A Dichotomy for Maximum PCSPs on Graphs*

Tamio-Vesa Nakajima
University of Oxford
tamio-vesa.nakajima@cs.ox.ac.uk

Stanislav Živný
University of Oxford
standa.zivny@cs.ox.ac.uk

Fix two non-empty loopless graphs G and H such that G maps homomorphically to H. The *Maximum Promise Constraint Satisfaction Problem* parameterised by G and H is the following computational problem, denoted by $\mathrm{MaxPCSP}(G, H)$: Given an input (multi)graph X that admits a map to G preserving a ρ-fraction of the edges, find a map from X to H that preserves a ρ-fraction of the edges. As our main result, we give a complete classification of this problem under Khot's Unique Games Conjecture: The only tractable cases are when G is bipartite and H contains a triangle.

Along the way, we establish several results, including an efficient approximation algorithm for the following problem: Given a (multi)graph X which contains a bipartite subgraph with ρ edges, what is the largest triangle-free subgraph of X that can be found efficiently? We present an SDP-based algorithm that finds one with at least 0.8823ρ edges, thus improving on the subgraph with 0.878ρ edges obtained by the classic Max-Cut algorithm of Goemans and Williamson.

1 Introduction

Given two undirected graphs[1] G and H, a *homomorphism* from G to H is an edge preserving map h from $V(G)$ to $V(H)$; that is, if $(u, v) \in E(G)$ then $(h(u), h(v)) \in E(H)$. A classic result of Hell and Nešetřil established a computational *dichotomy* for the so-called H-colouring problem [HN90], for a fixed graph H: if H is bipartite then deciding whether an input graph G is homomorphic to H is solvable in polynomial time, and for every other H this problem is NP-complete. Going beyond graphs, Feder and Vardi conjectured that a similar dichotomy holds for all finite relational structures [FV98], not only for graphs and for relational structures over the Boolean domain [Sch78].[2] Bulatov [Bul17] and, independently, Zhuk [Zhu20], confirmed the tractability part of the dichotomy, which together with the NP-hardness part [BJK05], answered the Feder-Vardi conjecture in the affirmative. The homomorphism problem is also known as the *constraint satisfaction problem* (CSP) [Jea98]. CSPs can be equivalently defined as problems seeking an assignment of values to the given variables subject to

*An extended abstract of a part of this work will appear in Proceedings of ICALP (A) 2025. This work was supported by UKRI EP/X024431/1 and Clarendon Fund Scholarship.

All data is provided in full in the results section of this paper.

[1] All graphs in this article are loopless and non-empty, meaning having at least one edge (and thus at least two vertices).

[2] We will not need relational structures, but, intuitively, one should think of them as generalisations of (hyper)graphs in which one is given a ground set and a collection of relations on the ground set.

the given constraints. The fixed target structure in the homomorphism problem corresponds to the set of allowed (domain) values and the set of allowed relations in the constraints. Concrete examples of CSPs include solvability of linear equations over finite fields and variants of (hyper)graph colourings.

A well-studied line of work focuses on *approximability* of CSPs [KSTW00, TSSW00]. A classic example here is the Max-Cut problem. In Max-Cut, the variables correspond to the vertices of the input graph, the values are just 0 and 1 (corresponding to the two sides of a cut), and the constraints are binary disequalities associated with the edges of the graph. Given a CSP, the computational task could be to find a solution maximising the number of satisfied constraints as in Max-Cut, or finding a (perfect) solution satisfying all constraints as discussed in the previous paragraph.

A *promise* CSP (PCSP) is a CSP in which each constraint comes in two forms, a strong one and a weak one. The promise is that a solution exists using the strong versions of the constraints, while the (possibly easier) task is to find a solution using the weak constraints. A recent line of work by Austrin, Guruswami, and Håstad [AGH17], Brakensiek and Guruswami [BG21], and Barto, Bulín, Krokhin, and Opršal [BBKO21] initiated a systematic study of PCSPs with perfect completeness, i.e., finding a solution satisfying all weak constraints given the promise that a solution satisfying all strong constraints exists. Canonical examples include approximate graph [GJ76] and hypergraph [DRS05, ABP20, BBB21] colouring problems, e.g., finding a 5-colouring of a given 3-colourable graph [KOWŽ23]. PCSPs are a vast generalisation of CSPs and their complexity is not well understood, not even on the Boolean domain [FKOS19, BGS23a] or for graphs. In particular, Brakensiek and Guruswami conjectured that only bipartite graphs lead to tractable PCSPs on graphs [BG21] (cf. Conjecture 3 in Section 3 for a precise statement). Resolving their conjecture would in particular include resolving the notoriously difficult approximate graph colouring problem, cf. [AFO+25] for exciting recent progress.

In this work, we will focus on the *approximability of maximisation PCSPs*. The ultimate goal is to understand the precise approximation factor for all MaxPCSPs, and thus identify where the transition from tractability to intractability occurs. This is an ambitious, long-term goal that would encompass many existing fundamental results.

An example of a MaxPCSP is the following problem. Given a graph G that admits a 2-colouring of the vertices with a ρ-fraction of the edges coloured properly, find a 3-colouring of G with an $\alpha\rho$-fraction of the edges coloured properly, where $0 < \alpha \leq 1$ is the approximation factor. As one of the results in the present paper, we will show that there is a 1-approximation algorithm; i.e., given a graph with a 2-colouring with ρ-fraction of non-monochromatic edges, one can efficiently find a 3-colouring of the graph with the same fraction of non-monochromatic edges.

As our main result, we will establish a dichotomy for 1-approximation of graph PCSPs under Khot's Unique Games Conjecture (UGC) [Kho02].

Theorem 1. *Let G and H be two fixed graphs such that there is a homomorphism from G to H. If G is bipartite and H contains a triangle then* $\mathrm{MaxPCSP}(G, H)$ *is 1-approximable. Otherwise, 1-approximation of* $\mathrm{MaxPCSP}(G, H)$ *is* NP*-hard assuming the* UGC.

Along the way to prove Theorem 1, we will design two efficient approximation algorithms. We shall discuss one of them briefly here, with an overview of both algorithms and all results in Section 3.

Given an undirected (multi)graph G, what is the bipartite subgraph of G with the most edges? This is nothing but the already mentioned Max-Cut problem, one of the most fundamental problems in computer science. Max-Cut was among the 21 problems shown to be NP-hard by Karp [Kar72]. Papadimitriou and Yannakakis showed that Max-Cut is APX-hard [PY91] and thus does not admit a polynomial-time approximation scheme, unless P = NP. However, there are several simple 0.5-approximation algorithms. Goemans and Williamson used semidefinite programming and randomised rounding to design a 0.878-approximation algorithm [GW95]. Khot, Kindler, Mossel, O'Donnell, and Oleszkiewicz established the optimality of this algorithm [KKMO07, MOO10] under the UGC.

What if the task is merely finding a large triangle-free subgraph (rather than a bipartite one)?

While the Goemans-Williamson algorithm can still be used, as one of our results we design an algorithm with a better approximation guarantee: If G contains a bipartite subgraph with ρ edges, our algorithm efficiently finds a triangle-free subgraph of G with 0.8823ρ edges.[3] Our algorithm is a randomised combination of the Goemans-Williamson original "random hyperplane algorithm", and an algorithm that first selects "long edges" (meaning edges for which the angle between the corresponding vectors from the SDP solution is above a certain threshold) and then applies a random hyperplane rounding, selecting "shorter edges" (still longer than some other threshold). The probability of the biased coin that selects one of the two algorithms depends on certain geometric quantities which guarantee that the resulting subgraph is indeed triangle-free. We complement our tractability result for this problem by showing that it is NP-hard to find a triangle-free subgraph with $(25/26+\varepsilon)\rho \approx (0.961+\varepsilon)\rho$ edges. This result is obtained by a reduction from Håstad's 3-bit PCP [Hås01].

Related work The notion of MaxPCSPs is a natural generalisation of the well-studied notion of MaxCSPs. For finite-domain MaxCSPs, it is known that a certain rounding of the basic SDP relaxation gives, up to some ε, the UGC-optimal approximation ratio (in time doubly exponential in $1/\varepsilon$) [Rag08, RS09]. However, the Raghavendra-Steurer algorithm does not immediately give a 1-approximation algorithm due to the above-mentioned ε, even for MaxCSPs. Moreover, that result is established only for finite-domain MaxCSPs. On other hand, our results include a 1-approximation for MaxPCSP(K_2, K_3), and an algorithm for infinite-domain structures, namely for MaxPCSP$(K_2, \mathfrak{G}_3)$, which captures the bipartite vs. triangle-free subgraph discussed above (cf. Section 2 for a precise definition).

Approximation of concrete MaxPCSPs has been studied for decades, include several papers on almost approximate graph colouring [EH08, DKPS10, KS12, HMS23], approximate colouring [NŽ23], and promise Max-3-LIN [BLŽ25]. Our work initiates a systematic investigation, giving a complete classification for 1-approximation of the graph case.

Recent work of Brakensiek, Guruswami, and Sandeep [BGS23b] studied robust approximation of MaxPCSPs; in particular, they state that Raghavendra's above-mentioned theorem on approximate MaxCSPs [Rag08] applies verbatim to MaxPCSPs. This in combination with the work of Brown-Cohen and Raghavendra [BR15] gives a framework for studying approximation of MaxPCSPs. An alternative framework for studying approximation of MaxPCSPs has recently been put forward by Barto, Butti, Kazda, Viola, and Živný [BBK+24].

Paper organisation After defining MaxPCSPs formally and few other basic concepts in Section 2, we will state all our results precisely in Section 3. The rest of the paper is then split in different parts of the proofs, including two tractability results in Section 4 and Section 5 and hardness results in Section 6 and Section 7

[3] We note that our algorithm is easy to extend to the case where edges have positive weights.

2 Preliminaries

For two nonzero vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$, we denote by $\angle(\mathbf{x}, \mathbf{y})$ the angle between $\mathbf{x}$ and $\mathbf{y}$ in radians; i.e., $\angle(\mathbf{x}, \mathbf{y}) = \arccos\left(\frac{\mathbf{x}\cdot\mathbf{y}}{||\mathbf{x}||||\mathbf{y}||}\right)$. The following useful fact is well-known, cf. [Euc26, Book XI, Proposition 21].

Lemma 2. *For any three nonzero vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \mathbb{R}^N$, we have $\angle(\mathbf{x}_1, \mathbf{x}_2) + \angle(\mathbf{x}_2, \mathbf{x}_3) + \angle(\mathbf{x}_3, \mathbf{x}_1) \leq 2\pi$.*

Graphs and (partial) homomorphisms. All graphs will be nonempty, undirected and loopless but with possibly multiple edges. Fix two graphs $G = (V, E)$, $H = (U, F)$, and $\rho \in \mathbb{N}$. We say that there exists a *partial homomorphism* of weight ρ from G to H, and write $G \xrightarrow{\rho} H$, if there exists a mapping $h : V \to U$ such that for ρ edges $(x, y) \in E$ we have $(h(x), h(y)) \in F$. If $G \xrightarrow{|E|} H$, we say that there exists a homomorphism from G to H and write $G \to H$. (Note that for any G, H, I, if $G \xrightarrow{\rho} H \to I$ then $G \xrightarrow{\rho} I$.)

We denote by K_2 a clique on two vertices. A partial homomorphism $h : G \xrightarrow{\rho} K_2$ represents a cut of weight ρ, namely the edges (x, y) with $h(x) \neq h(y)$. Equivalently, it represents a bipartite subgraph of G with weight ρ. We now introduce a graph that similarly captures triangle-free subgraphs. Let $\mathfrak{G}_3$ be the direct sum of all finite triangle-free graphs. In other words, for every finite triangle-free graph $G = (V, E)$, the graph $\mathfrak{G}_3$ contains vertices x_G for $x \in V$, and edges (x_G, y_G) for $(x, y) \in E$. Then, for finite G, a partial homomorphism $h : G \xrightarrow{\rho} \mathfrak{G}_3$ represents a triangle-free subgraph of G with weight ρ: all the edges that connect vertices that are mapped by h to neighbouring vertices in $\mathfrak{G}_3$ form a triangle-free subgraph of G.

Maximum PCSPs. Fix two (possibly infinite) graphs $G \to H$. Then the *maximum promise constraint satisfaction problem* (MaxPCSP) for undirected graphs, denoted by MaxPCSP(G, H), is defined as follows. In the search version of the problem, we are given a (multi)graph X such that $X \xrightarrow{\rho} G$, and must find $h : X \xrightarrow{\rho} H$; this problem can be approximated with the approximation ratio α if we can find $h : X \xrightarrow{\lceil\alpha\rho\rceil} H$. In the decision version, we are given a (multi)graph X and a number $\rho \in \mathbb{N}$ and must output YES if $X \xrightarrow{\rho} G$, and NO if not even $X \xrightarrow{\rho} H$. This problem can be approximated with approximation ratio α if we can decide between $X \xrightarrow{\rho} G$ and not even $X \xrightarrow{\lceil\alpha\rho\rceil} H$. (In all cases, ρ is *not* part of the input.)

In particular, approximating the problem MaxPCSP$(K_2, \mathfrak{G}_3)$ with approximation ratio α means the following. In the search version: given a graph G that contains a cut of weight ρ, find a triangle-free subgraph of weight $\alpha\rho$. In the decision version: given a graph G and a number $\rho \in \mathbb{N}$, output YES if it has a cut of weight ρ, and NO if it has no triangle-free subgraph of weight $\alpha\rho$.

We define the problem PCSP(G, H) identically to MaxPCSP(G, H), except that it is guaranteed that ρ is the number of edges of G. Thus observe that PCSP(G, H) reduces to MaxPCSP(G, H) trivially, in the sense that there is a polynomial-time reduction from PCSP(G, H) to MaxPCSP(G, H) that does not change the input.

Suppose $G \to G' \to H' \to H$. Then, it follows that PCSP(G, H) polynomial-time reduces to PCSP(G', H') and MaxPCSP(G, H) polynomial-time reduces to MaxPCSP(G', H') (and the same holds for α-approximation). Furthermore, the decision version of PCSP(G, H) and MaxPCSP(G, H) polynomial-time reduces to the search version of PCSP(G, H) and MaxPCSP(G, H), respectively. In other words, the decision version is no harder than the search version. Hence by proving our tractability results for the search version, and our hardness results for the decision version, we prove them for both versions of the problems.

SDP. For the Max-Cut problem, which is just $\mathrm{MaxPCSP}(K_2, K_2)$, the *basic SDP relaxation* for a graph $G = (V, E)$ with n vertices, which can be solved within additive error ε in polynomial time with respect to the size of G and $\log(1/\varepsilon)$,[4] is as follows:

$$\begin{aligned} \max \quad & \sum_{(u,v)\in E} \frac{1 - \mathbf{x}_u \cdot \mathbf{x}_v}{2} \\ \text{s.t.} \quad & \|\mathbf{x}_u\|^2 = 1, \\ & \mathbf{x}_u \in \mathbb{R}^n. \end{aligned} \tag{1}$$

Goemans and Williamson [GW95] gave a rounding algorithm for the SDP (1) with approximation ratio

$$\alpha_{GW} = \left(\max_{0\le\tau\le\pi} \frac{\pi}{2} \frac{1 - \cos\tau}{\tau} \right)^{-1} = 0.878\cdots,$$

thus beating the trivial approximation ratio of 1/2 obtained by, e.g., a random cut. Their algorithm solves the SDP (1), selects a uniformly random hyperplane in $\mathbb{R}^N$, and returns the cut induced by the hyperplane.

