}_j}| = \mathcal{O}\big(\frac{|V|(K+\log|V|)}{\varepsilon^2}\big)$, where $K := \max_{j \in \{1,\ldots,\ell\}} k_j$. We construct an ε-sparsifier $\mathcal{I}_\varepsilon = (V, \Pi_\varepsilon, w_\varepsilon)$ of $\mathcal{I}$ where

$$\Pi_\varepsilon := \bigsqcup_{j=1}^{\ell} \Pi_\varepsilon^{\mathsf{P}_j}$$

and

$$w_\varepsilon(\pi) = w_\varepsilon^{\mathsf{P}_j}(\pi) \text{ for } \pi \in \Pi_\varepsilon^{\mathsf{P}_j}.$$

Notice that $\Pi_\varepsilon = \sqcup_{j=1}^{\ell} \Pi_\varepsilon^{\mathsf{P}_j} \subset \sqcup_{j=1}^{\ell} \Pi^{\mathsf{P}_j} = \Pi$, and moreover

$$\begin{aligned}|\Pi_\varepsilon| &= \sum_{j=1}^{\ell} |\Pi_\varepsilon^{\mathsf{P}_j}| = \sum_{j=1}^{\ell} \mathcal{O}\Big(\frac{|V|(K+\log|V|)}{\varepsilon^2}\Big) = \mathcal{O}\Big(\sum_{j=1}^{\ell} \frac{|V|(K+\log|V|)}{\varepsilon^2}\Big) \\ &= \mathcal{O}\Big(\ell \frac{|V|(K+\log|V|)}{\varepsilon^2}\Big) = \mathcal{O}\Big(\frac{|V|(K+\log|V|)}{\varepsilon^2}\Big)\end{aligned}$$

since ℓ is constant. Finally, given that for all $j \in \{1, \ldots, \ell\}$ and for all assignments $A : V \to D$ we have $\mathsf{P}_{j,\mathcal{I}_\varepsilon^{\mathsf{P}_j}} \in (1 \pm \varepsilon)\mathsf{P}_{j,\mathcal{I}^{\mathsf{P}_j}}$, notice that

$$\begin{aligned} Val_{\mathcal{I}_\varepsilon}(A) &= \sum_{\pi \in \Pi_\varepsilon} \sum_{j=1}^{\ell} w_\varepsilon(\pi)\mathsf{P}_j(A(\pi)) = \sum_{j=1}^{\ell} \sum_{\pi \in \Pi_\varepsilon^{\mathsf{P}_j}} w_\varepsilon^{\mathsf{P}_j}(\pi)\mathsf{P}_j(A(\pi)) = \sum_{j=1}^{\ell} \mathsf{P}_{j,\mathcal{I}_\varepsilon^{\mathsf{P}_j}} \\ &\in (1 \pm \varepsilon) \sum_{j=1}^{\ell} \mathsf{P}_{j,\mathcal{I}^{\mathsf{P}_j}} = (1 \pm \varepsilon) \sum_{j=1}^{\ell} \sum_{\pi \in \Pi^{\mathsf{P}_j}} w^{\mathsf{P}_j}(\pi)\mathsf{P}_j(A(\pi)) \\ &= (1 \pm \varepsilon) \sum_{\pi \in \Pi} \sum_{j=1}^{\ell} w(\pi)\mathsf{P}_j(A(\pi)) = (1 \pm \varepsilon) Val_{\mathcal{I}}(A) \end{aligned}$$

therefore, $\mathcal{I}_\varepsilon$ is an ε-sparsifier of $\mathcal{I}$. $\square$

The following propositions show that a predicate's sparsifiability is not affected by the constraint set being a multiset.

Proposition 2.5.2. Suppose $\mathsf{P}_1, \dots, \mathsf{P}_\ell : D^k \to \{0,1\}$ are sparsifiable predicates and let $\Gamma = \{\mathsf{P}_j\}_{j=1}^{\ell}$. Then, for any instance $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\Gamma)$ where Π is allowed to be a multi-set and ℓ is constant, there exists a sub-instance $\mathcal{I}_\varepsilon = (V, \Pi_\varepsilon, w_\varepsilon)$ with $\Pi_\varepsilon \subset \Pi$, $|\Pi_\varepsilon| = \tilde{\mathcal{O}}\big(\frac{|V|}{\varepsilon^2}\big)$ and, for any assignment $A : V \to D$,

$$Val_{\mathcal{I}_\varepsilon}(A) \in (1 \pm \varepsilon) Val_{\mathcal{I}}(A).$$

Proof. Let $\mathcal{I} = (V, \Pi, w) \in \mathrm{VCSP}(\Gamma)$ with Π a multiset. We construct a new instance $\mathcal{I}' = (V, \Pi', w') \in \mathrm{VCSP}(\Gamma)$ as follows. We let Π' be the underlying set of Π, that is, a set where each element of Π occurs exactly once. Further define the set

$$E := \{e \in V^k : \exists j \in \{1, \dots, \ell\} : \langle e, \mathsf{P}_j \rangle \in \Pi\}.$$

For any k-tuple $e \in E$ and for any predicate P_j we define the multiset $\Pi_e^{\mathsf{P}_j} := \{\pi \in \Pi : \pi = \langle e, \mathsf{P}_j \rangle\}$. Notice that

$$\bigcup_{e \in E} \bigcup_{j=1}^{\ell} \Pi_e^{\mathsf{P}_j} = \Pi \quad \text{and} \quad \bigcap_{e \in E} \bigcap_{j=1}^{\ell} \Pi_e^{\mathsf{P}_j} = \emptyset \quad :$$

$\{\Pi_e^{\mathsf{P}_j}\}_{j=1, e \in E}^{\ell}$ forms a partition of Π.
We define $w' : \Pi' \to \mathbb{R}^+$ such that, for $\pi' = \langle e, \mathsf{P}_j \rangle \in \Pi'$,

$$w'(\pi') = w'(\langle e, \mathsf{P}_j \rangle) := \sum_{\pi \in \Pi_e^{\mathsf{P}_j}} w(\pi) \quad :$$

essentially, the sum of the weights of all the constraints that have the same scope and are subject to the same predicate as π'. Now, for any assignment $A : V \to D$,

$$\begin{aligned} Val_{\mathcal{I}'}(A) &= \sum_{\substack{\pi' \in \Pi' \\ \pi' = \langle e, \mathsf{P}_j \rangle}} w'(\pi') \mathsf{P}_j(A(\pi')) = \sum_{\substack{\pi' \in \Pi' \\ \pi' = \langle e, \mathsf{P}_j \rangle}} \Big(\sum_{\pi \in \Pi_e^{\mathsf{P}_j}} w(\pi) \Big) \mathsf{P}_j(A(\pi')) \\ &= \sum_{\substack{\pi \in \Pi \\ \pi = \langle e, \mathsf{P}_j \rangle}} w(\pi) \mathsf{P}_j(A(\pi)) = Val_{\mathcal{I}}(A) \end{aligned}$$

Now the only predicates appearing in the constraints of $\mathcal{I}'$ are $\mathsf{P}_1, \dots, \mathsf{P}_\ell$, which are assumed to be sparsifiable. Hence by Theorem 2.5.1, there exists an ε-sparsifier for $\mathcal{I}'$, which we denote by $\mathcal{I}'_\varepsilon = (V, \Pi'_\varepsilon, w'_\varepsilon)$. Then, for any assignment $A : V \to D$, we have

$$Val_{\mathcal{I}'_\varepsilon}(A) \in (1 \pm \varepsilon) Val_{\mathcal{I}'}(A) = (1 \pm \varepsilon) Val_{\mathcal{I}}(A).$$

Notice that $\Pi'_\varepsilon \subset \Pi' \subset \Pi$ and $|\Pi'_\varepsilon| = \mathcal{O}\big(\frac{|V|}{\varepsilon^2}\big)$, therefore, $\mathcal{I}'_\varepsilon$ is also an ε-sparsifier for $\mathcal{I}$. $\square$

Proposition 2.5.3. Suppose $\mathsf{P} : D^k \to \{0, 1\}$ is resistant to sparsification. Then, for any weighted directed k-uniform multihypergraph $H = (V, E, w)$, H does not have an ε-P-sparsifier.

Proof. Let $H = (V, E, w)$ be a weighted directed multihypergraph. We introduce the notation $\mathcal{DH}(n)$, $\mathcal{DH}^m(n)$ to denote the set of directed hypergraphs and multihypergraphs respectively on n vertices. We define a function $\Phi : \mathcal{DH}^m(n) \to \mathcal{DH}(n)$ such that for $H = (V, E, w) \in \mathcal{DH}^m(n)$ we have

$$\Phi(H) = (V, E^\Phi, w^\Phi)$$

where E^Φ is the underlying set of E. For any hyperedge $e = (v_1, \dots, v_k) \in E$ we define the multiset $E_e := \{f \in E : f = (v_1, \dots, v_k)\}$. Notice that $\{E_e\}_{e \in E}$ forms a partition of E. We define $w^\Phi : E^\Phi \to \mathbb{R}^+$ such that, for $e \in E^\Phi$,

$$w^\Phi(e) = \sum_{f \in E_e} w(f) \quad :$$

essentially, the sum of the weights of all hyperedges on the same k-tuple in E. Notice that, by the construction of Φ, we must have $|E^\Phi| \leq |E|$ and, for any assignment $A : V \to D$,

$$\begin{aligned} Val_{\Phi(H)}(A) &= \sum_{e \in E^\Phi} w^\Phi(e) \mathsf{P}(A(e)) = \sum_{e \in E^\Phi} \Big(\sum_{f \in E_e} w(f) \Big) \mathsf{P}(A(e)) \\ &= \sum_{f \in E} w(f) \mathsf{P}(A(f)) = Val_H(A) \end{aligned}$$

by noticing that $\mathsf{P}(A(f)) = \mathsf{P}(A(e))$ for all $f \in E_e$ and for all assignments A.
Now suppose by contradiction that $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ is an ε-P-sparsifier of H, and consider $\Phi(H_\varepsilon) = (V, E_\varepsilon^\Phi, w_\varepsilon^\Phi)$. Notice that

$$E_\varepsilon^\Phi \subset E^\Phi,$$

$$|E_\varepsilon^\Phi| \leq |E_\varepsilon| = \mathcal{O}\Big(\frac{|V|(k + \log |V|)}{\varepsilon^2}\Big)$$

and, for any assignment $A : V \to D$,

$$Val_{\Phi(H_\varepsilon)}(A) = Val_{H_\varepsilon}(A) \in (1 \pm \varepsilon) Val_H(A) = Val_{\Phi(H)}(A)$$

implying that $\Phi(H_\varepsilon)$ is an ε-P-sparsifier of $\Phi(H)$, which is a contradiction as $\Phi(H) \in \mathcal{DH}(n)$ is a regular hypergraph and P is resistant to sparsification on hypergraphs. □

2.5.2 Map Reductions

In this section we present a number of results aimed at proving sparsifiability of predicates of higher arity by showing that they are reducible to other predicates of possibly different arity and domain.

Proposition 2.5.4. Let $\mathsf{P} : D^k \to \{0,1\}$ and $\mathsf{P}' : D'^k \to \{0,1\}$ be predicates. Suppose there is a map $h : D' \to D$ such that for all $x_1, \ldots, x_k \in D'$ we have

$$\mathsf{P}'(x_1, \ldots, x_k) = 1 \iff \mathsf{P}(h(x_1), \ldots, h(x_k)) = 1.$$

Then, P is sparsifiable implies that P' is sparsifiable.

Proof. Suppose that P is sparsifiable. We need to prove that P' is also sparsifiable. Since P is sparsifiable, for any weighted directed k-uniform hypergraph $H = (V, E, w)$ and for any $\varepsilon \in (0,1)$ there exists an ε-P-sparsifier $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ with $E_\varepsilon \subset E$ and $|E_\varepsilon| = \mathcal{O}\big(\frac{|V|(k+\log|V|)}{\varepsilon^2}\big)$ such that, for any assignment $A : V \to D$,

$$\mathrm{Val}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathrm{Val}_H(A).$$

Claim 2.5.1. H_ε is also an ε-P'-sparsifier.

Proof. For any assignment $A' : V \to D'$, define the assignment

$$A : V \to D, \quad A(v) = h(A'(v)) \quad \text{for } v \in V.$$

Then,

$$\begin{aligned}\mathsf{P}'_H(A') &= \sum_{e \in E} w(e)\mathsf{P}'(A'(e)) = \sum_{e \in E} w(e)\mathsf{P}(h(A'(e))) \\ &= \sum_{e \in E} w(e)\mathsf{P}(A(e)) = \mathsf{P}_H(A)\end{aligned}$$

and similarly $\mathsf{P}'_{H_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A)$, so that

$$\mathsf{P}'_{H_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}_H(A) = (1 \pm \varepsilon)\mathsf{P}'_H(A')$$

which implies that H_ε is an ε-P'-sparsifier of H. □

□

Remark 2.5.1. Given Proposition 2.5.4, one could conjecture that every sparsifiable predicate on any domain can be reduced to a Boolean sparsifiable predicate via a map. However, this does not hold. For instance, $3\text{-}\mathsf{Cut} : \{0,1,2\}^2 \to \{0,1\}$ is sparsifiable (Proposition 1.2.1), but there is no map $h : \{0,1,2\} \to \{0,1\}$ and predicate $\mathsf{P} : \{0,1\}^2 \to \{0,1\}$ as described above. Indeed, if such h existed, we would need

$$\mathsf{P}(h(z), h(z)) = 0 \text{ for all } z \in \{0,1,2\}$$

and

$$\mathsf{P}(h(x), h(y)) = 1 \text{ for all } x \neq y.$$

But by the pigeonhole principle, we must have that for some $x, y \in \{0,1,2\}$, $h(x) = h(y)$ and hence $\mathsf{P}(h(x), h(y)) = \mathsf{P}(h(x), h(x)) = 0$, a contradiction.

