New Algorithms and Hardness Results for Robust Satisfiability of (Promise) CSPs

Joshua Brakensiek* Lorenzo Ciardo† Venkatesan Guruswami‡ Aaron Potechin§
Stanislav Živný¶

Abstract. In this paper, we continue the study of robust satisfiability of promise CSPs (PCSPs), initiated in (Brakensiek, Guruswami, Sandeep, STOC 2023), and obtain the following results:

For the PCSP 1-IN-3-SAT vs NAE-SAT with negations, we prove that it is hard, under the Unique Games conjecture (UGC), to satisfy $1 - \Omega(1/\log(1/\epsilon))$ constraints in a $(1-\epsilon)$-satisfiable instance. This shows that the exponential loss incurred by the BGS algorithm for the case of Alternating-Threshold polymorphisms is necessary, in contrast to the polynomial loss achievable for Majority polymorphisms.

For any Boolean PCSP that admits Majority polymorphisms, we give an algorithm satisfying $1 - O(\sqrt{\epsilon})$ fraction of the weaker constraints when promised the existence of an assignment satisfying $1 - \epsilon$ fraction of the stronger constraints. This significantly generalizes the Charikar–Makarychev–Makarychev algorithm for 2-SAT, and matches the optimal trade-off possible under the UGC. The algorithm also extends, with the loss of an extra $\log(1/\epsilon)$ factor, to PCSPs on larger domains with a certain structural condition, which is implied by, e.g., a family of Plurality polymorphisms.

We prove that assuming the UGC, robust satisfiability is preserved under the addition of equality constraints. As a consequence, we can extend the rich algebraic techniques for decision/search PCSPs to robust PCSPs. The methods involve the development of a correlated and robust version of the general SDP rounding algorithm for CSPs due to (Brown-Cohen, Raghavendra, ICALP 2016), which might be of independent interest.

1 Introduction

The CSP dichotomy theorem has precisely identified which problems in the rich class of constraint satisfaction problems (CSPs) are polynomial-time solvable, with the rest being NP-complete [19, 51]. Strikingly, this landmark result shows that simple gadget reductions from 3-SAT are the only obstructions to the existence of an efficient algorithm for a CSP, and conversely the existence of a single "non-trivial polymorphism" suffices for a polynomial time satisfiability algorithm. Informally, a polymorphism is an operator that combines multiple satisfying assignments to the predicates defining the CSP into another satisfying assignment.

On the algorithmic side, in essence there are only two broad approaches: local consistency (also captured by a few levels of the Sherali–Adams hierarchy of linear programs) [6] and generalizations of Gaussian elimination [20, 38]. However, the overall algorithm in the CSP dichotomy theorem is highly non-trivial due to the intricate ways in which these two basic algorithmic paradigms might have to be combined to solve an arbitrary tractable CSP.

The picture is considerably simpler and clearer (but is still non-trivial) when focusing on polynomial time *robust satisfiability algorithms* for tractable CSPs, a concept first considered in the beautiful work of Zwick [52]. In addition to finding perfectly satisfying assignments when they exist, such algorithms are also robust in the sense that, when given as input almost-satisfiable instances (namely those that admit an assignment failing to satisfy

*UC Berkeley (josh.brakensiek@berkeley.edu). Research supported in part by a Simons Investigator award, NSF CCF-2211972, and NSF DMS-2503280.

†TU Graz (lorenzo.ciardo@tugraz.at)

‡Simons Institute and UC Berkeley (venkatg@berkeley.edu). Research supported in part by a Simons Investigator award and NSF CCF-2211972.

§University of Chicago (potechin@uchicago.edu)

¶University of Oxford (standa.zivny@cs.ox.ac.uk). Research supported by UKRI EP/X024431/1.

only an ϵ fraction of the constraints), they find an assignment satisfying all but $g(\epsilon)$ fraction of the constraints, for some loss $g(\epsilon)$ that vanishes as $\epsilon \to 0$.

For ease of terminology, let us refer to CSPs that admit such efficient robust satisfiability algorithms as *robust CSPs*. We further call them $(\epsilon, g(\epsilon))$-robust CSPs to indicate the loss incurred by the robust algorithm. Every robust CSP is also tractable—this follows from two works of Barto and Kozik [6,7]. However, the converse is not true and there are tractable CSPs that lack robust satisfiability algorithms. The quintessential such CSPs are defined by linear relations over an Abelian group. Satisfiability of such CSPs can be efficiently ascertained via Gaussian elimination, but by the celebrated inapproximability results of Håstad [36], they are not robust.[1] On the other hand, tractable CSPs that are solved by local consistency (called bounded-width CSPs in the literature) are in fact robust. For Boolean CSPs, this was effectively implicit in Zwick's original work that pioneered robust satisfiability [52]. For CSPs over any fixed finite domains, this was shown by Barto and Kozik [7]. An earlier breakthrough result of Barto and Kozik [6] had shown that any CSP that cannot express linear equations (in a certain formal sense) is solved by local consistency algorithms. Thus, we have the pleasing picture that CSPs solved by local consistency—one of the two basic algorithmic strategies—are precisely those that are robust (this statement was explicitly conjectured in [35]). Further, for all such CSPs, there is a robust satisfiability algorithm based on semidefinite programming. We therefore have a single unified approach for robust satisfiability compared to the highly complex situation for exact satisfiability of CSPs. In a way, the robust CSP dichotomy offers a crisper and more comprehensible complexity criterion.

Given this backdrop concerning robust CSPs, and motivated by the quest for understanding robustness of SDP-based algorithms more broadly, Brakensiek, Guruswami, and Sandeep [13] initiated the study of robust algorithms for *promise CSPs*, which we introduce next. Promise CSPs (also PCSPs for short) are a vast generalization of CSPs that have received significant attention in recent years [2–4, 8, 11, 12, 15, 16, 24–29, 32, 34, 37, 41, 42, 44–47]. A promise CSP is defined by a fixed collection of relation pairs (P_i, Q_i) over some domain pair (D, E), with $P_i \subseteq Q_i$.[2] Given a CSP instance based on the relations, the goal is to find an assignment satisfying the (weak) constraints given by the Q_i relations if promised that an assignment satisfying the (strong) constraints given by the P_i relations exists (but is not known). A classic example of a promise CSP is the *approximate graph coloring* problem [33]: given a k-colorable graph, find an ℓ-coloring of it, where $3 \le k \le \ell$. In our terminology, this is just the PCSP with a single pair of relations (P, Q), where P is the disequality relation on a k-element set, and Q is the disequality relation on an ℓ-element set.

Another, more recent, example of a PCSP is the $(2 + \epsilon)$-SAT problem, introduced and studied by Austrin, Guruswami, and Håstad [1] (who also coined the expression promise CSP): given an instance of k-SAT with the promise that an assignment exists that satisfies at least g literals in each clause, where $1 \le g \le k$, find a standard satisfying assignment (satisfying at least 1 literal in each clause). In this case, the P relations encode Boolean clause assignments with Hamming weight at least g, where the Q relations encode Boolean clause assignments with Hamming weight at least 1. Another example is the 1-IN-3-SAT vs NAE-SAT problem, identified in the influential paper of Brakensiek and Guruswami [10] that initiated a systematic study of Boolean PCSPs. Here one is given a satisfiable instance of 1-IN-3-SAT and the goal is to find an assignment that satisfies 1 or 2 variables per clause. Astoundingly, this problem is solvable in polynomial time (via an algorithm not previously considered in the context of CSPs) [10]. Further, such an algorithm cannot be obtained via a reduction to (finite domain) CSPs [5]!