3 Results

Our main result is the following.

Theorem 1. *Let G and H be two fixed graphs such that there is a homomorphism from G to H. If G is bipartite and H contains a triangle then* $\mathrm{MaxPCSP}(G, H)$ *is 1-approximable. Otherwise, 1-approximation of* $\mathrm{MaxPCSP}(G, H)$ *is* NP*-hard assuming the* UGC.

This is an optimisation variant of a conjecture by Brakensiek and Guruswami on the tractability boundary of promise CSPs on undirected graphs.

Conjecture 3 ([BG21]). *Let G and H be two fixed graphs such that there is a homomorphism from G to H. If G is bipartite then* $\mathrm{PCSP}(G, H)$ *is tractable. Otherwise,* $\mathrm{PCSP}(G, H)$ *is* NP*-hard.*

The currently known cases supporting Conjecture 3 are NP-hardness of $\mathrm{PCSP}(K_3, K_5)$ [BBKO21], $\mathrm{PCSP}(K_k, K_{\binom{k}{\lfloor k/2\rfloor}} - 1)$ for $k \ge 4$ [KOWŽ23], and $\mathrm{PCSP}(C_{2k+1}, K_4)$ for $k \ge 1$ [AFO+25], where C_{2k+1} denotes a cycle on $2k+1$ vertices. As $\mathrm{PCSP}(G, H)$ reduces to $\mathrm{MaxPCSP}(G, H)$, Conjecture 3 implies that $\mathrm{MaxPCSP}(G, H)$ is NP-hard whenever G is non-bipartite. We establish this result under the UGC but *not* relying on Conjecture 3. It follows from our Theorem 1 that not all cases of $\mathrm{MaxPCSP}(G, H)$ with bipartite G are 1-approximable, and thus the tractability boundary lies elsewhere for 1-approximation.

We note that establishing Theorem 1 appears easier than resolving Conjecture 3, similarly to how the complexity of exact solvability of MaxCSPs [TŽ16] was resolved before the complexity of decision CSPs [Bul17, Zhu20].

An important part in proving Theorem 1 is the NP-hardness of finding a triangle-free subgraph. For this problem, we also establish a non-trivial approximation result.

[4] Throughout we will ignore issues of real precision.

Theorem 4. $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ *is* 0.8823*-approximable (in the search version) in polynomial time, and it is* NP*-hard to* $(25/26 + \varepsilon)$*-approximate (even in the decision version) for any fixed* $\varepsilon > 0$.

Note that crucially $0.8823 > 0.878\ldots$, thus our algorithm beats the Goemans-Williamson algorithm. Theorem 4 is proved in two parts: the tractability side in Section 4 and the hardness side in Section 6. We quickly give an intuitive explanation of why such an algorithm is possible. Define

$$\tau_{GW} = \arg\max_{0\le\tau\le\pi} \frac{\pi}{2}\frac{1-\cos\tau}{\tau} \approx 0.742\pi.$$

The function $\tau \mapsto (\pi/2)(1-\cos\tau)/\tau$, depicted in Figure 1, is increasing up to τ_{GW}, then decreasing. Why would $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ be easier to approximate than $\mathrm{MaxPCSP}(K_2, K_2)$? Consider the

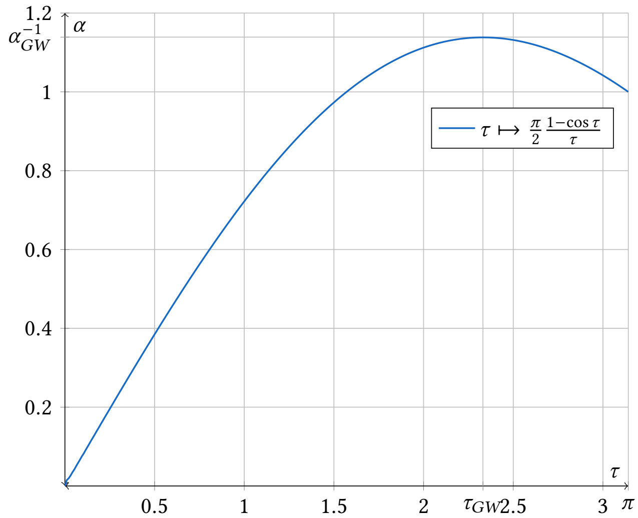


Figure 1: Function giving rise to α_{GW}, τ_{GW}.

Goemans-Williamson algorithm for $\mathrm{MaxPCSP}(K_2, K_2)$; the worst-case performance of this algorithm appears in a graph where the embedding into $\mathbb{R}^N$ given by solving the SDP (1) gives all edges an angle of approximately τ_{GW}. Observe that $\tau_{GW} > 2\pi/3$ — so immediately (cf. Lemma 2) this instance is triangle-free. So, for this instance an algorithm for $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ could just return the entire graph! Indeed, in order to create an instance that contains triangles one needs to introduce shorter edges. This suggests that a hybrid algorithm, that either selects "long edges" or some appropriate selection of "shorter edges" (still longer than some threshold), should have better performance. The details can be found in Section 4, while the NP-hardness part of Theorem 4 is proved in Section 6.

Our next result is a combination of the Goemans-Williamson SDP for Max-Cut [GW95] and a rounding scheme due to Frieze and Jerrum for Max-3-Cut [FJ97].

Theorem 5. $\mathrm{MaxPCSP}(K_2, K_3)$ *is 1-approximable in polynomial time (in the search version).*

This algorithm is somewhat similar to the Goemans-Williams algorithm, except that rather than selecting a uniformly random hyperplane, it selects three normally distributed vectors, and partitions the vertices according to which vector they are closest to (where closeness is measured in terms of inner products). Thus, while this algorithm solves the same SDP as the algorithm from Theorem 4, it rounds the solution differently. This rounding scheme is the same as the rounding scheme of [FJ97]; it is also very similar to one of the rounding schemes of [KMS98]. The details can be found in Section 5.

The key in proving NP-hardness in Theorem 1 is the following result, established by generalising the proof of Dinur, Mossel, and Regev[DMR09] with a multilayered unique games conjecture, in the style of [BWŽ21]. The details can be found in Section 7.

Theorem 6. *For every $k \geq 1$ and $\ell \geq 3$, 1-approximation of* $\mathrm{MaxPCSP}(C_{2k+1}, K_\ell)$ *is* NP*-hard assuming the* UGC.

We now have all tools to prove Theorem 1.

Proof of Theorem 1. If G is bipartite and H contains a triangle then $G \to K_2 \to K_3 \to H$, so it follows that $\mathrm{MaxPCSP}(G, H)$ reduces to $\mathrm{MaxPCSP}(K_2, K_3)$, which is 1-approximable by Theorem 5.

If G is bipartite and H is triangle-free then $K_2 \to G \to H \to \mathfrak{G}_3$, so $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ reduces to $\mathrm{MaxPCSP}(G, H)$. Hence, by Theorem 4, it is NP-hard even to approximate $\mathrm{MaxPCSP}(G, H)$ with approximation ratio $25/26 + \varepsilon$ (in the decision version), and thus in particular 1-apprxomation of $\mathrm{MaxPCSP}(G, H)$ is NP-hard (in the decision version).

If G is not bipartite then we have $C_{2k+1} \to G$ for some k, and, since H is loopless, we have $H \to K_\ell$ for some $\ell \geq 3$. Hence NP-hardness of 1-approximation under the the UGC follows from Theorem 6. ◻

We note that our hardness result depends in an essential way on the fact that the input graph can have multiple edges; we equivalently could have allowed non-negative integer weights on the edges. This variant of the problem is most natural when looking at it as a constraint satisfaction problem. It is interesting to ask what the complexity of $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ is if the input graph is both weightless and without multiple edges.

4 Approximation of MaxPCSP$(K_2, \mathfrak{G}_3)$

In this section, we will prove the tractability part of the following result.

Theorem 4. $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ *is* 0.8823*-approximable (in the search version) in polynomial time, and it is* NP*-hard to* $(25/26 + \varepsilon)$*-approximate (even in the decision version) for any fixed* $\varepsilon > 0$.

We will need the following technical lemma.

Lemma 7. *There exist $\alpha, P, Q, \tau \in \mathbb{R}$ with $P + Q = 1, P \geq 0, Q \geq 0$, $\tau \in [2\pi/3, \tau_{GW}]$, such that the following hold*

$$P\frac{\theta}{\pi} + Q \geq \alpha\frac{1 - \cos\theta}{2} \qquad \theta \in [\tau, \pi] \tag{2}$$

$$P\frac{\varphi}{\pi} + Q\frac{\varphi}{\pi} \geq \alpha\frac{1 - \cos\varphi}{2} \qquad \varphi \in [\pi - \tau/2, \tau] \tag{3}$$

$$P\frac{\psi}{\pi} \geq \alpha\frac{1 - \cos\psi}{2} \qquad \psi \in [0, \pi - \tau/2]. \tag{4}$$

In particular, we can take $\alpha \geq 0.88232$, $\tau = 2.18746$, $Q = 1 - P$ and

$$P = \frac{\alpha\pi}{2}\left(\frac{1 - \cos(\pi - \tau/2)}{\pi - \tau/2}\right) \approx 0.987535.$$

Proof. Firstly (4) is most tight when $\psi = \pi - \tau/2$, so it reduces to

$$P \geq \frac{\alpha\pi}{2} \underbrace{\left(\frac{1-\cos(\pi-\tau/2)}{\pi-\tau/2}\right)}_{X(\tau)}.$$

Next, (3) is actually independent of P, Q, since $P + Q = 1$ and thus the left-hand side is just φ/π. Since we will take $\tau \leq \tau_{GW}$, it is also most tight when $\varphi = \tau$ (cf. Figure 1), so it becomes

$$\alpha \leq \frac{2}{\pi}\frac{\tau}{1-\cos\tau}.$$

For (2) we see that it becomes easier to satisfy if P is smaller and Q is larger, so we will choose P to be the minimum value it could have vis-à-vis (3) and (4), namely

$$P = \frac{\alpha\pi}{2}X(\tau).$$

Hence, as $P + Q = 1$, the first constraint becomes

$$\frac{\alpha\theta}{2}X(\tau) + 1 - \frac{\alpha\pi}{2}X(\tau) \geq \alpha\frac{1-\cos\theta}{2}.$$

Simplifying, we get

$$\alpha\frac{\theta-\pi}{2}X(\tau) + 1 \geq \alpha\frac{1-\cos\theta}{2}.$$

This is equivalent to

$$\alpha \leq \frac{2}{(\pi-\theta)X(\tau) - \cos\theta + 1}$$

Separating out α, we have

$$\alpha \leq \min_\theta \frac{2}{(\pi-\theta)X(\tau)-\cos\theta+1} = 2\left(\max_\theta(\pi-\theta)X(\tau)-\cos\theta+1\right)^{-1}$$

Now we can observe that $0.75 > X(\tau) > 0.6$ for $2\pi/3 \leq \tau \leq \pi$ numerically. The derivative of the function we are maximising is $-X(\tau) + \sin\theta$ — this is necessarily positive at $\theta = 2\pi/3$ and negative for $\theta = \pi$, because of the approximation of $X(\tau)$ above. So the maximum is hit at $\pi - \arcsin X(\tau)$, as this is the solution to the equation within the bound for θ. Thus the bound is

$$\alpha \leq \frac{2}{X(\tau)\arcsin X(\tau) + \cos\arcsin X(\tau) + 1}$$

So we now want to find τ that maximises α such that

$$\alpha \leq \frac{2}{X(\tau)\arcsin X(\tau) + \cos\arcsin X(\tau) + 1} \tag{5}$$

$$\alpha \leq \frac{2}{\pi}\frac{\tau}{1-\cos\tau} \tag{6}$$

We can see this situation in Figure 2. Numerically we compute that if we choose $\tau = 2.18746$, then we get $\alpha \geq 0.88232$. □

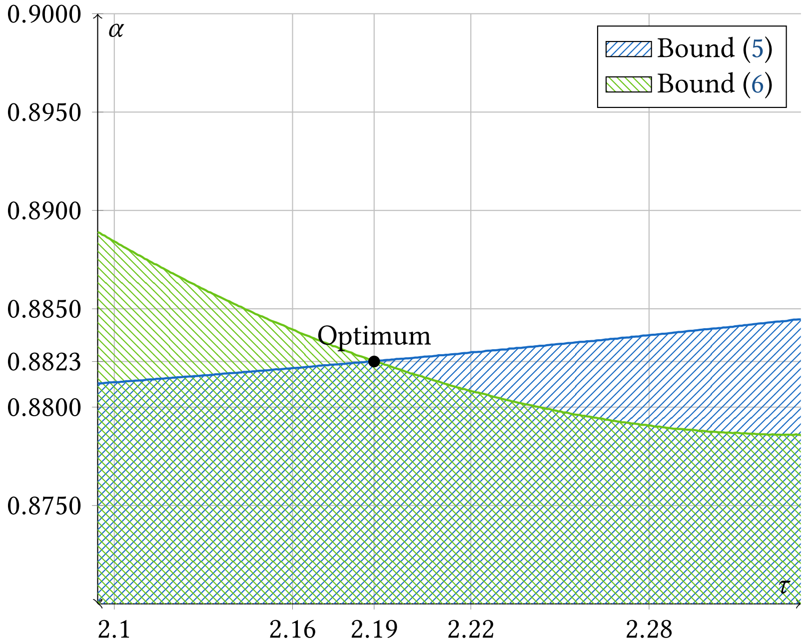


Figure 2: Bounds from Lemma 7

We now prove the desired tractability result.[5]

Theorem 8. $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ *can be* 0.8823*-approximated in polynomial time.*

Proof. Take α, P, Q, τ as in Lemma 7. Consider the following randomised algorithm.

1. Input: a graph $G = (V, E)$, which admits a cut of weight $\rho \geq |E|/2$.
2. Solving SDP (1) to within error ε, we get a set of vectors $\mathbf{x}_v$ for $v \in V$, with $||\mathbf{x}_v||^2 = 1$ and

$$\sum_{(u,v)\in E} \frac{1 - \mathbf{x}_u \cdot \mathbf{x}_v}{2} \geq \rho - \varepsilon.$$

3. Flip a biased coin, randomly choosing from the following two cases.
   - (i) With probability P, sample a uniformly random hyperplane H, and compute the set of edges (u, v) with $\mathbf{x}_u$ on the opposite side of H as $\mathbf{x}_v$. Return this set of edges.
   - (ii) With probability Q, return all the edges (u, v) with $\angle(\mathbf{x}_u, \mathbf{x}_v) > \tau$, then sample a uniformly random hyperplane H and additionally return all the edges (u, v) with $\angle(\mathbf{x}_u, \mathbf{x}_v) \geq \pi - \tau/2$ and with $\mathbf{x}_u, \mathbf{x}_v$ on opposite sides of H.