Remark 2.5.2. The conditions of Proposition 2.5.4 are *not* sufficient to show the converse, i.e., that P' is sparsifiable implies that P is sparsifiable. To illustrate this we provide a counterexample. Consider the predicates

$$\mathsf{P} = \mathsf{And} : \{0,1\}^2 \to \{0,1\} \quad \mathsf{P}(x,y) = 1 \iff x = y = 1$$

and

$$\mathsf{P}' = \overrightarrow{1} : \{0,1\}^2 \to \{0,1\}, \quad \mathsf{P}'(x,y) = 1 \quad \forall x,y \in \{0,1\}.$$

Define the map $h : \{0,1\} \to \{0,1\}$, $h(0) = h(1) = 1$. It can be easily verified that $\mathsf{P}'(x,y) = 1 \iff \mathsf{P}(h(x),h(y)) = \mathsf{P}(1,1) = 1$, hence all the conditions of Proposition 2.5.4 hold. However, $\mathsf{P}' = \overrightarrow{1}$ is sparsifiable, but $\mathsf{P} = \mathsf{And}$ is not.

Remark 2.5.3. In Proposition 2.5.4, it is *not* sufficient to relax the condition

$$\mathsf{P}'(x,y) = 1 \iff \mathsf{P}(h(x),h(y)) = 1$$

to

$$\mathsf{P}'(x,y) = 1 \implies \mathsf{P}(h(x),h(y)) = 1.$$

To illustrate this we provide a counterexample. Consider the predicates

$$\mathsf{P} = \overrightarrow{1} : \{0,1\}^2 \to \{0,1\}, \quad \mathsf{P}(x,y) = 1 \quad \forall x,y \in \{0,1\}$$

and

$$\mathsf{P}' = \mathsf{And} : \{0,1\}^2 \to \{0,1\}, \quad \mathsf{P}'(x,y) = 1 \iff x = y = 1.$$

Then, any map $h : \{0,1\} \to \{0,1\}$ satisfies $\mathsf{P}'(x,y) = 1 \implies \mathsf{P}(h(x),h(y)) = 1$. However, $\mathsf{P} = \overrightarrow{1}$ is sparsifiable, but $\mathsf{P}' = \mathsf{And}$ is not.

Remark 2.5.4. In Proposition 2.5.4, it is also *not* sufficient to modify the condition

$$\mathsf{P}'(x,y) = 1 \iff \mathsf{P}(h(x),h(y)) = 1$$

to the existence of two maps

$$h : D' \to D, \quad g : D \to D'$$

such that

$$\mathsf{P}'(x,y) = 1 \implies \mathsf{P}(h(x),h(y)) = 1 \quad \text{and} \quad \mathsf{P}(x,y) = 1 \implies \mathsf{P}'(g(x),g(y)) = 1.$$

To illustrate this we provide a counterexample. Consider the predicates

$$\mathsf{P} = \overrightarrow{1} : \{0,1\}^2 \to \{0,1\}, \quad \mathsf{P}(x,y) = 1 \quad \forall x,y \in \{0,1\}$$

and

$$\mathsf{P}' = \mathsf{And} : \{0,1\}^2 \to \{0,1\}, \quad \mathsf{P}'(x,y) = 1 \iff x = y = 1.$$

Then, any map $h : \{0,1\} \to \{0,1\}$ satisfies $\mathsf{P}'(x,y) = 1 \implies \mathsf{P}(h(x),h(y)) = 1$. Moreover, define the map $g : \{0,1\} \to \{0,1\}$, $g(0) = g(1) = 1$. Notice that $\mathsf{P}(x,y) = 1$ implies $\mathsf{P}'(g(x),g(y)) = \mathsf{And}(1,1) = 1$. However, $\mathsf{P} = \overrightarrow{1}$ is sparsifiable, but $\mathsf{P}' = \mathsf{And}$ is not.

Notation. For a set $U \subset D^k$ and a map $h : D \to C$, we define

$$h(U) := \{(h(u_1),\ldots,h(u_k)) : (u_1,\ldots,u_k) \in U\}.$$

Corollary 2.5.1. A permutation on the arguments of P does not affect sparsifiability.

Proof. Suppose $D = [r]$ and let $\sigma \in S_r$ be a permutation of the elements of D. Let $\mathsf{P} : D^k \to \{0,1\}$ be a predicate and define $\mathsf{Q} : D^k \to \{0,1\}$ such that $\mathsf{Q}^{-1}(1) = \sigma(\mathsf{P}^{-1}(1))$. Notice that then $\sigma : D \to D$ is such that, for all $x_1,\ldots,x_k \in D$,

$$\mathsf{P}(x_1,\cdots,x_k) = 1 \iff \mathsf{Q}(\sigma(x_1),\cdots,\sigma(x_k)) = 1$$

and hence by Proposition 2.5.4, Q is sparsifiable implies P is sparsifiable.
Moreover, permutations on finite sets are bijections and hence σ has an inverse $\sigma^{-1} : D \to D$. Then,

$$\begin{aligned}\mathsf{Q}(x_1,\cdots,x_k) = 1 &\iff \mathsf{Q}(\sigma(\sigma^{-1}(x_1)),\cdots,\sigma(\sigma^{-1}(x_k))) = 1\\ &\iff \mathsf{P}(\sigma^{-1}(x_1),\cdots,\sigma^{-1}(x_k)) = 1\end{aligned}$$

and hence by Proposition 2.5.4, P is sparsifiable implies Q is sparsifiable.
Therefore, P is sparsifiable if and only if Q is sparsifiable.

□

Proposition 2.5.5. Let $\mathsf{P} : D^k \to \{0,1\}$ and $\mathsf{P}' : D'^{\,k+1} \to \{0,1\}$ be predicates. Suppose there is a map $h : D' \to D$ such that for all $x_1,\ldots,x_k,z \in D'$,

$$\mathsf{P}'(x_1,\ldots,x_k,z) = 1 \iff \mathsf{P}(h(x_1),\ldots,h(x_k)) = 1.$$

Then, P is sparsifiable implies that P' is sparsifiable.

Proof. Suppose that P is sparsifiable. We need to prove that P' is also sparsifiable. For any weighted directed $(k+1)$-uniform hypergraph $H' = (V,E',w')$ we can construct a corresponding k-uniform multihypergraph $H = (V,E,w)$ as follows. Define an injective map

$$\phi : E' \to E, \quad \phi(v_1,\ldots,v_k,v_{k+1}) = (v_1,\ldots,v_k).$$

We define $E := \phi(E')$ to be the image of this map. Notice that due to the construction of ϕ, E is possibly a multiset. However, since we assume that P is sparsifiable, Proposition

2.5.2 implies that the argument used in this proof is still valid. Similarly, $w : E \to \mathbb{R}^+$ is defined such that $w(\phi(e')) = w'(e')$ for any hyperedge $\phi(e') \in E$.
Now, assuming that P is sparsifiable, let $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ be an ε-P-sparsifier of H. From H_ε, we construct a partial subhypergraph $H'_\varepsilon = (V, E'_\varepsilon, w'_\varepsilon)$ of H', where

$$E'_\varepsilon = \{e' \in E' : \phi(e') \in E_\varepsilon\}$$

and

$$w'_\varepsilon(e') = w_\varepsilon(\phi(e')) \text{ for any } e' \in E'_\varepsilon.$$

Notice that $E'_\varepsilon \subset E'$ and that $|E'_\varepsilon| = |E_\varepsilon| = \mathcal{O}\big(\frac{|V|(k+\log|V|)}{\varepsilon^2}\big) = \mathcal{O}\big(\frac{|V|((k+1)+\log|V|)}{\varepsilon^2}\big)$.

$$\begin{tikzcd} E' \arrow[r, dotted, "\supset"] \arrow[d, "\phi"'] & E'_\varepsilon \\ E \arrow[r, "\supset"] & E_\varepsilon \arrow[u] \end{tikzcd}$$

Claim 2.5.2. H'_ε is an ε-P'-sparsifier of H'.

Proof. For any assignment $A' : V \to D'$, define the assignment

$$A : V \to D, \quad A(v) = h(A'(v)) \quad \text{for } v \in V.$$

Then,

$$\begin{aligned} \mathsf{P}'_{H'}(A') &= \sum_{e' \in E'} w'(e')\mathsf{P}'(A'(e')) = \sum_{e' \in E'} w(\phi(e'))\mathsf{P}(h(A'(e'))) \\ &= \sum_{e' \in E'} w(\phi(e'))\mathsf{P}(A(\phi(e'))) = \sum_{e \in E} w(e)\mathsf{P}(A(e)) = \mathsf{P}_H(A) \end{aligned}$$

and similarly $\mathsf{P}'_{H'_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A)$.
Recall that, since H_ε is an ε-P-sparsifier of H, for any assignment $A : V \to D$ we have

$$\mathsf{P}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}_H(A),$$

therefore

$$\mathsf{P}'_{H'_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}_H(A) = (1 \pm \varepsilon)\mathsf{P}'_{H'}(A')$$

which implies that H'_ε is an ε-P'-sparsifier of H'. □

□

In what follows we extend Proposition 2.5.5 to more general predicates $\mathsf{P}' : D'^{\,k+t} \to \{0, 1\}$ for any $t \in \mathbb{N}$.

Proposition 2.5.6. Let $\mathsf{P} : D^k \to \{0,1\}$ and $\mathsf{P}' : D'^{\,k+t} \to \{0,1\}$ be predicates. Suppose there is a map $h : D' \to D$ such that for all $x_1, \ldots, x_k, z_1, \ldots, z_t \in D'$,

$$\mathsf{P}'(x_1, \ldots, x_k, z_1, \ldots, z_t) = 1 \iff \mathsf{P}(h(x_1), \ldots, h(x_k)) = 1.$$

Then, P is sparsifiable implies that P' is sparsifiable.

Proof. By induction on t.
The case $t = 1$ follows immediately from Proposition 2.5.5. So suppose that the statement holds for all $t \leq s-1$. We need to prove that it holds for $t = s$. We assume that there is a map $h : D' \to D$ such that for all $x_1, \ldots, x_k, z_1, \ldots, z_s \in D'$,

$$\mathsf{P}'(x_1, \ldots, x_k, z_1, \ldots, z_s) = 1 \iff \mathsf{P}(h(x_1), \ldots, h(x_k)) = 1.$$

Notice that the value of $\mathsf{P}'(y_1, \ldots, y_{k+s})$ is entirely determined by its first k arguments: then, for any $d, d' \in D'$, we have that $\mathsf{P}'(y_1, \ldots, y_{k+s-1}, d) = \mathsf{P}'(y_1, \ldots, y_{k+s-1}, d')$. We denote this value by $\mathsf{P}'(y_1, \ldots, y_{k+s-1}, \bullet)$.
Define a new predicate $\mathsf{P}'' : D' \to \{0,1\}^{k+s-1}$,

$$\mathsf{P}''(y_1, \ldots, y_{k+s-1}) = \mathsf{P}'(y_1, \ldots, y_{k+s-1}, \bullet) \qquad \text{for all } y_1, \ldots, y_{k+s-1} \in D'.$$

Notice that then, for all $x_1, \ldots, x_k, z_1, \ldots, z_s \in D'$,

$$\begin{aligned} \mathsf{P}(h(x_1), \ldots, h(x_k)) = 1 &\iff \mathsf{P}'(x_1, \ldots, x_k, z_1, \ldots, z_{s-1}, z_s) = \mathsf{P}'(x_1, \ldots, x_k, z_1, \ldots, z_{s-1}, \bullet) = 1 \\ &\iff \mathsf{P}''(x_1, \ldots, x_k, z_1, \ldots, z_{s-1}) = 1 \end{aligned}$$

so by the induction hypothesis,

$$\mathsf{P} \text{ is sparsifiable } \implies \mathsf{P}'' \text{ is sparsifiable.} \tag{2.1}$$

Now consider the predicates $\mathsf{P}'' : D'^{\,k+s-1} \to \{0,1\}$, $\mathsf{P}' : D'^{\,k+s} \to \{0,1\}$, and define the map $h' = id_{D'} : D' \to D'$. Then for any $x_1, \ldots, x_{k+s-1}$ and any $z \in D'$ we have

$$\mathsf{P}''(h'(x_1), \ldots, h'(x_{k+s-1})) = \mathsf{P}''(x_1, \ldots, x_{k+s-1}) = 1 \iff \mathsf{P}'(x_1, \ldots, x_{k+s-1}, z) = 1$$

and hence by Proposition 2.5.5,

$$\mathsf{P}'' \text{ is sparsifiable } \implies \mathsf{P}' \text{ is sparsifiable.} \tag{2.2}$$

Putting together (2.1) and (2.2), we obtain that

$$\mathsf{P} \text{ is sparsifiable } \implies \mathsf{P}' \text{ is sparsifiable.}$$

□

Proposition 2.5.7. Let $\mathsf{P} : D^k \to \{0,1\}$ and $\mathsf{P}' : D'^{\,k-1} \to \{0,1\}$ be predicates. Suppose there is a map $h : D' \to D$ such that for all $x_1, \ldots, x_{k-1}, y \in D'$ it hold that

$$\mathsf{P}'(x_1, \ldots, x_{k-1}) = 1 \iff \mathsf{P}(h(x_1), \ldots, h(x_{k-1}), h(y)) = 1.$$

Then, P is sparsifiable implies that P' is sparsifiable.

