SPARSIFICATION OF BINARY CSPS [*]

SILVIA BUTTI[†] AND STANISLAV ŽIVNÝ[‡]

Abstract.
A cut ε-sparsifier of a weighted graph G is a re-weighted subgraph of G of (quasi)linear size that preserves the size of all cuts up to a multiplicative factor of ε. Since their introduction by Benczúr and Karger [STOC'96], cut sparsifiers have proved extremely influential and found various applications. Going beyond cut sparsifiers, Filtser and Krauthgamer [SIDMA'17] gave a precise classification of which binary Boolean CSPs are sparsifiable. In this paper, we extend their result to binary CSPs on arbitrary finite domains.

1. Introduction. The pioneering work of Benczúr and Karger [4] showed that every edge-weighted undirected graph $G = (V, E, w)$ admits a cut-sparsifier. In particular, assuming that the edge weights are positive, for every $0 < \varepsilon < 1$ there exists (and in fact can be found efficiently) a re-weighted subgraph $G_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ of G with $|E_\varepsilon| = O(\varepsilon^{-2} n \log n)$ edges such that

$$\forall S \subseteq V, \qquad \mathsf{Cut}_{G_\varepsilon}(S) \in (1 \pm \varepsilon)\mathsf{Cut}_G(S),$$

where $n = |V|$ and $\mathsf{Cut}_G(S)$ denotes the total weight of edges in G with exactly one endpoint in S. The bound on the number of edges was later improved to $O(\varepsilon^{-2} n)$ by Batson, Spielman, and Srivastava [3]. Moreover, the bound $O(\varepsilon^{-2} n)$ is known to be tight by the work of Andoni, Chen, Krauthgamer, Qin, Woodruff, and Zhang [2].

The original motivation for cut sparsification was to speed up algorithms for cut problems and graph problems more generally. The idea turned out to be very influential, with several generalisations and extensions, including, for instance, sketching [1, 2], sparsifiers for cuts in hypergraphs [10, 12], and spectral sparsification [16, 15, 14, 9, 13].

Filtser and Krauthgamer [8] considered the following natural question: which binary Boolean CSPs are sparsifiable? In order to state their results as well as our new results, we will now define binary constraint satisfaction problems.

An instance of the binary[1] *constraint satisfaction problem* (CSP) is a quadruple $I = (V, D, \Pi, w)$, where V is a set of variables, D is a finite set called the *domain*,[2] Π is a set of constraints, and $w : \Pi \to \mathbb{R}_+$ are positive weights for the constraints. Each constraint $\pi \in \Pi$ is a pair $((u, v), P)$, where $(u, v) \in V^2$, called the constraint *scope*,

---

[*]Submitted to the editors on 4th February 2019.

Funding: An extended abstract of this work appeared in *Proceedings of the 36th International Symposium on Theoretical Aspects of Computer Science (STACS 2019)* [6]. Stanislav Živný was supported by a Royal Society University Research Fellowship. Work mostly done while Silvia Butti was at the University of Oxford. The project that gave rise to these results received the support of a fellowship from "a Caixa" Foundation (ID 100010434). The fellowship code is LCF/BQ/DI18/11660056. This project has received funding from the European Union's Horizon 2020 research and innovation programme grant agreement No 714530 and under the Marie Skłodowska-Curie grant agreement No 713673. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.

[†]Department of Information and Communication Technologies, Universitat Pompeu Fabra, Spain (silvia.butti@upf.edu).

[‡]Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford, OX1 3QD, UK (standa.zivny@cs.ox.ac.uk).

[1]Some papers use the term *two-variable*.

[2]Some papers use the term *alphabet*.

is a pair of distinct variables from V, and $P : D^2 \to \{0,1\}$ is a binary predicate. A CSP instance is called *Boolean* if $|D| = 2$, i.e., if the domain is of size two.[3]

For a fixed binary predicate P, we denote by $\mathrm{CSP}(P)$ the class of CSP instances in which all constraints use the predicate P. Note that if we take $D = \{0,1\}$ and P defined on D^2 by $P(x,y) = 1$ iff $x \neq y$ then $\mathrm{CSP}(P)$ corresponds to the cut problem.

We say that a constraint $\pi = ((u,v),P)$ is *satisfied* by an assignment $A : V \to D$ if $P(A(u),A(v)) = 1$. The value of an instance $I = (V,D,\Pi,w)$ under an assignment $A : V \to D$ is defined to be the total weight of satisfied constraints:

$$\mathrm{Val}_I(A) = \sum_{\pi=((u,v),P)\in\Pi} w(\pi)P(A(u),A(v)).$$

For $0 < \varepsilon < 1$, an ε-sparsifier of $I = (V,D,\Pi,w)$ is a re-weighted subinstance $I_\varepsilon = (V,D,\Pi_\varepsilon \subseteq \Pi, w_\varepsilon)$ of I such that

$$\forall A : V \to D, \qquad \mathrm{Val}_{I_\varepsilon}(A) \in (1 \pm \varepsilon)\,\mathrm{Val}_I(A).$$

The goal is to obtain a sparsifier with the minimum number of constraints, i.e., $|\Pi_\varepsilon|$.

A binary predicate P is called *sparsifiable* if for every instance $I \in \mathrm{CSP}(P)$ on $n = |V|$ variables and for every $0 < \varepsilon < 1$ there is an ε-sparsifier for I with $O(\varepsilon^{-2}n)$ constraints.

We call a (not necessarily Boolean or binary) predicate P a *singleton* if $|P^{-1}(1)| = 1$.

Filtser and Krauthgamer showed, among other results, the following classification. Let P be a binary Boolean predicate. Then, P is sparsifiable if and only if P is not a singleton.[4] In other words, the only predicates that are not sparsifiable are those with support of size one.

Contributions. As our main contribution, we identify in Theorem 2.2 the precise borderline of sparsifiability for binary predicates on arbitrary finite domains, thus extending the work from [8] on Boolean predicates. Let P be a binary predicate defined on an arbitrary finite domain D. Then, P is sparsifiable if and only if P does not "contain" a singleton subpredicate. More precisely, we say that P "contains" a singleton subpredicate if there are two (not necessarily disjoint) subdomains $B, C \subseteq D$ with $|B| = |C| = 2$ such that the restriction of P onto $B \times C$ is a singleton predicate.

The crux of Theorem 2.2 is the sparsifiability part, which is established by a reduction to cut sparsifiers. Unlike in the classification of binary Boolean predicates from [8], we do not rely on a case analysis that differs for different sparsifiable predicates but instead give a simpler argument for all sparsifiable predicates. The idea is to reduce (the graph of) any CSP instance, as was done in [8], to a new graph via the so-called bipartite double cover [5]. However, there is no natural assignment in the new graph (as it was in the Boolean case in [8]). In order to overcome this, we define a graph G^P whose edges correspond to the support of the predicate P. Using a simple combinatorial argument, we show (in Proposition 2.7) that, under the assumption that P does not "contain" a singleton subpredicate, the bipartite complement of G^P is a collection of bipartite cliques. This special structure allows us to find a good assignment in the new graph.

[3]Some papers use the term *binary* to mean domains of size two. In this paper, *Boolean* always refers to a domain of size two and *binary* always refers to the arity of the constraint(s).

[4]Filtser and Krauthgamer use the term *valued CSPs* for what we defined as CSPs. We prefer CSPs in order to distinguish them from the much more general framework of valued CSPs studied in [11].

In view of Filtser and Krauthgamer's work [8], one might conjecture that P is sparsifiable if and only if P is not a singleton. While it is easy to show that if a (possibly non-binary and non-Boolean) predicate P is a singleton then P is not sparsifiable (cf. Section A.4 in the appendix), our results show that the borderline of sparsifiability lies elsewhere. In particular, by Theorem 2.2, there are binary non-Boolean predicates that are not sparsifiable but are not singletons. Also, there are non-binary Boolean predicates that are not sparsifiable but are not singletons (cf. Section A.4).

We remark that the term "sparsification" is also used in an unrelated line of work in which the goal is, given a CSP instance, to reduce the number of constraints without changing satisfiability of the instance; see, e.g., [7].

2. Classification of Binary Predicates. Throughout the paper we denote by $n = |V|$ the number of variables of a given CSP instance.

The following classification of binary Boolean predicates is from [8].

