On the proof of the 2-to-2 Games Conjecture

Subhash Khot

Abstract. This article gives an overview of the recent proof of the 2-to-2 Games Conjecture in [68, 39, 38, 69] (with additional contributions from [75, 18, 67]). The proof requires an understanding of expansion in the Grassmann graph.

The 2-to-2 Games Conjecture is a lesser variant of the more well-known Unique Games Conjecture in Theoretical Computer Science. These conjectures have applications to "hardness of approximation", namely, the phenomenon that for many NP-hard problems, even computing approximate solutions to them remains NP-hard. They have several additional connections to computational complexity, algorithm design, analysis, and geometry.

The proof of the 2-to-2 Games Conjecture proves the Unique Games Conjecture "half-way" (at least in a certain technical sense) and in author's opinion, provides a strong evidence in favor of the latter.

Contents

1. Introduction

Since it was proposed in 2002 by the author [63], the Unique Games Conjecture has found many connections between computational complexity,

Research supported by NSF grant CCF-1813438, Simons Collaboration on Algorithms and Geometry grant, and the Simons Investigator Award.

algorithms, analysis, and geometry. It is now regarded as a central question in Theoretical Computer Science (TCS). Until recently, most researchers viewed the conjecture skeptically. However a recent proof of the closely related 2-to-2 Games Conjecture has led to a dramatic reversal in the popular opinion. This article gives an overview of the broader context, the proof of the 2-to-2 Games Conjecture, and a crucial ingredient in the proof, namely, a result about expansion in the Grassmann graph. We first briefly summarize how these connections play out, which are then elaborated in subsequent sections. Anticipating that the intended audience of this article is not necessarily familiar with TCS, we instead focus more on the context and the combinatorial aspects.

1.1. Approximation algorithms and hardness of approximation. The well-known P $\neq$ NP hypothesis says that a large class of computational problems known as "NP-hard" problems do not have efficient algorithms.[1] An algorithm is called efficient if it runs in time polynomial in the length of the input (by the term algorithm we would mean an efficient one unless stated otherwise). Since we do not hope to compute exact solutions to NP-hard problems efficiently, the next natural question is whether we can compute approximate solutions efficiently and how good an approximation can be achieved. An approximation algorithm for an NP-hard problem is an efficient algorithm that computes a solution that is guaranteed to be within a certain multiplicative factor of the optimum (known as the approximation factor). We are interested in both upper and lower bounds: designing algorithms with a guarantee on the approximation (upper bounds) as well as results showing that no efficient algorithm exists that achieves an approximation guarantee beyond a certain threshold (lower bounds). It is the latter question, namely the lower bounds, that is the focus of this article. Such results are known as "inapproximability" or "hardness of approximation" results, proved under a standard computational hypothesis such as P $\neq$ NP.

1.2. Illustrative example of 3Lin problem. Let us consider the 3Lin problem as an illustration. We are given a system of linear equations over $\mathbb{F}_2$ with three variables in each equation and the goal is to find an assignment that satisfies the maximum number of equations. This is known to be an NP-hard problem. We note that if there is an assignment that satisfies *all* equations, it can certainly be found efficiently using Gaussian elimination. What makes the problem hard is that even the best possible assignment could satisfy only, say, three-quarters of the equations.

Consider a trivial approximation algorithm that achieves a multiplicative approximation guarantee of 2. The algorithm simply assigns a random value in $\mathbb{F}_2$ to each variable and in expectation satisfies half of the equations.

[1]Strictly speaking, we are talking about the "NP-complete" problems. A NP-hard problem is at least as hard as every problem in NP. A NP-complete problem, in addition, is also in NP itself. We find it more convenient to use the term "NP-hard" though all the problems we consider are indeed in NP. See Section 2.1.

The optimal assignment may satisfy all (or nearly all) equations and thus the assignment produced by the algorithm is within factor 2 of the optimal assignment.