Proof. Suppose that P is sparsifiable. We need to prove that P' is also sparsifiable. For any weighted directed $(k-1)$-uniform hypergraph $H' = (V, E', w')$ we can construct a corresponding k-uniform hypergraph $H = (V, E, w)$ as follows. Pick some arbitrary variable $u \in V$ and define a map

$$\phi_u : E' \to E, \quad \phi_u(v_1, \ldots, v_{k-1}) = (v_1, \ldots, v_{k-1}, u).$$

We define $E := \phi_u(E')$ to be the image of this map. Similarly, $w : E \to \mathbb{R}^+$ is defined such that $w(\phi_u(e')) = w'(e')$ for any hyperedge $\phi_u(e') \in E$.
Now, assuming that P is sparsifiable, let $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ be an ε-P-sparsifier of H. From H_ε, we construct a partial subhypergraph $H'_\varepsilon = (V, E'_\varepsilon, w'_\varepsilon)$ of H' where

$$E'_\varepsilon = \{e' \in E' : \phi_u(e') \in E_\varepsilon\}$$

and

$$w'_\varepsilon(e') = w_\varepsilon(\phi_u(e')) \text{ for any } e' \in E'_\varepsilon.$$

Notice that $E'_\varepsilon \subset E'$ and that $|E'_\varepsilon| = |E_\varepsilon| = \mathcal{O}\big(\frac{|V|(k+\log|V|)}{\varepsilon^2}\big) = \mathcal{O}\big(\frac{|V|((k-1)+\log|V|)}{\varepsilon^2}\big)$.

$$\begin{tikzcd}
E' \arrow[r, dotted, "\supset"] \arrow[d, "\phi_u"'] & E'_\varepsilon \\
E \arrow[r, "\supset"] & E_\varepsilon \arrow[u]
\end{tikzcd}$$

Claim 2.5.3. H'_ε is an ε-P'-sparsifier of H'.

Proof. For any assignment $A' : V \to D'$, define the assignment

$$A : V \to D, \quad A(v) = h(A'(v)) \quad \text{for } v \in V.$$

Then,

$$\begin{aligned}\mathsf{P}'_{H'}(A') &= \sum_{e'\in E'} w'(e')\mathsf{P}'(A'(e')) = \sum_{e'\in E'} w(\phi_u(e'))\mathsf{P}(h(A'(e'))) \\ &= \sum_{e'\in E'} w(\phi_u(e'))\mathsf{P}(A(\phi_u(e'))) = \sum_{e\in E} w(e)\mathsf{P}(A(e)) = \mathsf{P}_H(A)\end{aligned}$$

and similarly $\mathsf{P}'_{H'_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A)$.
Recall that, since H_ε is an ε-P-sparsifier of H, for any assignment $A : V \to D$ we have

$$\mathsf{P}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}_H(A),$$

therefore

$$\mathsf{P}'_{H'_\varepsilon}(A') = \mathsf{P}_{H_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}_H(A) = (1 \pm \varepsilon)\mathsf{P}'_{H'}(A')$$

which implies that H'_ε is an ε-P'-sparsifier of H'. □

□

Chapter 3

Spectral Sparsification

3.1 Spectral Sparsifiers and Continuous VCSPs

In previous sections we have discussed extensively how valued CSPs are essentially weighted directed hypergraphs augmented by some relations on the hyperedges. When we restrict ourselves to instances $\mathcal{I} \in \mathrm{VCSP}(\mathsf{P})$ where P is binary and symmetric, we are essentially looking at weighted undirected graphs whose edges are all subject to the same predicate.[1] As mentioned in Section 1.2, there are two main approaches to graph sparsification. The first, which we have implicitly referred to throughout Chapter 2, is Benczúr and Karger's Combinatorial Sparsification [BK96], which achieves sparsifiers of size quasilinear in n. On the other hand, the notion of Spectral Sparsification has recently proved to be very fruitful for finding linear sparsifiers. It was first introduced in 2011 by Spielman and Teng [ST11]. Successively, Batson, Spielman and Srivastava [BSS12] gave a construction of a Spectral Sparsifier that is the first deterministic graph sparsifier of linear size. In this section, we define the concept of spectral sparsification, and show that two continuous binary predicates admit a spectral sparsifier. To the best of our knowledge, this is the first attempt towards sparsification of continuous VCSPs.

Definition 3.1.1. Let $G = (V, E, w)$ be an undirected positively weighted graph on n vertices. Define

$$w_{ij} = \begin{cases} w(\{v_i, v_j\}) & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Let $A_G = \left(a_{ij}\right)_{i,j=1}^n = \left(w_{ij}\right)_{i,j=1}^n$ be the weighted adjacency matrix of G and $D_G = (d_{ij})_{i,j=1}^n$

[1] The equivalence is given by considering the underlying directed graph of $\mathcal{I}$, ignoring the direction of each edge, and - given that the resulting graph is possibly a multigraph - substituting for each set of edges between vertices u and v a single undirected edge with weight equal to the sum of the weights of said edges. It is straigthforward that the value of the resulting undirected graph under any assignment corresponds to the value of $\mathcal{I}$ under the same assignment.

be the diagonal matrix of weighted degrees, where

$$d_{ij} = \begin{cases} \displaystyle\sum_{\{v_i,v_k\}\in E} w_{ik} & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

The *Laplacian* of graph G is defined as

$$L_G := D_G - A_G.$$

Definition 3.1.2. Let $G = (V, E, w)$ be an undirected positively weighted graph on n vertices. Define the weighted adjacency matrix A_G and the diagonal matrix of weighted degrees D_G as in Definition 3.1.1. The *Negated Laplacian* of G is defined as

$$U_G := D_G + A_G.$$

Notice that, since for an undirected graph $w_{ij} = w_{ji}$, A_G, L_G, U_G and trivially D_G are symmetric matrices.

Definition 3.1.3. Let $\kappa > 0$. We say that H is a *κ-approximation* of G if, for all $x \in \mathbb{R}^n$,

$$x^T L_G x \leq x^T L_H x \leq \kappa \cdot x^T L_G x$$

where L_G and L_H are the Laplacians of G and H respectively.

Definition 3.1.4. An *ε-spectral-sparsifier* of a graph $G = (V, E, w)$ is a subgraph $H = (V, E_\varepsilon, w_\varepsilon)$ with $|E_\varepsilon| = \mathcal{O}\big(\frac{n \log n}{\varepsilon^2}\big)$ that is an $(1 + \varepsilon)$-approximation of G.

The following result is due to Batson, Spielman and Srivastava:

Theorem 3.1.1 ([BSS12], Theorem 1.1)**.** For every $d > 1$, every weighted undirected graph $G = (V, E, w)$ on n vertices has a reweighted subgraph $H = (V, F, \tilde{w})$ with $|F| = \lceil d(n-1) \rceil$ that satisfies

$$x^T L_G x \leq x^T L_H x \leq \left(\frac{d + 1 + 2\sqrt{d}}{d + 1 - 2\sqrt{d}} \right) \cdot x^T L_G x \qquad \forall x \in \mathbb{R}^n.$$

For completeness, we now prove two well known technical results.

Proposition 3.1.1. For all weighted undirected graphs G and for all $x \in \mathbb{R}^n$ it holds that

$$x^T L_G x = \sum_{\{v_i,v_j\}\in E} w_{ij}(x_i - x_j)^2.$$

Proof. Denote $L_G = (\ell_{ij})_{i,j=1}^n$ and recall that $\ell_{ij} = d_{ij} - a_{ij}$, implying that

$$\ell_{ij} = \begin{cases} d_{ii} & \text{if } i = j \\ -a_{ij} & \text{if } i \neq j. \end{cases}$$

Then

$$\begin{aligned}
x^T L_G x &= \sum_{i=1}^{n}\sum_{j=1}^{n} x_i x_j \ell_{ij} = \sum_{\substack{i=1\\j=i}}^{n} x_i x_j \ell_{ij} + \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j \ell_{ij} = \sum_{i=1}^{n} x_i^2 d_{ii} - \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j a_{ij}\\
&= \sum_{i=1}^{n} x_i^2 \Big(\sum_{\{v_i,v_j\}\in E} w_{ij}\Big) - \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j w_{ij} = \sum_{i=1}^{n} \sum_{\substack{j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 - x_i x_j\big)\\
&= \frac{1}{2}\Big(\sum_{i=1}^{n} \sum_{\substack{j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 - x_i x_j\big) + \sum_{j=1}^{n} \sum_{\substack{i=1\\\{v_j,v_i\}\in E}}^{n} w_{ji}\big(x_j^2 - x_i x_j\big)\Big)\\
&= \frac{1}{2}\Big(\sum_{\substack{i,j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 - 2x_i x_j + x_j^2\big)\Big) = \frac{1}{2}\Big(\sum_{\substack{i,j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i - x_j\big)^2\Big)\\
&= \frac{1}{2}\Big(2\sum_{\{v_i,v_j\}\in E}^{n} w_{ij}\big(x_i - x_j\big)^2\Big) = \sum_{\{v_i,v_j\}\in E}^{n} w_{ij}\big(x_i - x_j\big)^2.
\end{aligned}$$

□

Proposition 3.1.2. For all weighted undirected graphs G and for all $x \in \mathbb{R}^n$ it holds that

$$x^T U_G x = \sum_{\{v_i,v_j\}\in E} w_{ij}(x_i + x_j)^2.$$

Proof. Denote $U_G = (u_{ij})_{i,j=1}^n$ and recall that $u_{ij} = d_{ij} + a_{ij}$, implying that

$$u_{ij} = \begin{cases} d_{ii} & \text{if } i = j\\ a_{ij} & \text{if } i \neq j. \end{cases}$$

Then

$$\begin{aligned}
x^T U_G x &= \sum_{i=1}^{n}\sum_{j=1}^{n} x_i x_j u_{ij} = \sum_{\substack{i=1\\j=i}}^{n} x_i x_j u_{ij} + \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j u_{ij} = \sum_{i=1}^{n} x_i^2 d_{ii} + \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j a_{ij}\\
&= \sum_{i=1}^{n} x_i^2 \Big(\sum_{\{v_i,v_j\}\in E} w_{ij}\Big) + \sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} x_i x_j w_{ij} = \sum_{i=1}^{n} \sum_{\substack{j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 + x_i x_j\big)\\
&= \frac{1}{2}\Big(\sum_{i=1}^{n} \sum_{\substack{j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 + x_i x_j\big) + \sum_{j=1}^{n} \sum_{\substack{i=1\\\{v_j,v_i\}\in E}}^{n} w_{ji}\big(x_j^2 + x_i x_j\big)\Big)\\
&= \frac{1}{2}\Big(\sum_{\substack{i,j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i^2 + 2x_i x_j + x_j^2\big)\Big) = \frac{1}{2}\Big(\sum_{\substack{i,j=1\\\{v_i,v_j\}\in E}}^{n} w_{ij}\big(x_i + x_j\big)^2\Big)\\
&= \frac{1}{2}\Big(2\sum_{\{v_i,v_j\}\in E}^{n} w_{ij}\big(x_i + x_j\big)^2\Big) = \sum_{\{v_i,v_j\}\in E}^{n} w_{ij}\big(x_i + x_j\big)^2.
\end{aligned}$$

□

Now consider the continuous predicates

$$\mathsf{P}^r : [0,r]^2 \to [0,r^2], \quad \mathsf{P}^r(x,y) = |x-y|^2$$

and

$$\mathsf{Q}^r : [0,r]^2 \to [0,4r^2], \quad \mathsf{Q}^r(x,y) = |x+y|^2.$$

Note that, due to the codomain being continuous, we can no longer view the predicate as a relation, nor do we have a well defined notion of a constraint being "satisfied".

Proposition 3.1.3. P^r is sparsifiable for all $r \in \mathbb{R}^+$.