THEOREM 2.1 ([8, Theorem 3.7]). *Let $P : \{0,1\}^2 \to \{0,1\}$ be a binary Boolean predicate. Let $0 < \varepsilon < 1$.*

1. *If P is a singleton then there exists an instance I of $\mathrm{CSP}(P)$ such that every ε-sparsifier of I has $\Omega(n^2)$ constraints.*
2. *Otherwise, for every instance I of $\mathrm{CSP}(P)$ there exists an ε-sparsifier of I with $O(\varepsilon^{-2}n)$ constraints.*

We denote by $\binom{D}{2} = \{B \subseteq D : |B| = 2\}$ the set of two-element subsets of D. For a binary predicate $P : D^2 \to \{0,1\}$ and $B, C \in \binom{D}{2}$, $P|_{B\times C}$ denotes the restriction of P onto $B \times C$.

The following is our main result, generalising Theorem 2.1 to arbitrary finite domains.

THEOREM 2.2 (**Main**). *Let $P : D^2 \to \{0,1\}$ be a binary predicate, where D is a finite set with $|D| \geq 2$. Let $0 < \varepsilon < 1$.*

1. *If there exist $B, C \in \binom{D}{2}$ such that $P|_{B\times C}$ is a singleton then there exists an instance I of $\mathrm{CSP}(P)$ such that every ε-sparsifier of I has $\Omega(n^2)$ constraints.*
2. *Otherwise, for every instance I of $\mathrm{CSP}(P)$ there exists an ε-sparsifier of I with $O(\varepsilon^{-2}n)$ constraints.*

The rest of this section is devoted to proving Theorem 2.2.

First we introduce some useful notation. We set $[r] = \{0, 1, \ldots, r-1\}$. We denote by $X \sqcup Y$ the disjoint union of X and Y. For any $r \geq 2$, we define $r\text{-}\mathsf{Cut} : [r]^2 \to \{0,1\}$ by $r\text{-}\mathsf{Cut}(x, y) = 1$ if and only if $x \neq y$.

Given an instance $I = (V, D, \Pi, w) \in \mathrm{CSP}(P)$, we denote by G^I the *corresponding graph* of I; that is, $G^I = (V, E, w)$ is a weighted directed graph with $E = \{(u, v) : ((u, v), P) \in \Pi\}$ and $w(u, v) = w((u, v), P)$. Conversely, given a weighted directed graph $G = (V, E, w)$ and a predicate $P : D^2 \to \{0,1\}$, the *corresponding* $\mathrm{CSP}(P)$ *instance* is $I^{G,P} = (V, D, \Pi, w)$, where $\Pi = \{(e, P) : e \in E\}$ and $w(e, P) = w(e)$. Hence, we can equivalently talk about instances of $\mathrm{CSP}(P)$ or (weighted directed) graphs. Thus, an *ε-P-sparsifier* of a graph $G = (V, E, w)$ is a subgraph $G_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ whose corresponding $\mathrm{CSP}(P)$ instance $I^{G_\varepsilon,P}$ is an ε-sparsifier of the corresponding $\mathrm{CSP}(P)$ instance $I^{G,P}$ of G.

Case (1) of Theorem 2.2 is established by the following result.

THEOREM 2.3. *Let $P : D^2 \to \{0,1\}$ be a binary predicate. Assume that there exist $B, C \in \binom{D}{2}$ such that $P|_{B\times C}$ is a singleton. For any n there is a $\mathrm{CSP}(P)$ instance*

I with $2n$ variables and n^2 constraints such that for any $0 < \varepsilon < 1$ it holds that any ε-sparsifier of I has n^2 constraints.

Proof. Suppose $B = \{b, b'\}$, $C = \{c, c'\}$ and assume without loss of generality that ${P|_{B\times C}}^{-1}(1) = \{(b, c)\}$; that is, the support of $P|_{B\times C}$ is equal to $\{(b, c)\}$. Consider a CSP(P) instance $I = (V, D, \Pi, w)$, where

- $V = X \sqcup Y$, $X = \{x_1, \dots, x_n\}$, and $Y = \{y_1, \dots, y_n\}$;
- $\Pi = \{\pi_{ij} = ((x_i, y_j), P) : 1 \le i, j \le n\}$;
- w are arbitrary positive weights.

We have $|\Pi| = n^2$. We note that B and C may not be disjoint. We consider the family of assignments $A_{ij} : V \to B \cup C$ for $1 \le i, j \le n$ such that $A_{ij}(x_i) = b$, $A_{ij}(x) = b'$ for every $x \in X \setminus \{x_i\}$, $A_{ij}(y_j) = c$, and $A_{ij}(y) = c'$ for every $y \in Y \setminus \{y_j\}$. Then, we have

$$P(A_{ij}(u, v)) = \begin{cases} P(b, c) &= 1 \qquad \text{if } u = x_i, v = y_j, \\ P(b, c') &= 0 \qquad \text{if } u = x_i, v \in Y \setminus \{y_j\}, \\ P(b', c) &= 0 \qquad \text{if } u \in X \setminus \{x_i\}, v = y_j, \\ P(b', c') &= 0 \qquad \text{if } u \in X \setminus \{x_i\}, v \in Y \setminus \{y_j\}. \end{cases}$$

Therefore,

$$\mathrm{Val}_I(A_{ij}) = \sum_{\pi \in \Pi} w(\pi) P(A_{ij}(\pi)) = w(\pi_{ij}) > 0.$$

Hence, if $I_\varepsilon = (V, D, \Pi_\varepsilon, w_\varepsilon)$ is an ε-sparsifier of I, we must have that $\pi_{ij} \in \Pi_\varepsilon$ for every $1 \le i, j \le n$, as otherwise we would have

$$\mathrm{Val}_{I_\varepsilon}(A_{ij}) = \sum_{\pi \in \Pi_\varepsilon} w_\varepsilon(\pi) P(A_{ij}(\pi)) = 0 \notin (1 \pm \varepsilon) \mathrm{Val}_I(A_{ij}).$$

Therefore, we have $\Pi_\varepsilon = \Pi$ and hence $|\Pi_\varepsilon| = |\Pi| = n^2$. ◻

The main tool used in the proof of Theorem 2.1 (2) from [8] is a graph transformation known as the bipartite double cover [5], which allows for a reduction to cut sparsifiers [3].

Definition 2.4. *For a weighted directed graph $G = (V, E, w)$, the* bipartite double cover *of G is the weighted directed graph $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$, where*

- *$V^\gamma = \{v^{(0)}, v^{(1)} : v \in V\}$;*
- *$E^\gamma = \{(u^{(0)}, v^{(1)}) : (u, v) \in E\}$;*
- *$w^\gamma(u^{(0)}, v^{(1)}) = w(u, v)$.*

Given an assignment $A : V \to [r]$, we let $\mathcal{A} = (A_0, \dots, A_{r-1})$ be the induced r-partition of V, where $A_j = A^{-1}(j)$. For a binary predicate $P : [r]^2 \to \{0, 1\}$ and an instance $I = (V, [r], \Pi, w) \in \mathrm{CSP}(P)$, we define $\mathrm{Val}_I(\mathcal{A}) = \mathrm{Val}_I(A)$. Moreover, for a weighted directed graph G and a binary predicate P, we define $\mathrm{Val}_{G,P}(\mathcal{A}) = \mathrm{Val}_{I^{G,P}}(\mathcal{A})$. We denote the set of all r-partitions of V by $Part_r(V)$.

For any r-partition $\mathcal{A} = (A_0, \dots, A_{r-1})$ of the vertices of V, let $A_i^{(j)} = \{v^{(j)} : v \in A_i\}$. Thus $\mathcal{A}^\gamma = (A_0^{(0)}, A_0^{(1)}, \dots, A_{r-1}^{(0)}, A_{r-1}^{(1)})$ is a $2r$-partition of the vertices of V^γ.

We use an argument from the proof of Theorem 2.1 (2) from [8] and apply it to non-Boolean predicates.

Proposition 2.5. *Let $P : [r]^2 \to \{0, 1\}$ and $P' : [r']^2 \to \{0, 1\}$ be binary predicates. Suppose that there is a function $f_P : Part_r(V) \to Part_{r'}(V^\gamma)$ such that for any weighted directed graph G on V and for any r-partition $\mathcal{A} \in Part_r(V)$ it holds* *that*

$$\mathrm{Val}_{G,P}(\mathcal{A}) = \mathrm{Val}_{\gamma(G),P'}(f_P(\mathcal{A})),$$

where $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$ is the bipartite double cover of G. If there is an ε-P'-sparsifier of $\gamma(G)$ of size $g(n)$ then there is an ε-P-sparsifier of G of size $g(n)$.