First, let us verify that our algorithm returns a triangle-free subgraph. First, in Case (i), we return a bipartite subgraph, so we certainly return a triangle-free subgraph. The reasoning for Case (ii) is more geometric. Consider any three edges returned in this case. If the three edges are of angle between $\pi - \tau/2$ and τ, then they cannot form a triangle, since the edges of this kind that we return form a

[5] We remark in passing that *no* SDP-based algorithm can have performance greater than $8/9 = 0.888\ldots$, since for the triangle K_3 the SDP value is $9/4$, yet the largest triangle-free subgraph has weight 2 (and $2/(9/4) = 8/9$).

bipartite graph (determined by H). Conversely, suppose that at least one edge has angle greater than τ, and the other two have angle at least $\pi - \tau/2$. Then the sum of the angles of the edges are $> 2\pi$, and hence by Lemma 2, they cannot form a triangle.

Now, we compute the expected performance of our algorithm. Recall that for a uniformly random hyperplane H, two unit vectors $\mathbf{x}, \mathbf{y}$ are on opposite sides of H with probability $\angle(\mathbf{x}, \mathbf{y})/\pi$ [GW95]. Consider any edge (u, v) in our graph. We will show that the probability that the edge is included in the cut is at least $\alpha(1 - \mathbf{x}_u \cdot \mathbf{x}_v)/2 = \alpha(1 - \cos\angle(\mathbf{x}_u, \mathbf{x}_v))/2$; there are three cases depending on $\angle(\mathbf{x}_u, \mathbf{x}_v)$.

$\angle(\mathbf{x}_u, \mathbf{x}_v) \in (\tau, \pi]$. In this case, the edge is included with probability $\angle(\mathbf{x}_u, \mathbf{x}_v)/\pi$ in Case (i), and with probability 1 in Case (ii). Thus, and by applying (2) from Lemma 7, we find that the edge is included with probability

$$P\frac{\angle(\mathbf{x}_u, \mathbf{x}_v)}{\pi} + Q \geq \alpha\frac{1 - \cos\angle(\mathbf{x}_u, \mathbf{x}_v)}{2}.$$

$\angle(\mathbf{x}_u, \mathbf{x}_v) \in [\pi - \tau/2, \tau]$. In this case, the edge is included with probability $\angle(\mathbf{x}_u, \mathbf{x}_v)/\pi$ both in Case (i) and in Case (ii). Thus, by (3) from Lemma 7, the edge is included with probability

$$P\frac{\angle(\mathbf{x}_u, \mathbf{x}_v)}{\pi} + Q\frac{\angle(\mathbf{x}_u, \mathbf{x}_v)}{\pi} \geq \alpha\frac{1 - \cos\angle(\mathbf{x}_u, \mathbf{x}_v)}{2}.$$

$\angle(\mathbf{x}_u, \mathbf{x}_v) \in [0, \pi - \tau/2)$. In this case, the edge is included with probability $\angle(\mathbf{x}_u, \mathbf{x}_v)/\pi$ in Case (i) and never included in Case (ii). Thus, by (4) from Lemma 7, the edge is included with probability

$$P\frac{\angle(\mathbf{x}_u, \mathbf{x}_v)}{\pi} \geq \alpha\frac{1 - \cos\angle(\mathbf{x}_u, \mathbf{x}_v)}{2}.$$

Hence overall we include an edge (u, v) with probability at least $\alpha(1 - \mathbf{x}_u \cdot \mathbf{x}_v)/2$. Since

$$\sum_{(u,v)\in E} \frac{1 - \mathbf{x}_u \cdot \mathbf{x}_v}{2} \geq \rho - \varepsilon,$$

it follows that we return a triangle-free subgraph with expected weight $\alpha(\rho - \varepsilon)$.

Now, we derandomise our algorithm. First, rather than randomly choosing between Case (i) and Case (ii), we can just run both cases and return the better of the two solutions. Second, using the techniques of [MR99] (or, more efficiently [BK05]), we can derandomise the choice of random hyperplane in polynomial time, at the cost of losing ε' potential value, in polynomial time in $\log(1/\varepsilon')$. Hence, in polynomial time in the size of the graph and $\varepsilon, \varepsilon'$ we return a triangle-free subgraph with weight $\alpha(\rho - \varepsilon) - \varepsilon' \geq \alpha\rho - (\varepsilon + \varepsilon')$.

To complete the proof, note that we take $\alpha \geq 0.88232$, but we only advertise an approximation ratio of 0.8823. Hence, the subgraph we return has weight $0.8823\rho + 2 \cdot 10^{-5}\rho - (\varepsilon + \varepsilon')$, and since $\rho \geq |E|/2 \geq 1/2$ (since if $|E| = 0$ the problem is trivial), this is at least $0.8823\rho + (10^{-5} - \varepsilon - \varepsilon')$. Hence it is sufficient to choose $\varepsilon + \varepsilon' = 10^{-5}$. The algorithm then runs in polynomial time in the size of G, and finds a triangle-free subgraph with weight 0.8823ρ, as required. □

It is also interesting to consider what the power of our approximation algorithm is in the *almost satisfiable regime*, i.e. if an input graph that has a cut of value $1 - \varepsilon$. It turns out that in this case we output a triangle-free subgraph with $1 - O(\varepsilon)$ edges, significantly more than the $1 - O(\sqrt{\varepsilon})$ edges outputted by the Goemans-Williamson algorithm [GW95]. This is not very hard to see, it follows immediately from the fact that our algorithm can choose all edges of angle $> \tau$ immediately.

Theorem 9. *The derandomised algorithm from Theorem 8, if run on an input graph G with a cut with a $(1-\varepsilon)$-fraction of edges, produces a triangle-free subgraph with $(1-O(\varepsilon))$-fraction of edges.*

Indeed, the (extremely loose) analysis below gives us that it returns a triangle-free subgraph with a $(1-15\varepsilon)$-fraction of edges at least.

Proof. Suppose we solve the SDP (1) within error ε, hence getting a solution with value at least $(1-\varepsilon)^2 \geq 1-3\varepsilon$ times the number of edges in G. Suppose that of the edges, a proportion of a of them have angle $> \tau$, and a proportion of b of them have angle $\leq \tau$. We will output at least all of the edges with angle $> \tau$, so we must show that $a = 1 - O(\varepsilon)$. Observing that in the worst case all edges counting towards a have angle π and all edges counting towards b have angle τ, we have

$$a + b = 1$$
$$a + b \underbrace{\frac{1-\cos(\tau)}{2}}_{C=C(\tau)} \geq 1 - 3\varepsilon.$$

Note that τ is just some absolute constant, and thus C is also just some absolute constant. Indeed we can approximate $C \approx 0.789$. Subtracting C times the first equation from the second we get $(1-C)a = (1-C) - 3\varepsilon$, and then dividing again by $1-C$ to get $a = 1-(3/(1-C))\varepsilon \geq 1-15\varepsilon = 1-O(\varepsilon)$, as required. □

Interestingly, the non-derandomised algorithm has worse performance! Since it chooses at random between selecting all long edges deterministically and cutting according to the Goemans-Williamson algorithm, the performance degrades to $1 - O(\sqrt{\varepsilon})$.

5 Approximation of MaxPCSP(K_2, K_3)

We first introduce some useful notation. For any predicate φ, we let $[\varphi] = 1$ if φ is true, and 0 otherwise.

For an event φ we let $\Pr[\varphi]$ be the probability that φ is true. For a random variable X, we let $\mathbb{E}[X]$ denote its expected value. Note that $\mathbb{E}[[\varphi]] = \Pr[\varphi]$. For any two distributions $\mathcal{D}, \mathcal{D}'$ with domains A, A', we let $\mathcal{D} \times \mathcal{D}'$ denote the product distribution, whose domain is $A \times A'$. For any distribution $\mathcal{D}$ over $\mathbb{R}$ and $a, b \in \mathbb{R}$, the distribution $a\mathcal{D} + b$ is the distribution of $aX + b$ when $X \sim \mathcal{D}$. We use the standard probability theory abbreviations i.i.d. (independent and identically distributed) and p.m.f. (probability mass function).

We introduce a few classic distributions we will need. The uniform distribution $\mathcal{U}(D)$ over a discrete set D is the distribution with p.m.f. $f : D \to [0,1]$ given by $f(x) = 1/|D|$. Note that $\mathcal{U}(D^n)$ is the same as $\mathcal{U}(D)^n$, a fact which we will use implicitly. We let $\mathrm{NBin}(n)$ denote a normalised binomial distribution: it is the distribution of $X_1 + \cdots + X_n$, where $X_i \sim \mathcal{U}(\{-1/\sqrt{n}, 1/\sqrt{n}\})$. The domain of this distribution is $\{(-n+2k)/\sqrt{n} \mid 0 \leq k \leq n\}$, the probability mass function is $(-n+2k)/\sqrt{n} \mapsto \binom{n}{k}/2^n$, the expectation is 0, and the variance is 1. If $\mu, \sigma \in \mathbb{R}$, then we let $\mathcal{N}(\mu, \sigma^2)$ denote the normal distribution with mean μ and variance σ^2. Fixing d, if $\mu \in \mathbb{R}^d, \Sigma \in \mathbb{R}^{d\times d}$, then we let $\mathcal{N}(\mu, \Sigma)$ denote the multivariate normal distribution with mean μ and covariance matrix Σ. We let $\mathbf{I}_d$ denote the $d \times d$ identity matrix. Observe that if $\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$, where $\mathbf{x} \in \mathbb{R}^d$, then for any matrix $A \in \mathbb{R}^{d' \times d}$ we have that $A\mathbf{x} \sim \mathcal{N}(A\mu, A\Sigma A^T)$. Furthermore if $\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$ with Σ positive semidefinite, then by finding the Cholesky decomposition $\Sigma = \mathbf{A}\mathbf{A}^T$, where $\mathbf{A} \in \mathbb{R}^{d\times d}$, we find that $\mathbf{x}$ is identically distributed to $\mathbf{A}\mathbf{x}' + \mu$, where $\mathbf{x}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$.

Our goal is to prove the following result.

Theorem 5. MaxPCSP(K_2, K_3) *is 1-approximable in polynomial time (in the search version).*

Our proof will be split into three parts: First we prove some technical bounds which we will need. Next, we provide a randomised algorithm. Finally, we derandomise the algorithm.

Technical bounds. For the proof, we will need a technical lemma, stated as Lemma 11 below. The proof of Lemma 11 is an application of the following result of Cheng.

Theorem 10 ([Che68][Che69, Equation (2.18)]). *Suppose* $\mathbf{u} = (u_1, u_2, u_3, u_4) \sim \mathcal{N}(\mathbf{0}, \Sigma)$ *are drawn from a quadrivariate normal distribution with mean zero and covariance matrix*

$$\Sigma = \begin{pmatrix} 1 & a & b & ab \\ a & 1 & ab & b \\ b & ab & 1 & a \\ ab & b & a & 1 \end{pmatrix},$$

where $a, b \in [-1, 1]$*. Then* $\Pr_{\mathbf{u}}[u_1 \geq 0, u_2 \geq 0, u_3 \geq 0, u_4 \geq 0]$ *is*

$$\frac{1}{16} + \frac{\arcsin a + \arcsin b + \arcsin ab}{4\pi} + \frac{(\arcsin a)^2 + (\arcsin b)^2 - (\arcsin ab)^2}{4\pi^2}.$$

Lemma 11. *Fix* $\alpha, \beta \in \mathbb{R}$ *such that* $\alpha^2 + \beta^2 = 1$*. Suppose* $x_1, x_2, x_3, y_1, y_2, y_3 \sim \mathcal{N}(0, 1)$ *i.e. they are i.i.d. standard normal variables. The probability that*

$$\begin{aligned} x_1 &\geq x_2 \\ x_1 &\geq x_3 \\ \alpha x_1 + \beta y_1 &\geq \alpha x_2 + \beta y_2 \\ \alpha x_1 + \beta y_1 &\geq \alpha x_3 + \beta y_3 \end{aligned}$$

is precisely

$$P(\alpha) = \frac{1}{9} + \frac{\arcsin \alpha + \arcsin \frac{\alpha}{2}}{4\pi} + \frac{(\arcsin \alpha)^2 - (\arcsin \frac{\alpha}{2})^2}{4\pi^2}.$$

Proof. Define the following normally distributed random variables:

$$\begin{aligned} u_1 &= (x_1 - x_2)/\sqrt{2}, \\ u_2 &= (x_1 - x_3)/\sqrt{2}, \\ u_3 &= (\alpha x_1 + \beta y_1 - \alpha x_2 - \beta y_2)/\sqrt{2}, \\ u_4 &= (\alpha x_1 + \beta y_1 - \alpha x_3 - \beta y_3)/\sqrt{2}. \end{aligned}$$

Observe that, by simple computation, and since $\alpha^2 + \beta^2 = 1$, we have that $\mathbf{u} = (u_1, u_2, u_3, u_4) \sim \mathcal{N}(\mathbf{0}, \Sigma)$, where

$$\Sigma = \begin{pmatrix} 1 & \frac{1}{2} & \alpha & \frac{\alpha}{2} \\ \frac{1}{2} & 1 & \frac{\alpha}{2} & \alpha \\ \alpha & \frac{\alpha}{2} & 1 & \frac{1}{2} \\ \frac{\alpha}{2} & \alpha & \frac{1}{2} & 1 \end{pmatrix}.$$

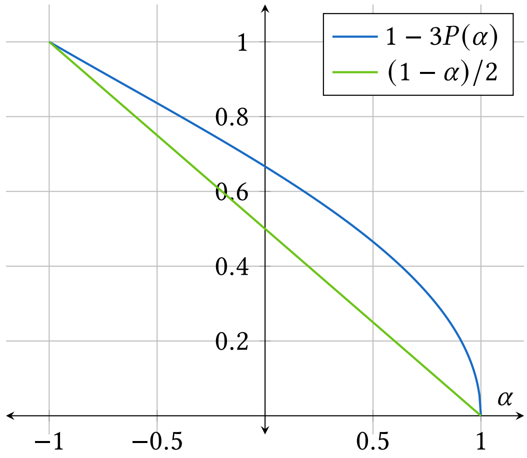


Figure 3: Plot of expressions from Lemma 12.

The probability we want is just $\Pr_{\mathbf{u}}[u_1 \geq 0, u_2 \geq 0, u_3 \geq 0, u_4 \geq 0]$. Apply Theorem 10 with $a = \frac{1}{2}$ and $b = \alpha$ to find that the required probability is

$$\frac{1}{16} + \frac{\arcsin \frac{1}{2} + \arcsin \alpha + \arcsin \frac{\alpha}{2}}{4\pi} + \frac{\left(\arcsin \frac{1}{2}\right)^2 + (\arcsin \alpha)^2 - (\arcsin \frac{\alpha}{2})^2}{4\pi^2} = \frac{1}{9} + \frac{\arcsin \alpha + \arcsin \frac{\alpha}{2}}{4\pi} + \frac{(\arcsin \alpha)^2 - (\arcsin \frac{\alpha}{2})^2}{4\pi^2},$$

as required. □

We will need a bound on the $P(\alpha)$ function from Lemma 11.

Lemma 12. *For* $-1 \leq \alpha \leq -1$, $1 - 3P(\alpha) \geq \frac{1-\alpha}{2}$.

The functions involved are shown in Figure 3.