Proof. Recall that, by Proposition 3.1.1, for all graphs G and for all $x \in \mathbb{R}^n$ we have

$$x^T L_G x = \sum_{\{v_i,v_j\}\in E} w_{ij}(x_i - x_j)^2.$$

Let $A : V \to [0,r]$ be an assignment on the vertices of G and $\phi^A \in [0,r]^n$ be the vector defined by $\phi_i^A = A(v_i)$. Then for all weighted undirected graphs G, we have

$$\begin{aligned}(\phi^A)^T L_G \phi^A &= \sum_{\{v_i,v_j\}\in E} w_{ij}(\phi_i^A - \phi_j^A)^2 = \sum_{\{v_i,v_j\}\in E} w_{ij}(A(v_i) - A(v_j))^2 \\ &= \sum_{\{v_i,v_j\}\in E} w_{ij}\mathsf{P}^r(A(v_i), A(v_j)) = \sum_{e\in E} w(e)\mathsf{P}^r(A(e)) = \mathsf{P}^r_G(A).\end{aligned}$$

So, by choosing

$$d = \frac{\varepsilon^2 + 8\varepsilon + 8 + 4(2+\varepsilon)\sqrt{\varepsilon+1}}{\varepsilon^2},$$

we get $d > 1$ and $\frac{d+1+2\sqrt{d}}{d+1-2\sqrt{d}} = 1+\varepsilon$. We can then apply Theorem 3.1.1 to obtain a subgraph H of G with $\lceil d(n-1)\rceil = \mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ edges such that,for all assignments $A : V \to [0,r]$,

$$\mathsf{P}^r_G(A) = (\phi^A)^T L_G \phi^A \le (\phi^A)^T L_H \phi^A \le (1+\varepsilon)(\phi^A)^T L_G \phi^A = (1+\varepsilon)\mathsf{P}^r_G(A)$$

and hence

$$\mathsf{P}^r_H(A) = (\phi^A)^T L_H \phi^A \in \big(\mathsf{P}^r_G(A), (1+\varepsilon)\mathsf{P}^r_G(A)\big) \subset (1\pm\varepsilon)\mathsf{P}^r_G(A)$$

implying that H is an ε-P^r-sparsifier of G. □

Definition 3.1.5. A matrix $M = (m_{ij})_{i,j=1}^n \in \mathbb{R}^n$ is said to be *Balanced Symmetric Diagonally Dominant* (BSDD) if $M = M^T$ and

$$\forall i \in n \quad m_{ii} = \sum_{j\neq i} |m_{ij}|.$$

Definition 3.1.6. A matrix $M' = (m'_{ij})_{i,j=1}^n$ is said to be *governed* by a matrix $M = (m_{ij})_{i,j=1}^n$ if

$$m'_{ij} \neq 0 \implies m_{ij} \neq 0 \quad \text{and} \quad m'_{ij} \cdot m_{ij} \geq 0.$$

The following theorem, presented in [FK17], is originally due to Andoni et al. [ACK+16].

Theorem 3.1.2 ([FK17],Theorem 6.4)**.** For every BSDD matrix M and every $\varepsilon \in (0,1)$ there exists a BSDD matrix M' which is governed by M, has $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ non-zero entries, and satisfies

$$x^T M' x \in (1 \pm \varepsilon) x^T M x \quad \text{for all } x \in \mathbb{R}^n.$$

The following proposition is an adaptation to continuous predicates of the UnCut spectral sparsifier construction described in [FK17, Section 6.4].

Proposition 3.1.4. Q^r is sparsifiable for all $r \in \mathbb{R}^+$.

Proof. Recall that, by Proposition 3.1.2, for all weighted undirected graphs G and for all $x \in \mathbb{R}^n$ we have

$$x^T U_G x = \sum_{\{v_i,v_j\} \in E} w_{ij}(x_i + x_j)^2.$$

Let $A : V \to [0,r]$ be an assignment on the vertices of G and $\phi^A \in [0,r]^n$ be the vector defined by $\phi_i^A = A(v_i)$. Then for all weighted undirected graphs G we have

$$\begin{aligned}(\phi^A)^T U_G \phi^A &= \sum_{\{v_i,v_j\} \in E} w_{ij}(\phi_i^A + \phi_j^A)^2 = \sum_{\{v_i,v_j\} \in E} w_{ij}(A(v_i) + A(v_j))^2 \\ &= \sum_{\{v_i,v_j\} \in E} w_{ij}\mathsf{Q}^r(A(v_i), A(v_j)) = \sum_{e \in E} w(e)\mathsf{Q}^r(A(e)) = \mathsf{Q}^r_G(A).\end{aligned}$$

Noticing that U_G is BSDD and that if H is a subgraph of G than U_H is governed by U_G, we can apply Theorem 3.1.2 to show that there exists a BSDD matrix U'_G with $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$ non-zero entries which is governed by U_G such that

$$x^T U'_G x \in (1 \pm \varepsilon) x^T U_G x \quad \text{for all } x \in \mathbb{R}^n.$$

Notice that U'_G defines a subgraph $G_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of G such that

$$E_\varepsilon := \{\{v_i, v_j\} : (U'_G)_{ij} > 0\}$$

and

$$w_\varepsilon(\{v_i, v_j\}) := (U'_G)_{ij}.$$

Then, $U'_G = U_{G_\varepsilon}$ and, for any assignment $A : V \to [0,r]$,

$$\mathsf{Q}^r_{G_\varepsilon}(A) = (\phi^A)^T U_{G_\varepsilon} \phi^A \in (1 \pm \varepsilon)(\phi^A)^T U_G \phi^A = (1 \pm \varepsilon)\mathsf{Q}^r_G(A)$$

implying that G_ε is an ε-Q^r-sparsifier of G. ∎

3.1.1 Applications of Spectral Sparsification

In this paragraph we propose a possible application of the continuous VCSP spectral sparsifiers found in Section 3.1.

Suppose we have a Directed Acyclic Graph (DAG) $G = (V, E, w)$ which models a set of tasks. More specifically, each edge represents a task, where the tail of the edge represents the start of the task and the head of the edge represents the end of the task. Let e_{ij} denote edge (v_i, v_j). If e_{ij} and e_{ik} belong to E, this means that task e_{ij} and task e_{ik} must start at the same time, and similarly if e_{ik} and e_{jk} belong to E, this means that task e_{ik} and task e_{jk} must end at the same time. If e_{ij} and e_{jk} belong to E, this means that task e_{jk} must start exactly when task e_{ij} is completed. The fact that G is acyclic prevents any time paradox in this representation. The weight of each edge represents the absolute cost of keeping the corresponding task "active" - that is, maintaining a task that has been started but not completed - per unit time. For instance, a task might represent transporting some frozen goods from one place to another. The weight of the task represents the cost of the energy needed to keep the goods frozen for a day.

Consider some domain D which represents a period of time starting at $\min\{d \in D\}$ and ending at $\max\{d \in D\}$. The problem is to find an assignment of the vertices in V to D that indicates when tasks start and end, and which minimises the total cost of the task set. Notice that D can be discrete (and hence an assignment A would indicate, for instance, that tasks e_{ij} and e_{ik} must begin on day $A(v_i) = 1$) or continuous. Since G is acyclic, it has a topological sort and hence such an assignment exists. Notice that the topological sort may not be unique.

Now, the time during which task e_{ij} is "active", and hence the time during which we must calculate its cost, is given by $|A(v_i) - A(v_j)|$. More formally, we want to find an assignment $A : V \to [0, r]$ for some $r \in \mathbb{R}^+$ such that

$$\sum_{e_{ij} \in E} w(e_{ij}) |A(v_i) - A(v_j)|$$

is minimised. Given that $f(x) = x^2$ is monotone increasing on $x \geq 0$, we have that

$$\sum_{e_{ij} \in E} w(e_{ij}) |A(v_i) - A(v_j)| \text{ is minimised} \iff \sum_{e_{ij} \in E} w(e_{ij}) |A(v_i) - A(v_j)|^2 \text{ is minimised.}$$

Notice that

$$\sum_{e_{ij} \in E} w(e_{ij}) |A(v_i) - A(v_j)|^2 = \sum_{e_{ij} \in E} w(e_{ij}) \mathsf{P}^r(A(v_i, v_j)) = \mathsf{P}^r_G(A).$$

Then, if we have a sparse subgraph G_ε such that $\mathsf{P}^r_{G_\varepsilon}(A) \in (1 \pm \varepsilon)\mathsf{P}^r_G(A)$ for some small $\varepsilon \in (0, 1)$, we can find all topological sorts A on G_ε, calculate $\min_A \mathsf{P}^r_{G_\varepsilon}(A)$, and use such A on G, which gives us the (approximately) minimum value of $\mathsf{P}^r_G(A)$. Given that finding a topological sort has a $\mathcal{O}(n + m)$ runtime, using a graph sparsifier gives a new runtime of $\mathcal{O}\big(n + \frac{n}{\varepsilon^2}\big) = \mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$.

Chapter 4

Conclusion

In this dissertation we have tackled the question posed in [FK17] of identifying which classes of VCSP instances admit sparsification. We have successfully solved the problem for binary discrete predicates by proving a necessary and sufficient condition for a binary predicate to be sparsifiable. We have further identified two continuous binary predicates which admit a spectral sparsifier. A number of open questions naturally arise from our work.

First, more work needs to be done towards a full classification of non-binary VCSPs according to sparsifiability. We have already made some steps in this direction: Proposition 2.1.1 together with Theorem 1.2.2 show that the class of k-ary predicates that can be rewritten in terms of the k-ary cut predicate NAE is sparsifiable. On the other hand, Corollary 2.2.1 defines a whole class of predicates that cannot be sparsifiable, namely those that contain a singleton ℓ-cube for some constant ℓ. However, there are predicates which do not fall in either of these categories, so new techniques are needed to tackle this problem. We hope that the results we present in Section 2.5 will prove useful in this direction.
A potential approach would be to develop new combinatorial sparsification techniques, not reliant on reduction to cut sparsifier, in order to encapsulate those predicates that may be sparsifiable but cannot be reduced to a hypergraph cut. The k-ary Parity predicate is an example of such a predicate.[1]
Another promising approach is to find methods to reduce a k-ary CSP to a binary one, where the classification problem is fully solved. Efforts in this direction - although not with the sparsification problem in mind - where made by Cohen et al. [CCJ$^+$17], who showed that every VCSP instance can be converted to an equivalent VCSP instance which only uses unary and binary constraints via a transformation which relies on the dual encoding of the underlying hypergraph. However, all non-trivial unary constraints are non-sparsifiable (as we could see them as containing a singleton point), making this transformation difficult to use for positive results. Moreover, the number of vertices in the dual encoding of a hypergraph is actually $|V| + |E|$, meaning that the graph sparsifier of such graph would

[1] A detailed proof that Parity is one such predicate can be found in Appendix C.

need to be sublinear in its number of vertices in order to produce a valid sparsifier of the original hypergraph. We know that the general lower bound for graph sparsification is $\mathcal{O}\left(\frac{n}{\varepsilon^2}\right)$, however it is possible that certain classes of graphs admit a sparsifier of sublinear size.
A final approach is that of finding a suitable generalisation of the graph Laplacian to hypergraphs and work with this to extract more spectral sparsification results. Although the spectral theory of hypergraphs is still not well understood, recently Chan et al. [CLTZ18] have proposed a hypergraph Laplacian operator which generalises the graph Laplacian, and which could be taken as a starting point for future work in this direction.

Another interesting question altogether is that of understanding the nature of the "gap" between sparsifiable and non-sparsifiable predicates. While we know that for the binary case the classes of not sparsifiable and non-sparsifiable predicates coincide, and hence resistance to sparsification is a special case of non-sparsifiability, the distribution of the k-ary predicates that are not sparsifiable is not well understood. At the moment, we are not aware of any predicate which is provably neither sparsifiable nor non-sparsifiable (that is, whose ε-sparsifiers have all size asymptotically strictly larger than $\tilde{\mathcal{O}}\left(\frac{|V|}{\varepsilon^2}\right)$ but are also possibly of size $o(|V|^k)$. However, we have no proof that such a predicate does not exist. While Proposition 2.2.1 gives lower bounds for the size of the sparsifier of the class of predicates which contain singleton ℓ-cubes, we do not know whether those bounds are tight.

Finally, we have made the first steps towards the sparsification of continuous VCSPs by means of a spectral sparsifier construction. Whether there are other continuous predicates which admit a sparsifier, spectral or not, is still unknown. To tackle this problem, one could begin by analysing which properties of the predicate - such as continuity and submodularity - affect the sparsifiability of the corresponding VCSP class.

To conclude, given that making use of cut sparsifiers has proved to be advantageous in a variety of fields in Computer Science, we believe that extending the sparsification framework to general valued CSPs will not only prove to be an interesting theoretical question, but also represent a crucial advancement in the efficient approximate solution of a number of practical problems.

Appendix A

A.1 A Generalised Definition of Valued CSP

The following generalised definition of a VCSP is given in [RN03].
An instance of a *Valued Constraint Satisfaction Problem* (VCSP) is a quadruple

$$\mathcal{I} = (V, D, \Pi, w)$$

where

- $V = \{v_1, \ldots, v_n\}$ is a set of variables,
- $D = \{D_1, \ldots, D_n\}$ is a set of domains, one for each variable,
- $\Pi = \{\pi_1, \ldots, \pi_m\}$ is a set of constraints,
- $w : \Pi \to \mathbb{R}^+$ is a strictly positive weight function on the set of constraints.