Proof. Given $G = (V, E, w)$, let $\gamma(G)_\varepsilon = (V, E^\gamma_\varepsilon, w^\gamma_\varepsilon)$ be an ε-P'-sparsifier (of size $g(n)$) of the bipartite double cover $\gamma(G)$ of G. Define a subgraph $G_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of G by $E_\varepsilon = \{(u, v) : (u^{(0)}, v^{(1)}) \in E^\gamma_\varepsilon\}$ and $w_\varepsilon(u, v) = w^\gamma_\varepsilon(u^{(0)}, v^{(1)})$. Note that $\gamma(G_\varepsilon) = \gamma(G)_\varepsilon$ and $E_\varepsilon \subseteq E$.

Then, we have

$$\begin{aligned}\mathrm{Val}_{G_\varepsilon,P}(\mathcal{A}) &= \mathrm{Val}_{\gamma(G_\varepsilon),P'}(f_P(\mathcal{A}))\\ &= \mathrm{Val}_{\gamma(G)_\varepsilon,P'}(f_P(\mathcal{A})) \in (1 \pm \varepsilon)\,\mathrm{Val}_{\gamma(G),P'}(f_P(\mathcal{A})) = (1 \pm \varepsilon)\,\mathrm{Val}_{G,P}(\mathcal{A}).\end{aligned}$$

Hence G_ε is also an ε-P-sparsifier of G of size $g(n)$. ∎

We now focus on proving Case (2) of Theorem 2.2. Assume that for any $B, C \in \binom{D}{2}$, $P|_{B\times C}$ is not a singleton. Our strategy is to show that in this case the value of a CSP(P) instance under any assignment can be expressed as the value of a corresponding CSP(ℓ-Cut) instance (for some $\ell \leq 2|D|$) under the same assignment.

For an undirected graph $G = (V, E)$ and a subset $U \subseteq V$, we denote the vertex-induced subgraph on U by $G[U]$ and its edge set by $E[U]$. For a possibly disconnected undirected 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 $\deg_G(v)$.

Definition 2.6. *Let $G = (U \sqcup V, E)$ be an undirected bipartite graph. The* bipartite complement $\overline{G} = (U \sqcup V, \overline{E})$ *of G has the following edge set:*

$$\overline{E} = \{\{u, v\} : u \in U, v \in V, \{u, v\} \notin E\}.$$

The following property of bipartite graphs will be crucial in the proof of Theorem 2.8.

Proposition 2.7. *Let $G = (U \sqcup V, E)$ be a bipartite graph with $|U| = |V| = r$, $r \geq 2$. Assume that for any $u, u' \in U$ and $v, v' \in V$ we have $|E[\{u, u', v, v'\}]| \neq 1$. Then, for any $v \in U \sqcup V$ with $\deg_{\overline{G}}(v) > 0$, $\overline{G}_v$ is a complete bipartite graph with partition classes $\{U \cap V(\overline{G}_v)\}$ and $\{V \cap V(\overline{G}_v)\}$.*

Proof. For contradiction, assume that there are $u \in U$ and $v \in V$ such that $\{u, v\} \notin \overline{E}$ but u and v belong to the same connected component of $\overline{G}$. Choose u and v with the shortest possible distance between them. Let $u = u_0, u_1, \dots, u_k = v$ be a shortest path between u and v in $\overline{G}$, where $k \geq 3$ is odd. We will show that $|\overline{E}[\{u_0, u_1, u_{k-1}, u_k\}]| = 3$, which contradicts the assumption that

$$|E[\{u_0, u_1, u_{k-1}, u_k\}]| \neq 1.$$

If $k = 3$ then the claim holds since we assumed that $\{u_0, u_1\}, \{u_1, u_2\}, \{u_2, u_3\} \in \overline{E}$ and $\{u_0, u_3\} \notin \overline{E}$.

Let $k \geq 5$. We will be done if we show that $\{u_1, u_{k-1}\} \in \overline{E}$, as by our assumptions $\{u_0, u_1\}, \{u_{k-1}, u_k\} \in \overline{E}$ and $\{u_0, u_k\} \notin \overline{E}$. To this end, note that $\{u_0, u_{k-2}\} \in \overline{E}$ as otherwise u_0 and u_{k-2} would be a pair of vertices with the required properties but of distance $k - 2$, contradicting our choice of u and v. Thus,

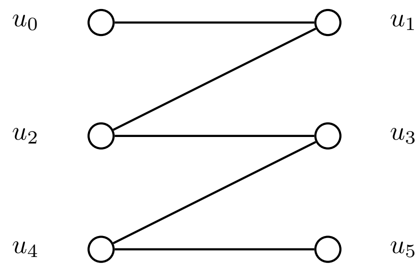

FIG. 1. *Illustration of the proof of Proposition 2.7 for $k = 5$.*

$\{u_1, u_{k-1}\} \in \overline{E}$ as otherwise we would have $|\overline{E}[\{u_0, u_1, u_{k-2}, u_{k-1}\}]| = 3$, which contradicts $|E[\{u_0, u_1, u_{k-2}, u_{k-1}\}]| \neq 1$. (See Figure 1 for an illustration of the case $k = 5$.) ∎

Case (2) of Theorem 2.2 is established by the following result.

THEOREM 2.8. *Let $P : D^2 \to \{0,1\}$ be a binary predicate such that for any $B, C \in \binom{D}{2}$ we have that $P|_{B\times C}$ is not a singleton. Then, for every $0 < \varepsilon < 1$ and every instance I of* CSP(P) *there is a sparsifier of I with $O(\varepsilon^{-2}n)$ constraints.*

Proof. Let $I = (V, D, \Pi, w)$ be an instance of CSP(P) with $r = |D|$. Without loss of generality, we assume that $D = [r]$. Let $G = G^I = (V, E, w)$ be the corresponding (weighted directed) graph of I, and let $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$ be the bipartite double cover of G. Recall that for an assignment $A : V \to [r]$, we denote $A_i = A^{-1}(i)$. Thus, $\mathcal{A} = (A_0, \dots, A_{r-1})$ forms an r-partition of V.

Our goal is to show the existence of a function $f_P : Part_r(V) \to Part_\ell(V^\gamma)$ (for some fixed $\ell \le 2r$) such that

$$\forall A : V \to [r], \qquad \mathrm{Val}_{G,P}(\mathcal{A}) = \mathrm{Val}_{\gamma(G),\ell\text{-}\mathsf{Cut}}(f_P(\mathcal{A})). \tag{2.1}$$

Assuming the existence of f_P, we can finish the proof as follows. Batson, Spielman, and Srivastava established the existence of a sparsifier of size $O(\varepsilon^{-2}n)$ for any instance of CSP(2-Cut) [3]. By [8, Section 6.2], this implies the existence of a sparsifier of size $O(\varepsilon^{-2}n)$ for any instance of CSP(ℓ-Cut). Consequently, by Proposition 2.5 and (2.1), there is a sparsifier of size $O(\varepsilon^{-2}n)$ for the instance $I^{G,P} = I$.

In the proof of Theorem 2.1 (2) in [8], functions f_P are given for any binary Boolean predicate P with support size $|P^{-1}(1)| \in \{0, 2, 4\}$. In what follows we give a construction of f_P for an *arbitrary* binary predicate $P : [r]^2 \to \{0,1\}$ with $r \ge 2$ from the statement of the theorem.

Although the bipartite double cover is commonly defined as a directed graph, in this proof we will consider the *undirected* bipartite double cover $\gamma(G)$ of G.[5] We also define an auxiliary graph $G^P = (V^P, E^P)$, where

$$V^P = \{v_0, v_0', \dots, v_{r-1}, v_{r-1}'\},$$

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

[5]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.

Let ℓ be the number of connected components of $\overline{G^P}$, the bipartite complement of G^P. By definition, $\ell \le |V^P| = 2r$.