The study of PCSPs calls for significant new algorithmic and hardness techniques. Studying such techniques in the broader context of PCSPs has also led to new results for (standard, non-promise) CSPs, e.g., a single algorithm blending together linear programming with linear Diophantine equations [14] that solves all tractable Boolean CSPs. Despite a lot of attention and recent progress on PCSPs, the complexity landscape is vast and mostly not understood. In fact, the complexity of Boolean PCSPs is itself a major challenge, in contrast to the CSP world where Schaefer proved a dichotomy for Boolean CSPs already in the 1970s [50]. Following a

[1]In fact, even for almost-satisfiable instances, it is NP-hard to beat the approximation ratio achieved by the trivial algorithm that simply outputs a random assignment.

[2]More generally, there must be a map $h : D \to E$ that is a homomorphism from each P_i to Q_i.

classification of Boolean symmetric PCSPs allowing negations from [10], Ficak, Kozik, Olšák, and Stankiewicz obtained a classification of Boolean PCSPs with symmetric relations [31]. Moreover, Brakensiek, Guruswami, and Sandeep obtained a (conditional) classification of monotone Boolean PCSPs [12].[3]

Returning to robust satisfiability, given the crisp picture of robust CSPs—namely, either the natural SDP gives an efficient robust algorithm or none exists—the study of robust PCSPs is a natural goal, as proposed by Brakensiek, Guruswami, and Sandeep (BGS) [13]. A robust satisfiability algorithm for a PCSP defined by relation pairs (P_i, Q_i) means the following: given an instance such that $(1-\epsilon)$ fraction of the constraints are promised to be satisfiable according to the stronger relations P_i, there is an algorithm to weakly satisfy (according to the relations Q_i) $(1-g(\epsilon))$ fraction of the constraints, where $g(\epsilon) \to 0$ as $\epsilon \to 0$. As with CSPs, in this case we say that the PCSP is robust or $(\epsilon, g(\epsilon))$-robust.

BGS focused on Boolean PCSPs where the known satisfiability algorithms[4] can be attributed to the existence of polymorphisms. A polymorphism for a relation pair (P, Q) is a homomorphism from a (categorical) power P^m of P to Q, with $\mathrm{Pol}(P, Q)$ representing the set of all polymorphisms. More precisely, BGS focused on three families of polymorphisms: Majority (MAJ), Alternating Threshold (AT), and Parity [10]. For any odd $L \in \mathbb{N}$, we let $\mathrm{MAJ}_L, \mathrm{AT}_L, \mathrm{PAR}_L : \{-1,1\}^L \to \{-1,1\}$ be defined as

$$\mathrm{MAJ}_L(x_1, \ldots, x_L) := \mathbf{1}\left[\sum_{i=1}^{L} x_i \geq 0\right],$$

$$\mathrm{AT}_L(x_1, \ldots, x_L) := \mathbf{1}\left[\sum_{i=1}^{L} (-1)^{i-1} x_i \geq 0\right],$$

$$\mathrm{PAR}_L(x_1, \ldots, x_L) := \mathbf{1}\left[\sum_{i=1}^{L} x_i \equiv L \mod 4\right].$$

We let $\mathrm{MAJ} := \{\mathrm{MAJ}_L : L \in \mathbb{N} \text{ odd}\}$, $\mathrm{AT} := \{\mathrm{AT}_L : L \in \mathbb{N} \text{ odd}\}$, $\mathrm{PAR} := \{\mathrm{PAR}_L : L \in \mathbb{N} \text{ odd}\}$ be the respective sets of polymorphisms. In two of these cases, Majority and AT, BGS showed that the associated PCSPs are robust via an algorithm based on semidefinite programming [13]. They also showed that a partial converse holds: for PCSPs defined by a single pair (P, Q) of symmetric Boolean relations (plus allowing negations), if the relation pair (P, Q) lacks some odd-arity MAJ and some odd-arity AT as a polymorphism, then the PCSP is not robust (assuming the Unique Games conjecture).

The quantitative aspects of the robust satisfaction algorithms in [13] for the two cases, Majority and AT, however, diverged significantly. For Majority, the BGS algorithm, which is really the same as the Charikar–Makarychev–Makarychev algorithm for 2-SAT but analyzed in greater generality assuming only a Majority polymorphism, guaranteed that at most $\widetilde{O}(\epsilon^{1/3})$ fraction of the constraints are violated. This is weaker than the $(\epsilon, O(\sqrt{\epsilon}))$ robustness guarantee for 2-SAT [23]—which is tight [40, 43] under the Unique Games conjecture [39, 40, 43].[5] A natural question then is whether the BGS loss guarantee for Majority polymorphisms can be improved to $O(\sqrt{\epsilon})$, which would then give the right polymorphic generalization of the CMM robust algorithm for 2-SAT.

The situation for robust algorithms for Boolean PCSPs with AT polymorphisms is worse, as the BGS algorithm only showed $(\epsilon, O(1/\log(1/\epsilon)))$-robustness. A natural question then is whether this exponential loss is necessary, or whether one can achieve polynomial loss also for the AT case similar to the Majority case. We address and resolve both of these questions in this work.

1.1 Our Results Our contributions in this paper fall into three parts. First, we show that the robust algorithm for Alternating Threshold due to Brakensiek–Guruswami–Sandeep [13] has a near-matching hardness result under the UGC. This is based on a novel integrality gap for (1-IN-3-SAT, NAE-SAT). Second, we show

[3]The classification assumes the Rich 2-to-1 conjecture of Braverman, Khot, and Minzer [17].

[4]Recall that robust PCSPs must first of all be tractable.

[5]Historically, showing evidence for the near-optimality of the Goemans–Williamson robust algorithm for Max-Cut, which also achieves $O(\sqrt{\epsilon})$ loss, was the original motivation for the formulation of the Unique Games conjecture in [39].

that any promise template with the Majority polymorphism has a robust algorithm with loss $g(\epsilon) = O(\sqrt{\epsilon})$, which is asymptotically tight, and improves over BGS's analysis of $O(\epsilon^{1/3})$. We further extend this analysis to show that similar algorithms achieve a robustness of $O(\sqrt{\epsilon}\log(1/\epsilon))$ for Plurality and related polymorphisms. Finally, we show that the robustness of PCSPs is (approximately) preserved under a large family of gadget reductions under the UGC. We do this by solving a seemingly elementary but technically complex problem: given a promise template with a robust algorithm, show that adding the equality relation to the template (approximately) preserves the robustness of the problem.

Hardness for Alternating Threshold. A key result of Brakensiek–Guruswami–Sandeep [13] is that for any promise template (P, Q) with $\mathrm{AT} \subseteq \mathrm{Pol}(P, Q)$, $\mathrm{PCSP}(P, Q)$ is robust with $g(\epsilon) = O(\frac{\log\log(1/\epsilon)}{\log(1/\epsilon)})$. Interestingly, for CSPs, similar asymptotics appear with the OR and AND families[6] of polymorphisms, and these are known to be tight [35, 52]. We show that AT exhibits a similar behavior by proving UGC hardness.

Theorem 1.1 (AT hardness, informal). *Assuming UGC,* fiPCSP(*1-in-3-SAT*, *NAE-SAT*) *is not* $(\epsilon, \Omega(1/\log(1/\epsilon)))$*-robust.*

Here fiPCSP refers to PCSPs that allow for variables to be negated and set as constants (see [13]). The use of negations is necessary, as Brakensiek–Guruswami–Sandeep [13] observed that (1-in-3-SAT, NAE-SAT) without negations is robust with polynomial loss. Theorem 1.1 is proved in the full version of this paper. Using Raghavendra's theorem [48], we prove Theorem 1.1 by constructing an explicit integrality gap—the same high-level strategy as used by Guruswami–Zhou [35] for HORN-3-SAT, although the execution and analysis in our setting are significantly more complex. We explain further details in Subsection 1.2.