Each domain D_i consists of the set of values that variable v_i is allowed to take. Each constraint $\pi_i \in \Pi$ consists of a pair $\langle scope, rel \rangle$ where $scope = \mathbf{v} \in V^k$ is an ordered k-tuple of distinct variables and rel is a k-ary relation on the cross product of the corresponding domains. We take k to be constant. A relation rel with scope $(u_1, \ldots, u_k) \in V^k$ can be seen as a k-ary function

$$\mathsf{P} : D(u_1) \times \ldots \times D(u_k) \to \{0, 1\},$$

$$\mathsf{P}\big((x_1, \ldots, x_k)\big) = 1 \iff (x_1, \ldots, x_k) \in rel \subset D(u_1) \times \ldots \times D(u_k)$$

where $D(u_j)$ is the domain of variable u_j. An assignment is a function which maps each variable to a particular value in the corresponding domain: that is, a map

$$A : V \to \bigcup_{i=1}^{n} D_i \quad \text{such that} \quad A(v_i) \in D_i \quad \text{for } i = 1, \ldots, n. \tag{A.1}$$

Given a tuple $\mathbf{v} = (u_1, \ldots, u_k) \in V^k$ and a predicate $\mathsf{P} : D(u_1) \times \ldots \times D(u_k) \to \{0, 1\}$, we say that $\pi = \langle \mathbf{v}, \mathsf{P} \rangle$ is satisfied by an assignment $A : V \to \cup_{i=1}^{n} D_i$ if $\mathsf{P}(A(u_1), \ldots, A(u_k)) = 1$. The value of an instance $\mathcal{I}$ under assignment A is defined exactly as in Section 1.1. Similarly, an ε-sparsifier is defined as in Section 1.1, with the exception that assignments must satisfy property (A.1).

Remark A.1.1. If $\pi = \langle (u_1, \ldots, u_k), \mathsf{P} \rangle$ is a contraint in π and $D(\mathsf{P})$ is the domain of P, then we must have $D(u_1) \times \ldots \times D(u_k) \subset D(\mathsf{P})$.

Definition A.1.1. We say that a predicate $\mathsf{P} : D(\mathsf{P}) \to \{0, 1\}$ is *sparsifiable* if for all $\varepsilon \in (0, 1)$ and for all instances $\mathcal{I} = (V, D, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ there exists an ε-sparsifier $\mathcal{I}_\varepsilon$ of $\mathcal{I}$.

Definition A.1.2. We say that a k-ary predicate $\mathsf{P} : D(\mathsf{P}) \to \{0, 1\}$ *is non-sparsifiable* if there exists an instance $\mathcal{I} = (V, D, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ and an $\varepsilon \in (0, 1)$ such that all sub-instances $\mathcal{I}_\varepsilon = (V, D, \Pi_\varepsilon \subset \Pi, w)$ of $\mathcal{I}$ with property (1.1) have $|\Pi_\varepsilon| = \Omega(|V|^k)$.

Definition A.1.3. We say that a predicate $\mathsf{P} : D(\mathsf{P}) \to \{0, 1\}$ is *strongly non-sparsifiable* if for all instances $\mathcal{I} = (V, D, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ there exists an $\varepsilon \in (0, 1)$, such that all sub-instances $\mathcal{I}_\varepsilon = (V, D, \Pi_\varepsilon \subset \Pi, w)$ with preperty (1.1) have $|\Pi_\varepsilon| = \Omega(|\Pi|)$.

Definition A.1.4. We say that a predicate $\mathsf{P} : D(\mathsf{P}) \to \{0, 1\}$ is *resistant to sparsification* if for all instances $\mathcal{I} = (V, D, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ that are not sparse there exists an $\varepsilon \in (0, 1)$ such that $\mathcal{I}$ does not have an ε-sparsifier.

A.2 Extending Results

In [FK17], Filtser and Krauthgamer show that, if $B = C = \{0, 1\}$, a binary predicate $\mathsf{P} : B \times C \to \{0, 1\}$ is sparsifiable if and only if $|\mathsf{P}^{-1}(1)| \neq 1$. We want to generalise this result to arbitrary binary predicates $\mathsf{P} : B \times C \to \{0, 1\}$ where $|B| = |C| = 2$.

Proposition A.2.1. Let $\mathsf{P} : B \times C \to \{0, 1\}$ be a predicate with $|B| = |C| = 2$. If $|\mathsf{P}^{-1}(1)| = 1$, then P is non-sparsifiable.

Proof. Suppose $B = \{b_1, b_2\}$, $C = \{c_1, c_2\}$ and assume without loss of generality that $\mathsf{P}^{-1}(1) = \{(b_1, c_1)\}$. Consider the VCSP instance $\mathcal{I} = (V, D, \Pi, w)$ where

- $V = R \sqcup S$, $R = \{r_1, \ldots, r_{\frac{n}{2}}\}$, and $S = \{s_1, \ldots, s_{\frac{n}{2}}\}$ for some even $n \in \mathbb{N}$;
- $D = \{B, C\}$, $D(r) = B$ for all $r \in R$, $D(s) = C$ for all $s \in S$;
- $\Pi = \{\pi_{ij} := \langle (r_i, s_j), \mathsf{P} \rangle : i, j \in \{1, \ldots, \frac{n}{2}\}\}$.

Notice that $|\Pi| = \frac{n^2}{4}$. Also notice that the set $B \cap C$ may or may not be empty, however we do assume that $b_1 \neq b_2$ and $c_1 \neq c_2$.
Now consider the family of assignments $A_{ij} : V \to B \cup C$ for $i, j = 1, \ldots, \frac{n}{2}$ such that $A_{ij}(r_i) = b_1$, $A_{ij}(s_j) = c_1$, and $A_{ij}(r) = b_2$ for all $r \in R \setminus \{r_i\}$, $A_{ij}(s) = c_2$ for all $s \in S \setminus \{s_j\}$. Then, we have

$$\mathsf{P}(A_{ij}(u, v)) = \begin{cases} \mathsf{P}(b_1, c_1) = 1 & \text{if } u = r_i, v = s_j \\ \mathsf{P}(b_1, c_2) = 0 & \text{if } u = r_i, v \in S \setminus \{s_j\} \\ \mathsf{P}(b_2, c_1) = 0 & \text{if } u \in R \setminus \{r_i\}, v = s_j \\ \mathsf{P}(b_2, c_2) = 0 & \text{if } u \in R \setminus \{r_i\}, v \in S \setminus \{s_j\} \end{cases}$$

Therefore,

$$Val_{\mathcal{I}}(A_{ij}) = \sum_{\pi \in \Pi} w(\pi)\mathsf{P}(A_{ij}(\pi)) = w(\pi_{ij}) > 0$$

and hence if $\mathcal{I}_\varepsilon = (V, D, \Pi_\varepsilon, w_\varepsilon)$ is an ε-sparsifier of $\mathcal{I}$, we must have that $\pi_{ij} \in \Pi_\varepsilon$ for all $i, j = 1, \ldots, n$, otherwise we would have

$$Val_{\mathcal{I}_\varepsilon}(A_{ij}) = \sum_{\pi \in \Pi_\varepsilon} w_\varepsilon(\pi)\mathsf{P}(A_{ij}(\pi)) = 0 \notin (1 \pm \varepsilon)Val_{\mathcal{I}}(A_{ij}).$$

Therefore we must have $\Pi_\varepsilon = \Pi$ and hence $|\Pi_\varepsilon| = |\Pi| = \frac{n^2}{4} = \Omega(n^2)$, implying that P is non-sparsifiable. $\square$

Proposition A.2.2. Let $\mathsf{P} : B \times C \to \{0, 1\}$ be a binary predicate with $|B| = |C| = 2$. If $|\mathsf{P}^{-1}(1)| \neq 1$, then P is sparsifiable.

Proof. (Adapted from Theorem 1.2.4.) Consider an instance $\mathcal{I} = (V, D, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ and notice that $D = \{D_1, \ldots, D_n\} = \{B, C\}$. Define a new binary predicate $\mathsf{Q} : (B \cup C)^2 \to \{0, 1\}$ with $\mathsf{Q}^{-1}(1) = \mathsf{P}^{-1}(1)$ and let $r = |B \cup C|$. Assume that there exists a functions $f_{\mathsf{Q}} : Part_r(V) \to Part_\ell(V^\gamma)$ for some $\ell \in \mathbb{N}$ such that for some weighted directed graph G on V and for any $\mathcal{S} \in Part_r(V)$ it holds that

$$\mathsf{Q}_G(\mathcal{S}) = \ell\text{-}\mathsf{Cut}_{\gamma(G)}(f_{\mathsf{Q}}(\mathcal{S})), \tag{A.2}$$

where $\gamma(G)$ is the bipartite double cover of G. Then, we can apply the argument used in the proof of Proposition 2.1.1 to show that G has an ε-Q-sparsifier[1], and hence so does $\mathcal{I}^{G,\mathsf{Q}}$.

Now consider an instance $\mathcal{I} = (V, \{B, C\}, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$. We can construct a corresponding instance $\mathcal{I}^{\mathsf{Q}} = (V, B \cup C, \Pi^{\mathsf{Q}}, w^{\mathsf{Q}}) \in \mathsf{VCSP}(\mathsf{Q})$ by setting

$$\Pi^{\mathsf{Q}} := \{\langle (u, v), \mathsf{Q}\rangle : \langle (u, v), \mathsf{P}\rangle \in \Pi\}$$

and

$$w^{\mathsf{Q}}(\langle\langle (u, v), \mathsf{Q}\rangle) := w(\langle\langle (u, v), \mathsf{P}\rangle) \text{ for all } \pi^{\mathsf{Q}} = (\langle\langle (u, v), \mathsf{Q}\rangle) \in \Pi^{\mathsf{Q}}.$$

Assuming that equation (A.2) holds for $G = G^{\mathcal{I}^{\mathsf{Q}}}$, let $\mathcal{I}_\varepsilon^{\mathsf{Q}}$ be an ε-Q-sparsifier of $\mathcal{I}^{\mathsf{Q}} = \mathcal{I}^{G,\mathsf{Q}}$. From $\mathcal{I}_\varepsilon^{\mathsf{Q}}$ we construct a corresponding instance $\mathcal{I}_\varepsilon \in \mathrm{VCSP}(\mathsf{P})$ by setting

$$\Pi_\varepsilon := \{\langle (u, v), \mathsf{P}\rangle : \langle (u, v), \mathsf{Q}\rangle \in \Pi_\varepsilon^{\mathsf{Q}}\}$$

and

$$w_\varepsilon(\langle\langle (u, v), \mathsf{P}\rangle) := w_\varepsilon^{\mathsf{Q}}(\langle\langle (u, v), \mathsf{Q}\rangle) \text{ for all } \pi = (\langle\langle (u, v), \mathsf{P}\rangle) \in \Pi_\varepsilon.$$

[1]Notice that we are *not* claiming that Q is sparsifiable, given that equation (A.2) does not need to hold for all weighted directed graphs.

uv	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8	Q_9	Q_{10}	Q_{11}
b_1b_1	0	0	0	0	0	0	0	0	0	0	0	0
b_1b_2	0	0	0	0	0	0	0	0	0	0	0	0
b_2b_1	0	0	0	0	0	0	0	0	0	0	0	0
b_2b_2	0	0	0	0	0	0	0	0	0	0	0	0
$\mathbf{b_1c_1}$	**0**	**0**	**0**	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**
$\mathbf{b_1c_2}$	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**1**	**0**	**1**	**1**	**1**
$\mathbf{b_2c_1}$	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**1**
$\mathbf{b_2c_2}$	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**	**1**	**1**	**0**	**1**
c_1b_1	0	0	0	0	0	0	0	0	0	0	0	0
c_1b_2	0	0	0	0	0	0	0	0	0	0	0	0
c_2b_1	0	0	0	0	0	0	0	0	0	0	0	0
c_2b_2	0	0	0	0	0	0	0	0	0	0	0	0
c_1c_1	0	0	0	0	0	0	0	0	0	0	0	0
c_1c_2	0	0	0	0	0	0	0	0	0	0	0	0
c_2c_1	0	0	0	0	0	0	0	0	0	0	0	0
c_2c_2	0	0	0	0	0	0	0	0	0	0	0	0

Table A.1: Explicit representation of all possible predicates $\mathsf{Q} : (B \cup C)^2 \to \{0, 1\}$ which "extend" a corresponding predicate $\mathsf{P} : B \times C \to \{0, 1\}$ with $|\mathsf{P}^{-1}(1)| \neq 1$.