The desired function f_P satisfying (2.1) corresponds to a map $c : V^P \to [\ell]$ on the vertices of G^P with the following property:

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

We call such maps *colourings*. Indeed, the colouring c induces, for $\mathcal{A}$, an assignment $A^\gamma : V^\gamma \to [\ell]$ of the vertices of $\gamma(G)$ which satisfies

$$A^\gamma(u) = c(v_{A(u)}) \quad \text{and} \quad A^\gamma(u') = c(v'_{A(u)})$$

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 $f_P(\mathcal{A}) = (U_0, \dots, U_{\ell-1})$. Now for any $u, v \in V$ and for any assignment $A : V \to [r]$, we have

$$\begin{aligned} P(A(u), A(v)) = 1 &\iff \{v_{A(u)}, v'_{A(v)}\} \in E^P \\ &\iff c(v_{A(u)}) \neq c(v'_{A(v)}) \\ &\iff A^\gamma(u) \neq A^\gamma(v') \\ &\iff \ell\text{-}\mathsf{Cut}(A^\gamma(u), A^\gamma(v')) = 1. \end{aligned}$$

Moreover, by the definition of the bipartite double cover, we have $w(u, v) = w^\gamma(u, v')$ for all $u, v \in V$, implying that

$$\begin{aligned} \mathrm{Val}_{G,P}(\mathcal{A}) = \mathrm{Val}_{G,P}(A_0, \dots, A_{r-1}) &= \sum_{(u,v)\in E} w(u, v) P(A(u), A(v)) \\ &= \sum_{(u,v')\in E^\gamma} w^\gamma(u, v') \ell\text{-}\mathsf{Cut}(A^\gamma(u), A^\gamma(v')) = \mathrm{Val}_{\gamma(G),\ell\text{-}\mathsf{Cut}}(A^\gamma) \\ &= \mathrm{Val}_{\gamma(G),\ell\text{-}\mathsf{Cut}}(U_0, \dots, U_{\ell-1}) = \mathrm{Val}_{\gamma(G),\ell\text{-}\mathsf{Cut}}(f_P(\mathcal{A})) \end{aligned}$$

as required.

While a colouring does not exist for an arbitrary bipartite graph, we now argue that a colouring does exist if the auxiliary graph G^P arises from a predicate P from the statement of the theorem. Since for any $B, C \in \binom{[r]}{2}$ we have $|P|_{B\times C}{}^{-1}(1)| \neq 1$, G^P satisfies the assumptions of Proposition 2.7. Therefore, the ℓ separate connected components which form its bipartite complement $\overline{G^P}$ are complete bipartite graphs. We can assign one of the ℓ colours to each connected component to get a colouring for the graph G^P. We now show that this colouring satisfies $(*)$. (See Figure 2 for an example of G^P, $\overline{G^P}$, and the colouring with $\ell = 3$ satisfying $(*)$ for a particular predicate P on a four-element domain.)

Indeed, if $\{v_i, v'_j\} \in E^P$ then $\{v_i, v'_j\}$ is not an edge in $\overline{G^P}$. Hence v_i and v'_j are in different connected components of $\overline{G^P}$ and thus v_i and v'_j are assigned different colours. Similarly, if $\{v_i, v'_j\} \notin E^P$ then $\{v_i, v'_j\}$ is an edge in $\overline{G^P}$. Hence v_i and v'_j are in the same connected component of $\overline{G^P}$ and thus are assigned the same colour. □

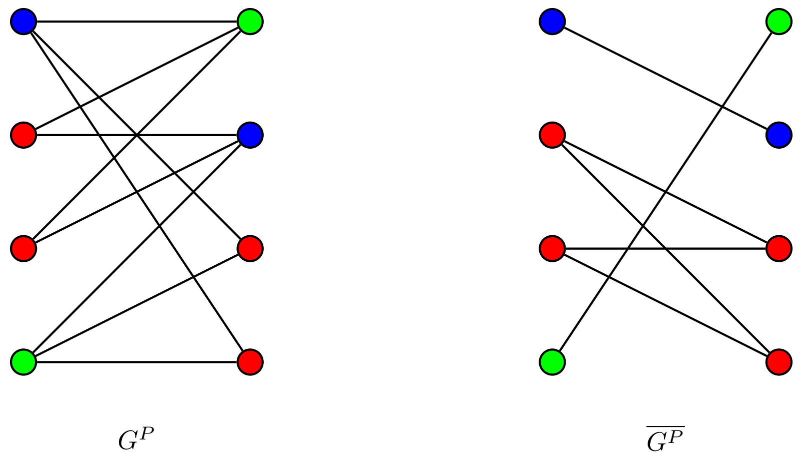

FIG. 2. *An example of* G^P *and* $\overline{G^P}$ *from the proof of Theorem 2.8. The (vertex) colouring indicates the bicliques of* $\overline{G^P}$.

3. Conclusion. For simplicity, we have only presented our main result on binary CSPs over a single domain. However, it is not difficult to extend our result to the so-called *multisorted* binary CSPs, in which different variables come with possibly different domains. We discuss this in the appendix.

We have classified *binary* CSPs (on finite domains) but much more work seems required for a full classification of non-binary CSPs. We have made some initial steps. For any $k \geq 3$, the k-ary Boolean "not-all-equal" predicate $k\text{-}\mathsf{NAE} : \{0,1\}^k \to \{0,1\}$ is defined by $k\text{-}\mathsf{NAE}^{-1}(0) = \{(0,\dots,0),(1,\dots,1)\}$. Kogan and Krauthgamer showed that the k-NAE predicates, which correspond to hypergraph cuts, are sparsifiable [10, Theorem 3.1]. By extending bipartite double covers for graphs in a natural way to k-partite k-fold covers (in Section A.3) we obtain sparsifiability for the class of k-ary predicates that can be rewritten in terms of k-NAE. On the other hand, we identify (in Section A.4) a whole class of predicates that are not sparsifiable, namely those k-ary predicates that contain a singleton ℓ-cube for some $\ell \leq k$. However, most predicates do not fall in either of these two categories; that is, predicates that cannot be proved sparsifiable via k-partite k-fold covers but also cannot be proved non-sparsifiable via the current techniques. An example of such predicates are the "parity" predicates (cf. Section A.5 of the appendix).

Acknowledgements. The authors thank all reviewers of the this paper and the extended abstract [6].

REFERENCES

[1] K. J. Ahn and S. Guha, *Graph sparsification in the semi-streaming model*, in Proceedings of the 36th International Colloquium on Automata, Languages and Programming (ICALP'09), Part II, vol. 5556 of Lecture Notes in Computer Science, Springer, 2009, pp. 328–338, https://doi.org/10.1007/978-3-642-02930-1_27.

[2] A. Andoni, J. Chen, R. Krauthgamer, B. Qin, D. P. Woodruff, and Q. Zhang, *On Sketching Quadratic Forms*, in Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science (ITCS'16), 2016, pp. 311–319, https://doi.org/10.1145/2840728.2840753.

[3] J. D. Batson, D. A. Spielman, and N. Srivastava, *Twice-Ramanujan Sparsifiers*, SIAM

Journal on Computing, 41 (2012), pp. 1704–1721, https://doi.org/10.1137/090772873.

[4] A. A. Benczúr and D. 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 (STOC'96), 1996, pp. 47–55, https://doi.org/10.1145/237814.237827.

[5] R. A. Brualdi, F. Harary, and Z. Miller, *Bigraphs versus digraphs via matrices*, Journal of Graph Theory, 4 (1980), pp. 51–73, https://doi.org/10.1002/jgt.3190040107.

[6] S. Butti and S. Živný, *Sparsification of binary CSPs*, in Proceedings of the 36th International Symposium on Theoretical Aspects of Computer Science (STACS'19), vol. 126 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2019, pp. 17:1–17:8, https://doi.org/10.4230/LIPIcs.STACS.2019.17.

[7] H. Chen, B. M. P. Jansen, and A. Pieterse, *Best-case and Worst-case Sparsifiability of Boolean CSPs*, in Proceedings of the 13th International Symposium on Parameterized and Exact Computation (IPEC'18), vol. 115 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, pp. 15:1–15:13, https://doi.org/10.4230/LIPIcs.IPEC.2018.15.

[8] A. Filtser and R. Krauthgamer, *Sparsification of two-variable valued constraint satisfaction problems*, SIAM Journal on Discrete Mathematics, 31 (2017), pp. 1263–1276, https://doi.org/10.1137/15M1046186.

[9] W. S. Fung, R. Hariharan, N. J. A. Harvey, and D. Panigrahi, *A general framework for graph sparsification*, in Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC'11), ACM, 2011, pp. 71–80, https://doi.org/10.1145/1993636.1993647.

[10] D. Kogan and R. Krauthgamer, *Sketching Cuts in Graphs and Hypergraphs*, in Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science (ITCS'15), 2015, pp. 367–376, https://doi.org/10.1145/2688073.2688093.

[11] V. Kolmogorov, A. A. Krokhin, and M. Rolínek, *The Complexity of General-Valued CSPs*, SIAM Journal on Computing, 46 (2017), pp. 1087–1110, https://doi.org/10.1137/16M1091836.

[12] I. Newman and Y. Rabinovich, *On multiplicative lambda-approximations and some geometric applications*, SIAM Journal on Computing, 42 (2013), pp. 855–883, https://doi.org/10.1137/100801809.