Improved Analysis for Majority and Beyond. Our next result is a robust algorithm for PCSPs with Majority polymorphisms with an improved loss function.

Theorem 1.2 (MAJ robustness, informal). *For any promise template* (P, Q) *with* $\mathrm{MAJ} \subseteq \mathrm{Pol}(P, Q)$, $\mathrm{PCSP}(P, Q)$ *is* $(\epsilon, O(\epsilon^{1/2}))$*-robust.*

This result is proved in the full version. We note that the algorithm used in Theorem 1.2 is identical to the one used in [13]. The main improvement in the analysis comes from a more refined analysis of multivariate normal distributions. See Subsection 1.2 for further details.

Note that Theorem 1.2 only applies to Boolean PCSPs. A commonly studied non-Boolean variant of the Majority polymorphisms is the *Plurality* polymorphisms. Typical examples of (P)CSPs admitting such polymorphisms are *unique games* [39] as well as the so-called SetSAT problem—a non-Boolean generalization of $(2+\epsilon)$-SAT [1] introduced in [15].

Theorem 1.3 (PLUR robustness, informal). *For any promise template* (P, Q) *with* $\mathrm{PLUR} \subseteq \mathrm{Pol}(P, Q)$, $\mathrm{PCSP}(P, Q)$ *is* $(\epsilon, O(\epsilon^{1/2}\log(1/\epsilon)))$*-robust.*

This result is proved in the full version. We note that Theorem 1.3 is merely a special case of our main result in the full version, which applies to any *separable* PCSP. We describe this broader family more precisely in Subsection 1.2.

Robust Gadget Reductions (adding EQUALITY). In virtually all classifications of (variants of) CSPs, an essential tool is *gadget reductions* between CSP templates. For example, in the CSP dichotomy, the hardness side of the CSP is done using gadget reductions from 3-SAT [21]. Robust (P)CSPs are no exception, and gadget reductions are frequently used to study the relationship between templates [13, 30]. However, there is a significant distinction between the ordinary CSP dichotomy [19, 51] and the one for robust (P)CSPs [7]: the allowance of *equality constraints*, which we denote by EQ. For exact satisfiability of (P)CSPs, if we specify that some variables are to be equal, we can efficiently compute the connected components of the equality relation and distill the problem down to a smaller number of variables (and without any equality constraints). However, for robust (P)CSPs, equality is a rather subtle concept. If only $1-\epsilon$ constraints are satisfied in the optimal assignment,

[6]For any $L \in \mathbb{N}$, $\mathrm{OR}_L(x_1, \ldots, x_L) = 1$ if $x_i = 1$ for some $i \in [L]$ and $\mathrm{AND}_L(x_1, \ldots, x_L) = 1$ if $x_i = 1$ for all $i \in [L]$.

we do not know whether each equality constraint should be trusted or ignored. That said, it still seems quite reasonable to assume that adding equality constraints should only mildly change the robustness of the resulting PCSP.

QUESTION 1.4 (Barto–Kozik [7]). *Let* Γ *be a CSP template, i.e., a set of relations. Assume that* $\mathrm{CSP}(\Gamma)$ *is* $(\epsilon, f(\epsilon))$*-robust. Is* $\mathrm{CSP}(\Gamma \cup \{\mathrm{EQ}\})$ *then* $(\epsilon, O(f(\epsilon)))$*-robust?*

Despite Barto–Kozik [7] giving a complete classification of all robust CSPs,[7] they do not answer Question 1.4 except in the very weak sense that $\mathrm{CSP}(\Gamma \cup \{\mathrm{EQ}\})$ is $(\epsilon, O(\log\log(1/\epsilon)/\log(1/\epsilon)))$-robust, independent of f. In this paper, we establish Question 1.4 is *nearly* true for both CSPs and PCSPs:

THEOREM 1.5 (Robustness of Equality, informal). *Assume UGC. For any promise template* Γ*, if* $\mathrm{PCSP}(\Gamma)$ *is* $(\epsilon, f(\epsilon))$*-robust, then* $\mathrm{PCSP}(\Gamma \cup \{\mathrm{EQ}\})$ *is* $(\epsilon, O(f(\epsilon^{1/6})))$*-robust.*

As an immediate corollary, we can now use the most general gadget reductions available for studying (P)CSPs to study robust (P)CSPs, modulo a polynomial loss in robustness.

THEOREM 1.6 (Gadget Reductions, informal). *Assume UGC. Let* Γ *and* Γ' *be promise templates such that there is a gadget reduction*[8] *from* $\mathrm{PCSP}(\Gamma')$ *to* $\mathrm{PCSP}(\Gamma)$*. If* $\mathrm{PCSP}(\Gamma)$ *is* $(\epsilon, f(\epsilon))$*-robust, then* $\mathrm{PCSP}(\Gamma')$ *is* $(\epsilon, O(f(\epsilon^{1/6})))$*-robust.*

These results are proved in the full version. In the technical overview (Subsection 1.2), we describe the techniques we use to establish our results, including an adaptation of the algorithm of Brown-Cohen and Raghavendra [18] for approximate (P)CSPs.

1.2 Technical Overview

Hardness for Alternating Threshold. Our integrality gap instance for 1-IN-3-SAT vs NAE-SAT has a similar high-level idea to the following LP integrality gap instance for Horn-SAT [35]:

1. We have variables $\{x_{j1}, x_{j2} : j \in [k]\}$ where $k = \lceil 1/\log_2(\epsilon) \rceil + 1$.
2. We have the unary constraints x_{11}, x_{12}, and $\neg x_{k1}$ where 1 is True and -1 is False.
3. For all $j \in [k-1]$, we have the constraints $\neg x_{j1} \vee \neg x_{j2} \vee x_{(j+1)1}$ and $\neg x_{j1} \vee \neg x_{j2} \vee x_{(j+1)2}$.

Clearly this instance is unsatisfiable, as we insist that x_{11}, x_{12} are True, and then x_{k1}, x_{k2} should also be True due to the chain of implication constraints, but we insist that x_{k1} is False. As there are only $O(k) = O(\log(1/\epsilon))$ constraints, the integral value of this instance is at most $1 - \Omega(1/\log(1/\epsilon))$.

On the other hand, the following LP solution gives value at least $1 - \epsilon$ to all of the constraints:

1. For each $j \in [k]$, we give x_{j1} and x_{j2} bias $1 - 2^{j+1-k}$.
2. For the constraint x_{11}, we can set $x_{11} = 1$ with probability $1 - 2^{1-k}$ and -1 with probability 2^{1-k} and we will have that $\mathbb{E}[x_{11}] = 1 - 2^{1-k} - 2^{1-k} = 1 - 2^{2-k}$. Thus, the LP gives a value of $1 - 2^{1-k} \geq 1 - \epsilon$ for this constraint. By symmetry, the LP also gives a value of $1 - 2^{1-k} \geq 1 - \epsilon$ for the constraint x_{12}.
3. For each $j \in [k-1]$, for the constraint $\neg x_{j1} \vee \neg x_{j2} \vee x_{(j+1)1}$, we can take the distribution where
   a. With probability $1 - 2^{j+1-k}$, we set $x_{j1} = x_{j2} = x_{(j+1)1} = 1$.
   b. With probability 2^{j-k}, we set $x_{j1} = 1$, $x_{j2} = -1$, and $x_{(j+1)1} = 1$.
   c. With probability 2^{j-k}, we set $x_{j1} = -1$, $x_{j2} = 1$, and $x_{(j+1)1} = 1$.