On the other hand, a well-known result of Håstad [52] shows that such a trivial algorithm is the best we can hope for! Specifically, let $\varepsilon > 0$ be an arbitrarily small constant.[2] Then unless P = NP, there is no efficient algorithm that given an instance of 3Lin that has an assignment satisfying $1-\varepsilon$ fraction of the equations, finds an assignment that satisfies $\frac{1}{2}+\varepsilon$ fraction of the equations! We would like to emphasize the remarkable implication of this result: if there were an algorithm $\mathcal{A}$ that, on a $(1-\varepsilon)$-satisfiable instance, finds a $(\frac{1}{2}+\varepsilon)$-satisfying assignment, then P = NP, and hence there is another algorithm $\mathcal{A}'$ that actually finds the best possible assignment on every instance, and in particular, $(1-\varepsilon)$-satisfying assignment on a $(1-\varepsilon)$-satisfiable instance. Moreover, as turns out to be the case, the algorithm $\mathcal{A}'$ can be constructed somehow using the algorithm $\mathcal{A}$ as a sub-routine. Stated differently, on a $(1-\varepsilon)$-satisfiable instance, the task of finding a $(\frac{1}{2}+\varepsilon)$-satisfying assignment is as hard as the seemingly harder task of finding a $(1-\varepsilon)$-satisfying assignment, both tasks being NP-hard. Stated yet differently, the 3Lin problem is hard to approximate within factor $2-\varepsilon$, and in light of the trivial 2-approximation algorithm, we have the approximation and hardness results that perfectly match at the threshold of factor 2, demonstrating optimality of each other.

1.3. The PCP theorem. The complementary study of approximation algorithms and hardness of approximation aims at characterizing precise approximation thresholds for NP-hard problems of interest, i.e. the threshold at which the approximation factor and the hardness factor (essentially) match.

The hardness of approximation results build on a foundational result known as the Probabilistically Checkable Proofs (PCP) Theorem [42, 8, 6]. The theorem can be viewed from a hardness viewpoint as well as from a proof checking viewpoint. From the hardness viewpoint, it states that there exists an absolute constant $\beta < 1$ such that, given a 3SAT formula ϕ, it is NP-hard to distinguish whether it is satisfiable or whether it is at most β-satisfiable (see Definition 2.1 and Section 2.3 for a definition of the 3SAT problem and a formal statement of the result). From the proof checking viewpoint, it states the following surprising fact: every NP-statement has a polynomial size proof that can be checked efficiently by a probabilistic verifier that reads only a constant number of bits from the proof! The verifier is complete and sound in the sense that a correct proof of a correct statement is accepted with probability 1 and any proof of an incorrect statement is accepted with probability at most, say $\frac{1}{2}$.

The equivalence between the hardness and the proof checking viewpoints, though not difficult to see, has led to many illuminating insights

[2]Unless stated otherwise, $\varepsilon > 0$ will henceforth denote an arbitrarily small constant and the statements are meant to hold for every such ε.

and strong hardness results over the last three decades. The proof checking viewpoint (whose roots go back to the work on interactive proofs) played a decisive role in the discovery of the PCP Theorem. However, for the sake of uniformity and ease of presentation, we adopt the hardness viewpoint here. We refer the reader to the surveys [**5**, **53**, **97**, **64**] for an overview of the extensive and influential body work on PCPs and hardness results.

Soon after the discovery of the PCP Theorem, researchers started seeking precise approximation thresholds for NP-hard problems of interest, i.e. the threshold at which the approximation factor and the hardness factor (essentially) match. The hardness result for the 3Lin problem described above was a major success in this regard, showing that factor 2 is the precise threshold. However such precise thresholds are known only for a handful of other problems, e.g. 3SAT, Clique, Set Cover (see Section 2.4, [**52**, **51**, **41**]). For a vast majority of problems, there remains a significant gap between the best known approximation factor and the best known hardness factor.