[13] T. Soma and Y. Yoshida, *Spectral Sparsification of Hypergraphs*, in Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'19), SIAM, 2019, pp. 2570–2581, https://doi.org/10.1137/1.9781611975482.159.

[14] D. A. Spielman and N. Srivastava, *Graph sparsification by effective resistances*, SIAM Journal on Computing, 40 (2011), pp. 1913–1926, https://doi.org/10.1137/080734029.

[15] D. A. Spielman and S. Teng, *Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems*, in Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC'04), ACM, 2004, pp. 81–90, https://doi.org/10.1145/1007352.1007372.

[16] D. A. Spielman and S. Teng, *Spectral Sparsification of Graphs*, SIAM Journal on Computing, 40 (2011), pp. 981–1025, https://doi.org/10.1137/08074489X.

Appendix A. Extensions.

A.1. Constraint Satisfaction Problems. There are several natural and well-studied extensions of the binary CSP framework: (i) *non-binary* CSPs, in which constraints are of arity larger than two; (ii) *multisorted* CSPs, in which different variables have possibly different domains; and (iii) CSPs with different types of constraint predicates, leading to *constraint languages*.

Definition A.1. *An instance of the* constraint satisfaction problem *(CSP) is a quadruple* $I = (V, \mathcal{D}, \Pi, w)$ *where* $V = \{v_1, \ldots, v_n\}$ *is a set of variables,* $\mathcal{D} = \{D(v_1), \ldots, D(v_n)\}$ *is a set of domains, one for each variable,* Π *is a set of constraints, and* $w : \Pi \to \mathbb{R}_+$ *are positive weights for the constraints. Each constraint* $\pi \in \Pi$ *is a pair* $(\mathbf{v}, P)$*, where* $\boldsymbol{v} = (v_{i_1}, \ldots, v_{i_k}) \in V^k$ *is an ordered k-tuple of distinct variables from* V *and* $P : D(v_{i_1}) \times \ldots \times D(v_{i_k}) \to \{0, 1\}$ *is a k-ary predicate on the Cartesian product of the corresponding domains.*

The elements from $\cup_{D\in\mathcal{D}} D$ are called *labels*.

For a fixed predicate P, we denote by $\mathrm{CSP}(P)$ the class of CSP instances in which all constraints use the predicate P.

We say that an assignment $A : V \to \cup_{D\in\mathcal{D}} D$ is *valid* if each variable $v \in V$ is assigned a label that belongs to the intersection of the domains of all the constraint predicates whose scope contains v. For a vector $\mathbf{v} \in V^k$ and an assignment $A : V \to \cup_{D\in\mathcal{D}} D$, we denote by $A(\mathbf{v})$ the entry-wise application of A to $\mathbf{v}$. Given a predicate $P : D(v_1) \times \ldots \times D(v_k) \to \{0, 1\}$, we say that a constraint $\pi = (\mathbf{v}, P)$ is *satisfied* by an assignment A if $P(A(\mathbf{v})) = 1$.

The *value* of an instance $I = (V, \mathcal{D}, \Pi, w)$ under assignment $A : V \to \cup_{D\in\mathcal{D}} D$ is given by the total weight of the constraints satisfied by A:

$$\mathrm{Val}_I(A) = \sum_{\pi=(\mathbf{v},P)\in\Pi} w(\pi)P(A(\mathbf{v})). \tag{A.1}$$

For $0 < \varepsilon < 1$, an ε-*sparsifier* of $I = (V, \mathcal{D}, \Pi, w)$ is a re-weighted subinstance $I_\varepsilon = (V, \mathcal{D}, \Pi_\varepsilon \subseteq \Pi, w_\varepsilon)$ of I such that for all valid assignments A of the variables in V,

$$\mathrm{Val}_{I_\varepsilon}(A) \in (1 \pm \varepsilon)\,\mathrm{Val}_I(A). \tag{A.2}$$

Given an instance $I = (V, \mathcal{D}, \Pi, w) \in \mathrm{CSP}(P)$ for a k-ary P, we will call the *corresponding hypergraph* of I the weighted directed k-uniform hypergraph $H^I = (V, E, w)$, where $E = \{\mathbf{v} : (\mathbf{v}, P) \in \Pi\}$ and $w(\mathbf{v}) = w(\mathbf{v}, P)$. Conversely, given a weighted directed k-uniform hypergraph $H = (V, E, w)$ and a predicate $P : D^k \to \{0, 1\}$, the *corresponding* CSP(P) *instance* is $I^{H,P} = (V, \mathcal{D}, \Pi, w)$, where $\mathcal{D} = \{D\}$, $\Pi = \{(e, P) : e \in E\}$, and $w(e, P) = w(e)$. Hence, we can equivalently talk about instances of CSP(P) or hypergraphs. Thus, an ε-P-*sparsifier* of a hypergraph $H = (V, E, w)$ is a partial subhypergraph[6] $H_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ whose corresponding CSP(P) instance $I^{H_\varepsilon,P}$ is an ε-sparsifier of the corresponding CSP(P) instance $I^{H,P}$ of H.

A.2. Multisorted Binary Predicates. The following result is a multisorted extension of Theorem 2.2.

THEOREM A.2. *Let $P : D \times E \to \{0, 1\}$ be a binary predicate, where D and E are finite sets with $|D|, |E| \geq 2$. Let $0 < \varepsilon < 1$.*

1. *If there exist $B \in \binom{D}{2}$ and $C \in \binom{E}{2}$ such that $P|_{B\times C}$ is a singleton then there exists an instance I of CSP(P) such that every ε-sparsifier of I has $\Omega(n^2)$ constraints.*
2. *Otherwise, for every instance I of CSP(P) there exists an ε-sparsifier of I with $O(\varepsilon^{-2}n)$ constraints.*

An inspection of the proof of Theorem 2.3 reveals that the proof establishes Case (1) of Theorem A.2. The proof of Theorem A.2 (2) is essentially identical to the proof of Theorem 2.8. The main difference is that, using the notation from the proof of Theorem 2.8, the bipartite double cover $\gamma(G) = (V^\gamma, E^\gamma, w^\gamma)$ of G may contain vertices of degree zero. Let $Z = \{v^\gamma : \deg_{\gamma(G)}(v) = 0\}$ be all such vertices. Let $\tau(G) = (V^\tau, E^\tau, w^\tau)$ be the subgraph of $\gamma(G)$ induced by $V^\gamma \setminus Z$. Then, for any valid assignment $A : V^\gamma \to D \cup E$ we have

$$\mathrm{Val}_{\tau(G),P}(A') = \mathrm{Val}_{\gamma(G),P}(A), \tag{A.3}$$

[6] A partial subhypergraph is obtained by removing hyperedges while keeping the vertex set unchanged.

where A' is the restriction of A to V^τ. Working with $\tau(G)$ instead of $\gamma(G)$, the rest of the proof proceeds identically to the proof of Theorem 2.8, except for applying Proposition 2.7 to bipartite graphs whose left part is D and the right part is E. This last step is fine since the proof of Proposition 2.7 is not affected if the given bipartite graph has parts of different sizes.

Remark A.3. *For a fixed* set Γ *of predicates, we denote by* $\mathrm{CSP}(\Gamma)$ *the class of CSP instances in which all constraints use predicates from* Γ. Γ *is often called a constraint language. Filtser and Krauthgamer considered sparsifiability of binary Boolean CSPs of the from* $\mathrm{CSP}(\Gamma)$*, i.e., CSPs with multiple binary Boolean predicates [8, Section 5]. Under the assumption that no two constraints act on the same list of variables, any instance* I *of* $\mathrm{CSP}(\Gamma)$ *can be partitioned into disjoint CSP instances according to the predicates in the constraints. By finding a sparsifier for each of these instances, the union of the sparsifiers yields a sparsifier for* I*. Thus our main sparsifiability result (Case (2) of Theorem 2.2 and its multisorted generalisation, Case (2) of Theorem A.2) trivially extends to* $\mathrm{CSP}(\Gamma)$ *for any* Γ *that consists of predicates that do not contain singleton subpredicates.*

A.3. Hypergraph Covers. We generalise the notion of the bipartite double cover for graphs from [5] in a natural way to that of a k-partite k-fold cover for hypergraphs, as this will be useful in the proof of Theorem A.11. The case of $k = 2$ in the following definition corresponds to the bipartite double cover.