[7] We observe that Barto–Kozik were able to establish this classification by using gadget reductions that do *not* allow equality. Using Theorem 1.6, one could (in theory) simplify parts of Barto–Kozik's proof, although the proof of Theorem 1.6 is much more complicated than the workarounds needed by Barto–Kozik.

[8] More precisely, there is a *minion homomorphism* from $\mathrm{Pol}(\Gamma)$ to $\mathrm{Pol}(\Gamma')$.

With this distribution, $\mathbb{E}[x_{j1}] = \mathbb{E}[x_{j2}] = (1 - 2^{j-k}) - 2^{j-k} = 1 - 2^{j+1-k}$ and $\mathbb{E}[x_{(j+1)1}] = (1 - 2^{j+1-k}) - 2^{j+1-k} = 1 - 2^{j+2-k}$. Thus, the LP gives this constraint a value of 1. By symmetry, the LP also gives a value of 1 to the constraint $\neg x_{j1} \vee \neg x_{j2} \vee x_{(j+1)2}$.

4. The bias for x_{k1} is $1 - 2 = -1$ so x_{k1} is always set to -1 which satisfies the constraint $\neg x_{k1}$.

One way to think about this integrality gap instance is as follows. In order to avoid violating a constraint, the following must hold.

1. Variables with bias $1 - 2^{j+1-k}$ must be rounded to 1.
2. For all $j \in [k-1]$, if the variables with bias $1 - 2^{j+1-k}$ are rounded to 1 then the variables with bias $1 - 2^{j+2-k}$ are rounded to 1.
3. Variables with bias -1 must be rounded to -1.

Since we can obtain a contradiction in $O(\log(1/\epsilon))$ steps, at least $\Omega(1/\log(1/\epsilon))$ of the constraints must be violated.

We will use a similar idea for our integrality gap instance. We will construct our instance so that while the SDP value is at least $1 - \epsilon$, if we want to avoid violating a significant number of constraints,

1. Almost all of the variables with bias $1 - 2^{-k}$ must be set to 1 and almost all of the variables with bias $2^{-k} - 1$ must be set to -1.
2. For all $j \in [k]$, if almost all of the variables with bias $1 - 2^{-j}$ are set to 1 then almost all of the variables with bias $2^{1-j} - 1$ are set to -1. Similarly, for all $j \in [k]$, if almost all of the variables with bias $2^{-j} - 1$ are set to -1 then almost all of the variables with bias $1 - 2^{1-j}$ are set to 1.

We then observe that these conditions imply that almost all of the variables with bias 0 are set to 1 and almost all of the variables with bias 0 are set to -1, which is impossible. Since we can obtain a contradiction in $O(\log(1/\epsilon))$ steps, at least $\Omega(1/\log(1/\epsilon))$ of the constraints must be violated.

In order to have 1-IN-3-SAT constraints, it turns out that we need general vectors of the form $\{x\mathbf{v}_0 + \sqrt{1-x^2}\mathbf{w}\}$ where $\mathbf{w}$ is orthogonal to $\mathbf{v}_0$. A natural choice for this is to use all $\mathbf{w} \in S^{d-1}$ for some large d which depends on ϵ. While this gives an integrality gap instance, it has infinite size and is not that easy to analyze since it involves functions on the sphere rather than functions with multivariate Gaussian inputs. Thus, we modify this integrality gap instance as follows:

1. Instead of using vectors $\mathbf{w} \in S^{d-1}$, we will use vectors $\mathbf{w} \sim \mathcal{N}(0, 1/d)^d$.
2. For all but a negligible portion of the constraints, our vectors $\mathbf{w} \sim \mathcal{N}(0, 1/d)^d$ are very close to unit vectors so we can discard the negligible number of constraints where the vectors are badly behaved.
3. We discretize our instance by splitting our space into regions and mapping all vectors in each region to a representative vector in that region.

Through a careful analysis, we show that for this modified instance, $\Omega(1/\log(1/\epsilon))$ fraction of the constraints must be violated and even after these modifications, the SDP value for our instance is at least $1 - \epsilon$.

Improved Analysis for Majority. We now switch to designing robust algorithms for families of PCSPs. To begin, we discuss the algorithm used by Brakensiek, Guruswami, Sandeep [13] for promise templates with Majority polymorphisms. This algorithm was inspired by the algorithm used by Charikar, Makarychev, Makarychev [23] for robust MAX 2-SAT.

For convenience, we relabel the Boolean domain as $\{-1, +1\}$. Fix a template (P, Q) with MAJ $\subseteq$ Pol(P, Q). Consider an instance of PCSP(P, Q) on variables $x_1, \ldots, x_n$ and clauses $C_1, \ldots, C_m$. The algorithm begins by solving the Basic SDP for this instance by finding unit vectors $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^{n+1}$ (with $\mathbf{v}_0$ representing the "truth" vector) such that the average value of the vector assignment to the clauses is $1 - \epsilon$, where $\epsilon > 0$ is the specified robustness parameter. Next, we sample a random multivariate normal vector $\mathbf{r} \in \mathcal{N}(0^{n+1}, I_{n+1})$. Then, for all $i \in [n]$, we round x_i to $+1$ if $\langle \mathbf{v}_i, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle \geq 0$ and -1 otherwise. Here, we improve $\epsilon^{2/3}$ to $\sqrt{\epsilon}$ via a new analysis.

We briefly explain the key ideas in Brakensiek, Guruswami, Sandeep [13] in the analysis of this algorithm. Using a reduction in their paper, we may also assume without loss of generality that $Q = \{-1, +1\}^k \setminus \{(-1)^k\}$. In other words, assume that the majority vote of any list of assignments to P is never all -1's. For simplicity, fix a clause C_i on variables $x_1, \ldots, x_k$ such that the SDP vectors $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_k$ give a value of $1 - \epsilon$. It may be the case that for all $i \in [n]$, $\langle \mathbf{v}_i, \mathbf{v}_0 \rangle \approx -\Theta(\epsilon)$, so purely rounding $\langle \mathbf{v}_i, \mathbf{v}_0 \rangle$ will fail to satisfy any of the clauses. A key observation by BGS is that as long as the vectors have completeness $1 - \epsilon$, there exists a probability distribution $(w_1, \ldots, w_k)$ such that $\sum_{i=1}^k w_i \langle \mathbf{v}_i, \mathbf{v}_0 \rangle \geq -\epsilon$. Let $\mathbf{u} := \sum_{i=1}^k w_i \mathbf{v}_i$. By concentration, we can assume with probability $1 - \epsilon^{O(1)}$ that $|\langle \mathbf{u}, \mathbf{r} \rangle| = O(\log(1/\epsilon))$. As a key observation, note that since $\langle \mathbf{u}, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle = \sum_{i=1}^k w_i \langle \mathbf{v}_i, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle$, if all k variables round to -1, then $\langle \mathbf{u}, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle$ is negative. However, $\langle \mathbf{u}, \mathbf{v}_0 \rangle \geq -\epsilon$ and the standard deviation of $\langle \mathbf{u}, \mathbf{r} \cdot \epsilon^{2/3} \rangle$ is at most $\epsilon^{1/3}$. Thus, if $\langle \mathbf{u}, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle$ is negative, it is barely negative. Hence, this tightly constrains the value of each $\langle \mathbf{v}_i, \mathbf{v}_0 + \mathbf{r} \cdot \epsilon^{2/3} \rangle$, which is unlikely due to anti-concentration of the normal distribution.