Proof. Define

$$f(\alpha) = 1 - 3P(\alpha) - \frac{1-\alpha}{2} = \frac{1}{6} + \frac{\alpha}{2} - \frac{3}{4\pi}\left(\arcsin \alpha + \arcsin \frac{\alpha}{2}\right) - \frac{3}{4\pi^2}\left((\arcsin \alpha)^2 - \left(\arcsin \frac{\alpha}{2}\right)^2\right).$$

We want to show that $f(x) \geq 0$ for $x \in [-1, -1]$. First we show that $f(x) \geq 0$ for $x \in [-1, 0]$. Numerically, we can find that

$$\max_{-1<x<1} f'''(x) \approx -0.1454 < 0,$$

at $x \approx -0.5681$. Thus $f''(x)$ is decreasing, and $f'(x)$ is concave. Thus, by Jensen's inequality, for $x \in (-1, 0)$,

$$f'(x) \geq -x \lim_{t \to -1^+} f'(t) + (1+x)f'(0),$$

(as f' is not defined at -1). But $f'(0) \approx 0.1419 > 0$, and $\lim_{t\to-1^+} f'(t) \approx 0.1642 > 0$, so $f'(x) > 0$ for $x \in (-1, 0)$. It follows that f is increasing on $[-1, 0]$, which is sufficient to show that $f(x) \geq 0$ for $x \in [-1, 0]$, as $f(-1) = 0$.

Now, we consider $x \in [0, 1]$. Observe again that we know that $f''(x)$ is decreasing. But since $f''(0) \approx -0.1139 < 0$, it follows that $f''(x) < 0$ for $x \in [0, 1)$; so f is concave on $[0, 1]$. Now, again applying Jensen's inequality, we find that for $x \in [0, 1]$,

$$f(x) \geq xf(0) + (1 - x)f(1).$$

Now, $f(0) = 1/6 > 0$, and $f(1) = 0$, so $f(x) \geq 0$ for $x \in [0, 1]$.

Thus our conclusion follows in all cases. □

Randomised algorithm. We first provide a randomised version of our algorithm, accurate up to some ε.

Theorem 13 (Randomised version of Theorem 5). *There exists a randomised algorithm which, given a graph $G = (V, E)$ that has a cut with ρ edges and an accuracy parameter ε, finds a 3-colouring of G that satisfies $\rho - \varepsilon$ edges in expectation, in polynomial time with respect to the size of G and* $\log(1/\varepsilon)$.

Proof. Our algorithm is as follows. In essence, it solves the SDP of Goemans and Williamson [GW95] i.e. (1), then randomly rounds as the 3-colouring algorithm of Frieze and Jerrum [FJ97].

1. Input: a graph $G = (V, E)$, which admits a cut of weight ρ. Suppose $n = |V|$.
2. Solve SDP (1) to within error ε. Thus we get a set of vectors $\mathbf{x}_v \in \mathbb{R}^n$ for $v \in V$, with $||\mathbf{x}_v||^2 = 1$ and
$$\sum_{(u,v)\in E} \frac{1 - \mathbf{x}_u \cdot \mathbf{x}_v}{2} \geq \rho - \varepsilon.$$
3. Sample three i.i.d. normally distributed n-dimensional vectors $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$.
4. Set the colour of node u to $\arg\max_i \mathbf{x}_u \cdot \mathbf{a}_i$. (Break ties arbitrarily.)

Let us compute the expected number of edges which are satisfies by this 3-colouring. Consider an edge $(u, v) \in E$; in terms of $\frac{1-\mathbf{x}_u \cdot \mathbf{x}_v}{2}$, what is the probability that (u, v) is properly coloured? This is the same as the probability that $\arg\max_i \mathbf{x}_u \cdot \mathbf{a}_i \neq \arg\max_i \mathbf{x}_v \cdot \mathbf{a}_i$, which, by symmetry, is equal to

$$1 - 3 \Pr_{\mathbf{a}_1,\mathbf{a}_2,\mathbf{a}_3} [\mathbf{x}_u \cdot \mathbf{a}_1 \geq \mathbf{x}_u \cdot \mathbf{a}_2, \mathbf{x}_u \cdot \mathbf{a}_1 \geq \mathbf{x}_u \cdot \mathbf{a}_3, \mathbf{x}_v \cdot \mathbf{a}_1 \geq \mathbf{x}_v \cdot \mathbf{a}_2, \mathbf{x}_v \cdot \mathbf{a}_1 \geq \mathbf{x}_v \cdot \mathbf{a}_3]. \tag{7}$$

Since $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are selected from a rotationally symmetric distribution, we can rotate everything to be in a 2-dimensional plane without affecting the probability in (7). Furthermore, we can rotate so that $\mathbf{x}_u$ is moved to $(1, 0)$, and $\mathbf{x}_v$ is at (α, β), where $\alpha = \mathbf{x}_u \cdot \mathbf{x}_v$ and $\beta = \sqrt{1 - \alpha^2}$ (note that this rotation is possible since it preserves the angle between $\mathbf{x}_u$ and $\mathbf{x}_v$, and their lengths). Since (the projection to this plane of) the vectors $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are bivariate standard normal variables, we can see them as pairs $(a_1, b_1), (a_2, b_2), (a_3, b_3)$, where $a_1, a_2, a_3, b_1, b_2, b_3 \sim \mathcal{N}(0, 1)$ are i.i.d. standard normal variables. Then, we can rewrite (7) as

$$1 - 3 \Pr_{\substack{a_1,a_2,a_3\\b_1,b_2,b_3}} [a_1 \geq a_2, a_1 \geq a_3, \alpha a_1 + \beta b_1 \geq \alpha a_2 + \beta b_2, \alpha a_1 + \beta b_1 \geq \alpha a_3 + \beta b_3]. \tag{8}$$

By Lemma 11, the expression in (8) is $1 - 3P(\alpha)$, which is, by Lemma 12, at least $\frac{1-\alpha}{2} = \frac{1-\mathbf{x}_u \cdot \mathbf{x}_v}{2}$. Thus, by linearity of expectation, the expected number of edges we satisfy is at least $\sum_{(u,v)\in E} \frac{1-\mathbf{x}_u \cdot \mathbf{x}_v}{2} \geq \rho - \varepsilon$, as required. □

Derandomised algorithm. We will now show how to derandomise our algorithm, using the method of conditional expectations, which was also used by Mahajan and Ramesh [MR99]. The approach of Bhargava and Kosaraju [BK05] derandomises conditional probabilities by an approximation of normal distributions via polynomials; we approximate simply just with a scaled bionomial distribution. We believe that the results of the literature are sufficient to prove the derandomisation theorem we need (which crucially needs to work for our 1-approximation setting); however we propose a simpler derandomisation method that we believe will be easier to apply in general. (Indeed, our method avoids integration alltogether.) There is an interesting duality between our approach and that of Mahajan and Ramesh: we discretise the normal distribution, whereas they discretise the SDP vectors.

Our goal will be the following general derandomisation theorem.

Theorem 14. *Fix a constant d. There exists an algorithm that does the following. Suppose we are given $n, m \in \mathbb{N}$, $\mathbf{x}_{ij} \in \mathbb{R}^n$ and $y_{ij}, \varepsilon \in \mathbb{R}$ for all $i \in [m]$, $j \in [d]$. Suppose $\mathbf{a} = (a_1, \dots, a_n) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ and that*

$$\sum_{i=1}^{m} \Pr_{\mathbf{a}}\left[\bigwedge_{j=1}^{d} \mathbf{x}_{ij} \cdot \mathbf{a} > y_{ij}\right] \geq \rho$$

for some $\rho \in \mathbb{R}$. Then the algorithm computes some particular $\mathbf{a}^* = (a_1^*, \dots, a_n^*) \in \mathbb{R}^n$ such that

$$\sum_{i=1}^{m} \left[\bigwedge_{j=1}^{d} \mathbf{x}_{ij} \cdot \mathbf{a}^* > y_{ij}\right] \geq \rho - \varepsilon,$$

in polynomial time with respect to $n, m, 1/\varepsilon$.

To facilitate the proof of Theorem 14, we will need a multidimensional version of the Berry-Esseen theorem. We will use the following version with explicit constants, due to Raič [Rai19].

Theorem 15 ([Rai19, Theorem 1.1]). *Suppose $\mathbf{t}_1, \dots, \mathbf{t}_N \in \mathbb{R}^d$ are independent random variables with mean zero, such that the sum of their covariance matrices is $\mathbf{I}_d$. Let $\mathbf{s} = \mathbf{t}_1 + \cdots + \mathbf{t}_N$. Suppose $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$, and let $C \subseteq \mathbb{R}^d$ be convex and measurable. Then*

$$|\Pr_{\mathbf{s}}[\mathbf{s} \in C] - \Pr_{\mathbf{a}}[\mathbf{a} \in C]| \leq \left(42\sqrt[4]{d} + 16\right) \sum_{i=1}^{N} \mathbf{E}\left[||\mathbf{t}_i||^3\right].$$

The following is an easy and well-known corollary of Theorem 15: We can approximate a multivariate normal distribution with binomial distributions. For completeness, we provide a proof.

Corollary 16. *Let $d \in \mathbb{N}$ be a constant and take $\varepsilon \in (0, 1)$. Take*

$$N = N_\varepsilon \geq \left(\frac{42d^{7/4} + 16d^{3/2}}{\varepsilon}\right)^2 = \frac{\xi_d}{\varepsilon^2}, \tag{9}$$

where $\xi_d = O(d^{7/2})$ depends only on d. Suppose $s_1, \dots, s_N \sim \mathrm{NBin}(N)$ are i.i.d., and let $\mathbf{s} = (s_1, \dots, s_d)$. Let $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. Then for all convex measurable sets $C \subseteq \mathbb{R}^d$ we have

$$|\Pr_{\mathbf{s}}[\mathbf{s} \in C] - \Pr_{\mathbf{a}}[\mathbf{a} \in C]| \leq \varepsilon.$$

Proof. Note that each component of $\mathbf{s}$ is i.i.d. and distributed as the sum of N independent trials that take value $\pm 1/\sqrt{N}$ equiprobably. In other words, we can see $\mathbf{s}$ as the sum $\mathbf{t}_1 + \cdots + \mathbf{t}_N$, where $\mathbf{t}_1, \ldots, \mathbf{t}_N \sim \mathcal{U}(\{-1/\sqrt{N}, 1/\sqrt{N}\}^d)$ are i.i.d. Observe that the covariance matrix of $\mathbf{t}_i$ is $\mathbf{I}_d/N$, so the sum of these covariance matrices for all i is $\mathbf{I}_d$. Furthermore $||\mathbf{t}_i|| = \sqrt{(\pm 1/\sqrt{N})^2 + \cdots + (\pm 1/\sqrt{N})^2} = \sqrt{d/N}$ with probability 1. Now, apply Theorem 15 to $\mathbf{s} = \sum_{i=1}^N \mathbf{t}_i$. We find that

$$|\Pr_{\mathbf{s}}[\mathbf{s} \in C] - \Pr_{\mathbf{a}}[\mathbf{a} \in C]| \leq \left(42\sqrt[4]{d} + 16\right) \sum_{i=1}^{N} \mathbb{E}\left[||\mathbf{t}_i||^3\right] = \left(42\sqrt[4]{d} + 16\right) N \sqrt{\frac{d}{N}}^3 = \frac{42d^{7/4} + 16d^{3/2}}{\sqrt{N}}.$$

Substituting (9), it follows that $|\Pr_{\mathbf{s}}[\mathbf{s} \in C] - \Pr_{\mathbf{a}}[\mathbf{a} \in C]| \leq \varepsilon$, as required. □

Theorem 17. *Fix a constant d, and take $\mathbf{x}_1, \ldots, \mathbf{x}_d \in \mathbb{R}^n$, $y_1, \ldots, y_d \in \mathbb{R}$, $z_1, \ldots, z_d \in \mathbb{R}$, $\varepsilon \in \mathbb{R}$. Consider the function*

$$p(t) = \Pr_{\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)}\left[\bigwedge_{i=1}^{d} \mathbf{x}_i \cdot \mathbf{a} + z_i t > y_i\right].$$

There exists a step function $\widehat{p}$ with $\mathrm{poly}(1/\varepsilon)$ steps, where the steps and the values at those steps are computable in polynomial time with respect to $1/\varepsilon$ and n, such that $|\widehat{p}(t) - p(t)| \leq \varepsilon$ for all $t \in \mathbb{R}$.

Proof. Observe that the tuple $(\mathbf{x}_1 \cdot \mathbf{a}, \ldots, \mathbf{x}_d \cdot \mathbf{a})$ (interpreted as a column vector) is a d-variate normally distributed vector; namely, if we let

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_d^T \end{pmatrix}$$

be a block matrix whose rows are $\mathbf{x}_1^T, \ldots, \mathbf{x}_d^T$, then $(\mathbf{x}_1 \cdot \mathbf{a}, \ldots, \mathbf{x}_d \cdot \mathbf{a}) = \mathbf{X}\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{X}\mathbf{I}_D\mathbf{X}^T) = \mathcal{N}(\mathbf{0}, \mathbf{X}\mathbf{X}^T)$.

We can compute the covariance matrix, namely $\mathbf{X}\mathbf{X}^T$, in polynomial time with respect to n. Now, by computing the Cholesky decomposition of this positive semidefinite matrix, we can find $\mathbf{X}' \in \mathbb{R}^{d \times d}$ such that $\mathbf{X}'\mathbf{X}'^T = \mathbf{X}\mathbf{X}^T$. Thus $(\mathbf{x}_1 \cdot \mathbf{a}, \ldots, \mathbf{x}_d \cdot \mathbf{a}) \sim \mathcal{N}(\mathbf{0}, \mathbf{X}'\mathbf{X}'^T)$. Letting $\mathbf{x}_1'^T, \ldots, \mathbf{x}_d'^T$ be the rows of $\mathbf{X}'$, we find that $(\mathbf{x}_1 \cdot \mathbf{a}, \ldots, \mathbf{x}_d \cdot \mathbf{a})$ is identically distributed to $(\mathbf{x}_1' \cdot \mathbf{a}', \ldots, \mathbf{x}_d' \cdot \mathbf{a}')$, when $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ and $\mathbf{a}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$, since both follow the distribution $\mathcal{N}(0, \mathbf{X}'\mathbf{X}'^T)$. Thus,

$$p(t) = \Pr_{\mathbf{a}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)}\left[\bigwedge_{i=1}^{d} \mathbf{x}_i' \cdot \mathbf{a}' + z_i t > y_i\right].$$

In other words, we have reduced the dimensionality of our problem from n to d, a constant.

Note that the set defined by $\bigwedge_{i=1}^d \mathbf{x}_i' \cdot \mathbf{a}' + z_i t > y_i$ is necessarily convex and measurable, being the intersection of finitely many half-spaces. Thus we can apply Corollary 16. Let $N = N_\varepsilon$ and $s_1, \ldots, s_N \sim \mathrm{NBin}(N)$, and suppose $\mathbf{s} = (s_1, \ldots, s_d)$. Then, we know that

$$\left|p(t) - \Pr_{\mathbf{s}}\left[\bigwedge_{i=1}^{d} \mathbf{x}_i' \cdot \mathbf{s} + z_i t > y_i\right]\right| \leq \varepsilon.$$

This suggests using the following definition:

$$\widehat{p}(t) = \Pr_{\mathbf{s}}\left[\bigwedge_{i=1}^{d} \mathbf{x}_i' \cdot \mathbf{s} + z_i t > y_i\right],$$

as this must satisfy the condition $|\widehat{p}(t) - p(t)| \le \varepsilon$. What remains is to show that $\widehat{p}$ is a step function, and that these steps can be efficiently computed.