Definition A.4. *For a weighted directed* k*-uniform hypergraph* $H = (V, E, w)$*, the* k-partite k-fold cover *of* H *is the weighted directed* k*-uniform hypergraph* $\gamma(H) = (V^\gamma, E^\gamma, w^\gamma)$*, where*

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

Given an assignment $A : V \to [r]$, we let $\mathcal{A} = (A_0, \ldots, A_{r-1})$ be the induced r-partition of V, where $A_j = A^{-1}(j)$. For a predicate $P : [r]^k \to \{0, 1\}$ and an instance $I = (V, [r], \Pi, w) \in \mathrm{CSP}(P)$, we define $\mathrm{Val}_I(\mathcal{A}) = \mathrm{Val}_I(A)$. Moreover, for a weighted directed k-uniform hypergraph H and a k-ary predicate P, we define $\mathrm{Val}_{H,P}(\mathcal{A}) = \mathrm{Val}_{I^{H,P}}(\mathcal{A})$. We denote the set of all r-partitions of V by $Part_r(V)$.

For any r-partition $\mathcal{A} = (A_0, \ldots, A_{r-1})$ of the vertices of V, let $A_i^{(j)} = \{v^{(j)} : v \in A_i\}$. Thus $\mathcal{A}^\gamma = (A_0^{(0)}, \ldots, A_0^{(k-1)}, \ldots, A_{r-1}^{(0)}, \ldots, A_{r-1}^{(k-1)})$ is a kr-partition of the vertices of V^γ.

We use an argument from the proof Theorem 2.1 (2) from [8] and apply it to non-binary, non-Boolean predicates.

Proposition A.5. *Let* $P : [r]^k \to \{0, 1\}$ *and* $P' : [r']^k \to \{0, 1\}$ *be* k*-ary predicates. Suppose that there is a function* $f_P : Part_r(V) \to Part_{r'}(V^\gamma)$ *such that for any weighted directed* k*-uniform hypergraph* H *on* V *and for any* r*-partition* $\mathcal{A} \in Part_r(V)$ *it holds that*

$$\mathrm{Val}_{H,P}(\mathcal{A}) = \mathrm{Val}_{\gamma(H),P'}(f_P(\mathcal{A})),$$

where $\gamma(H) = (V^\gamma, E^\gamma, w^\gamma)$ *is the* k*-partite* k*-fold cover of* H*. If there is an* ε*-*P'*-sparsifier of* $\gamma(H)$ *of size* $g(n)$ *then there is an* ε*-*P*-sparsifier of* H *size* $g(n)$*.*

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 $\gamma(H)$. Define a partial subhypergraph $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of H by $E_\varepsilon = \{(v_1, \ldots, v_k) : (v_1^{(0)}, \ldots, v_k^{(k-1)}) \in E_\varepsilon^\gamma\}$ and by $w_\varepsilon((v_1, \ldots, v_k)) =$

$w^\gamma_\varepsilon(v_1^{(0)}, \ldots, v_k^{(k-1)})$. Note that $\gamma(H_\varepsilon) = \gamma(H)_\varepsilon$ and $E_\varepsilon \subseteq E$.

Then, we have

$$\begin{aligned}\mathrm{Val}_{H_\varepsilon,P}(\mathcal{A}) &= \mathrm{Val}_{\gamma(H_\varepsilon),P'}(f_P(\mathcal{A})) \\ &= \mathrm{Val}_{\gamma(H)_\varepsilon,P'}(f_P(\mathcal{A})) \in (1\pm\varepsilon)\,\mathrm{Val}_{\gamma(H),P'}(f_P(\mathcal{A})) = (1\pm\varepsilon)\,\mathrm{Val}_{H,P}(\mathcal{A}).\end{aligned}$$

Hence H_ε is also an ε-P-sparsifier of H of size $g(n)$. ☐

A.4. Non-Sparsifiability and Singleton Predicates. We identify two simple sufficient conditions for a predicate *not* to be sparsifiable, namely the *singleton ℓ-cube* (Proposition A.7) and the *unused label* (Proposition A.9). We then use these conditions to show that singleton predicates are not sparsifiable.

The idea of a singleton ℓ-cube is essentially an ℓ-ary singleton subpredicate with Boolean domain.

Definition A.6. *A k-ary predicate $P : D^k \to \{0,1\}$* contains a singleton ℓ-cube *for some $2 \le \ell \le k$ if there exist subdomains $\{D_j = \{d_0^j, d_1^j\}\}_{j=1}^{\ell} \in \binom{D}{2}$, indices $\{n_j\}_{j=1}^{\ell} \in \{0,1\}$, and a permutation σ on $\{1,2,\ldots,k\}$ such that there exist $x_{\ell+1},\ldots,x_k \in D$ which satisfy*

$$P(\sigma(d^1_{n_1}, \ldots, d^\ell_{n_\ell}, x_{\ell+1}, \ldots, x_k)) = 1$$

and for all $y_{\ell+1},\ldots,y_k \in D$, for all $i_j \in \{0,1\}$,

$$P(\sigma(d^1_{i_1}, \ldots, d^\ell_{i_\ell}, y_{\ell+1}, \ldots, y_k)) = 1 \quad\Longrightarrow\quad i_j = n_j \text{ for all } j = 1,\ldots,\ell.$$

Proposition A.7 (**Singleton ℓ-cube**).
Let $P : D^k \to \{0,1\}$ be a k-ary predicate which contains a singleton ℓ-cube. Then, there exists a weighted directed k-uniform hypergraph $H = (V,E,w)$ with $|V| = n$ such that for every $0 < \varepsilon < 1$ and for every partial subhypergraph $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of H which satisfies (A.2), we have $|E_\varepsilon| = \Omega(n^\ell)$.

Proof. Let $\{D_j = \{d_0^j, d_1^j\}\}_{j=1}^{\ell}$ and $\{n_j\}_{j=1}^{\ell}$ be as in Definition A.6. Without loss of generality, assume that σ is the identity permutation.

Let $H = (V,E,w)$ be a weighted directed k-uniform hypergraph on $n = kq$ vertices with $V = V_1 \sqcup \ldots \sqcup V_k$, $|V_i| = q$ for $i = 1,\ldots,k$, and $E = \{(u_1,\ldots,u_k) : u_i \in V_i\}$. Notice that $|E| = q^k$. Take an arbitrary hyperedge $f = (v_1,\ldots,v_k) \in E$. By construction, $v_j \in V_j$ for all j. Furthermore, pick some $x_{\ell+1},\ldots,x_k$ such that $P(d^1_{n_1},\ldots,d^\ell_{n_\ell}, x_{\ell+1},\ldots,x_k) = 1$.

Define the assignment

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

Notice that $P(A^f(u_1,\ldots,u_k)) = 1 \iff u_j = v_j$ for all $j \le \ell$. Therefore, at least one of the $q^{k-\ell}$ edges whose first ℓ variables are $v_1,\ldots,v_\ell$ must belong to E_ε for (A.2) to be satisfied. We repeat the same argument for all q^ℓ combinations of vertices $(v_1,\ldots,v_\ell) \in V_1 \times \ldots \times V_\ell$. Thus $|E_\varepsilon| \ge q^\ell = \Theta(n^\ell)$, as k is a constant, and $|E_\varepsilon| = \Omega(n^\ell)$ as required. ☐

Example A.8. *Let $P : \{0,1\}^3 \to \{0,1\}$ be the ternary Boolean predicate defined by $P^{-1}(1) = \{(0,0,0),(0,0,1)\}$. Note that P is not a singleton. P contains a singleton 2-cube (e.g., on the first two coordinates) and thus it is not sparsifiable by Proposition A.7.*

Our second sufficient condition for not being sparsifiable is the idea of an unused label. An unused label is an element of the domain which never appears in the tuples that belong to the predicate's support set.

Proposition A.9 (**Unused label**).
Let $P : D^k \to \{0,1\}$ be a k-ary predicate with $P^{-1}(1) \neq \emptyset$. Suppose that there exists $z \in D$ such that, for all $x_1, \ldots, x_{k-1} \in D$ and for all permutations σ on $\{1, 2, \ldots, k\}$, $P(\sigma(x_1, \ldots, x_{k-1}, z)) = 0$. Then, for every weighted directed k-uniform hypergraph $H = (V, E, w)$, for every $0 < \varepsilon < 1$, and for every partial subhypergraph $H_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ of H which satisfies (A.2), we have $|E_\varepsilon| = \Omega(|E|)$.

Proof. Let $H = (V, H, w)$, $0 < \varepsilon < 1$, and $H_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ be as in the statement. We will show that $|E_\varepsilon| = \Omega(|E|)$.