One such problem is Vertex Cover. Given a graph $G = (V, E)$, a subset $C \subseteq V$ is called a vertex cover if for every edge $e = (u, v) \in E$, either u or v is in C. Finding a minimum vertex cover is a well-known NP-hard problem [**60**]. It admits a simple 2-approximation, namely, an efficient algorithm that outputs a vertex cover C of size at most twice the minimum. The algorithm picks an arbitrary edge $e = (u, v) \in E$, adds both u, v to C, removes all edges that are incident on either u or v, and repeats this step. It is easily seen that this yields a 2-approximation: the edges $e_1, \ldots, e_k$ picked by the algorithm successively form a matching in the graph and while the vertex cover C output by the algorithm has $2k$ vertices, the minimum vertex cover must contain at least k vertices, one from each edge e_i. Whether there exists an algorithm with approximation factor strictly below 2 is among the flagship questions in approximability. Surprisingly, there is now good reason to believe that no such algorithm exists, i.e. it is conceivable that approximating Vertex Cover within factor $2 - \varepsilon$ is NP-hard!

1.4. The Unique Games and 2-to-2 Games Conjectures. As noted, the quest towards proving optimal hardness results was stalled after the remarkable but relatively few successes. In [**63**], the author introduced the Unique Games Conjecture as a plausible avenue to make further progress.[3] The conjecture turned out to be very successful, way beyond the author's initial expectations. Optimal hardness results are now known for a vast majority of problems of interest assuming the Unique Games Conjecture. In particular, in [**74**], it was shown that the Unique Games Conjecture implies NP-hardness of approximating Vertex Cover within $2 - \varepsilon$.

The conjecture states that a specific, canonical problem called Unique Games is hard to approximate. It can then be "reduced" to other problems

[3]The Unique Games problem was studied earlier in [**43**] in the context of parallel repetition, but not in the context of its computational hardness. The notion of "2-to-2" constraints appeared implicitly in [**36**] concurrently with [**63**].

showing their hardness as well. An instance of the Unique Games is a system of linear equations over the group $\mathbb{F}_2^\ell$ where every equation is of the form (here $\oplus$ denotes the group addition)

$$x_i \oplus x_j = c_{ij},$$

$\{x_1, \ldots, x_n\}$ are variables, and $c_{ij} \in \mathbb{F}_2^\ell$ are constants. The goal is to find an assignment to the variables that satisfies a *good* fraction of the equations.

The Unique Games Conjecture states that for every constant $\varepsilon > 0$, there is a large enough constant $\ell = \ell(\varepsilon)$, such that given an instance of Unique Games over $\mathbb{F}_2^\ell$ that has an assignment satisfying $1 - \varepsilon$ fraction of the equations, it is NP-hard to find an assignment that satisfies (even) an ε fraction of the equations.

In addition to implying hardness of approximation results, the Unique Games Conjecture has found many connections to computational complexity, algorithms, analysis, and geometry. It has led to many *unconditional* results that would hold true even if the conjecture itself were proven incorrect. A few examples include the Majority Is Stablest Theorem in Boolean function analysis [**66**, **84**], non-embeddability of negative type metrics into L_1, disproving the Goemans-Linial Conjecture [**76**], construction of optimal tiling [**92**, **77**], and connection between eigenvalues and small-set expansion [**9**].

Until recently however, most researchers viewed the conjecture skeptically and many arguments were cited as evidence against it. The perception has suddenly changed however thanks to a recent proof of the closely related 2-to-2 Games Conjecture. This conjecture is very similar to the Unique Games Conjecture except that the constraints are "2-to-2", namely, of the type

$$x_i \oplus x_j \in \{c_{ij}, c'_{ij}\}.$$

As before, $\{x_1, \ldots, x_n\}$ are variables, and $c_{ij}, c'_{ij} \in \mathbb{F}_2^\ell$ are constants. The term "2-to-2" refers to the property that for every assignment to the variable x_i, there are two acceptable values to the variable x_j, and vice versa (the term "unique" in the Unique Games is for a similar reason).

The 2-to-2 Games Conjecture (now a Theorem), proved recently in a sequence of papers [**68**, **39**, **38**, **69**], states that for every constant $\varepsilon > 0$, there is a large enough constant $\ell = \ell(\varepsilon)$, such that given an instance of 2-to-2 Games over $\mathbb{F}_2^\ell$ that has an assignment satisfying $1 - \varepsilon$ fraction of the constraints, it is NP-hard to find an assignment that satisfies (even) an ε fraction of the constraints.