Intuitively, this is the case since the probability distribution we define $\widehat{p}$ over is discrete. More precisely, letting $D = \{-N/\sqrt{N}, (-N+2)/\sqrt{N}, \ldots, (N-2)/\sqrt{N}, N/\sqrt{N}\}^d$ be the domain of $\mathbf{s}$, and letting q be the p.m.f. of $\mathbf{s}$ (note that it can be efficiently computed, since $\mathbf{s}$ is essentially distributed according to a product distribution of normalised binomials), we note that

$$\widehat{p}(t) = \Pr_{\mathbf{s}}\left[\bigwedge_{i=1}^{d} \mathbf{x}'_i \cdot \mathbf{s} + z_i t > y_i\right] = \sum_{\mathbf{u}\in D} q(\mathbf{u}) \left[\bigwedge_{i=1}^{d} \mathbf{x}'_i \cdot \mathbf{u} + z_i t > y_i\right]$$
$$= \sum_{\mathbf{u}\in D} q(\mathbf{u}) \prod_{i=1}^{d} \left[\mathbf{x}'_i \cdot \mathbf{u} + z_i t > y_i\right].$$

Now observe that, for each $\mathbf{u} \in D$, the function

$$t \mapsto \left[\mathbf{x}'_i \cdot \mathbf{u} + z_i t > y_i\right]$$

is a step function with at most one step: if $z_i = 0$ then the function is a constant (whose value is easy to compute); otherwise the step is at $(1/z_i)(y_i - \mathbf{x}'_i \cdot \mathbf{u})$, where the step being increasing or decreasing is determined by the sign of z_i (and again the values of the function are easy to compute). It therefore follows that $\widehat{p}$ is a step function that has at most $|D| = dN_\varepsilon^d = \mathsf{poly}(1/\varepsilon)$ steps, and that each of the values that the function takes can be computed in polynomial time with respect to $1/\varepsilon$ and n. □

Proof of Theorem 14. We give a recursive algorithm. If $n = 0$ then there is nothing to output, so assume $n \ge 1$. Let $\mathbf{a} = (t, \mathbf{a}')$, and $\mathbf{x}_{ij} = (z_{ij}, \mathbf{x}'_{ij})$ — in other words, separate out the first variable. We are given that

$$\sum_{i=1}^{m} \Pr_{\mathbf{a}',t}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' + z_{ij} t > y_{ij}\right] = \sum_{i=1}^{m} \Pr_{\mathbf{a}}\left[\bigwedge_{j=1}^{d} \mathbf{x}_{ij} \cdot \mathbf{a} > y_{ij}\right] \ge \rho,$$

when $\mathbf{a} = (\mathbf{a}', t) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. But then this must be true for some particular value of t, say t^*, i.e.

$$\sum_{i=1}^{m} \Pr_{\mathbf{a}'}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' + z_{ij} t^* > y_{ij}\right] \ge \rho. \tag{10}$$

Apply Theorem 17 to each of the probabilities above viewed as functions of t, with $\varepsilon' = \varepsilon/2nm$; we thus build step functions $p_1, \ldots, p_m$ in polynomial time with respect to n and $1/\varepsilon' = 2nm/\varepsilon$, such that

$$\left|p_i(t) - \Pr_{\mathbf{a}'}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' + z_{ij} t > y_{ij}\right]\right| \le \frac{\varepsilon}{2nm}$$

for all t. Add these equations for $i = 1, \ldots, m$ to find

$$\left|\sum_{i=1}^{m} p_i(t) - \sum_{i=1}^{m} \Pr_{\mathbf{a}'}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' + z_{ij} t > y_{ij}\right]\right| \le \frac{\varepsilon}{2n}. \tag{11}$$

Observe that $\sum_{i=1}^{m} p_i$ is a step function with polynomially many steps with respect to n, m, whose values are also computable in polynomial time. Thus it is easy to find some value $\hat{t}$ that maximises the expression $\sum_{i=1}^m p_i(\hat{t})$. By (10) and (11) we have that $\sum_{i=1}^m p_i(\hat{t}) \geq \sum_{i=1}^m p_i(t^*) \geq \rho - \frac{\varepsilon}{2n}$, and by (11) again we find that

$$\sum_{i=1}^{m} \Pr_{\mathbf{a}'}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' + z_{ij}\hat{t} > y_{ij}\right] \geq \rho - \frac{\varepsilon}{n}.$$

Equivalently,

$$\sum_{i=1}^{m} \Pr_{\mathbf{a}'}\left[\bigwedge_{j=1}^{d} \mathbf{x}'_{ij} \cdot \mathbf{a}' > y_{ij} - z_{ij}\hat{t}\right] \geq \rho - \frac{\varepsilon}{n},$$

and we can recursively find optimal values for the remaining random variables in $\mathbf{a}'$. Observe that our recursive depth is n, that at each level we use polynomial time with respect to $n, m, 1/\varepsilon$, and that, finally, at each step we lose ε/n from the sum of our probabilities. These facts together imply the correctness of our general derandomisation procedure.

We note in passing that the total time complexity of our method is exponential in d; however this does not matter, as we consider d a constant. □

This is enough to derandomise our algorithm.

Proof of Theorem 5. Let $G = (V, E)$, where $V = [n]$ (without loss of generality) and $m = |E|$. Suppose this graph has a cut of size ρ. By the analysis of our randomised algorithm from Theorem 4 with $\varepsilon = 1/3$, using SDP we can find, in polynomial time with respect to G, a set of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ such that $||\mathbf{x}_i||^2 = 1$ and, if $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ are normally distributed variables, then

$$\sum_{(u,v)\in E} \Pr_{\mathbf{a}_1,\mathbf{a}_2,\mathbf{a}_3}\left[\arg\max_i \mathbf{a}_k \cdot \mathbf{x}_u \neq \arg\max_i \mathbf{a}_i \cdot \mathbf{x}_v\right] \geq \rho - \frac{1}{3}.$$

Now, let $\mathbf{a} = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{3n})$, and define $\mathbf{x}_{ui}$ such that $\mathbf{a} \cdot \mathbf{x}_{ui} = \mathbf{a}_i \cdot \mathbf{x}_u$; in other words, pad out $\mathbf{x}_u$ with $2n$ zeroes. In what follows, let $\oplus$ denote addition mod 3 over $[3]$. We first claim that the event

$$\arg\max_i \mathbf{a}_i \cdot \mathbf{x}_u \neq \arg\max_i \mathbf{a}_i \cdot \mathbf{x}_v,$$

can be seen as the disjoint union of 6 intersections of 4 half spaces in the space of $\mathbf{a}$. To express it in this way, first fix the value of the respective sides to $c \neq c'$, where $c, c' \in [3]$. Observe that the event that $\arg\max_i \mathbf{a}_i \cdot \mathbf{x}_u = c$ is the same as $\mathbf{a}_c \cdot \mathbf{x}_u > \mathbf{a}_{c\oplus 1} \cdot \mathbf{x}_u \wedge \mathbf{a}_c \cdot \mathbf{x}_u > \mathbf{a}_{c\oplus 2} \cdot \mathbf{x}_u$. Now, using the notation from before, this is equivalent to $\mathbf{a} \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 1}) > 0 \wedge \mathbf{a} \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 2}) > 0$. It follows that

$$\begin{aligned}\rho - \frac{1}{3} \leq \sum_{(u,v)\in E} \Pr_{\mathbf{a}_1,\mathbf{a}_2,\mathbf{a}_3}&\left[\arg\max_i \mathbf{a}_i \cdot \mathbf{x}_u \neq \arg\max_i \mathbf{a}_i \cdot \mathbf{x}_v\right] \\ &= \sum_{(u,v)\in E} \sum_{c\neq c'} \Pr_{\mathbf{a}}[\mathbf{a} \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 1}) > 0 \wedge \mathbf{a} \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 2}) > 0 \\ &\qquad \wedge \mathbf{a} \cdot (\mathbf{x}_{vc'} - \mathbf{x}_{v,c'\oplus 1}) > 0 \wedge \mathbf{a} \cdot (\mathbf{x}_{vc'} - \mathbf{x}_{v,c'\oplus 2}) > 0].\end{aligned}$$

By Theorem 14 with $\varepsilon = \frac{1}{3}$, we can find particular values $\mathbf{a}^*$ such that

$$\begin{aligned}\sum_{(u,v)\in E} \sum_{c\neq c'} &[\mathbf{a}^* \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 1}) > 0 \wedge \mathbf{a}^* \cdot (\mathbf{x}_{uc} - \mathbf{x}_{u,c\oplus 2}) > 0 \\ &\wedge \mathbf{a}^* \cdot (\mathbf{x}_{vc'} - \mathbf{x}_{v,c'\oplus 1}) > 0 \wedge \mathbf{a}^* \cdot (\mathbf{x}_{vc'} - \mathbf{x}_{v,c'\oplus 2}) > 0] \geq \\ &\geq \rho - \frac{1}{3} - \frac{1}{3} = \rho - \frac{2}{3}.\end{aligned}$$

Defining $(\mathbf{a}_1^*, \mathbf{a}_2^*, \mathbf{a}_3^*) = \mathbf{a}^*$, this is equivalent to

$$\sum_{(u,v)\in E} [\arg\max_i \mathbf{a}_i^* \cdot \mathbf{x}_u \neq \arg\max_i \mathbf{a}_i^* \cdot \mathbf{x}_v] \geq \rho - \frac{2}{3}.$$

In other words, if we set the colour of vertex u to $\arg\max_i \mathbf{a}_i^* \cdot \mathbf{x}_u$, then we will correctly colour at least $\rho - 2/3$ edges. Now, note that $\rho \in \mathbb{N}$. Since the lumber of correctly coloured edges must also be an integer, and it is at least $\rho - 2/3$, it follows that it is at least ρ. Thus our algorithm returns a 3-colouring of value ρ, as required. □

6 Hardness of MaxPCSP$(K_2, \mathfrak{G}_3)$

In this section, we will prove the hardness part of the following result.

Theorem 4. MaxPCSP$(K_2, \mathfrak{G}_3)$ *is* 0.8823*-approximable (in the search version) in polynomial time, and it is* NP*-hard to* $(25/26 + \varepsilon)$*-approximate (even in the decision version) for any fixed* $\varepsilon > 0$.

Our general strategy will be to gadget reduce from the 3-bit PCP of Håstad [Hås01], similarly to [TSSW00] or [BGS98]. The main difficulty comes in from the fact that it is not possible to "negate" variables in an obvious way, since "negation" is not globally preserved by the property of being triangle-free, as opposed to that of being bipartite. Some mild complications will be forced by this. Recall first the definition of exactly-3 linear equations.

Definition 18. In the problem E3Lin$_\delta$, one is given a system of mod-2 linear equations with exactly 3 variables per equation; i.e. $x+y+z \equiv 0 \bmod 2$ or $x+y+z \equiv 1 \bmod 2$. If it is possible to simultaneously solve a $1-\delta$ fraction of all the equations, one must answer Yes; otherwise, if it is not even possible to simultaneously solve a $\frac{1}{2}+\delta$ fraction of the equations, one must answer No.

Theorem 19 ([Hås01]). *For every small enough* δ*, the problem* E3Lin$_\delta$ *is* NP*-hard.*

To deal with our negation problems, we will need a "balanced" version of this problem.

Definition 20. In the problem BalancedE3Lin$_\delta$, one is given a system of mod-2 linear equations with exactly 3 variables per equation; i.e. $x+y+z \equiv 0 \bmod 2$ or $x+y+z \equiv 1 \bmod 2$. Furthermore, the number of equations of the two types is equal. A *balanced solution* to such a system of equations is one that satisfies exactly as many equations of form $x+y+z \equiv 0 \bmod 2$ as those of form $x+y+z \equiv 1 \bmod 2$.

If it is possible to find a balanced solution that satisfies a $1-\delta$ fraction of all the equations, one must answer Yes; otherwise, it if is not even possible to find any (*possibly even unbalanced*) solution that satisfies a $\frac{1}{2}+\delta$ fraction of the equations, one must answer No.

We believe that [Hås01] proves, without being explicit about it, NP-hardness of BalancedE3Lin$_\delta$, although it is not straightforward to see it from the proof in [Hås01]. For completeness, we provide a simple, self-contained reduction.

Lemma 21. *For every small enough* δ*, the problem* BalancedE3Lin$_\delta$ *is* NP*-hard.*

Proof. We reduce from E3Lin$_\delta$ to BalancedE3Lin$_\delta$.

Reduction. Given a system of m equations $\mathcal{E}$ on n variables, which contains the equations $x_i + y_i + z_i \equiv p_i \bmod 2$ for $i \in [m]$,[6] we define the system of equations $\mathcal{E}'$ on n variables, which contains the equations $x_i' + y_i' + z_i' \equiv 1 - p_i \bmod 2$ for $i \in [m]$. We then return the system of equations $\mathcal{E} \sqcup \mathcal{E}'$ i.e. the disjoint union of the two systems.

Completeness. Suppose that $\mathcal{E}$ has a solution $x_i \mapsto c(x_i)$ that satisfies a $1 - \delta$ fraction of the equations. Then the system $\mathcal{E} \sqcup \mathcal{E}'$ has a balanced solution that also satisfies a $1 - \delta$ fraction of its equations, namely the one that sends x_i to $c(x_i)$ and x_i' to $1 - c(x_i)$. This solution is balanced since every equation $x_i + y_i + z_i \equiv p_i \bmod 2$ that it solves within $\mathcal{E}$ can be paired up with an equation $x_i' + y_i' + z_i' \equiv 1 - p_i \bmod 2$ which is solved in $\mathcal{E}'$.

Soundness. Suppose $\mathcal{E} \sqcup \mathcal{E}'$ has a solution $x_i \mapsto c(x_i)$, $x_i' \mapsto d(x_i')$, which satisfies a $\frac{1}{2} + \delta$ fraction of the equations. Such a solution must either satisfy a $\frac{1}{2} + \delta$ fraction of the equations within $\mathcal{E}$, or a $\frac{1}{2} + \delta$ fraction of the equations within $\mathcal{E}'$. In the first case, c is the required solution for the original problem; in the second, $x_i \mapsto 1 - d(x_i')$ is the required solution. □

We now define the notion of "gadget" that we will need for this particular reduction. This is along the same lines as [BGS98, TSSW00], but (i) generalised to deal with promise problems and (ii) specialised to our particular promise problem.

For the following, if $G = (V, E)$ is any bipartite graph, and $V' \subseteq V$, then we say that a function $c : V' \to \{0, 1\}$ is compatible with G if it is possible to extend c into a 2-colouring of G.

Definition 22. A gadget with performance $\alpha \in \mathbb{N}$ and parity $p \in \{0, 1\}$ is a graph $G = (V, E)$, with $0, x, y, z \in V$, where the following hold.

1. For any function $c : \{0, x, y, z\} \to \{0, 1\}$ such that $c(0) + c(x) + c(y) + c(z) \equiv p \bmod 2$, there exists a bipartite subgraph H of G with α edges, such that c is compatible with H.

2. Any triangle-free subgraph of G has at most α edges.

3. Every triangle-free subgraph H of G with strictly more than $\alpha - 1$ edges puts $0, x, y, z$ in the same connected component C, and the distance from 0 to x, y, z respectively is at most 2. Furthermore C is bipartite, and for any $c : \{0, x, y, z\} \to \{0, 1\}$ that is compatible with C, we have that $c(0) + c(x) + c(y) + c(z) \equiv p \bmod 2$.