Consider some tuple $(a_1, \ldots, a_k) \in P^{-1}(1)$. By assumption, z does not appear in any tuple which belongs to $P^{-1}(1)$ and therefore we must have $a_j \neq z$ for all j. Pick a hyperedge $f = (u_1, \ldots, u_k) \in E$ and let $U = \{u_1, \ldots, u_k\}$. Define the assignment $A : V \to D$ by $A(u_j) = a_j$ for $j = 1, \ldots, k$ and by $A(v) = z$ for all $v \in V \setminus U$. Notice that the a_j may not necessarily be all distinct.

For $d \in D$, let δ_d be the number of times d appears in $(a_1, \ldots, a_k)$. Further define

$$M = \prod_{d \in D, \delta_d \neq 0} \delta_d!$$

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

$$\mathrm{Val}_{H,P}(A) = \sum_{e \in E} w(e) P(A(e)) = \sum_{i=1}^{M_E} w(e_i) > 0.$$

Since H_ε is an ε-P-sparsifier of H, at least one of $e_1, \ldots, e_{M_E}$ must be in E_ε, since otherwise we would have

$$\mathrm{Val}_{H_\varepsilon,P}(A) = \sum_{e \in E_\varepsilon} w_\varepsilon(e) P(A(e)) = 0 \notin (1 \pm \varepsilon)\,\mathrm{Val}_{H,P}(A).$$

Noticing that this argument holds for any hyperedge $f \in E$ and that $M \leq k!$, we have

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

Therefore, we have $|E_\varepsilon| \geq |E|/k!$ and thus $|E_\varepsilon| = \Omega(|E|)$. ∎

Notice that, if a k-ary predicate P has an unused label, then P contains a singleton k-cube. For singleton predicates with a very specific support (consisting of the same label), Proposition A.9 is directly applicable.

Proposition A.10. *Let $P : D^k \to \{0,1\}$ be a k-ary singleton predicate with $|D| \geq 2$ such that $P^{-1}(1) = \{(a, a, \ldots, a)\}$ for some $a \in D$. Then, for every weighted directed k-uniform hypergraph $H = (V, E, w)$, for every $0 < \varepsilon < 1$, and for every partial subhypergraph $H_\varepsilon = (V, E_\varepsilon \subseteq E, w_\varepsilon)$ of H which satisfies (A.2), we have $|E_\varepsilon| = \Omega(|E|)$.*

Proof. By assumption, the support set of P is not empty 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 σ on $\{1, 2, \ldots, k\}$, we have $P(\sigma(x_1, \ldots, x_{k-1}, z)) = 0$. Thus, z is an unused label and, by Proposition A.9, the claim follows. ∎

Since every singleton k-ary predicate P contains a k-cube, by Proposition A.7, there exists a weighted directed k-uniform hypergraph $H = (V, E, w)$ with $|V| = n$ such that for every $0 < \varepsilon < 1$ and for every partial subhypergraph $H_\varepsilon = (V, E_\varepsilon, w_\varepsilon)$ of H which satisfies (A.2), we have $|E_\varepsilon| = \Omega(n^k)$. In particular, the k-ary singleton predicate $\mathsf{nOR} : D^k \to \{0,1\}$ has this property, where nOr is defined by $\mathsf{nOr}^{-1}(1) = \{(0,0,\ldots,0)\}$. We use the concept of k-partite k-fold covers from Appendix A.3 (and in particular Proposition A.5) to show that if any instance of CSP(P) has a (small) sparsifier then so does CSP(nOR), which establishes that singleton predicates cannot be sparsifiable.

Theorem A.11. *Let $P : D^k \to \{0,1\}$ be a k-ary singleton predicate. If there is an ε-P-sparsifier of size $g(n)$ then there is an ε-nOr-sparsifier of size $O(g(n))$.*

Proof. Without loss of generality, $D = [r]$ and $P^{-1}(1) = \{(a_1,\ldots,a_k)\}$. Let $H = (V, E, w)$ be a weighted directed k-uniform hypergraph.

We will show the existence of a function $f_P : Part_r(V) \to Part_r(V^\gamma)$ such that for any $\mathcal{A} \in Part_r(V)$ it holds that

$$\mathrm{Val}_{H,\mathsf{nOr}}(\mathcal{A}) = \mathrm{Val}_{\gamma(H),P}(f_P(\mathcal{A})).$$

The statement of the theorem then follows by Proposition A.5.

Let $A_j^P = \bigcup_{i=1}^k A^{(i-1)}_{(j-a_i) \pmod r}$. Define

$$f_P(A_0,\ldots,A_{r-1}) = (A_0^P,\ldots,A_{r-1}^P).$$

Moreover, define an assignment $A : V \to D$ by $A(v) = j \iff v \in A_j$. By definition,

$$\mathrm{Val}_{H,\mathsf{nOr}}(A) = \mathrm{Val}_{H,\mathsf{nOr}}(A_0,\ldots,A_{r-1}).$$

Define the assignment $A^\gamma : V^\gamma \to D$ by $A^\gamma(v^{(i)}) = j \iff v^{(i)} \in A_j^P$. We have

$$\mathrm{Val}_{\gamma(H),P}(A^\gamma) = \mathrm{Val}_{\gamma(H),P}(A_0^P,\ldots,A_{r-1}^P) = \mathrm{Val}_{\gamma(H),P}(f_P(A_0,\ldots,A_{r-1})).$$

For a hyperedge $e = (v_1,\ldots,v_k) \in E$, define $\gamma(e) = e^\gamma = (v_1^{(0)},\ldots,v_k^{(k-1)})$. We have

$$\begin{aligned}\mathsf{nOr}(A(e)) = 1 &\iff A(v_1) = \ldots = A(v_k) = 0\\ &\iff (v_1,\ldots,v_k) \in A_0 \times \ldots \times A_0\\ &\iff \gamma((v_1,\ldots,v_k)) = (v_1^{(0)},\ldots,v_k^{(k-1)}) \in A_0^{(0)} \times \ldots \times A_0^{(k-1)}.\end{aligned}$$

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

$$\begin{aligned}A_0^{(i-1)} \subseteq A_j^P &\iff A_0^{(i-1)} = A^{(i-1)}_{j-a_i \pmod r} && \text{for some } j \in [r]\\ &\iff j = a_i \pmod r \iff j = a_i && (\text{since } 0 \le a_i, j < r).\end{aligned}$$

Therefore, assuming that $\mathsf{nOr}(A(e)) = 1$,

$$\begin{aligned}A_0^{(i-1)} \subseteq A_j^P &\implies (v_1^{(0)},\ldots,v_k^{(k-1)}) \in A_{a_1}^P \times \ldots \times A_{a_k}^P\\ &\iff A^\gamma(v_1^{(0)},\ldots,v_k^{(k-1)}) = (a_1,\ldots,a_k)\\ &\iff P(A^\gamma(v_1^{(0)},\ldots,v_k^{(k-1)})) = P(A^\gamma(\gamma(v_1,\ldots,v_k)))\\ &\qquad\qquad = P(A^\gamma(\gamma(e))) = 1.\end{aligned}$$

Therefore, for any $e \in E$,

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

which implies

$$\begin{aligned}\mathrm{Val}_{H,\mathsf{nOr}}(A_0, \dots, A_{r-1}) &= \mathrm{Val}_{H,\mathsf{nOr}}(A) = \sum_{e \in E} w(e)\mathsf{nOr}(A(e)) = \sum_{e \in E} w(e)P(A^\gamma(\gamma(e))) \\ &= \sum_{e \in E} w^\gamma(\gamma(e))P(A^\gamma(\gamma(e))) = \sum_{e^\gamma \in E^\gamma} w^\gamma(e^\gamma)P(A^\gamma(e^\gamma)) \\ &= \mathrm{Val}_{\gamma(H),P}(A^\gamma) = \mathrm{Val}_{\gamma(H),P}(f_P(A_0, \dots, A_{r-1})).\end{aligned}$$

□

A.5. Parity Predicates. The k-ary parity predicate $\mathsf{Par} : [r]^k \to \{0,1\}$ is defined by

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

It is trivial to show that the parity predicates do not contain an unused label. We will show that, for any $k \geq 3$, 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.

PROPOSITION A.12. *For all $2 \leq \ell \leq k$, where $k \geq 3$, the k-ary parity predicate Par does not contain a singleton ℓ-cube.*

Proof. By Definition A.6, the containment of a singleton ℓ-cube for some $\ell \geq 3$ implies the containment of a singleton 2-cube. Thus it suffices to show that Par does not contain any singleton 2-cube.