We note that a 2-to-2 constraint $x_i \oplus x_j \in \{c_{ij}, c'_{ij}\}$ can be replaced by two "unique" constraints $x_i \oplus x_j = c_{ij}$ and $x_i \oplus x_j = c'_{ij}$. This automatically gives a "half-way" version of the Unique Games Conjecture that given a $(\frac{1}{2} - \varepsilon)$-satisfiable Unique Games instance, it is NP-hard to find a ε-satisfying assignment. All the arguments that were cited as evidence against the Unique Games Conjecture apply to the "half-way" version as well and

the latter we now know to be NP-hard! Thus we now have, in author's opinion, a strong evidence towards the correctness of the Unique Games Conjecture.

An interesting aspect of the proof of the 2-to-2 Games Theorem is that it led to new Fourier analytic tools and an understanding of expansion in the so-called Grassmann graph. The Grassmann graph $\mathsf{Gr}_{k,\ell}$ is defined as follows. Its vertex set consists of all ℓ-dimensional subspaces L of $\mathbb{F}_2^k$ and (L, L') is an edge if and only if $\mathsf{dim}(L \cap L') = \ell - 1$. Expansion $\Phi(S)$ of a set of vertices S is the probability that picking a random vertex in S and a random edge out of it, one lands outside of S. A set S is said to have imperfect expansion if $\phi(S) \leqslant 1 - \varepsilon$ for some constant $\varepsilon > 0$. In [38, 69], a complete characterization of sets of imperfect expansion is obtained, which completes the proof of the 2-to-2 Games Conjecture proposed earlier in [68, 39]. The subsequent sections of this article explain, in more detail, the overall context and the combinatorial aspects involving the Grassmann graph.

2. Preliminary background

In this section, we review the basic notions and the context leading to the proof of the 2-to-2 Games Conjecture. While much of this is repeating what has already been said in the introductory section, we hope that the reader benefits from a more detailed exposition.

2.1. P, NP, and computational (hardness) hypotheses. The class P consists of all computational problems that can be solved in deterministic polynomial time. A related class BPP, which includes P, consists of all computational problems that can be solved in randomized polynomial time. Here the algorithm is allowed to use randomness and its output is guaranteed to be correct with high probability. BPP is viewed as the class of *efficiently* solvable problems; as far as efficiency is concerned, we do not mind if an algorithm is deterministic or randomized. There is a strong indication that P and BPP could coincide and that every randomized algorithm could be turned into a deterministic one up to a polynomial loss in running time (see [55] for a survey)!

The class NP consists of all computational problems that can be solved in nondeterministic polynomial time. Equivalently, NP is the class of problems for which their solution can be "checked" in deterministic polynomial time. We avoid entering into a discussion of nondeterminism and proof checking and instead work with concrete, canonical problems that capture the essence of the class NP. A problem $\mathcal{I}$ is called NP-complete if firstly, it is in NP and secondly, it is "at least as hard as" every other problem in NP, meaning every other problem in NP can be "reduced" to the single problem $\mathcal{I}$. It is a remarkable fact (the well-known Cook-Levin Theorem) that NP-complete problems exist and moreover many natural problems happen to be NP-complete. Perhaps the most well-known and canonical NP-complete problem is $\mathsf{3SAT}$.

DEFINITION 2.1. *A* 3SAT *instance* ϕ *consists of* n *Boolean, i.e.* {True, False}*-valued, variables* $x_1, \ldots, x_n$ *and* m *clauses* $\{C_1, \ldots, C_m\}$*. Each clause is of the form* $\ell_i \vee \ell_j \vee \ell_k$ *for distinct* i*,* j*,* k*. The* ℓ_i *is referred to as a literal which is either a variable* x_i *or in negated form* $\overline{x}_i$*. The goal is to decide whether there exists a* {True, False} *assignment to the variables that satisfies all the clauses (referred to as a satisfying assignment) and to find one if one exists.*

We now make some clarifying comments which are taken for granted in TCS literature, but which the reader might not be familiar with. An instance ϕ is called satisfiable if there exists a satisfying assignment and unsatisfiable otherwise.