With a more careful analysis of these rounding probabilities, we can change the rounding threshold to $\langle \mathbf{v}_i, \mathbf{v}_0 + \mathbf{r} \cdot \sqrt{\epsilon} \rangle$ and get a $1 - O(\sqrt{\epsilon})$ success probability (soundness). The key idea is, instead of directly comparing each $\langle \mathbf{u}, \mathbf{r} \rangle$ to the individual distributions $\langle \mathbf{v}_i, \mathbf{r} \rangle$, we use a more careful decomposition of the vectors. In particular, define $\mathbf{v}_i^{=}$ to be the component of $\mathbf{v}_i$ parallel to $\mathbf{u}$ and let $\mathbf{v}_i^{\perp}$ be the component of $\mathbf{v}_i$ perpendicular to $\mathbf{u}$. Since $\mathbf{v}_i^{=}$ and $\mathbf{u}$ are related by a scalar, $\langle \mathbf{v}_i^{=}, \mathbf{r} \rangle$ and $\langle \mathbf{u}, \mathbf{r} \rangle$ are also related by a scalar. However, $\langle \mathbf{v}_i^{\perp}, \mathbf{r} \rangle$ is independent of $\langle \mathbf{u}, \mathbf{r} \rangle$. Using this observation, we can split our argument into three high-level cases.

First, if $\|\mathbf{u}\|_2^2 = \Omega(\epsilon \log(1/\epsilon))$, then $\langle \mathbf{u}, \mathbf{v}_0 \rangle$ will dominate $\langle \mathbf{u}, \mathbf{r}\sqrt{\epsilon} \rangle$, so the chances that $\langle \mathbf{u}, \mathbf{v}_0 + \mathbf{r}\sqrt{\epsilon} \rangle \leq 0$ are quite small.

Second, if a perpendicular component is large, that is $\|w_i \mathbf{v}_i^{\perp}\|_2 = \Omega(1)$ for some $i \in [k]$, then even if we condition on $\langle \mathbf{u}, \mathbf{r} \rangle$, the value of $\langle w_i \mathbf{v}_i, \mathbf{v}_0 + \sqrt{\epsilon}\mathbf{r} \rangle$ still has considerable variance. In particular, most likely $\langle w_i \mathbf{v}_i, \mathbf{v}_0 + \sqrt{\epsilon}\mathbf{r} \rangle$ will either be (1) too positive, in which case i is rounded correctly, or (2) too negative, in which case the average of $\sum_{j \neq i} w_j \langle \mathbf{v}_j, \mathbf{v}_0 + \sqrt{\epsilon}\mathbf{r} \rangle$ is positive, so some other j is rounded correctly.

These two cases themselves are enough to get a $\sqrt{\epsilon} \log(1/\epsilon)$ loss. To shave the log, in the third and final case, we finely partition the space of potential "bad" outcomes and show that these in total contribute at most $O(\sqrt{\epsilon})$ loss to the rounding. This is the most technical part of the argument.

New algorithms for Plurality and Separable Families. We extend these rounding techniques for Majority to non-Boolean domains and more general rounding functions. To do this, we abstract out the essential feature of the analysis of Majority: the existence of a hyperplane separation between the strong form of the constraint P and SDP-configurations whose rounding lies outside of the weak form Q. We define templates (P, Q) with a generalization of this property that we call *separable* families. The precise description is given in the full version but, at a high level, here is the idea. In the Boolean Majority case, our analysis relies upon a linear function separating $\langle \mathbf{v}_i, \mathbf{v}_0 \rangle$ from an absent tuple of Q, which in turn can be expressed via the inner product with a weight vector w. In non-Boolean domains D, we have a separate vector $\mathbf{v}_{i,d}$ for each variable x_i and each domain element $d \in D$. Hence, we encode both P and the SDP-configurations whose rounding lies outside of Q as two convex bodies living in the matrix space $\mathbb{R}^{k \times D}$, where k is the arity of the constraint. The first is the convex hull of the *one-hot* encodings $\Pi_{\mathbf{p}}$ of tuples in $\mathbf{p} \in P$ (where the (i, d)-th entry equals 1 if $p_i = d$ and 0 otherwise). The second is the preimage under the given rounding function ρ of "bad" tuples—those lying outside of Q. If the polymorphisms of (P, Q) are rich enough, one can show that these two convex sets are disjoint—in which case, they must admit a hyperplane separation. The latter is naturally expressed via a linear functional over $R^{k \times D}$ determined by the Frobenius inner product times a suitable weight matrix W—the non-Boolean analogue of the weight vector w. If, for some rounding function ρ, the template (P, Q) admits such a separation, we call it *ρ-separable.*[9] Provided that ρ satisfies an extra *conservativity* condition (roughly speaking, the winning element of a given distribution must have weight bounded away from zero), we are able to use the hyperplane separation property to show that (P, Q) is robustly solved by SDP with loss $O(\sqrt{\epsilon} \log(1/\epsilon))$, by generalising part of the

[9]Such predicates have some resemblance to the "regional polymorphisms" defined by Brakensiek–Guruswami [9].

analysis performed in the Boolean Majority case. Note that the loss we achieve in this setting is slightly worse than the one for Majority, by a $\log(1/\epsilon)$ factor (although, even for Majority, better than the loss achieved in [13]). This is due to the fact that the trick of splitting the SDP vectors into parallel and orthogonal components does not carry over in the non-Boolean domain.

A notable example of a separable family is the Unique Games problem. Unique Games has been known to have a robust algorithm for a long time due to the algorithm of Charikar, Makarychev, and Makarychev [22] (although this algorithm is rather different from the one used by the same authors for 2-SAT). The fact that their 2-SAT algorithm can be extended to Unique Games may be of independent interest. The underlying polymorphism driving this is *Plurality*, which takes the most commonly occurring element in a list of domain elements, even if its frequency is much less than 1/2. More interestingly, our polymorphism-based result captures *all* PCSPs admitting Plurality—in particular, the family of so-called SetSAT PCSPs identified in [15] as a natural non-Boolean generalization of $(2+\epsilon)$-SAT [1]. Unlike Unique Games, these problems were previously not known to be robustly solvable with any loss.

Robust Gadget Reductions (Adding Equality). We now discuss the proofs of Theorem 1.5 and Theorem 1.6. Assuming Theorem 1.5, Theorem 1.6 is straightforward to establish by combining existing gadget reductions for robust (P)CSPs [7, 13, 30] with state-of-the-art gadget reductions for (P)CSPs [5]. The precise details are worked out in the full version.

As such, we focus on sketching the proof of Theorem 1.5. We crucially build off the algorithm of Brown-Cohen and Raghavendra [18] ("BCR algorithm") for solving approximate MAX (P)CSPs.[10] For the purposes of this high-level overview, we assume that the our promise template is a single pair of Boolean relations (P, Q) with $P \subseteq Q \subseteq \{-1, 1\}^k$.