Suppose by contradiction that there exist subdomains $\{D_j = \{d_0^j, d_1^j\}\}_{j \in \{1,2\}} \in \binom{[r]}{2}$, indices $n_1, n_2 \in \{0,1\}$, and a permutation σ on $\{1, 2, \dots, k\}$ such that there exist $x_3, \dots, x_k \in [r]$ which satisfy

$$\mathsf{Par}(\sigma(d_{n_1}^1, d_{n_2}^2, x_3, \dots, x_k)) = 1$$

and for all $y_3, \dots, y_k \in [r]$, for all $i_j \in \{0,1\}$,

$$\mathsf{Par}(\sigma(d_{i_1}^1, d_{i_2}^2, y_3, \dots, y_k)) = 1 \quad \implies \quad i_j = n_j \text{ for all } j = 1, 2. \tag{A.4}$$

Case 1: $d_0^1 - d_1^1 = 0 \pmod 2$.

Then,

$$d_{n_1}^1 + d_{n_2}^2 + \sum_{j=3}^{k} x_k = d_{1-n_1}^1 + d_{n_2}^2 + \sum_{j=3}^{k} x_k \pmod 2$$

and hence

$$\mathsf{Par}(\sigma(d_{1-n_1}^1, d_{n_2}^2, x_3, \dots, x_k)) = \mathsf{Par}(\sigma(d_{n_1}^1, d_{n_2}^2, x_3, \dots, x_k)) = 1,$$

contradicting (A.4).

Case 2: $d_0^1 - d_1^1 = 1 \pmod 2$.

Then,

$$d_{n_1}^1 + d_{n_2}^2 + \sum_{j=3}^{k} x_k = d_{1-n_1}^1 + d_{n_2}^2 + (x_3 + 1) + \sum_{j=4}^{k} x_k \pmod 2$$

and hence

$$\mathsf{Par}(\sigma(d^1_{1-n_1}, d^2_{n_2}, x_3 + 1 \pmod 2, x_4, \dots, x_k)) = \mathsf{Par}(\sigma(d^1_{n_1}, d^2_{n_2}, x_3, \dots, x_k)) = 1,$$

again contradicting (A.4). ◻

Proposition A.13. *Let* $\mathit{Par} : [r]^k \to \{0,1\}$ *be the* k*-ary parity predicate, where* $k \geq 3$*. Then, for all weighted directed* k*-uniform hypergraphs* $H = (V, E, w)$ *with* $|V| \geq rk$*, for all* $r' \geq 2$*, and for all functions* $f : Part_r(V) \to Part_{r'}(V^\gamma)$*, there exists a partition of the vertices* $\mathcal{A} \in Part_r(V)$ *such that*

$$\mathrm{Val}_{H,\mathit{Par}}(\mathcal{A}) \neq \mathrm{Val}_{\gamma(H),r'\text{-}\mathit{NAE}}(f(\mathcal{A})),$$

where $\gamma(H)$ *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 $|V| \geq rk$, an integer $r' \geq 2$, and a function $f_{\mathsf{Par}} : Part_r(V) \to Part_{r'}(V^\gamma)$ such that for all partitions $\mathcal{A} \in Part_r(V)$ we have

$$\mathrm{Val}_{H,\mathsf{Par}}(\mathcal{A}) = \mathrm{Val}_{\gamma(H),r'\text{-}\mathsf{NAE}}(f(\mathcal{A})). \tag{A.5}$$

Let $A : V \to [r]$ be any assignment with the induced r-partition $\mathcal{A} = (A_0, \dots, A_{r-1}) \in Part_r(V)$ such that $|A_i| \geq k$ for all $i \in [r]$. Denote $f_{\mathsf{Par}}(\mathcal{A}) = (U_0, \dots, U_{r'-1})$. Define an assignment $A_{f_{\mathsf{Par}}(\mathcal{A})} : V^\gamma \to [r']$ such that, for all $i \in [r']$ and for all $j \in [k]$,

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

First of all, we need to show that the assignment $A_{f_{\mathsf{Par}}(\mathcal{A})}$ is well-defined. Notice that for all $i \in [r]$, for all $j \in [k]$, for all $u^{(j)}, v^{(j)} \in A^{(j)}_i$ 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 contradiction that there exist $i \in [r]$, $j \in [k]$, and $u^{(j)}, v^{(j)} \in A^{(j)}_i$ such that $u^{(j)} \in U_{\ell_u}$ and $v^{(j)} \in U_{\ell_v}$ with $\ell_u \neq \ell_v$. Assume without loss of generality that $j = 0$. Then, for all $v_2, \dots, v_k \in V$ and for $A_i \in \mathcal{A} \in Part_r(V)$, we would have

$$\begin{aligned}\mathsf{Par}(A(u, v_2 \dots, v_k)) &= \mathsf{Par}(A(u), A(v_2), \dots, A(v_k)) = \mathsf{Par}(i, A(v_2), \dots, A(v_k)) \\ &= \mathsf{Par}(A(v), A(v_2), \dots, A(v_k)) = \mathsf{Par}(A(v, v_2, \dots, v_k))\end{aligned}$$

and hence

$$\begin{aligned}r'\text{-}\mathsf{NAE}(A_{f_{\mathsf{Par}}(\mathcal{A})}(u^{(0)}, v_2^{(1)}, \dots, v_k^{(k-1)})) &= \mathsf{Par}(A(u, v_2, \dots, v_k)) = \mathsf{Par}(A(v, v_2, \dots, v_k)) \\ &= r'\text{-}\mathsf{NAE}(A_{f_{\mathsf{Par}}(\mathcal{A})}(v^{(0)}, v_2^{(1)}, \dots, v_k^{(k-1)})).\end{aligned}$$

Now pick $v_2, \dots, v_k$ such that $\mathsf{Par}(A(u, v_2, \dots, v_k)) = 0$. Then,

$$\begin{aligned}r'\text{-}\mathsf{NAE}(A_{f_{\mathsf{Par}}(\mathcal{A})}(u^{(0)}, v_2^{(1)}, \dots, v_k^{(k-1)})) &= \mathsf{Par}(A(u, v_2, \dots, v_k)) = 0 \\ &\implies v_2^{(1)}, \dots, v_k^{(k-1)} \in U_{\ell_u}\end{aligned} \tag{A.6}$$

and

$$\begin{aligned}r'\text{-}\mathsf{NAE}(A_{f_{\mathsf{Par}}(\mathcal{A})}(v^{(0)}, v_2^{(1)}, \dots, v_k^{(k-1)})) &= \mathsf{Par}(A(v, v_2, \dots, v_k)) \\ &= \mathsf{Par}(A(u, v_2, \dots, v_k)) = 0 \\ &\implies v_2^{(1)}, \dots, v_k^{(k-1)} \in U_{\ell_v}.\end{aligned} \tag{A.7}$$

Putting (A.6) and (A.7) 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 $u^{(j)}, v^{(j)} \in A_i^{(j)}$ and for all $\ell \in [r']$ we have $\{u^{(j)}, v^{(j)}\} \cap U_\ell \in \{\emptyset, \{u^{(j)}, v^{(j)}\}\}$ and hence $A_{f_{\mathsf{Par}}(\mathcal{A})}$ is well-defined.

Now we want to consider vertices which belong to sets A_i of different parity. Without loss of generality, pick k vertices $u_1, \ldots, u_k \in A_0$ and 3 vertices $v_1, v_2, v_3 \in A_1$. Then we have

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

and

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

Then, by (A.5) we must have

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

and

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

respectively.

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

$$X = A_1^{(0)} \sqcup A_0^{(1)} \sqcup A_0^{(2)} \sqcup \ldots \sqcup A_0^{(k-1)}$$

and

$$Y = A_1^{(0)} \sqcup A_1^{(1)} \sqcup A_1^{(2)} \sqcup A_0^{(3)} \sqcup A_0^{(4)} \sqcup \ldots \sqcup A_0^{(k-1)}$$

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,

$$A_1^{(0)} \subseteq (X \cap Y) \subseteq U_x \cap U_y$$

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

$$A_1^{(0)} \sqcup A_1^{(1)} \sqcup A_0^{(2)} \sqcup A_0^{(3)} \sqcup \ldots \sqcup A_0^{(k-1)} \subseteq X \cup Y \subseteq U_x$$

and hence

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

implying, by (A.5), that

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

a contradiction since

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

Therefore, such a map f_{Par} cannot exist. $\square$