- **Exact versus Approximate:** We will sometimes refer to the problem as Exact-3SAT to emphasize that the goal here is to find an *exact* solution, i.e. an assignment that satisfies *all* clauses (if one exists). Later we will consider the approximation problem where the goal is to find an assignment that satisfies say 90% of the clauses given that there exists a (fully) satisfying assignment.
- **Search versus Decision (Distinguishing) Problem:** The problem is stated as a search problem, where we actually seek a satisfying assignment (if one exists). In the decision version of the problem, we *only* seek to *decide* (*distinguish*) whether the instance ϕ is satisfiable or not, the answer being just Yes or No. Formally speaking, the classes P, BPP, NP are defined as classes of decision problems and formally, it is the decision version of 3SAT that is NP-complete.

  In this article, we do not differentiate between search and decision versions of problems since these are "morally" the same. For example, for Exact-3SAT, the search version formally reduces to the decision version (as easy fact) and hence the two versions are polynomial time equivalent to each other. Moreover, we will be mostly concerned with hardness of problems and showing that the decision version is hard makes the result only better.
- **NP-complete versus NP-hard:** We find it more convenient to use a related term "NP-hard". A NP-hard problem is at least as hard as every problem in NP but need not be in NP itself. However all the problems in this article will be in NP and hence usage of the term "NP-hard" will be synonymous with that of "NP-complete".
- **Reductions:** To show that a problem is NP-hard, we show that some NP-hard problem, say Exact-3SAT, reduces to it. Consider the Third-Clique problem for instance. A clique in a graph is a subset of vertices in which all pairs of vertices are connected by an edge. In the Third-Clique problem, given a graph G with $3n$ vertices, the goal is to decide whether there exists a clique of size n. This is a well-known NP-hard problem and a reduction from Exact-3SAT to Third-

Clique appears in standard textbooks. Specifically, a reduction here is a polynomial time algorithm that, given a Exact-3SAT instance ϕ, constructs a Third-Clique instance G, such that ϕ is satisfiable if and only if G has a clique of size one-third.

We now state some standard hypotheses in TCS. These are viewed as increasingly stronger hypotheses: for Exact-3SAT, these hypotheses successively state that an algorithm (deterministic or randomized) must take time that is super-polynomial, exponential (that is 2^{n^γ}), and "truly exponential" (that is $2^{\gamma n}$).

- P $\neq$ NP, BPP $\neq$ NP.
- NP $\not\subseteq \cap_{\gamma>0}\text{TIME}(2^{n^\gamma})$. In words, an algorithm (deterministic or randomized) for a NP-hard problem must take time 2^{n^γ} for some constant $\gamma > 0$ (that may depend on the problem).
- Exponential Time Hypothesis [56, 57]: An algorithm (deterministic or randomized) for Exact-3SAT with n variables must take time $2^{\gamma n}$ for some absolute constant $\gamma > 0$.

Some remarks are in order:

- The relationship between BPP and NP is a bit mysterious and we don't know for sure whether BPP is contained in NP. However, as mentioned before, P and BPP could very well coincide.
- Most TCS researchers strongly believe that P $\neq$ NP and also that NP $\not\subseteq \cap_{\gamma>0}\text{TIME}(2^{n^\gamma})$. The confidence in the Exponential Time Hypothesis is less but it is still viewed as a reasonable hypothesis. We note that Exact-3SAT does have a trivial 2^n-time algorithm that goes over all 2^n assignments and hence the Exponential Time Hypothesis is quite strong.
- Most hardness results in TCS rely on such hypotheses. While the situation is less than ideal, actually proving such hypotheses is exceedingly difficult, and TCS researchers choose to further their understanding modulo such hypotheses rather than not proceeding further at all!
- In this article, all hardness results are NP-hardness results and hence rely, by default, on the hypothesis P $\neq$ NP. Stated differently, if it were the case that P = NP, then all problems in this article would have efficient algorithms, none would be hard, leaving nothing to talk about!