Notice that $|\Pi_\varepsilon| = |\Pi_\varepsilon^{\mathsf{Q}}|$ and $\Pi_\varepsilon \subset \Pi$. Then, for any assignment $A : V \to B \cup C$ which satisfies property (A.1), we have

$$Val_{\mathcal{I}^{\mathsf{Q}}}(A) = \sum_{\pi^{\mathsf{Q}} \in \Pi^{\mathsf{Q}}} w^{\mathsf{Q}}(\pi^{\mathsf{Q}})\mathsf{Q}(A(\pi^{\mathsf{Q}})) = \sum_{\pi \in \Pi} w(\pi)\mathsf{P}(A(\pi)) = Val_I(A)$$

and similarly

$$Val_{\mathcal{I}_\varepsilon}(A) = \sum_{\pi \in \Pi_\varepsilon} w_\varepsilon(\pi)\mathsf{P}(A(\pi)) = \sum_{\pi^{\mathsf{Q}} \in \Pi_\varepsilon^{\mathsf{Q}}} w_\varepsilon^{\mathsf{Q}}(\pi^{\mathsf{Q}})\mathsf{Q}(A(\pi^{\mathsf{Q}})) = Val_{I_\varepsilon^{\mathsf{Q}}}(A),$$

so that

$$Val_{\mathcal{I}_\varepsilon}(A) = Val_{I_\varepsilon^{\mathsf{Q}}}(A) \in (1 \pm \varepsilon)Val_{\mathcal{I}^{\mathsf{Q}}}(A) = (1 \pm \varepsilon)Val_{\mathcal{I}}(A)$$

implying that $\mathcal{I}_\varepsilon^{\mathsf{Q}}$ is an ε-sparsifier of $\mathcal{I}^{\mathsf{Q}}$ with $|\Pi_\varepsilon| = |\Pi_\varepsilon^{\mathsf{Q}}| = \mathcal{O}\big(\frac{|V|}{\varepsilon^2}\big)$ constraints.

It remains to prove that such a function f_{Q} exists for all predicates $\mathsf{Q} : (B \cup C)^2 \to \{0, 1\}$ and for all VCSP instances $\mathcal{I}^{\mathsf{Q}}$ constructed as above.
We say a constraint is of type ij under assignment A if its scope is (u, v) with $A(u) = i$ and $A(v) = j$. Notice that by our construction of predicate Q, under any assignment A there can only be four types of constraints in $\mathcal{I}^{\mathsf{Q}}$: they are all of type b_1c_1, b_1c_2, b_2c_1 or b_2c_2.
In Table A.1 we list all twelve possible predicates $\mathsf{Q} : (B \cup C)^2 \to \{0, 1\}$ with $|\mathsf{Q}^{-1}(1)| \neq 1$, each corresponding to a predicate $\mathsf{P} : B \times C \to \{0, 1\}$ with $|\mathsf{P}^{-1}(1)| \neq 1$, and below we list the respective functions f_{Q}.

- $f_{\mathsf{Q}_0} = \{\emptyset, V\}$;
- $f_{\mathsf{Q}_1} = \{S^{(0)}_{b_2}, \overline{S^{(0)}_{b_2}}\}$;
- $f_{\mathsf{Q}_2} = \{S^{(1)}_{c_2}, \overline{S^{(1)}_{c_2}}\}$;
- $f_{\mathsf{Q}_3} = \{S^{(0)}_{b_1} \cup S^{(1)}_{c_1}, \overline{S^{(0)}_{b_1} \cup S^{(1)}_{c_1}}\}$;
- $f_{\mathsf{Q}_4} = \{S^{(0)}_{b_1} \cup S^{(1)}_{c_2}, \overline{S^{(0)}_{b_1} \cup S^{(1)}_{c_2}}\}$;
- $f_{\mathsf{Q}_5} = \{S^{(1)}_{c_1}, \overline{S^{(1)}_{c_1}}\}$;
- $f_{\mathsf{Q}_6} = \{S^{(0)}_{b_1}, \overline{S^{(0)}_{b_1}}\}$;
- $f_{\mathsf{Q}_7} = \{S^{(0)}_{b_2}, S^{(1)}_{c_2}, \overline{S^{(0)}_{b_2} \cup S^{(1)}_{c_2}}\}$;
- $f_{\mathsf{Q}_8} = \{S^{(0)}_{b_2}, S^{(1)}_{c_1}, \overline{S^{(0)}_{b_2} \cup S^{(1)}_{c_1}}\}$;
- $f_{\mathsf{Q}_9} = \{S^{(1)}_{b_1}, S^{(1)}_{c_2}, \overline{S^{(0)}_{b_1} \cup S^{(1)}_{c_2}}\}$;
- $f_{\mathsf{Q}_{10}} = \{S^{(0)}_{b_1}, S^{(1)}_{c_1}, \overline{S^{(0)}_{b_1} \cup S^{(1)}_{c_1}}\}$;
- $f_{\mathsf{Q}_{11}} = \{S^{(0)}_{b_1} \cup S^{(0)}_{b_2}, \overline{S^{(0)}_{b_1} \cup S^{(0)}_{b_2}}\}$.

$\square$

Appendix B

B.1 Sparsification of Generalised Binary Predicates

We aim to show that Theorem 2.4.1 also applies to general binary predicates $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$, where D_1 and D_2 are not necessarily of the same size. We begin by noticing that the proof of Proposition 2.2.1 would follow the exact same argument for a predicate $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$, and hence we can state an equivalent of Corollary 2.2.2 for generalised binary predicates $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$:

Corollary B.1.1. Let $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ be a binary predicate which contains a singleton square. Then, P is non-sparsifiable.

We proceed by proving a generalised version of Lemma 2.4.1 which holds for bipartite graphs with partition classes of any size. The main difference with the proof of Lemma 2.4.1 is that the induction is now on $|V|$. This will make the induction step somewhat simpler.

Lemma B.1.1. Let $G = (V,E)$ be a bipartite graph on $|V| = r_1 + r_2$ vertices, where $V = V_1 \sqcup V_2$ and $|V_j| = r_j \geq 2$ for $j = 1,2$. If for any $u_1, v_1 \in V_1$ and for any $u_2, v_2 \in V_2$ we have $|E(\{u_1, v_1, u_2, v_2\})| \neq 1$, then for any $v \in V$, $d_{\overline{G}}(v) > 0$ implies that $\overline{G}_v$ is a complete bipartite graph with partition classes $\{V_1 \cap V(\overline{G}_v)\}$, $\{V_2 \cap V(\overline{G}_v)\}$.

Proof. By induction on $|V|$.

|V|=4. This step is identical to the first part of the proof of Lemma 2.4.1. The reader can refer to Table 2.1.

Induction step. Suppose that the lemma holds for $4 \leq |V| \leq r_1 + r_2 - 1$. We need to prove that it holds for $|V| = r_1 + r_2$. Let $G = (V,E)$ be a bipartite graph with $V = V_1 \sqcup V_2$ and let $V_j = \{v_0^{(j)}, \ldots, v_{r_j-1}^{(j)}\}$ for $j = 1,2$. Since $r_1 + r_2 > 4$ and $r_1, r_2 \geq 2$ there exists $j \in \{1,2\}$ such that $r_j > 2$. Without loss of generality, assume $r_2 > 2$. Let $U_1 = V_1$ and $U_2 = V_2 \setminus \{v_{r_2-1}^{(2)}\}$,[1] and consider the vertex-induced subgraph $G(U) = (U, E(U))$ where $U = U_1 \sqcup U_2$. Notice that $|U| = r_1 + r_2 - 1$ and that $|U_j| \geq 2$ for $j = 1,2$. Suppose that

[1] By symmetry, the same argument holds if we choose $U_1 = V_1 \setminus \{v_{r_1-1}^{(1)}\}$ and $U_2 = V_2$ provided that $r_1 > 2$.

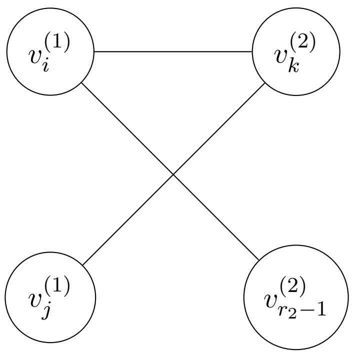
G satisfies the conditions of the Lemma. Then $G(U)$ satisfies such conditions too, and so by the induction hypothesis we have that for any $v \in U$, $d_{\overline{G(U)}}(v) > 0$ implies that $\overline{G(U)}_v$ is a complete bipartite graph with partition classes $\{U_1 \cap V(\overline{G}_v)\}$, $\{U_2 \cap V(\overline{G(U)}_v)\}$. Now consider $\overline{G(U)}$.

Denote each $\{v_i^{(1)}, v_j^{(2)}\}$ by e_{ij}. If for all $i = 0, \dots, r_1 - 1$ we have $e_{i,r_2-1} \notin \overline{E}$, then there is nothing to prove and the lemma holds.

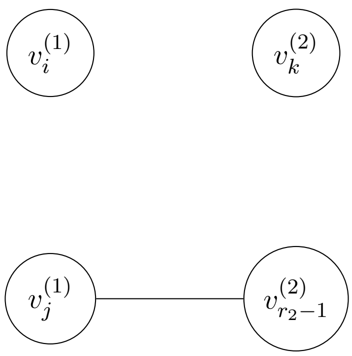
So suppose that $e_{i,r_2-1} \in \overline{E}$ for some $i \in [r_1]$. We need to prove that $e_{j,r_2-1} \in \overline{E}$ for all j such that $v_j^{(1)} \in V(\overline{G}_{v_i^{(1)}})$.

Suppose not. Then there exists some $j \in [r_1]$ such that $v_j^{(1)} \in V(\overline{G}_{v_i^{(1)}})$ and $e_{j,r_2-1} \notin \overline{E}$. But by the induction hypothesis $v_i^{(1)}, v_j^{(1)}$ belong to the complete bipartite graph $\overline{G(U)}_{v_i^{(1)}}$, so for all $v \in U_2 \cap V(\overline{G(U)}_{v_i^{(1)}})$ we have $\{v_i^{(1)}, v\}$,$\{v_j^{(1)}, v\} \in \overline{E}$. Notice that $U_2 \cap V(\overline{G(U)}_{v_i^{(1)}}) \neq \emptyset$ since $v_i^{(1)}, v_j^{(1)}$ are path-connected. So choose any $k \in \{0, \dots, r_2 - 2\}$ such that $v_k^{(2)} \in U_2 \cap V(\overline{G(U)}_{v_i^{(1)}})$.

Then, $e_{ik}, e_{jk} \in \overline{E}$ and we have the following picture for $\overline{G}(\{v_i^{(1)}, v_j^{(1)}, v_k^{(2)}, v_{r_2-1}^{(2)}\})$



which means that in $\overline{\overline{G}} = G$ we have the following picture for $G(\{v_i^{(1)}, v_j^{(1)}, v_k^{(2)}, v_{r_2-1}^{(2)}\})$



contradicting the initial assumption that for any $u_1, v_1 \in V_1$ and for any $u_2, v_2 \in V_2$ we have $|E(\{u_1, v_1, u_2, v_2\})| \neq 1$.

Then, we deduce that $v_{r_2-1}^{(2)}$ is connected to all the vertices in $V_1 \cap \overline{G}_{v_{r_2-1}^{(2)}}$ and hence $\overline{G}_{v_{r_2-1}^{(2)}}$ is complete bipartite with the required vertex set. $\square$

We now prove an equivalent of Proposition 2.4.1 for general binary predicates $\mathsf{P} : D_1 \times D_2 \to \{0, 1\}$. Notice that we will not be able to use the bipartite double cover as it is, so defining a new graph transformation $\Gamma(G)$ will be necessary.

Proposition B.1.1. Let $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ be a binary predicate which contains no singleton squares. Then, P is sparsifiable and there exists an ε-P-sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$.