Lemma 23. *Suppose we have n containers with capacities $c_1, \ldots, c_n \geq 0$. Suppose we distribute a volume of $c_1 + \cdots + c_n - n + a$ among the containers, distributing $v_i \leq c_i$ volume to container i. Then the number of containers i for which $v_i > c_i - 1$ is at least a.*

Proof. If we want to minimise the number of containers with $v_i > c_i - 1$, it is optimal first to distribute $c_i - 1$ volume to each container i, and then for those containers where we are forced to distribute more to distribute up to c_i. The first step uses up $c_1 + \cdots + c_n - n$ volume, leaving us with a volume to distribute. This volume of a cannot push fewer than $\lceil a \rceil$ of the containers above $c_i - 1$, since once pushed above this one can then add only 1 further unit to that container. □

The next theorem encodes our reduction from the 3-bit PCP of [Hås01] to MaxPCSP$(K_2, \mathfrak{G}_3)$. This reduction is standard, needing only some care to deal with the fact that the triangle-free graph selected in the soundness case must be "connected enough".

[6] As usual, we denote by $[m]$ the set $\{1, \ldots, m\}$.

Theorem 24. *Suppose that for $i \in \{0,1\}$ there exist gadgets G_i with performance α_i and parity i. Then it is* NP*-hard to approximate* $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ *with approximation ratio* $1 - 1/(\alpha_0 + \alpha_1) + \varepsilon$.

Proof. We reduce $\mathsf{BalancedE3Lin}_\delta$ to approximate $\mathrm{MaxPCSP}(K_2, \mathfrak{G}_3)$ with approximation ratio $1 - 1/(\alpha_0 + \alpha_1) + \varepsilon$. Our choice of δ will be bounded above by a value that depends on ε. We now describe our gadget reduction.

Reduction. Suppose we are given an instance of $\mathsf{BalancedE3Lin}_\delta$, with $2m$ constraints and n variables V. Suppose that the constraints are $x_i^0 + y_i^0 + z_i^0 \equiv 0 \bmod 2$ and $x_i^1 + y_i^1 + z_i^1 \equiv 1 \bmod 2$ for $i \in [m]$. Our reduction first creates $n+1$ vertices, one for every variable, plus a special vertex denoted by $0'$. For every constraint $x_i^p + y_i^p + z_i^p = p \bmod 2$ we create a copy of G_p, identifying vertices $0, x, y, z$ with $0', x_i^p, y_i^p, z_i^p$.

Completeness. Suppose that our $\mathsf{BalancedE3Lin}$ instance has a balanced solution that satisfies at least a $1 - \delta$ fraction of the constraints. We claim that the graph outputted by our algorithm has a cut with at least $m(1-\delta)(\alpha_0 + \alpha_1)$ edges. Indeed, this cut is guaranteed by Item 1 in Definition 22: place the vertices corresponding to variables in the original problem in the cut according to their value; then the vertices in the gadgets are placed according to the cut guaranteed by this assumption. The cut then has at least $m(1-\delta)(\alpha_0 + \alpha_1)$ edges because the solution is guaranteed to be balanced.

Soundness. Suppose that the graph outputted by our reduction has a triangle-free subgraph S with $m(\alpha_0 + \alpha_1 - 1 + 2\delta) = m(\alpha_0 + \alpha_1) - 2m + m + 2m\delta$ edges. We call a constraint with parity i "good" if the intersection of S with the gadget corresponding to that constraint has weight greater than $\alpha_i - 1$. Observe that at least $m + 2m\delta$ constraints must be good by Lemma 23: the gadgets for the $2m$ constraints are the containers; by Item 2 from Definition 22, m of the capacities are α_0, and m are α_1; finally, a constraint is "good" if it's container is allocated strictly more volume than it's capacity minus one. We now show how to create a solution to the original $\mathsf{BalancedE3Lin}$ instance that satisfies all the good constraints: this solution then satisfies a $(m + 2m\delta)/2m = 1/2 + \delta$ fraction of the constraints.

To create our solution, consider the subgraph S. Find the shortest path from $0'$ to every variable, then set the variable to the parity of the length of that path. (If there is no path, then we can set that variable arbitrarily.) Consider now any good constraint: suppose it is $x + y + z \equiv p \bmod 2$. Suppose S' is the intersection of S with the gadget for this constraint. Since S is triangle-free, S' must be triangle-free; since the constraint is good, the weight of S' is strictly greater than $\alpha_i - 1$. By Item 3 of Definition 22, S' is bipartite; furthermore S' connects $0'$ to x, y, z by paths of length at most 2. Thus the shortest path from $0'$ to x, y, z *in S (as well as S')* is also of length at most 2 as well. Since S is triangle-free, the parity of the length of the shortest path from $0'$ to x, y, z in S matches the parity of any path from $0'$ to x, y, z in S'. (To see why: if this were not the case, then for one of x, y, z, there are two paths from $0'$ of different parities, both of length at most 2. The first path is the one that exists in S by assumption, the second one is the one that exists in S' by Item 3 of Definition 22. This implies a triangle in S, a contradiction.) Colouring according to the sides of the bipartite graph S' satisfies the constraint according to Item 3 of Definition 22, and our colouring is the same as this. So all good constraints are satisfied.

Hardness factor. Thus we have shown that given a $\mathsf{BalancedE3Lin}$ instance with a balanced solution that satisfies a $1 - \delta$ fraction of the equations, our reduction yields a graph with a cut with $m(1-\delta)(\alpha_0+\alpha_1)$ edges; and if the graph we output has a triangle-free subgraph with at least $m(\alpha_0+\alpha_1-1+2\delta)$ edges the original instance must have had a solution that satisfies at least a $1/2+\delta$ fraction of equations. It follows that MaxPCSP$(K_2, \mathfrak{G}_3)$ is NP-hard to approximate with approximation ratio

$$\frac{m(\alpha_0+\alpha_1-1+2\delta)}{m(1-\delta)(\alpha_0+\alpha_1)} = \left(1 - \frac{1}{\alpha_0+\alpha_1} + O(\delta)\right)(1+O(\delta)) = 1 - \frac{1}{\alpha_0+\alpha_1} + O(\delta).$$

This can be made to be less than $1 - \frac{1}{\alpha_0+\alpha_1} + \varepsilon$ by setting δ small enough. □

We now exhibit the gadgets. The first gadget is identical to a gadget of Bellare, Goldreich and Sudan [BGS98] (although our analysis is slightly more complicated). In [BGS98], this gadget is called "PC-CUT", defined immediately before [BGS98, Claim 4.17]. The second gadget is a generalisation of the first. Recall that the gadgets of [BGS98] were improved in [TSSW00], and indeed the results of [TSSW00] indicate a generic method to find optimal gadgets for finite-domain CSPs. We do not believe this approach directly applies to our case because the property of being triangle-free is not captured by any finite CSP template (indeed, $\mathfrak{G}_3$ is infinite, and any homomorphism-equivalent structure must also be).

We will write our gadgets as graphs with non-negative integer weights for simplicity of presentation. These gadgets can then be implemented by adding edges multiple times.

Lemma 25. *There exists a gadget with performance 9 and parity 1.*

Proof. Consider the complete graph on $\{0, a, x, y, z\}$. Suppose $w(a) = 2, w(0) = w(x) = w(y) = w(z) = 1$, and give an edge (i, j) weight $w(i, j) = w(i)w(j)$. We now show that this satisfies the conditions from Definition 22.

1. Without loss of generality suppose $c(0) = 0$. We have two cases. First, suppose that $c(x) = c(y) = c(z) = 1$. Then our cut is $(\{x, y, z\}, \{a, 0\})$ which has weight 9. Conversely, suppose without loss of generality that $c(x) = c(y) = 0, c(z) = 1$. Then our cut is $(\{0, x, y\}, \{a, z\})$, which also has weight 9.

2. Consider any triangle-free subgraph of the gadget. This graph is either bipartite or it is C_5. If it is bipartite, suppose it has parts (A, B). Then

$$\sum_{i\in A, j\in B} w(i, j) = \left(\sum_{i\in A} w(i)\right)\left(\sum_{j\in B} w(j)\right),$$

   and the optimal cut puts a (with weight 2) and one other vertex on one side, and the other three vertices on the other side — this cut has weight 9, as required (all other cuts have weight at most 8). On the other hand, if the triangle-free subgraph is C_5 it has weight at most $2+2+1+1+1 = 7 \leq 9$.

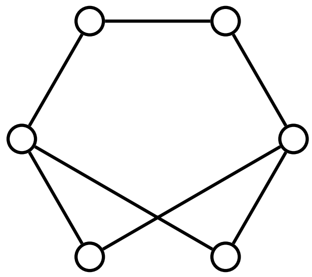
3. Consider any triangle-free subgraph of the gadget. By the analysis in the previous item, the only triangle-free subgraphs with weight greater than 8 are isomorphic to $K_{2,3}$, which put a and one of $0, x, y, z$ on one side, and the other three vertices on the other side. It is not difficult to check that all of these graphs connect 0 to x, y, z with paths of length at most 2, and that the sides of the cut exhibit the correct parity requirements. □

Lemma 26. *There exists a gadget with performance 17 and parity 0.*

Proof. Consider the complete graph on $\{0, 1, a, x, y, z\}$. Let $w(a) = w(1) = 2$, $w(0) = w(x) = w(y) = w(z) = 1$, $\delta(0, 1) = \delta(1, 0) = 1$, whereas otherwise $\delta(i, j) = 0$. Then define $w(i, j) = w(i)w(j) + \delta(i, j)$.

1. Without loss of generality suppose $c(0) = 0$. There are two cases. First suppose $c(x) = c(y) = c(z) = 0$. Then our cut is $(\{x, y, z, 0\}, \{a, 1\})$ with weight 17. Otherwise, suppose without loss of generality that $c(x) = 0, c(y) = c(z) = 1$. Then our cut is $(\{1, y, z\}, \{0, a, x\})$, with weight 17.

2. Consider all the triangle-free subgraphs of the gadget. This subgraph is either bipartite, or it is a subgraph of the following graph

   First we deal with the bipartite case. The weight of a complete bipartite graph with parts (A, B) is
   $$\left(\sum_{i\in A, j\in B} \delta(i, j)\right) + \left(\sum_{i\in A} w(i)\right)\left(\sum_{j\in B} w(j)\right).$$
   The first part of the sum is 1 if and only if 0 and 1 are in different sides of the cut, and the second is maximised exactly when the two sides of the cut are as even as possible i.e. have weight 4 and 4 respectively — cuts that satisfy both of these conditions weight at most $17 = 4 \times 4 + 1$ edges, while all other cuts have weight at most 16.

   Now consider a non-bipartite triangle-free subgraph. The largest this could be is the graph pictured above. First consider the contribution of $w(i)w(j)$ to the total weight. Suppose for contradiction that the contribution is > 14; since there are 7 edges, there must exist one edge whose contribution is > 2. Since $w(i)w(j) \in \{1, 2, 4\}$, and $w(i)w(j) = 4$ only for $\{i, j\} = \{a, 1\}$, a and 1 are adjacent within the subgraph. Since any way of placing $a, 1$ adjacently within the subgraph leads to them having at most 3 other neighbours, the contribution of $w(i)w(j)$ is at most $4 + 3 \times 2 + 3 \times 1 = 13 \not> 14$, a contradiction. So the contribution of $w(i)w(j)$ is at most 14, and the total weight is at most 15.

3. Consider now any triangle-free subgraph of the gadget with weight greater than 16. By the analysis in the previous item, this subgraph must be a complete bipartite subgraph where the two parts both have weight 4, and 0 and 1 are put on opposite sides of the cut. Consider the side of a: if it is on the same side as 1, then x, y, z are all on the opposite side i.e. together with 0. Conversely, if a is on the same side as 0, then two of x, y, z must be on the same side as 1, and the remaining vertex among x, y, z must be on the same side as 0. It is not difficult to check that these bipartite subgraphs satisfy the desired connectivity and parity conditions. □

7 Hardness of MaxPCSP(C_{2k+1}, K_ℓ)

In this section, we will prove the following result.

Theorem 6. *For every $k \geq 1$ and $\ell \geq 3$, 1-approximation of* MaxPCSP(C_{2k+1}, K_ℓ) *is* NP*-hard assuming the* UGC.

The proof is a generalisation of the proof in [DMR09] that establishes the NP-hardness of *almost 3-colouring* (i.e. given an n vertex graph G, output Yes if G has a 3-colourable $(1-\varepsilon)n$ vertex-induced subgraph, and No if G does not even have an independent set with more than εn vertices), assuming the UGC. We will need, unlike [DMR09], a *multilayered unique games conjecture*, in the style of [BWŽ21]. We first set up the necessary ingredients. The proof is based on *Markov-chain noise operators* — the one we will use is intimately related to C_{2k+1}.

Definition 27. Define M_n to be the matrix which has $\frac{1}{2}$ at position (i, j) if and only if $|i - j| \equiv 1 \bmod n$. I.e. M_n is the *circulant matrix* given by the vector $(0, \frac{1}{2}, 0, \dots, 0, \frac{1}{2})$, of length n. Note that M_n is a Markov chain on $[n]$, and (i, j) has nonzero transition probability if and only if (i, j) is an edge of C_n.

Lemma 28. *The uniform distribution is the stationary distribution of M_n.*

Proof. Easy to check, but also follows immediately since M_n is the random walk on C_n, which is an undirected, connected, regular graph. ◻

Lemma 29. *Let ω be the n-th root of unity. The eigenvalues of M_n are $\cos(2k\pi/n)$ for $k = 0, \dots, n-1$. In particular, if n is odd then exactly one eigenvalue has absolute value 1 and the rest have absolute value at most $\cos(1 - \pi/n) < 1$.*[7]

Proof. This follows from the formula for the eigenvalues of a circulant matrix. ◻

These are the key properties needed to apply the theory of [DMR09]. We will need a multi-layered Unique Games Conjecture, which we now state.

Definition 30. An ℓ-layered unique label-cover instance consists of a set of variables $X_1, \dots, X_\ell$, a domain $[D]$, and a multiset of constraints. Each constraint consists of ℓ variables $(x_1, \dots, x_\ell) \in X_1 \times \cdots \times X_\ell$, together with a family of permutations π_{ij} on $[D]$ for $1 \leq i < j \leq \ell$, such that $\pi_{ik} = \pi_{jk} \circ \pi_{ij}$. A solution is an assignment $c : (X_1 \cup \ldots \cup X_\ell) \to [D]$. The assignment c *strongly satisfies* a constraint given by $(x_1, \dots, x_\ell)$ and $(\pi_{ij})_{1 \leq i < j \leq \ell}$ if $\pi_{ij}(c(x_i)) = c(x_j)$ *for all* $1 \leq i < j \leq \ell$. The assignment c *weakly satisfies* this same constraint if $\pi_{ij}(c(x_i)) = c(x_j)$ *for at least one pair* $1 \leq i < j \leq \ell$. The strong value of an instance is the maximum fraction of constraints that can be simultaneously strongly satisfied by some assignment; the weak value is given by the maximum fraction of constraints that can be simultaneously weakly satisfied by some assignment.

Note that for $\ell = 2$, weak satisfaction and strong satisfaction (and hence weak and strong values) coincide. Hence for $\ell = 2$ we drop the weak/strong distinction.

Conjecture 31 (UGC [Kho02]). *For every ε there exists D such that, given a 2-layered unique label-cover instance with domain $[D]$, it is* NP*-hard to distinguish if the value is at least $1 - \varepsilon$ or not even ε.*

[7] This would *not* be true if n were even — there would be two eigenvalues with absolute value 1, namely ± 1, taking $k = 0$ and $k = n/2$.