We first describe the essential features of the BCR algorithm. For an instance of PCSP(P, Q) on variable set $x_1, \ldots, x_n$ and clauses $C_1, \ldots, C_m$, we can think of an SDP solution as a collection of unit vectors $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^{n+1}$. After finding an SDP solution with near-optimal value, the BCR algorithm proceeds by sampling two random objects: (i) a list of random vectors $\mathbf{r}_1, \ldots, \mathbf{r}_D \in \mathbb{R}^{n+1}$ sampled from a multivariate normal distribution, and (ii) a rounding function[11] $H : \mathbb{R}^D \to [-1, 1]$. For each $i \in [n]$, we then define a fractional assignment $z_i \in [-1, 1]$ via

$$z_i := H(\mathbf{v}_0 \cdot \mathbf{v}_1 + \mathbf{v}_i^{\perp} \cdot \mathbf{r}_1, \ldots, \mathbf{v}_i \cdot \mathbf{v}_0 + \mathbf{v}_i^{\perp} \cdot \mathbf{r}_D),$$

where we set $\mathbf{v}_i^{\perp} = \mathbf{v}_i - (\mathbf{v}_0 \cdot \mathbf{v}_i)\mathbf{v}_0$. We then get an integral solution to the PCSP by independently rounding x_i to $+1$ with probability $\frac{1+z_i}{2}$ and -1 otherwise.

Let $\mathcal{R}$ be the probability distribution over the choices of $(\mathbf{r}_1, \ldots, \mathbf{r}_D)$ and H. We can thus think of the BCR rounding scheme as a map $\mathrm{BCR} : \mathbb{R}^{n+1} \times \mathcal{R} \to [-1, 1]$ for which *global shared randomness* $R \in \mathcal{R}$ is picked at the start of the algorithm, and then for each $i \in [n]$, we set $z_i := \mathrm{BCR}(\mathbf{v}_i, R)$. The assumed robustness of the algorithm then translates into the following guarantee.

Key Property. For *every* SDP solution $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_n$ with value at least $1 - \epsilon$, the rounded assignment $\mathrm{BCR}(\mathbf{v}_1, R), \ldots, \mathrm{BCR}(\mathbf{v}_n, R)$ will satisfy $1 - f(\epsilon)$ of the constraints of our instance in expectation over the choice of $R \in \mathcal{R}$.

As is, the existing scheme may not be robust for PCSP(EQ) for the following reason: given two vectors $\mathbf{v}_i$ and $\mathbf{v}_j$ that are δ apart in Euclidean distance, the corresponding $\mathrm{BCR}(\mathbf{v}_i, R)$ and $\mathrm{BCR}(\mathbf{v}_j, R)$ might be very different for a typical $R \sim \mathcal{R}$.[12] In order to make the BCR rounding scheme also robust for equality, we exploit the Key Property in the following way. Consider a map $M_\delta : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ (not necessarily linear) such that, for

[10] Technically, Raghavendra's theorem and the result of Brown-Cohen–Raghavendra are only stated for CSPs, but as noted by Brakensiek, Guruswami, Sandeep [13], their arguments extend to PCSPs with minimal modification.

[11] The choice of rounding function is based on the existence of certain *approximate polymorphisms*, see the full version for a precise definition.

[12] In general H is very slightly smooth, so one can directly combine the BCR algorithm with a correlated rounding trick to get a robust algorithm for equality. The "catch" is that soundness of the robust algorithm depends on the arity of the approximate polymorphisms considered by BCR. However, no effective bound is given by BCR on the size of these approximate polymorphisms, resulting in guarantees much worse than Theorem 1.5.

every unit vector $\mathbf{v}$, the distance between $\mathbf{v}$ and $M_\delta(\mathbf{v})$ is at most δ. We can then consider the following scheme: $S_\delta(\mathbf{v}, R) := \mathrm{BCR}(M_\delta(\mathbf{v}), R)$. A key observation is that S_δ is still a robust rounding scheme with slightly worse parameters.

To see why, using S_δ to round a solution $\mathbf{v}_1, \ldots, \mathbf{v}_n$ is effectively the same as using BCR to round $M_\delta(\mathbf{v}_1), \ldots, M_\delta(\mathbf{v}_n)$. Since each $\mathbf{v}_i$ is close in Euclidean distance to $M_\delta(\mathbf{v}_i)$, for δ sufficiently small, the SDP value of the solution[13] $M_\delta(\mathbf{v}_1), \ldots, M_\delta(\mathbf{v}_n)$ is still approximately $1-\epsilon$, and as such we still satisfy roughly $1 - f(\epsilon)$ constraints on average.

More generally, $M_\delta : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ does not need to be deterministic, rather it can be any randomized map such that the input vector can never be more than δ far from the output vector. We call such a randomized map a δ-*spread* if for each unit vector $\mathbf{v}$, if the probability distribution $M_\delta(\mathbf{v})$ is supported within the ball $B(\mathbf{v}, \delta)$ and the probability degrades smoothly with distance.

For a given δ-spread (a distribution of M_δ's), the corresponding rounding scheme $S_\delta(\mathbf{v}, R) := \mathbb{E}_{M_\delta}[\mathrm{BCR}(M_\delta(\mathbf{v}), R)]$ is called a δ-*smoothing* of BCR. By the aforementioned logic, any δ-smoothing of BCR is still approximately $(\epsilon, f(\epsilon))$-robust. In particular, we now have a large collection of rounding schemes that are all robust for $\mathrm{PCSP}(P, Q)$.

Our next step is to pick one of these δ-smoothings that is also robust for equality. For each unit vector $\mathbf{v} \in \mathbb{R}^{n+1}$, we look at the following L^2 norm:

$$\|S_\delta(\mathbf{v})\|_2 := \sqrt{\mathbb{E}_{R\sim\mathcal{R}}[S_\delta(\mathbf{v}, R)^2]}.$$

Recall that the range of S_δ is $[-1, 1]$, so $S_\delta(\mathbf{v}, R)^2$ roughly measures the certainty the rounding scheme has for this value. Rather unintuitively, we select the δ-smoothing of BCR such that $\|S_\delta(\mathbf{v})\|_2$ is *minimized* for all unit vectors $\mathbf{v} \in \mathbb{R}^{n+1}$. We call this scheme REQ_δ, as we shall soon see it is Robust for EQuality. Roughly speaking, REQ_δ is the δ-smoothing of BCR with maximal entropy.

The key lemma we seek to show is that there are (small) constants $c_1, c_2 \geq 1$ such that if two vectors $\mathbf{v}$ and $\mathbf{w}$ are within distance δ^{c_1} of each other, then $\sqrt{\mathbb{E}_{R\sim\mathcal{R}}[(\mathrm{REQ}_\delta(\mathbf{v}, R) - \mathrm{REQ}_\delta(\mathbf{w}, R)^2]} \leq \delta^{c_2}$. The proof of this lemma uses the fact that the possible δ-smoothings around $\mathbf{v}$ are quite similar to the δ-smoothings around $\mathbf{w}$. If we think of these spaces of δ-smoothings as convex bodies, the L^2 minimizer of one of these convex bodies must then be in close proximity to the L^2 minimizer of the other convex body. This is enough to prove the key lemma.

However, REQ_δ by itself is not a robust rounding scheme for (P, Q) with equality. The reason why is that so far we have only established that if $\mathbf{v}_i$ and $\mathbf{v}_j$ are close, then on average over the choice of $R \in \mathcal{R}$, the outputs $z_i = \mathrm{REQ}_\delta(\mathbf{v}, R)$ and $z_j = \mathrm{REQ}_\delta(\mathbf{w}, R)$ are close. However, remember that the z_i's are rounded into an integral assignment with *independent* coin flips for each $i \in [n]$. In particular, if $z_i = z_j = 0$, then the equality constraint is satisfied only $1/2$ of the time.