2.2. Approximation algorithms and hardness of approximation. Since we do not hope to efficiently solve NP-hard problems exactly, a natural next step is to seek approximations. Let $\mathcal{I}$ denote an NP-hard problem. For an instance I of the problem with input size n, let $\mathsf{OPT}(I)$ denote the value of the optimal solution. For example, for a 3SAT instance ϕ, $\mathsf{OPT}(\phi)$ will denote the maximum fraction of clauses satisfied by any (that is the "best possible") assignment.

For a specific polynomial time approximation algorithm, let $\mathsf{ALG}(I)$ denote the value of the solution that the algorithm finds (or its expected value if the algorithm is randomized). Let $C > 1$ be a parameter that could be a function of n and regarded as the approximation factor (closer C is to 1, the better the approximation as per definition below).

DEFINITION 2.2. *An algorithm achieves an approximation factor of C if on every instance I,*

$$\begin{aligned} \mathsf{ALG}(I) &\geqslant \tfrac{1}{C} \cdot \mathsf{OPT}(I) && \text{if } \mathcal{I} \text{ is a maximization problem,} \\ \mathsf{ALG}(I) &\leqslant C \cdot \mathsf{OPT}(I) && \text{if } \mathcal{I} \text{ is a minimization problem.} \end{aligned}$$

We refer the reader to Vazirani's book [99] for an extensive treatment on approximation algorithms. We now indicate how hardness of approximation results are proved. In short, by a reduction as always! More specifically, a maximization problem $\mathcal{I}$ is proved to be hard to approximate by giving a reduction from a canonical NP-hard problem such as Exact-3SAT to a *gap version* of $\mathcal{I}$.

DEFINITION 2.3. *Let $0 < s < c$ be parameters. A (c, s)-gap version of a maximization problem $\mathcal{I}$, denoted as $\mathsf{Gap}\,\mathcal{I}_{c,s}$, is a promise problem where it is guaranteed that either $\mathsf{OPT}(I) \geqslant c$ or $\mathsf{OPT}(I) \leqslant s$ and the problem is to distinguish between the two cases.*

We say that $\mathsf{Gap}\,\mathcal{I}_{c,s}$ is NP-hard if there is a polynomial time reduction from Exact-3SAT to $\mathsf{Gap}\mathcal{I}_{c,s}$, i.e. a polynomial time reduction that given a Exact-3SAT instance ϕ, constructs an instance I of the problem $\mathcal{I}$ such that:

- (Yes/Completeness Case): If ϕ has a satisfying assignment, then $\mathsf{OPT}(I) \geqslant c$.
- (No/Soundness Case): If ϕ has no satisfying assignment, then $\mathsf{OPT}(I) \leqslant s$.

Such a reduction implies that approximating $\mathcal{I}$ within factor less than $\frac{c}{s}$ is NP-hard. While this is self-evident, we elaborate for clarity. Suppose on the contrary that there is an algorithm for $\mathcal{I}$ with approximation factor less than $\frac{c}{s}$. We show that $\mathcal{A}$ can be used to design an algorithm that decides whether ϕ is satisfiable, which is a NP-hard problem. Indeed, given ϕ, construct the instance I via the reduction, run the algorithm $\mathcal{A}$ on I, declare Yes if the value of the solution output by $\mathcal{A}$ is $> s$, and declare No otherwise. Clearly, if ϕ is satisfiable, the reduction guarantees that $\mathsf{OPT}(I) \geqslant c$, and since $\mathcal{A}$ has approximation factor less than $\frac{c}{s}$, it outputs a solution with value $> s$, and the overall algorithm declares Yes. On the other hand, if ϕ is unsatisfiable, the reduction guarantees that $\mathsf{OPT}(I) \leqslant s$, and the value of the solution output by $\mathcal{A}$ is necessarily $\leqslant s$, and the overall algorithm declares No.

We note here that hardness of approximation results show that the decision problem $\mathsf{Gap}\,\mathcal{I}_{c,s}$ is NP-hard. It follows that the search problem, namely, the task of finding a solution with value (at least) s given an instance with

optimum (at least) c, is also NP-hard (and hence we ignore the distinction between decision and search). Hardness of approximation results for minimization problems can be proved in a similar way. Sometimes, for a maximization problem, it is more convenient to talk of c-approximation for $c < 1$, meaning $\mathsf{ALG}(I) \geqslant c \cdot \mathsf{OPT}(I)$ (it will be clear from the context).