We will show that Conjecture 31 implies the following conjecture. Our proof closely follows [BWŽ21], which in turn builds on [DGKR05], but is generalised to deal with imperfect completeness.

Conjecture 32 (Multilayered UGC). *For every $\varepsilon, \ell \geq 2$ there exists some D such that, given an ℓ-layered unique label-cover instance with domain $[D]$, it is* NP*-hard to distinguish if the strong value is at least $1-\varepsilon$, or if the weak value is not even ε.*

Proof of Conjecture 32 from Conjecture 31. Our reduction is identical to that of [BWŽ21]. The proof differs in two ways (i) we must show that this reduction, if given a unique label cover, produces a multilayered label cover with bijective constraints, and (ii) we need to care of the completeness case, which is imperfect here unlike in [BWŽ21]. The proof of soundness is identical to that in [BWŽ21].

Reduction. Suppose we are given a 2-layered unique label-cover instance with variables A, B, domain D, and constraints $(a, b) \in E$ with permutations π_{ab} for $(a, b) \in E$. Our reduction produces an ℓ-layered unique label-cover instance, with variables $X_1, \dots, X_\ell$, where $X_i = A^{\ell-i} \times B^{i-1}$, on domain $[D]^{\ell-1}$. For every $(\ell-1)$-tuple of constraints $(a_1, b_1), \dots, (a_{\ell-1}, b_{\ell-1}) \in E$, where we let $\pi_i = \pi_{a_i, b_i}$, we create a constraint on variables $(x_1, \dots, x_{\ell-1})$, where

$$x_i = (a_1, \dots, a_{i-1}, b_i, \dots, b_{\ell-1}).$$

Now, we define π_{ij} as follows. Observe that x_i and x_j share the first $i-1$, as well as the last $\ell-j-1$ variables in common — the variables on the indices in between (i.e. i up to $j-1$) are different. Hence we write

$$\pi_{ij}(d_1, \dots, d_\ell) = (d_1, \dots, d_{i-1}, \pi_i(d_i), \dots, \pi_{j-1}(d_{j-1}), d_j, \dots, d_{\ell-1}).$$

Observe that every such constraint is a bijection on $[D]^\ell$ — indeed,

$$\pi_{ij}^{-1}(d_1, \dots, d_\ell) = (d_1, \dots, d_{i-1}, \pi_i^{-1}(d_i), \dots, \pi_{j-1}^{-1}(d_{j-1}), d_j, \dots, d_{\ell-1}).$$

Furthermore, it is easy to check that $\pi_{ik} = \pi_{jk} \circ \pi_{ij}$ for every $1 \leq i < j < k \leq \ell$. Noting that the number of vertices and edges is polynomial completes the reduction.

Completeness. Suppose that the original instance has value $1-\delta$. We wish to show that we can select δ small enough, independent of D, so that the final instance has strong value $1-\varepsilon$, for every ε, ℓ. Let c be the solution that witnesses the value of the original instance. We claim that c' given by

$$c'(x_1, \dots, x_{\ell-1}) = (c(x_1), \dots, c(x_{\ell-1}))$$

has strong value at least $(1-\delta)^{\ell-1}$. Setting $\delta = 1 - \sqrt[\ell-1]{1-\varepsilon}$ makes this value equal $1-\varepsilon$, as required. To see why this is the case, suppose we select a constraint of the output instance uniformly at random. We want to show that this constraint is satisfied with probability at least $(1-\delta)^{\ell-1}$. Observe that, by construction, selecting a constraint of the output uniformly at random is the same as selecting an $(\ell-1)$-tuple of constraints $(a_1, b_1), \dots, (a_{\ell-1}, b_{\ell-1}) \in E$ uniformly and independently at random. Each of these is satisfied by c with probability $1-\delta$; hence all constraints are satisfied by c with probability $(1-\delta)^{\ell-1}$. Furthermore, when c satisfies all the constraints (a_i, b_i) then c' satisfies the constraint in the output instance built from these constraints. This completes the proof.

Soundness. Identical to the proof in [BWŽ21]. (This requires that the UGC instance we start from is bi-regular, but this is possible by a standard transformation, cf. [Kho10] or [AB09, DGKR05].) □

The proof of Theorem 6 will be based on the *long code construction* [BGS98]. We will now describe the building blocks of our reduction.

Definition 33. Fix k, ℓ, D. A *cloud* of vertices, denoted by $\vec{f}$, is a set of vertices $f(a_1, \ldots, a_D)$ for $a_1, \ldots, a_D \in [2k+1]$. For ℓ clouds of variables $\vec{f_1}, \ldots, \vec{f_\ell}$ and a family of permutations $\pi_{ij} : [D] \to [D]$ for $1 \le i < j \le \ell$ as in Definition 30, we define the set of edges $E_\pi(\vec{f_1}, \ldots, \vec{f_\ell})$ as follows: for every $1 \le i < j \le \ell$, $a_1, \ldots, a_D, b_1, \ldots, b_D \in [2k+1]$ and for which a_t, b_t differ by ± 1 modulo $2k+1$, we include the edge $f_i(a_{\pi_{ij}(1)}, \ldots, a_{\pi_{ij}(D)}) - f_j(b_1, \ldots, b_D)$.

The following is the key theorem from [DMR09] that we will use. We will define some notions, however we refer to [DMR09] for a full treatment.

Definition 34. For a symmetric Markov operator T on $[q]$, and letting $f, g : [q]^n \to \mathbb{R}$, the value $\langle f, T^{\otimes n} g\rangle$ has the following interpretation. Let $x \in [q]^n$ be distributed uniformly at random, and let $y \in [q]^n$ be such that y_i is distributed according to the transition probabilities in T starting at x_i. Then $\langle f, T^{\otimes n} g\rangle$ is the expected value of $f(x)g(y)$.

The quantity $\langle F_\mu, U_\rho(1 - F_{1-\nu})\rangle_\gamma$, which we also denote by $\Gamma_\rho(\mu, \nu)$, has the following interpretation. Let x, y be two normally distributed variables with mean 0, variance 1 and covariance ρ. Then this value is the probability that $x \le \Phi^{-1}(\mu), y \ge \Phi^{-1}(1-\nu)$, where Φ is the cumulative distribution function of the normal distribution. Essentially, this value is nondecreasing in both μ and ν.

For a function $f : [q]^n \to \mathbb{R}$, the value $I_i^{\le t}(f)$ is the *low-degree influence* of coordinate i in f. In particular, if $f(x) \in [0, 1]$, it can be shown that $\sum_{i=1}^n I_i^{\le t}(f) \le t$ and $I_i^{\le t}(f) \ge 0$. Furthermore, the influence of a coordinate is defined compatibly with permuting coordinates i.e. if the influence of coordinate i is x, and we permute the coordinates of f so as to move i to position j, yielding a function g, then the influence of j in g is still x.

Theorem 35 ([DMR09]). *Fix q and let T be a symmetric Markov operator on $[q]$ with spectral radius $\rho < 1$ (by spectral radius we mean the second largest eigenvalue of T in absolute value). Then for any $\varepsilon > 0$ there exist $\delta > 0$ and $t \in \mathbb{N}$ so that if $f, g : [q]^n \to [0, 1]$ are two functions with*

$$\min(I_i^{\le t}(f), I_i^{\le t}(g)) < \delta,$$

for all i, then

$$\langle f, T^{\otimes n} g\rangle \ge \langle F_\mu, U_\rho(1 - F_{1-\nu})\rangle_\gamma - \varepsilon,$$

where $\mu = E[f], \nu = E[g]$.

Similarly to [DMR09], we will use this in the contrapositive, in particular in the following form.

Corollary 36. *Fix q and let T be a symmetric Markov operator on $[q]$ with spectral radius $\rho < 1$. For every ε there exists $\delta > 0$ and $t \in \mathbb{N}$ so that, if $f, g : [q]^n \to [0, 1]$ are two functions with $E[f] \ge \varepsilon, E[g] \ge \varepsilon$ and $\langle f, T^{\otimes n} g\rangle \le \delta$, then there exists $i \in [n]$ so that $I_i^{\le t}(f) \ge \delta$ and $I_i^{\le t}(g) \ge \delta$.*

Proof. Apply Theorem 35; the values ε', δ' are the values we use for Theorem 35. We take $\varepsilon' = \min(\varepsilon, \Gamma_\rho(\varepsilon, \varepsilon)/2)$ and take $\delta < \min(\delta', \Gamma_\rho(\varepsilon, \varepsilon)/2)$.

Suppose that $\langle f, T^{\otimes} g\rangle \leq \delta < \Gamma_\rho(\varepsilon,\varepsilon)/2$ — we must prove that $\langle f, T^{\otimes n} g\rangle < \Gamma_\rho(\mu,\nu) - \varepsilon'$, where $\mu = E[f], \nu = E[g]$. Since $\varepsilon \leq \mu, \varepsilon \leq \nu$ it is sufficient to show that $\langle f, T^{\otimes n} g\rangle < \Gamma_\rho(\varepsilon,\varepsilon) - \varepsilon'$. Since $\varepsilon' \leq \Gamma_\rho(\varepsilon,\varepsilon)/2$ it is sufficient to show that $\langle f, T^{\otimes n} g\rangle < \Gamma_\rho(\varepsilon,\varepsilon)/2$ — which is true by assumption. ◻

Lemma 37. *There exists s, δ which depend only on k, ℓ so that the following holds for any D.*

Consider an ℓ-colouring $\vec{f} \to [\ell]$ of the cloud of vertices $\vec{f}$. We denote the vertex $f(a_1,\dots,a_D)$ receiving colour c by $f(a_1,\dots,a_D) = c$. There exists a way to assign any such cloud a subset $I(\vec{f})$ of $[D]$ of size s such that the following holds.

Consider any $\ell+1$ clouds $\vec{f_1},\dots,\vec{f_{\ell+1}}$ and a family of permutations π_{ij} as in Definition 30. Suppose that these vertices are ℓ-coloured, and that the ℓ-colouring satisfies a $(1-\delta)$-fraction of the edges in $E_\pi(\vec{f_1},\dots,\vec{f_{\ell+1}})$. Then there exists $1 \leq i < j \leq \ell+1$ such that $\pi_{ij}(I(\vec{f_i})) \cap I(\vec{f_j}) \neq \emptyset$.

Proof. Consider some cloud $\vec{f}$. Note that this cloud must have a most frequent colour; let c be that colour, and let $f : [2k+1]^D \to \{0,1\}$ be the indicator function of this colour inside $\vec{f}$ i.e. $f(a_1,\dots,a_D)$ is 1 if $\vec{f}(a_1,\dots,a_D) = c$ and 0 otherwise. To construct $I(\vec{f})$, apply Corollary 36 to f with $T = M_{2k+1}$ and $\varepsilon' = \frac{1}{\ell}$; suppose we get k and δ' from Corollary 36. We set $\delta = \frac{\delta'}{(\ell+1)^2}$ and

$$I(\vec{f}) = \{i \in [D] \mid I_i^{\leq t}(f) \geq \delta\}.$$

Observe that $\sum_{i=1}^n I_i^{\leq t}(f) = 1$ and $I_i^{\leq t}(f) \geq 0$. So $|I(\vec{f})| \leq \frac{k}{\delta}$, and we set $s = \frac{k}{\delta}$.

Now we must prove that the chain condition holds. Consider clouds $\vec{f_1},\dots,\vec{f_{\ell+1}}$ and a family of permutations π_{ij} as in Definition 30. By the pidgeonhole principle, there exists $i < j$ such that the most frequent colour in $\vec{f_i}, \vec{f_j}$ coincides. Say that colour is i, and let $\vec{f} = \vec{f_i}, \vec{g} = \vec{g_i}, \pi = \pi_{ij}$ and let $f, g : [2k+1]^D \to \{0,1\}$ be the indicator functions of the colour c in $\vec{f}, \vec{g}$.

We define the function f^π as follows:

$$f^\pi(x_1,\dots,x_D) = f(x_{\pi(1)},\dots,x_{\pi(D)}).$$

Note that $\pi(I(f)) = I(f^\pi)$ since influences are compatible with permuting coordinates. Hence by Lemma 37 it suffices to show that $\langle f^\pi, M_{2k+1}^{\otimes n} g\rangle \leq \delta' = (\ell+1)^2\delta$. Suppose for contradiction that $\langle f^\pi, M_{2k+1}^{\otimes n} g\rangle > (\ell+1)^2\delta$.

Recall that $\langle f^\pi, M_{2k+1}^{\otimes n} g\rangle$ can be defined equivalently as follows. Suppose $x_1,\dots,x_D$ are drawn uniformly and independently at random from $[2k+1]$, and $y_1,\dots,y_D$ is drawn so that y_i follows the distribution of one step of M_{2k+1} starting from x_i. In other words, $y_i = (x_i \pm 1) \bmod 2k+1$, with the $\pm$ being $+$ or $-$ with probability $1/2$. Then $\langle f^\pi, M_{2k+1}^{\otimes n} g\rangle$ is the expected value of $f(x_{\pi(1)},\dots,x_{\pi(D)})g(y_1,\dots,y_D)$. But, by construction, this value is equal to the fraction of edges in $E_\pi(\vec{f_1},\dots,\vec{f_{\ell+1}})$ between $\vec{f}$ and $\vec{g}$ that have both endpoints coloured to c. Since the number of edges between $\vec{f_i}$ and $\vec{f_j}$ in $E_\pi(\vec{f_1},\dots,\vec{f_{\ell+1}})$ is equal for every choice of i and j, of which (coarsely) there are at most $(\ell+1)^2$, it follows that a $> \delta$ fraction of the edges in $E_\pi(\vec{f_1},\dots,\vec{f_{\ell+1}})$ have both endpoints coloured to c. This contradicts our assumptions. ◻

Theorem 38. *Conjecture 32 implies Theorem 6.*

Proof. We reduce the problem from Conjecture 32 to that of Theorem 6. Fix $k \geq 1, \ell \geq 3$; we will prove that it is hard to solve $\mathrm{MaxPCSP}(C_{2k+1}, K_\ell)$ in the case where $\rho = 1-\varepsilon$ for some very small ε. We will show that there exists some completeness/soundness for Conjecture 32, call it ε', for which our reduction will work.

Reduction. Suppose we are given an instance of $(\ell+1)$-layered unique label-cover with domain D. Suppose the variable set is $X_1, \ldots, X_{\ell+1}$, and the constraint set $E \subseteq X_1 \times \cdots \times X_{\ell+1}$. For every variable x, we introduce a cloud of vertices $\vec{f}_x$. For every constraint $(x_1, \ldots, x_{\ell+1}) \in E$, characterised by functions $(\pi_{ij})_{1 \le i < j \le \ell+1}$, we introduce the set of edges $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ — if an edge is included in multiple copies, we introduce it with a higher multiplicity.

Completeness. Suppose that the original instance has a strong value of $1-\varepsilon'$, witnessed by c. We claim that the reduced instance has a $(1-\varepsilon')$-fractional C_{2k+1}-colouring. Indeed let c' be this colouring, and colour vertex $f_x(a_1, \ldots, a_D)$ by

$$c'(f_x(a_1, \ldots, a_D)) = a_{c(x)}.$$

Now, we must show that a $1-\varepsilon'$ fraction of edges are coloured properly. We sample the edge as follows: first sample a constraint in the original instance uniformly at random, $(x_1, \ldots, x_{\ell+1})$, then sample some edge from $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ uniformly at random. Since each constraint spawns the same number of edges, this is the same as sampling an edge uniformly at random. Now, we will show that if the constraint $(x_1, \ldots, x_{\ell+1})$ was satisfied by c, then *all* the edges in $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ are satisfied — this is sufficient to show completeness.