To correct this issue, as our final step we modify the independent rounding into *correlated* rounding. In the Boolean setting, this involves picking a random *global* threshold $t \in [-1/2, 1/2]$ and rounding each x_i to 1 if $z_i > t$ and -1 otherwise. Now, if $z_i \approx z_j$, the corresponding equality constraint will almost always be satisfied. The tradeoff is that the robustness $f(\epsilon)$ for $\mathrm{PCSP}(P, Q)$ gets worse by a constant factor. See the full version for precise details.

Altogether, our rounding scheme is REQ_δ for $\delta = \epsilon^{O(1)}$ with correlated rounding. Due to various polynomial losses in the course of the proof, the new scheme is (approximately) $(\epsilon, f(\epsilon^{1/6}))$-robust.

Acknowledgments. The authors are grateful to the American Institute of Mathematics (AIM) SQuaREs program that funded and allowed for this collaboration. The authors also thank Marcin Kozik for many useful discussions.

References

[13]It may be the case that $M_\delta(\mathbf{v}_1), \ldots, M_\delta(\mathbf{v}_n)$ is no longer a valid SDP solution due to violating triangle inequalities. Circumventing this issue is highly technical and involves adapting a smoothing trick due to Raghavendra and Steurer [49]. We ignore this important issue for the purposes of this overview.

[1] P. Austrin, V. Guruswami, and J. Håstad, *(2+ϵ)-Sat is NP-hard*, SIAM J. Comput., 46 (2017), pp. 1554–1573, https://doi.org/10.1137/15M1006507, https://eccc.weizmann.ac.il/report/2013/159/.

[2] S. Avvakumov, M. Filakovský, J. Oprsal, G. Tasinato, and U. Wagner, *Hardness of 4-colouring k-colourable graphs*, in Proc. 57th Annual ACM Symposium on Theory of Computing (STOC'25), ACM, 2025, pp. 72–83, https://doi.org/10.1145/3717823.3718154.

[3] D. Banakh and M. Kozik, *Injective hardness condition for pcsps*, in Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024, P. Sobocinski, U. D. Lago, and J. Esparza, eds., ACM, 2024, pp. 8:1–8:10, https://doi.org/10.1145/3661814.3662072.

[4] L. Barto, D. Battistelli, and K. M. Berg, *Symmetric Promise Constraint Satisfaction Problems: Beyond the Boolean Case*, in Proc. 38th International Symposium on Theoretical Aspects of Computer Science (STACS'21), vol. 187 of LIPIcs, 2021, pp. 10:1–10:16, https://doi.org/10.4230/LIPIcs.STACS.2021.10, https://arxiv.org/abs/2010.04623.

[5] L. Barto, J. Bulín, A. A. Krokhin, and J. Opršal, *Algebraic approach to promise constraint satisfaction*, J. ACM, 68 (2021), pp. 28:1–28:66, https://doi.org/10.1145/3457606, https://arxiv.org/abs/1811.00970.

[6] L. Barto and M. Kozik, *Constraint satisfaction problems solvable by local consistency methods*, J. ACM, 61 (2014), https://doi.org/10.1145/2556646.

[7] L. Barto and M. Kozik, *Robustly solvable constraint satisfaction problems*, SIAM J. Comput., 45 (2016), pp. 1646–1669, https://doi.org/10.1137/130915479, https://arxiv.org/abs/1512.01157.

[8] L. Barto and M. Kozik, *Combinatorial gap theorem and reductions between promise csps*, in Proc. 2022 ACM-SIAM Symposium on Discrete Algorithms (SODA'22), SIAM, 2022, pp. 1204–1220, https://doi.org/10.1137/1.9781611977073.50.

[9] J. Brakensiek and V. Guruswami, *An algorithmic blend of lps and ring equations for promise csps*, in Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2019, pp. 436–455.

[10] J. Brakensiek and V. Guruswami, *Promise Constraint Satisfaction: Algebraic Structure and a Symmetric Boolean Dichotomy*, SIAM J. Comput., 50 (2021), pp. 1663–1700, https://doi.org/10.1137/19M128212X, https://arxiv.org/abs/1704.01937.

[11] J. Brakensiek and V. Guruswami, *The quest for strong inapproximability results with perfect completeness*, ACM Trans. Algorithms, 17 (2021), pp. 27:1–27:35, https://doi.org/10.1145/3459668.

[12] J. Brakensiek, V. Guruswami, and S. Sandeep, *Conditional Dichotomy of Boolean Ordered Promise CSPs*, TheoretiCS, Volume 2 (2023), p. 8967, https://doi.org/10.46298/theoretics.23.2.

[13] J. Brakensiek, V. Guruswami, and S. Sandeep, *SDPs and robust satisfiability of promise CSP*, in Proc. 55th Annual ACM Symposium on Theory of Computing (STOC'23), ACM, 2023, pp. 609–622, https://doi.org/10.1145/3564246.3585180, https://arxiv.org/abs/2211.08373.

[14] J. Brakensiek, V. Guruswami, M. Wrochna, and S. Živný, *The power of the combined basic LP and affine relaxation for promise CSPs*, SIAM J. Comput., 49 (2020), pp. 1232–1248, https://doi.org/10.1137/20M1312745, http://eccc.hpi-web.de/report/2020/004/.

[15] A. Brandts, M. Wrochna, and S. Živný, *The complexity of promise SAT on non-Boolean domains*, ACM Trans. Comput. Theory, 13 (2021), pp. 26:1–26:20, https://doi.org/10.1145/3470867, https://doi.org/10.1145/3470867.

[16] A. Brandts and S. Živný, *Beyond PCSP(1-in-3,NAE)*, Information and Computation, (2022), https://doi.org/10.1016/j.ic.2022.104954, https://arxiv.org/abs/2104.12800.

[17] M. Braverman, S. Khot, and D. Minzer, *On rich 2-to-1 games*, in Proc. 12th Innovations in Theoretical Computer Science Conference (ITCS'21), vol. 185 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, pp. 27:1–27:20, https://doi.org/10.4230/LIPIcs.ITCS.2021.27.

[18] J. Brown-Cohen and P. Raghavendra, *Correlation decay and tractability of CSPs*, in Proc. 43rd International Colloquium on Automata, Languages, and Programming (ICALP'16), vol. 55 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016, pp. 79:1–79:13, https://doi.org/10.4230/LIPICS.ICALP.2016.79.

[19] A. A. Bulatov, *A dichotomy theorem for nonuniform CSPs*, in Proc. 58th Annual IEEE Symposium on Foundations of Computer Science (FOCS'17), USA, 2017, IEEE, pp. 319–330, https://doi.org/10.1109/FOCS.2017.37, https://arxiv.org/abs/1703.03021.

[20] A. A. Bulatov and V. Dalmau, *A simple algorithm for Mal'tsev constraints*, SIAM J. Comput., 36 (2006), pp. 16–27, https://doi.org/10.1137/050628957.

[21] A. A. Bulatov, P. Jeavons, and A. A. Krokhin, *Classifying the complexity of constraints using finite algebras*, SIAM J. Comput., 34 (2005), pp. 720–742, https://doi.org/10.1137/S0097539700376676.

[22] M. Charikar, K. Makarychev, and Y. Makarychev, *Near-optimal algorithms for unique games*, in Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing, Seattle WA USA, May 2006, ACM, pp. 205–214, https://doi.org/10.1145/1132516.1132547.

[23] M. Charikar, K. Makarychev, and Y. Makarychev, *Near-optimal algorithms for maximum constraint satisfaction problems*, ACM Trans. Algorithms, 5 (2009), pp. 32:1–32:14, https://doi.org/10.1145/1541885.1541893, https://doi.org/10.1145/1541885.1541893.