2.3. The PCP theorem. In practice, a reduction as described above is actually a sequence of (potentially very involved) reductions. The first reduction in the sequence is the well-known PCP Theorem [**42**, **8**, **6**] discovered in early 1990s. In this sense, the PCP Theorem is the "mother of all hardness results". It yields a mild hardness result which is then amplified by subsequent reductions.

The PCP Theorem can be phrased as a reduction from Exact-3SAT to a gap version of 3SAT. For a 3SAT formula ϕ, let $\mathsf{OPT}(\phi)$ denote, as before, the maximum fraction of clauses that can be satisfied by any assignment. Thus $\mathsf{OPT}(\phi) = 1$ if and only if ϕ is satisfiable. The PCP Theorem states that for some universal constant $\theta < 1$, there is a (explicitly described) polynomial time reduction from Exact-3SAT to $\mathsf{Gap\text{-}3SAT}_{1,\theta}$. More specifically, the reduction maps a Exact-3SAT instance ϕ to another 3SAT instance ψ such that:

- (Yes/Completeness Case): If $\mathsf{OPT}(\phi) = 1$, then $\mathsf{OPT}(\psi) = 1$.
- (No/Soundness Case): If $\mathsf{OPT}(\phi) < 1$, then $\mathsf{OPT}(\psi) \leqslant \theta$.

We stress that in the Yes Case, it could be that ϕ has m clauses and $\mathsf{OPT}(\phi) = 1 - \frac{1}{m}$, that is there could be an assignment that satisfies *all but one* clause. However, in this case, any assignment to ψ fails on a *constant fraction* of its clauses. In other words, the reduction maps satisfiable instances to satisfiable ones (Yes Case) and unsatisfiable ones to *highly unsatisfiable* ones (No Case). We remark that the PCP Theorem is viewed as one of the landmarks in TCS and its original proof was rather involved and algebraic. A simpler, combinatorial proof was discovered much later in 2006 [**35**].

We stated the PCP Theorem as a reduction. There is an equivalent formulation of it in terms of *proof checking*. In fact, the proof checking viewpoint played a major role in its discovery and is very surprising. In this formulation, The PCP Theorem states that every NP statement has a polynomial size proof that can be checked by a probabilistic polynomial time verifier by reading only a constant number of bits in the proof! The verifier has the completeness and the soundness property: (completeness) every correct statement has a proof that is accepted with probability 1 and (soundness) every proof of an incorrect statement is accepted with only a small probability, say at most 1%.

The equivalence between the two views, namely reduction versus proof checking, is simple but illuminating, and has influenced much of the work in this area. In this article, we mostly stick to the reduction viewpoint for clarity.

2.4. Towards optimal hardness results: some successes. The PCP Theorem, as stated above, shows that 3SAT is hard to approximate within factor $\frac{1}{\theta} > 1$. After the discovery of the PCP Theorem, the focus shifted to proving optimal results, that is proving hardness results that match the best known algorithmic results. By late 1990s, some major successes were achieved in this regard, obtaining optimal hardness results for 3SAT [52], Clique [51], Set Cover [82, 41] (and also 3Lin [52] as described in the introductory section). We state these results below. We do not get into the details of their proofs, but for reader's benefit, we mention that some of the important developments included the introduction of the 2-Prover-1-Round Games problem [3] (of which Unique Games is a special case), the Parallel Repetition Theorem [91], the introduction of the Long Code and the basic reduction framework using Long Code [20], and use of Fourier analysis in analyzing the Long Code [51, 52].

2.4.1. 3Lin. The 3Lin problem was described in the introductory section. The optimum of a given instance is the maximum fraction of the equations satisfied by any assignment. The following result was also described in the introductory section [52].