Consider some arbitrary edge $f_{x_i}(a_{\pi_{ij}(1)}, \ldots, a_{\pi_{ij}(D)}) - f_j(b_1, \ldots, b_D)$ from a constraint $(x_1, \ldots, x_\ell)$ that was satisfied in the original instance. The endpoints are coloured by $a_{\pi_{ij}(c(x_i))}$ and $b_{c(x_j)}$. Since the original constraint was satisfied, we know that $\pi_{ij}(c(x_i)) = c(x_j)$, hence these colours are just $a_{c(x_j)}$ and $b_{c(x_j)}$. By construction, we know that these two value differ by ± 1 modulo $2k+1$, and hence the edge is satisfied.

Hence for the completeness to work, we need to take $\varepsilon' \le \varepsilon$.

Soundness. Suppose we have a $(1-\varepsilon)$-fractional ℓ-colouring of the reduced graph. We must show that the original instance has a weak solution of value ε'. Consider the unsatisfied edges; suppose that we select a constraint $(x_1, \ldots, x_{\ell+1})$ at random, and we count the expected number of unsatisfied edges is $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$. Noting that there are the same number of edges in $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ for every constraint $(x_1, \ldots, x_{\ell+1})$, by Markov's inequality, the probability that there are more than $\sqrt{\varepsilon}$ unsatisfied edges is at most $\sqrt{\varepsilon}$. Hence, for a $(1-\sqrt{\varepsilon})$-fraction of the constraints $(x_1, \ldots, x_{\ell+1})$, a $(1-\sqrt{\varepsilon})$-fraction of the edges in $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ are coloured correctly.

Now, apply Lemma 37 to get s, δ and an assignment function I. Take ε so that $\sqrt{\varepsilon} \le \delta$. We claim that assigning variable x a value from $I(f_x)$ uniformly at random will weakly satisfy a $\frac{1-\sqrt{\varepsilon}}{s^2}$-fraction of the constraints. Taking $\varepsilon' \le \frac{1-\sqrt{\varepsilon}}{s^2}$ will make the reduction work. For each of the $(1-\sqrt{\varepsilon})$-fraction of the constraints where a $(1-\sqrt{\varepsilon})$ fraction of the edges in $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ are coloured correctly, we claim that the constraint is weakly satisfied with probability at least $1/s^2$. Indeed, this follows from Lemma 37: the edges in $E_\pi(\vec{f}_{x_1}, \ldots, \vec{f}_{x_{\ell+1}})$ satisfy the preconditions of Lemma 37, so for some $1 \le i < j \le \ell+1$ we have that $\pi_{ij}(I(\vec{f}_{x_i})) \cap I(\vec{f}_{x_j}) \ne \emptyset$. So, with probability $1/s^2$ we choose the correct value for x_i and x_j, and weakly satisfy the constraint. □

Acknowledgements

We thank Jakub Opršal, who asked us about the complexity of MaxPCSPs for graphs beyond cliques, and for useful discussions. We also thank Pravesh Kothari for asking us about the performance of our maximum bipartite vs. triangle-free subgraph algorithm in the $1-\varepsilon$ regime, which led to Theorem 9.

References

[AB09] Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[ABP20] Per Austrin, Amey Bhangale, and Aditya Potukuchi. Improved inapproximability of rainbow coloring. In *Proc. 31st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'20)*, pages 1479–1495, 2020. arXiv:1810.02784, doi:10.1137/1.9781611975994.90.

[AFO+25] Sergey Avvakumov, Marek Filakovský, Jakub Opršal, Gianluca Tasinato, and Uli Wagner. Hardness of 4-Colourings G-Colourable Graphs. In *Proc. 57th Annual ACM Symposium on Theory of Computing (STOC'25)*. ACM, 2025. arXiv:2504.07592.

[AGH17] Per Austrin, Venkatesan Guruswami, and Johan Håstad. $(2+\varepsilon)$-Sat is NP-hard. *SIAM J. Comput.*, 46(5):1554–1573, 2017. doi:10.1137/15M1006507.

[BBB21] Libor Barto, Diego Battistelli, and Kevin M. Berg. Symmetric Promise Constraint Satisfaction Problems: Beyond the Boolean Case. In *Proc. 38th International Symposium on Theoretical Aspects of Computer Science (STACS'21)*, volume 187 of *LIPIcs*, pages 10:1–10:16, 2021. arXiv:2010.04623, doi:10.4230/LIPIcs.STACS.2021.10.

[BBK+24] Libor Barto, Silvia Butti, Alexandr Kazda, Caterina Viola, and Stanislav Živný. Algebraic approach to approximation. In *39th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'24)*, pages 10:1–10:14. ACM, 2024. arXiv:2401.15186, doi:10.1145/3661814.3662076.

[BBKO21] Libor Barto, Jakub Bulín, Andrei A. Krokhin, and Jakub Opršal. Algebraic approach to promise constraint satisfaction. *J. ACM*, 68(4):28:1–28:66, 2021. arXiv:1811.00970, doi:10.1145/3457606.

[BG21] Joshua Brakensiek and Venkatesan Guruswami. Promise Constraint Satisfaction: Algebraic Structure and a Symmetric Boolean Dichotomy. *SIAM J. Comput.*, 50(6):1663–1700, 2021. arXiv:1704.01937, doi:10.1137/19M128212X.

[BGS98] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free Bits, PCPs, and Nonapproximability — Towards Tight Results. *SIAM J. Comput.*, 27(3):804–915, 1998. doi:10.1137/S0097539796302531.

[BGS23a] Joshua Brakensiek, Venkatesan Guruswami, and Sai Sandeep. Conditional dichotomy of Boolean ordered promise CSPs. *TheoretiCS*, 2, 2023. arXiv:2102.11854, doi:10.46298/theoretics.23.2.

[BGS23b] Joshua Brakensiek, Venkatesan Guruswami, and Sai Sandeep. SDPs and Robust Satisfiability of Promise CSP. In *Proc. Annual 55th ACM Symposium on Theory of Computing (STOC'23)*, pages 609–622. ACM, 2023. arXiv:2211.08373, doi:10.1145/3564246.3585180.

[BJK05] Andrei Bulatov, Peter Jeavons, and Andrei Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, 2005. doi:10.1137/S0097539700376676.

[BK05] Ankur Bhargava and S. Rao Kosaraju. Derandomization of dimensionality reduction and SDP based algorithms. In *Proc. 8th International Workshop on Algorithms and Data Structures (WADS'05)*, pages 396–408. Springer Berlin Heidelberg, 2005. doi:10.1007/11534273_35.

[BLŽ25] Silvia Butti, Alberto Larrauri, and Stanislav Živný. Optimal inapproximability of promise equations over finite groups. In *Proc. 52nd International Colloquium on Automata, Languages, and Programming (ICALP'25)*, 2025. arXiv:2411.01630.

[BR15] Jonah Brown-Cohen and Prasad Raghavendra. Combinatorial optimization algorithms via polymorphisms. Technical report, 2015. arXiv:1501.01598.

[Bul17] Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs. In *Proc. 58th Annual IEEE Symposium on Foundations of Computer Science (FOCS'17)*, pages 319–330, USA, 2017. IEEE. arXiv:1703.03021, doi:10.1109/FOCS.2017.37.

[BWŽ21] Alex Brandts, Marcin Wrochna, and Stanislav Živný. The complexity of promise SAT on non-Boolean domains. *ACM Trans. Comput. Theory*, 13(4):26:1–26:20, 2021. arXiv:1911.09065, doi:10.1145/3470867.

[Che68] M. Cheng. The clipping loss in correlation detectors for arbitrary input signal-to-noise ratios. *IEEE Trans. Inf. Theory*, 14(3):382–389, 1968. doi:10.1109/TIT.1968.1054159.

[Che69] M. C. Cheng. The orthant probabilities of four gaussian variates. *Ann. Math. Stat.*, 40(1):152–161, 1969. doi:10.1214/aoms/1177697812.

[DGKR05] Irit Dinur, Venkatesan Guruswami, Subhash Khot, and Oded Regev. A new multilayered PCP and the hardness of hypergraph vertex cover. *SIAM J. Comput.*, 34(5):1129–1146, 2005. doi:10.1137/S0097539704443057.

[DKPS10] Irit Dinur, Subhash Khot, Will Perkins, and Muli Safra. Hardness of finding independent sets in almost 3-colorable graphs. In *Proc. 51th Annual IEEE Symposium on Foundations of Computer Science (FOCS'10)*, pages 212–221. IEEE Computer Society, 2010. doi:10.1109/FOCS.2010.84.

[DMR09] Irit Dinur, Elchanan Mossel, and Oded Regev. Conditional Hardness for Approximate Coloring. *SIAM J. Comput.*, 39(3):843–873, 2009. doi:10.1137/07068062X.

[DRS05] Irit Dinur, Oded Regev, and Clifford Smyth. The hardness of 3-uniform hypergraph coloring. *Comb.*, 25(5):519–535, September 2005. doi:10.1007/s00493-005-0032-4.

[EH08] Lars Engebretsen and Jonas Holmerin. More efficient queries in PCPs for NP and improved approximation hardness of maximum CSP. *Random Struct. Algorithms*, 33(4):497–514, 2008. doi:10.1002/RSA.20226.

[Euc26] Euclid. *The Thirteen Books of Euclid's Elements.* Cambridge University Press, Cambridge, second edition, 1926. Trans. Sir Thomas Little Heath.

[FJ97] Alan M. Frieze and Mark Jerrum. Improved Approximation Algorithms for MAX k-CUT and MAX BISECTION. *Algorithmica*, 18(1):67–81, 1997. doi:10.1007/BF02523688.

[FKOS19] Miron Ficak, Marcin Kozik, Miroslav Olšák, and Szymon Stankiewicz. Dichotomy for Symmetric Boolean PCSPs. In *Proc. 46th International Colloquium on Automata, Languages, and Programming (ICALP'19)*, volume 132, pages 57:1–57:12, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. arXiv:1904.12424, doi:10.4230/LIPIcs.ICALP.2019.57.

[FV98] Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1), 1998. doi:10.1137/S0097539794266766.

[GJ76] M. R. Garey and D. S. Johnson. The complexity of near-optimal graph coloring. *J. ACM*, 23(1):43–49, 1976. doi:10.1145/321921.321926.

[GW95] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995. doi:10.1145/227683.227684.

[Hås01] Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 7 2001. doi:10.1145/502090.502098.

[HMS23] Yahli Hecht, Dor Minzer, and Muli Safra. NP-Hardness of Almost Coloring Almost 3-Colorable Graphs. In *Proc. Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM'23)*, volume 275 of *LIPIcs*, pages 51:1–51:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.APPROX/RANDOM.2023.51.

[HN90] Pavol Hell and Jaroslav Nešetřil. On the Complexity of H-coloring. *J. Comb. Theory, Ser. B*, 48(1):92–110, 1990. doi:10.1016/0095-8956(90)90132-J.

[Jea98] Peter G. Jeavons. On the Algebraic Structure of Combinatorial Problems. *Theor. Comput. Sci.*, 200(1-2):185–204, 1998. doi:10.1016/S0304-3975(97)00230-2.

[Kar72] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations*, pages 85–103. Springer US, 1972. doi:10.1007/978-1-4684-2001-2_9.

[Kho02] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proc. 34th Annual ACM Symposium on Theory of Computing (STOC'02)*, pages 767–775. ACM, 2002. doi:10.1145/509907.510017.

[Kho10] Subhash Khot. On the unique games conjecture (invited survey). In *Proc. 25th Annual IEEE Conference on Computational Complexity (CCC'10)*, pages 99–121. IEEE Computer Society, 2010. doi:10.1109/CCC.2010.19.

[KKMO07] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs? *SIAM J. Comput.*, 37(1):319–357, 2007. doi:10.1137/S0097539705447372.

[KMS98] David R. Karger, Rajeev Motwani, and Madhu Sudan. Approximate graph coloring by semidefinite programming. *J. ACM*, 45(2):246–265, 1998. doi:10.1145/274787.274791.

[KOWŽ23] Andrei A. Krokhin, Jakub Opršal, Marcin Wrochna, and Stanislav Živný. Topology and adjunction in promise constraint satisfaction. *SIAM J. Computing*, 52(1):37–79, 2023. arXiv:2003.11351, doi:10.1137/20M1378223.

[KS12] Subhash Khot and Rishi Saket. Hardness of finding independent sets in almost q-colorable graphs. In *Proc. 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'12)*, pages 380–389. IEEE Computer Society, 2012. doi:10.1109/FOCS.2012.75.

[KSTW00] Sanjeev Khanna, Madhu Sudan, Luca Trevisan, and David P. Williamson. The approximability of constraint satisfaction problems. *SIAM J. Comput.*, 30(6):1863–1920, 2000. doi:10.1137/S0097539799349948.

[MOO10] Elchanan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. *Ann. of Math. (2)*, 171(1):295–341, 2010. doi:10.4007/annals.2010.171.295.

[MR99] Sanjeev Mahajan and H. Ramesh. Derandomizing approximation algorithms based on semidefinite programming. *SIAM J. Comput.*, 28(5):1641–1663, 1999. doi:10.1137/S0097539796309326.

[NŽ23] Tamio-Vesa Nakajima and Stanislav Živný. Maximum k- vs. ℓ-colourings of graphs, 2023. arXiv:2311.00440.

[PY91] Christos H. Papadimitriou and Mihalis Yannakakis. Optimization, approximation, and complexity classes. *J. Comput. Syst. Sci.*, 43(3):425–440, 1991. doi:10.1016/0022-0000(91)90023-X.

[Rag08] Prasad Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *Proc. 40th Annual ACM Symposium on Theory of Computing (STOC'08)*, pages 245–254, 2008. doi:10.1145/1374376.1374414.

[Rai19] Martin Raič. A multivariate Berry–Esseen theorem with explicit constants. *Bernoulli*, 25(4A):2824 – 2853, 2019. arXiv:1802.06475, doi:10.3150/18-BEJ1072.

[RS09] Prasad Raghavendra and David Steurer. How to round any CSP. In *Proc. 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS'09)*, pages 586–594. IEEE Computer Society, 2009. doi:10.1109/FOCS.2009.74.

[Sch78] Thomas Schaefer. The complexity of satisfiability problems. In *Proc. 10th Annual ACM Symposium on the Theory of Computing (STOC'78)*, pages 216–226, USA, 1978. ACM. doi:10.1145/800133.804350.

[TSSW00] Luca Trevisan, Gregory B. Sorkin, Madhu Sudan, and David P. Williamson. Gadgets, approximation, and linear programming. *SIAM J. Comput.*, 29(6):2074–2097, 2000. doi:10.1137/S0097539797328847.

[TŽ16] Johan Thapper and Stanislav Živný. The Complexity of Finite-Valued CSPs. *J. ACM*, 63(4):37:1–37:33, 2016. doi:10.1145/2974019.

[Zhu20] Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM*, 67(5):30:1–30:78, 2020. arXiv:1704.01914, doi:10.1145/3402029.