[24] L. Ciardo, M. Kozik, A. A. Krokhin, T. Nakajima, and S. Živný, *1-in-3 vs. not-all-equal: Dichotomy of a broken promise*, ACM Trans. Comput. Log., 26 (2025), pp. 10:1–10:22, https://doi.org/10.1145/3719007.

[25] L. Ciardo and S. Živný, *Approximate graph colouring and crystals*, in Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, 2023, pp. 2256–2267.

[26] L. Ciardo and S. Živný, *Approximate graph colouring and the hollow shadow*, in Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023, B. Saha and R. A. Servedio, eds., ACM, 2023, pp. 623–631, https://doi.org/10.1145/3564246.3585112.

[27] L. Ciardo and S. Živný, *CLAP: A new algorithm for promise CSPs*, SIAM J. Comput., 52 (2023), pp. 1–37, https://doi.org/10.1137/22M1476435, https://arxiv.org/abs/2107.05018.

[28] L. Ciardo and S. Živný, *Hierarchies of minion tests for PCSPs through tensors*, in Proc. 2023 ACM-SIAM Symposium on Discrete Algorithms (SODA'23), 2023, pp. 568–580, https://doi.org/10.1137/1.9781611977554.ch25, https://arxiv.org/abs/2207.02277.

[29] L. Ciardo and S. Živný, *Semidefinite programming and linear equations vs. homomorphism problems*, in Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024, B. Mohar, I. Shinkar, and R. O'Donnell, eds., ACM, 2024, pp. 1935–1943, https://doi.org/10.1145/3618260.3649635.

[30] V. Dalmau and A. A. Krokhin, *Robust satisfiability for csps: Hardness and algorithmic results*, ACM Trans. Comput. Theory, 5 (2013), pp. 15:1–15:25, https://doi.org/10.1145/2540090.

[31] M. Ficak, M. Kozik, M. Olšák, and S. Stankiewicz, *Dichotomy for Symmetric Boolean PCSPs*, in Proc. 46th International Colloquium on Automata, Languages, and Programming (ICALP'19), vol. 132, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019, pp. 57:1–57:12, https://doi.org/10.4230/LIPIcs.ICALP.2019.57, https://arxiv.org/abs/1904.12424.

[32] M. Filakovský, T.-V. Nakajima, J. Opršal, G. Tasinato, and U. Wagner, *Hardness of Linearly Ordered 4-Colouring of 3-Colourable 3-Uniform Hypergraphs*, in Proc. 41st International Symposium on Theoretical Aspects of Computer Science (STACS'24), vol. 289 of Leibniz International Proceedings in Informatics (LIPIcs), 2024, pp. 34:1–34:19, https://doi.org/10.4230/LIPIcs.STACS.2024.34, https://arxiv.org/abs/2312.12981.

[33] M. R. Garey and D. S. Johnson, *The complexity of near-optimal graph coloring*, J. ACM, 23 (1976), pp. 43–49, https://doi.org/10.1145/321921.321926.

[34] V. Guruswami and S. Sandeep, *d-To-1 Hardness of Coloring 3-Colorable Graphs with O(1) Colors*, in Proc. 47th International Colloquium on Automata, Languages, and Programming (ICALP'20), vol. 168 of LIPIcs, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020, pp. 62:1–62:12, https://doi.org/10.4230/LIPIcs.ICALP.2020.62.

[35] V. Guruswami and Y. Zhou, *Tight bounds on the approximability of almost-satisfiable horn SAT and exact hitting set*, Theory Comput., 8 (2012), pp. 239–267, https://doi.org/10.4086/toc.2012.v008a011.

[36] J. Håstad, *Some optimal inapproximability results*, J. ACM, 48 (2001), pp. 798–859, https://doi.org/10.1145/502090.502098.

[37] J. Håstad, B. Martinsson, T. Nakajima, and S. Živný, *A logarithmic approximation of linearly-ordered colourings*, in Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2024, August 28-30, 2024, London School of Economics, London, UK, A. Kumar and N. Ron-Zewi, eds., vol. 317 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, pp. 7:1–7:6, https://doi.org/10.4230/LIPICS.APPROX/RANDOM.2024.7.

[38] P. M. Idziak, P. Markovic, R. McKenzie, M. Valeriote, and R. Willard, *Tractability and learnability arising from algebras with few subpowers*, SIAM J. Comput., 39 (2010), pp. 3023–3037, https://doi.org/10.1137/090775646.

[39] S. Khot, *On the power of unique 2-prover 1-round games*, in STOC'02: Proceedings of the 34th Annual ACM Symposium on Theory of Computing, ACM, New York, 2002, pp. 767–775.

[40] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell, *Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs?*, SIAM J. Comput., 37 (2007), pp. 319–357, https://doi.org/10.1137/S0097539705447372.

[41] A. Krokhin, J. Opršal, M. Wrochna, and S. Živný, *Topology and adjunction in promise constraint satisfaction*, SIAM J. Comput., 52 (2023), pp. 37–79, https://doi.org/10.1137/20M1378223, https://arxiv.org/abs/2003.11351.

[42] A. Larrauri and S. Živný, *Solving promise equations over monoids and groups*, ACM Trans. Comput. Log., 26 (2025), pp. 3:1–3:24, https://doi.org/10.1145/3698106.

[43] E. Mossel, R. O'Donnell, and K. Oleszkiewicz, *Noise stability of functions with low influences: invariance and optimality*, Ann. of Math. (2), 171 (2010), pp. 295–341, https://doi.org/10.4007/annals.2010.171.295.

[44] A. Mottet, *Algebraic and algorithmic synergies between promise and infinite-domain csps*, in Proceedings of the 40th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2025, 2025.

[45] T. Nakajima, Z. Verwimp, M. Wrochna, and S. Živný, *Complexity of approximate conflict-free, linearly-ordered, and nonmonochromatic hypergraph colourings*, in 52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8-11, 2025, Aarhus, Denmark, K. Censor-Hillel, F. Grandoni, J. Ouaknine, and G. Puppis, eds., vol. 334 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025, pp. 169:1–169:10, https://doi.org/10.4230/LIPICS.ICALP.2025.169.

[46] T.-V. Nakajima and S. Živný, *Linearly ordered colourings of hypergraphs*, ACM Trans. Comput. Theory, 13 (2022), https://doi.org/10.1145/3570909, https://arxiv.org/abs/2204.05628.

[47] T.-V. Nakajima and S. Živný, *On the complexity of symmetric vs. functional PCSPs*, ACM Trans. Algorithms, 20 (2024), pp. 33:1–33:29, https://doi.org/10.1145/3673655.

[48] P. Raghavendra, *Optimal algorithms and inapproximability results for every CSP?*, in Proc. 40th Annual ACM Symposium on Theory of Computing (STOC'08), 2008, pp. 245–254, https://doi.org/10.1145/1374376.1374414.

[49] P. Raghavendra and D. Steurer, *How to round any csp*, in 2009 50th Annual IEEE Symposium on Foundations of Computer Science, IEEE, 2009, pp. 586–594.

[50] T. Schaefer, *The complexity of satisfiability problems*, in Proc. 10th Annual ACM Symposium on the Theory of Computing (STOC'78), 1978, pp. 216–226, https://doi.org/10.1145/800133.804350.

[51] D. Zhuk, *A proof of the CSP dichotomy conjecture*, J. ACM, 67 (2020), pp. 30:1–30:78, https://doi.org/10.1145/3402029.

[52] U. Zwick, *Finding almost-satisfying assignments*, in Proc. 13th Annual ACM Symposium on the Theory of Computing (STOC'98), ACM, 1998, pp. 551–560, https://doi.org/10.1145/276698.276869.