Proof. Let $I = (V, \Pi, w) \in \mathrm{VCSP}(\mathsf{P})$ and let $G = (V, E, w)$ be the underlying weighted directed graph of $\mathcal{I}$. Assume for simplicity that G is connected.[2] Consider its bipartite double cover $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$. Notice that there may be a number of vertices in V^γ that have degree equal to zero: define $Z := \{v \in V^\gamma : d_{\gamma(G)}(v) = 0\}$. We define the *restricted bipartite double* cover of G to be

$$\Gamma(G) := (V^\Gamma, E^\Gamma, w^\Gamma)$$

where $V^\Gamma = V^\gamma \setminus Z$ and $E^\Gamma = E^\gamma$, $w^\Gamma = w^\gamma$. Notice that, for any binary predicate $\mathsf{Q} : D_1 \times D_2 \to \{0,1\}$ and any valid assignment $A : V^\gamma \to D_1 \cup D_2$ we have

$$\mathsf{Q}_{\Gamma(G)}(A|_{V^\Gamma}) = \mathsf{Q}_{\gamma(G)}(A).$$

We will see later on that working with $\Gamma(G)$ will be easier than working with $\gamma(G)$.
Let $r = |D_1 \cup D_2|$ and $r_j = |D_j|$ for $j = 1, 2$. For a valid assignment $A : V \to D_1 \cup D_2$, recall that $S_d := A^{-1}(d)$ and hence $\mathcal{S} := \{S_0, \ldots, S_{r-1}\}_{d \in D_1 \cup D_2}$ forms an r-partition of V. Suppose that for a predicate $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ there is a function $f_{\mathsf{P}} : Part_r(V) \to Part_\ell(V^\Gamma)$ (for some $\ell \leq r_1 + r_2$) such that for any weighted directed graph G on V it holds that

$$\mathsf{P}_G(\mathcal{S}) = \ell\text{-}\mathsf{Cut}_{\Gamma(G)}(f_{\mathsf{P}}(\mathcal{S})).$$

Recall that the binary ℓ-Cut predicate is sparsifiable (Proposition 1.2.1), so let $\Gamma(G)_\varepsilon$ be an ε-ℓ-Cut-sparsifier of $\Gamma(G)$. We define G_ε analogously to the proof of Proposition 2.1.1 and notice that $\Gamma(G)_\varepsilon$ and $\Gamma(G_\varepsilon)$ have the same edge set. By the same argument used in Proposition 2.1.1, it follows that G_ε is an ε-P-sparsifier of G and that P has sparsifiers of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$. It remains to prove that such a function f_{P} exists for all binary predicates $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ which contain no singleton squares.
Consider a weighted directed graph $G = (V, E, w)$ and its undirected restricted bipartite double cover $\Gamma(G) = (V^\Gamma, E^\Gamma, w^\Gamma)$. Suppose $D_j = \{d_1^j, \ldots, d_{r_j}^j\}$ for $j = 1, 2$. Define an auxiliary graph $G^{\mathsf{P}, r_1, r_2} = G^{\mathsf{P}} := (V^{\mathsf{P}}, E^{\mathsf{P}})$, where

$$V^{\mathsf{P}} := \{v_{d_1^1}, \ldots, v_{d_{r_1}^1}, v'_{d_1^2}, \ldots, v'_{d_{r_2}^2}\},$$

$$E^{\mathsf{P}} := \{\{v_{d_i^1}, v'_{d_j^2}\} : \mathsf{P}(d_i^1, d_j^2) = 1\}.$$

Notice the slight change in the definition of G^{P} with respect to the definition of $G^{\mathsf{P},r}$ of Proposition 2.4.1 to accommodate for the different domains D_1 and D_2.
Let ℓ be the number of connected components of $\overline{G^{\mathsf{P}}}$. Clearly $\ell \leq |V^P| = r_1 + r_2$. Recall

[2]If not, we can divide $\mathcal{I}$ into its connected subinstances.

that a proper efficient ℓ-colouring of G^{P} is a map $c : V^{\mathsf{P}} \to \{0, \ldots, \ell-1\}$ such that

$$\forall d_i^1 \in D_1, \quad \forall d_j^2 \in D_2 \quad \begin{cases} \{v_{d_i^1}, v'_{d_j^2}\} \in E^{\mathsf{P},r} \implies c(v_{d_i^1}) \neq c(v'_{d_j^2}) \\ \{v_{d_i^1}, v'_{d_j^2}\} \notin E^{\mathsf{P},r} \implies c(v_{d_i^1}) = c(v'_{d_j^2}). \end{cases}$$

Now, a function $f_{\mathsf{P}} : Part_r(V) \to Part_\ell(V^\Gamma)$ which satisfies $\mathsf{P}_G(\mathcal{S}) = \ell\text{-}\mathsf{Cut}_{\Gamma(G)}(f_{\mathsf{P}}(\mathcal{S}))$ for all $\mathcal{S} \in Part_r(V)$ corresponds to a proper efficient ℓ-colouring $c : V^{\mathsf{P}} \to \{0, \ldots, \ell-1\}$ of the vertices of G^{P}. Indeed, the colouring c induces an assignment $A^\Gamma : V^\Gamma \to \{0, \ldots, \ell-1\}$ of the vertices of $\Gamma(G)$ which satisfies

$$A^\Gamma(u) = c(v_{A(u)}), \qquad A^\Gamma(u') = c(v'_{A(u)})$$

and which in turn induces a partition $\{U_i\}_{i=0}^{\ell-1}$ of V^Γ with $U_i := (A^\Gamma)^{-1}(i)$. We define $f_{\mathsf{P}}(\mathcal{S}) := \{U_0, \ldots, U_{\ell-1}\}$. Notice that the assignment A^Γ is well-defined: for all $u \in V^\Gamma$

$$\begin{aligned} u \in V^\Gamma = V^\gamma \setminus Z &\implies d_{\gamma(G)}(u) > 0 \\ &\implies \exists t' \in V^\gamma : (u, t') \in E^\gamma \\ &\implies (u, t) \in E \\ &\implies A(u) \in D_1 \quad \text{for all valid assignments } A : V \to D_1 \cup D_2 \\ &\implies v_{A(u)} \in V^{\mathsf{P}} \end{aligned}$$

and similarly for all $u' \in V^\Gamma$

$$\begin{aligned} u' \in V^\Gamma = V^\gamma \setminus Z &\implies d_{\gamma(G)}(u') > 0 \\ &\implies \exists t \in V^\gamma : (t, u') \in E^\gamma \\ &\implies (t, u) \in E \\ &\implies A(u) \in D_2 \quad \text{for all valid assignments } A : V \to D_1 \cup D_2 \\ &\implies v'_{A(u)} \in V^{\mathsf{P}}. \end{aligned}$$

Now for any $u, t \in V$ and for any valid assignment $A : V \to D_1 \cup D_2$, we have

$$\begin{aligned} \mathsf{P}(A(u), A(t)) = 1 &\iff \{v_{A(u)}, v'_{A(t)}\} \in E^{\mathsf{P}} \\ &\iff c(v_{A(u)}) \neq c(v'_{A(t)}) \\ &\iff A^\Gamma(u) \neq A^\Gamma(t') \\ &\iff \ell\text{-}\mathsf{Cut}(A^\Gamma(u), A^\Gamma(t')) = 1 \end{aligned}$$

Moreover, by the definition of the restricted bipartite double cover, we have

$$w(u, t) = w^\gamma(u, t') = w^\Gamma(u, t') \quad \text{for all } u, t \in V,$$

implying that

$$\begin{aligned}\mathsf{P}_G(S_0,\ldots,S_{r-1}) = \mathsf{P}_G(A) &= \sum_{(u,t)\in E} w(u,t)\mathsf{P}(A(u),A(t))\\ &= \sum_{(u,t')\in E^\Gamma} w^\Gamma(u,t')\ell\text{-}\mathsf{Cut}(A^\Gamma(u),A^\Gamma(t')) = \ell\text{-}\mathsf{Cut}_{\Gamma(G)}(A^\Gamma)\\ &= \ell\text{-}\mathsf{Cut}_{\Gamma(G)}(U_0,\ldots,U_{\ell-1}) = \ell\text{-}\mathsf{Cut}_{\Gamma(G)}(f_{\mathsf{P}}(\mathcal{S}))\end{aligned}$$

as required.

Now, a proper efficient colouring does not exist for any arbitrary bipartite graph. In the remainder of this proof, we proceed as in Proposition 2.4.1 to show that a proper efficient colouring exists if the auxiliary graph G^{P} arises from a predicate P which contains no singleton squares.

Consider the graph G^{P} for some predicate $\mathsf{P} : D_1\times D_2 \to \{0,1\}$ which contains no singleton squares. Notice that G^{P} is bipartite with $V^{\mathsf{P}} = V_1 \sqcup V_2$ where $V_1 = \{v_{d_1^1},\ldots,v_{d_{r_1}^1}\}$, $V_2 = \{v'_{d_1^2},\ldots,v'_{d_{r_2}^2}\}$. Given that for any $B_1 \in \binom{D_1}{2}$ and $B_2 \in \binom{D_2}{2}$ we have $|\mathsf{P}^{-1}_{|B_1\times B_2}(1)| \neq 1$, for any $b_1^1, b_2^1 \in V_1$, $b_1^2, b_2^2 \in V_2$ we must have $|E(\{b_1^1,b_2^1,b_1^2,b_2^2\})| \neq 1$, implying that G^{P} satisfies the conditions of Lemma B.1.1. Therefore, the ℓ separate connected components which form its bipartite complement $\overline{G^{\mathsf{P}}}$ are complete bipartite graphs. Then, we can assign one of ℓ colours to each connected component to get a proper efficient colouring for the graph G^{P}. Now, G^{P} has a proper efficient colouring and hence we can construct a suitable map f_{P} as specified above, implying that P is indeed sparsifiable and that its sparsifiers have size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$. □

We conclude by presenting an equivalent of Theorem 2.4.1 for generalised binary predicates $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ as a consequence of Corollary B.1.1 and Proposition B.1.1.

Theorem B.1.1. Let $\mathsf{P} : D_1 \times D_2 \to \{0,1\}$ be a binary predicate. Then P is sparsifiable if and only if for any subsets $B_1 \in \binom{D_1}{2}$, $B_2 \in \binom{D_2}{2}$, $\mathsf{P}_{|B_1\times B_2}$ is sparsifiable. Moreover, if P is sparsifiable, it has a sparsifier of size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$.

Proof. ($\Leftarrow$). Assume that for any $B_1 \in \binom{D_1}{2}$, $B_2 \in \binom{D_2}{2}$, $\mathsf{P}_{|B_1\times B_2}$ is sparsifiable. Then, by Theorem 1.2.6, $|\mathsf{P}^{-1}_{|B_1\times B_2}(1)| \neq 1$ for all $B_1 \in \binom{D_1}{2}$, $B_2 \in \binom{D_2}{2}$. This means that P contains no singleton squares, and therefore by Proposition B.1.1, P is sparsifiable and its sparsifiers have size $\mathcal{O}\big(\frac{n}{\varepsilon^2}\big)$.

($\Rightarrow$). Assume by contradiction that there exist $B_1 \in \binom{D_1}{2}$, $B_2 \in \binom{D_2}{2}$ such that $\mathsf{P}_{|B_1\times B_2}$ is not sparsifiable. By Theorem 1.2.6, we must have $|\mathsf{P}^{-1}_{|B_1\times B_2}(1)| = 1$. Therefore, $\mathsf{P}_{|B_1\times B_2}$ is a singleton predicate and hence P contains a singleton square. Then, by Corollary B.1.1, P is non-sparsifiable. □

Appendix C

C.1 The k-ary Parity Predicate

We aim to show that, for $k \geq 3$, the k-ary Parity predicate does not contain a singleton ℓ-cube for any $\ell \leq k$, yet it cannot be written in terms of a hypergraph cut predicate. Recall that the Parity predicate $\mathsf{Par}^0 : [r]^k \to \{0,1\}$ is defined by

$$\mathsf{Par}^0(x_1, \ldots, x_k) = 1 \quad \iff \quad \sum_{i=1}^{k} x_i = 0 \text{ (mod 2)}.$$

Proposition C.1.1. For all $2 \leq \ell \leq k$, the k-ary Parity predicate $\mathsf{P}^0 : [r]^k \to \{0,1\}$ does not contain a singletone ℓ-cube.

Proof. We begin by showing that P^0 does not contain a singleton square.
Suppose by contradiction that there exist $\{D_j = \{d_1^j, d_2^j\}\}_{j \in \{1,2\}} \in \binom{[r]}{2}$, integers $n(1), n(2) \in \{1,2\}$, and a permutation $\sigma \in S_k$ such that there exist $x_3, \ldots, x_k \in [r]$ which satisfy

$$\mathsf{P}^0(\sigma(d^1_{n(1)}, d^2_{n(2)}, x_3, \ldots, x_k)) = 1$$

and for all $y_3, \ldots, y_k \in D$, for all $i_j \in \{1,2\}$,

$$\mathsf{P}^0(\sigma(d^1_{i_1}, d^2_{i_2}, y_3, \ldots, y_k)) = 1 \quad \implies \quad i_j = n(j) \text{ for all } j = 1,2. \tag{C.1}$$

Now for $j = 1,2$, define

$$\neg n(j) = i \in \{1,2\} : i \neq n(j).$$

Suppose $d_1^1 - d_2^1 = 0$ (mod 2). Then,

$$d^1_{n(1)} + d^2_{n(2)} + \sum_{j=3}^{k} x_k = d^1_{\neg n(1)} + d^2_{n(2)} + \sum_{j=3}^{k} x_k \text{ (mod 2)}$$

and hence

$$\mathsf{P}^0(\sigma(d^1_{\neg n(1)}, d^2_{n(2)}, x_3, \ldots, x_k)) = \mathsf{P}^0(\sigma(d^1_{n(1)}, d^2_{n(2)}, x_3, \ldots, x_k)) = 1$$

contradicting property (C.1).

Then, suppose then that $d_1^1 - d_2^1 = 1 \pmod 2$. We have

$$d^1_{n(1)} + d^2_{n(2)} + \sum_{j=3}^{k} x_k = d^1_{\neg n(1)} + d^2_{n(2)} + (x_3 + 1) + \sum_{j=4}^{k} x_k \pmod 2$$

and hence

$$\mathsf{P}^0(\sigma(d^1_{\neg n(1)}, d^2_{n(2)}, x_3 + 1, x_4, \ldots, x_k)) = \mathsf{P}^0(\sigma(d^1_{n(1)}, d^2_{n(2)}, x_3, \ldots, x_k)) = 1$$

again contradicting property (C.1). Therefore, P^0 does not contain a singleton square. Notice that, by the definition of a singleton ℓ-cube, for all $2 \le \ell \le k$ every k-ary predicate which contains a singleton ℓ-cube also contains a singleton 2-cube. Then, P^0 does not contain a singleton ℓ-cube. □

Proposition C.1.2. Consider the k-ary Parity predicate $\mathsf{P}^0 : [r]^k \to \{0,1\}$ for $k \ge 3$. Then for all weighted directed k-uniform hypergraphs $H = (V, E, w)$ with $|V| \ge rk$, for all $r' \ge 2$, and for all functions $f : Part_r(V) \to Part_{r'}(V^\gamma)$, there exists a partition of the vertices $\mathcal{S} \in Part_r(V)$ such that

$$\mathsf{P}^0_H(\mathcal{S}) \ne r'\text{-}\mathsf{Cut}_{\gamma(H)}(f(\mathcal{S})),$$

where $r'\text{-}\mathsf{Cut} : [r']^k \to \{0,1\}$ is the k-ary r'-Cut predicate and $\gamma(H)$ is the k-partite k-fold cover of H.

Proof. We proceed by contradiction. Suppose that there exist a weighted directed k-uniform hypergraph $H = (V, E, w)$ with $|V| \ge rk$, an integer $r' \ge 2$, and a function $f_{\mathsf{P}^0} : Part_r(V) \to Part_{r'}(V^\gamma)$ such that for all partitions $\mathcal{S} \in Part_r(V)$ we have

$$\mathsf{P}^0_H(\mathcal{S}) = r'\text{-}\mathsf{Cut}_{\gamma(H)}(f(\mathcal{S})). \tag{C.2}$$

Pick some $\mathcal{S} = \{S_0, \ldots, S_{r-1}\} \in Part_r(V)$ with $|S_i| \ge k$ for all $i \in [r]$ and suppose $f_{\mathsf{P}^0}(\mathcal{S}) = \{U_0, \ldots, U_{r'-1}\}$. Let $A_{\mathcal{S}} : V \to [r]$ be the assignment defined by $A^{-1}_{\mathcal{S}}(i) = S_i$ for $i \in [r]$. Define a corresponding assignment $A_{f_{\mathsf{P}^0}(\mathcal{S})} : V^\gamma \to [r']$ such that, for all $i \in [r']$ and for all $j \in [k]$,

$$A_{f_{\mathsf{P}^0}(\mathcal{S})}(v^{(j)}) = i \iff S^{(j)}_{A(v)} \subseteq U_i.$$

First of all, we need to show that the assignment $A_{f_{\mathsf{P}^0}(\mathcal{S})}$ is well-defined. Notice that for all $i \in [r]$, for all $j \in [k]$, for all $u^{(j)}, v^{(j)} \in S_i^{(j)}$ and for all $\ell \in [r']$ we must have $\{u^{(j)}, v^{(j)}\} \cap U_\ell \in \{\emptyset, \{u^{(j)}, v^{(j)}\}\}$. For suppose by contradiciton that there exist $i \in [r]$, $j \in [k]$, and $u^{(j)}, v^{(j)} \in S_i^{(j)}$ such that $u^{(j)} \in U_{\ell_u}$ and $v^{(j)} \in U_{\ell_v}$ with $\ell_u \ne \ell_v$. Assume without loss of generality that $j = 0$. Then, for all $v_2, \ldots, v_k \in V$ and for $S_i \in \mathcal{S} \in Part_r(V)$, we would have

$$\begin{aligned}\mathsf{P}^0(A_{\mathcal{S}}(u, v_2 \ldots, v_k)) &= \mathsf{P}^0(A_{\mathcal{S}}(u), A_{\mathcal{S}}(v_2), \ldots, A_{\mathcal{S}}(v_k)) = \mathsf{P}^0(i, A_{\mathcal{S}}(v_2), \ldots, A_{\mathcal{S}}(v_k)) \\ &= \mathsf{P}^0(A_{\mathcal{S}}(v), A_{\mathcal{S}}(v_2), \ldots, A_{\mathcal{S}}(v_k)) = \mathsf{P}^0(A_{\mathcal{S}}(v, v_2, \ldots, v_k))\end{aligned}$$

and hence

$$\begin{aligned}r'\text{-}\mathsf{Cut}(A_{f_{\mathsf{P}^0}(\mathcal{S})}(u^{(0)}, v_2^{(1)}, \ldots, v_k^{(k-1)})) = \mathsf{P}^0(A_{\mathcal{S}}(u, v_2, \ldots, v_k)) &= \mathsf{P}^0(A_{\mathcal{S}}(v, v_2, \ldots, v_k)) \\ &= r'\text{-}\mathsf{Cut}(A_{f_{\mathsf{P}^0}(\mathcal{S})}(v^{(0)}, v_2^{(1)}, \ldots, v_k^{(k-1)}))\end{aligned}$$

Now pick $v_2, \ldots, v_k$ such that $\mathsf{P}^0(A_{\mathcal{S}}(u, v_2, \ldots, v_k)) = 0$. Then,

$$\begin{aligned}r'\text{-}\mathsf{Cut}(A_{f_{\mathsf{P}^0}(\mathcal{S})}(u^{(0)}, v_2^{(1)}, \ldots, v_k^{(k-1)})) &= \mathsf{P}^0(A_{\mathcal{S}}(u, v_2, \ldots, v_k)) = 0 \\ &\implies v_2^{(1)}, \ldots, v_k^{(k-1)} \in U_{\ell_u}\end{aligned} \tag{C.3}$$

and

$$\begin{aligned}r'\text{-}\mathsf{Cut}(A_{f_{\mathsf{P}^0}(\mathcal{S})}(v^{(0)}, v_2^{(1)}, \ldots, v_k^{(k-1)})) &= \mathsf{P}^0(A_{\mathcal{S}}(v, v_2, \ldots, v_k)) \\ &= \mathsf{P}^0(A_{\mathcal{S}}(u, v_2, \ldots, v_k)) = 0 \\ &\implies v_2^{(1)}, \ldots, v_k^{(k-1)} \in U_{\ell_v}\end{aligned} \tag{C.4}$$

Putting (C.3) and (C.4) together we get

$$\begin{aligned}v_2^{(1)}, \ldots, v_k^{(k-1)} \in U_{\ell_u} \cap U_{\ell_v} &\implies U_{\ell_u} \cap U_{\ell_v} \neq \emptyset \\ &\implies \ell_u = \ell_v\end{aligned}$$

contradicting our initial assumption that $\ell_u \neq \ell_v$. So for all $j \in [k]$, for all $u^{(j)}, v^{(j)} \in S_i^{(j)}$ and for all $\ell \in [r']$ we have $\{u^{(j)}, v^{(j)}\} \cap U_\ell \in \{\emptyset, \{u^{(j)}, v^{(j)}\}\}$ and hence $A_{f_{\mathsf{P}^0}(\mathcal{S})}$ is well-defined.

Now we want to consider vertices which belong to sets S_i of different parity. Without loss of generality, pick k vertices $u_1, \ldots, u_k \in S_0$ and 3 vertices $v_1, v_2, v_3 \in S_1$. Then we have

$$\begin{aligned}&\mathsf{P}^0(A_{\mathcal{S}}(v_1, u_2, \ldots, u_k)) = \mathsf{P}^0(1, 0, \ldots, 0) = 0 \\ \text{since} \quad &A_{\mathcal{S}}(v_1) + \sum_{j=2}^{k} A_{\mathcal{S}}(u_j) = 1 \pmod 2\end{aligned}$$

and

$$\begin{aligned}&\mathsf{P}^0(A_{\mathcal{S}}(v_1, v_2, v_3, u_4, \ldots, u_k)) = \mathsf{P}^0(1, 1, 1, 0, \ldots, 0) = 0 \\ \text{since} \quad &A_{\mathcal{S}}(v_1) + A_{\mathcal{S}}(v_2) + A_{\mathcal{S}}(v_3) + \sum_{j=4}^{k} A_{\mathcal{S}}(u_j) = 3 = 1 \pmod 2.\end{aligned}$$

Then, by property (C.2) we must have

$$r'\text{-}\mathsf{Cut}(A_{f_{P^0}(\mathcal{S})}(v_1^{(0)}, u_2^{(1)}, \ldots, u_k^{(k-1)})) = 0$$

and

$$r'\text{-}\mathsf{Cut}(A_{f_{P^0}(\mathcal{S})}(v_1^{(0)}, v_2^{(1)}, v_3^{(2)}, u_4^{(3)}, \ldots, u_k^{(k-1)})) = 0$$

respectively.
By the definition of r'-Cut, this implies that there exist $x, y \in [r']$ such that, for

$$X := S_1^{(0)} \sqcup S_0^{(1)} \sqcup S_0^{(2)} \sqcup \ldots \sqcup S_0^{(k-1)}$$

and

$$Y := S_1^{(0)} \sqcup S_1^{(1)} \sqcup S_1^{(2)} \sqcup S_0^{(3)} \sqcup S_0^{(4)} \sqcup \ldots \sqcup S_0^{(k-1)}$$

we have

$$X \cap U_x = X \quad \text{and} \quad Y \cap U_y = Y,$$

that is, hyperedges whose vertices lie wholly in X or wholly in Y do not contribute to the cut. But then,

$$S_1^{(0)} \subset (X \cap Y) \subseteq U_x \cap U_y$$

which implies $U_x \cap U_y \neq \emptyset$ and hence $x = y$. It follows that

$$S_1^{(0)} \sqcup S_1^{(1)} \sqcup S_0^{(2)} \sqcup S_0^{(3)} \sqcup \ldots \sqcup S_0^{(k-1)} \subset X \cup Y \subseteq U_x$$

and hence

$$r'\text{-}\mathsf{Cut}(A_{f_{P^0}(\mathcal{S})}(v_1^{(0)}, v_2^{(1)}, u_3^{(2)}, u_4^{(3)}, \ldots, u_k^{(k-1)})) = 0$$

implying by property (C.2) that

$$\mathsf{P}^0(A(v_1, v_2, u_3, u_4, \ldots, u_k)) = \mathsf{P}^0(1, 1, 0, 0, \ldots, 0) = 0,$$

a contradiction since

$$A_{\mathcal{S}}(v_1) + A_{\mathcal{S}}(v_2) + \sum_{j=3}^{k} A_{\mathcal{S}}(u_j) = 2 = 0 \pmod 2.$$

Therefore, such a map f_{P^0} cannot exist. $\square$

Bibliography

[ACK+16] Alexandr Andoni, Jiecao Chen, Robert Krauthgamer, Bo Qin, David P. Woodruff, and Qin Zhang. On Sketching Quadratic Forms. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science, Cambridge, MA, USA, January 14-16, 2016*, pages 311–319, 2016.

[BK96] András A. Benczúr and David R. Karger. Approximating s-t Minimum Cuts in $\tilde{O}(n^2)$ Time. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 47–55, 1996.

[BSS12] Joshua D. Batson, Daniel A. Spielman, and Nikhil Srivastava. Twice-Ramanujan Sparsifiers. *SIAM J. Comput.*, 41(6):1704–1721, 2012.

[CCJ+17] David A. Cohen, Martin C. Cooper, Peter G. Jeavons, Andrei A. Krokhin, Robert Powell, and Stanislav Živný. Binarisation for valued constraint satisfaction problems. *SIAM J. Discrete Math.*, 31(4):2279–2300, 2017.

[CCJK06] David A. Cohen, Martin C. Cooper, Peter Jeavons, and Andrei A. Krokhin. The Complexity of Soft Constraint Satisfaction. *Artif. Intell.*, 170(11):983–1016, 2006.

[CLTZ18] T.-H. Hubert Chan, Anand Louis, Zhihao Gavin Tang, and Chenzi Zhang. Spectral Properties of Hypergraph Laplacian and Approximation Algorithms. *J. ACM*, 65(3):15:1–15:48, 2018.

[EGIN97] David Eppstein, Zvi Galil, Giuseppe F. Italiano, and Amnon Nissenzweig. Sparsification - a technique for speeding up dynamic graph algorithms. *J. ACM*, 44(5):669–696, 1997.

[FK17] Arnold Filtser and Robert Krauthgamer. Sparsification of two-variable valued constraint satisfaction problems. *SIAM J. Discrete Math.*, 31(2):1263–1276, 2017.

[Kar94a] David R. Karger. Random sampling in cut, flow, and network design problems. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada*, pages 648–657, 1994.

[Kar94b] David R. Karger. Using Randomized Sparsification to Approximate Minimum Cuts. In *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms. 23-25 January 1994, Arlington, Virginia, USA.*, pages 424–432, 1994.

[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 2015, Rehovot, Israel, January 11-13, 2015*, pages 367–376, 2015.

[KZ17] Andrei Krokhin and Stanislav Živný. The Complexity of Valued CSPs. In Andrei Krokhin and Stanislav Živný, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 233–266. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017.

[RN03] Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach, 2nd Edition*, chapter 6. Prentice Hall series in Artificial Intelligence. Prentice Hall, 2003.

[RvBW06] Francesca Rossi, Peter van Beek, and Toby Walsh, editors. *Handbook of Constraint Programming*, volume 2 of *Foundations of Artificial Intelligence*. Elsevier, 2006.

[ST11] Daniel A. Spielman and Shang-Hua Teng. Spectral Sparsification of Graphs. *SIAM J. Comput.*, 40(4):981–1025, 2011.

[Vol09] Vitaly Ivanovich Voloshin. *Introduction to Graph and Hypergraph Theory*. Nova Science Publishers, New York, 2009.