THEOREM 2.4. *For an arbitrarily small constant $\varepsilon > 0$,* $\mathsf{Gap3Lin}_{1-\varepsilon,\frac{1}{2}+\varepsilon}$ *is NP-hard.*

2.4.2. 3SAT. Given a 3SAT instance ϕ, the goal here is to find an assignment that satisfies a maximum fraction of clauses (the optimum being this maximum). There exists a trivial $\frac{7}{8}$-approximation algorithm: just assign True or False at random to every variable. Out of 8 possible assignments to every clause of type $x_i \vee x_j \vee x_k$, 7 are satisfying assignments. Therefore a random assignment satisfies $\frac{7}{8}$ fraction of the clauses in expectation. Since the best possible assignment could satisfy (at most) all the clauses, this qualifies as $\frac{7}{8}$-approximation. Quite surprisingly, no polynomial time algorithm can hope to do better [52]!

THEOREM 2.5. *For an arbitrarily small constant $\varepsilon > 0$,* $\mathsf{Gap3SAT}_{1,\frac{7}{8}+\varepsilon}$ *is NP-hard.*

We emphasize here that the PCP Theorem, as noted, *only* shows that $\mathsf{Gap3SAT}_{1,\theta}$ is NP-hard for *some* absolute constant $\theta < 1$. One takes this as a starting point, builds a further reduction on top of it, and shows that in fact $\mathsf{Gap3SAT}_{1,\theta}$ is NP-hard for *every* $\theta = \frac{7}{8} + \varepsilon$. This is a remarkable self-improving, gap-amplifying reduction. The tools used (here as well as in most reductions post PCP Theorem) include, as mentioned, the Parallel Repetition Theorem, the Long Code, and Fourier analysis over the Boolean hypercube.

2.4.3. Clique.

DEFINITION 2.6. *An instance of the* Clique *problem is a n-vertex graph $G(V, E)$. The goal is to find a clique of maximum size (the optimum being*

this maximum size). A clique in a graph is a subset S of vertices such that there is an edge between every pair of vertices in S.

Clique is a notorious problem. The best known algorithm can achieve only a weak guarantee: given a graph that is promised to contain a clique of size $\frac{n}{\log n}$, the algorithm manages to find a clique of size $\log^2 n$ (up to $\log \log n$ factor). Again, no polynomial time algorithm can hope to do much better [51]!

THEOREM 2.7. *For an arbitrarily small constant $\varepsilon > 0$,* $\mathsf{GapClique}_{n^{1-\varepsilon},n^{\varepsilon}}$ *is NP-hard on n-vertex graphs.*

2.4.4. Set Cover.

DEFINITION 2.8. *An instance $\mathcal{F}$ of* Set Cover *consists of a ground set U with n elements and a family of subsets $\{S_1, \ldots, S_m\}$ such that $\cup_{i=1}^{m} S_i = U$. The goal is to find a set cover of minimum size (the optimum being this minimum size). A set cover is a sub-family $\{S_{i_1}, \ldots, S_{i_r}\}$ whose union still equals U.*

There is a $\ln n$ approximation algorithm for Set Cover. The algorithm greedily picks the largest set, removes it along with the elements it covers, and then repeats (by next picking a set that covers the largest number of remaining elements). Showing that this gives a $\ln n$ approximation requires a short, neat argument and can be found in standard text [99]. Again, no polynomial time algorithm can hope to do much better [41]![4]

THEOREM 2.9. *For an arbitrarily small constant $\varepsilon > 0$,* $\mathsf{GapSetCover}_{\frac{n}{k},(1-\varepsilon)\ln n \cdot \frac{n}{k}}$ *is NP-hard on* Set Cover *instances with n elements and all sets of size exactly k.*

2.5. Towards optimal hardness results: many challenges. In spite of the successes in hardness of approximation just mentioned, the goal of establishing optimal hardness results remains elusive for most problems of interest. We cite here three prominent examples, Vertex Cover, Max-Cut, and Max Acyclic Subgraph.

DEFINITION 2.10. *An instance of* Vertex Cover *is a graph $G(V, E)$. The goal is to find a vertex cover of minimum size (the optimum being this minimum size, but as a fraction relative to the total number of vertices in the graph). A vertex cover is a subset of vertices C such that for each edge $(u, v) \in E$, either $u \in C$ or $v \in C$ (or both).*

There is a trivial 2-approximation algorithm for Vertex Cover as described in the introductory section. It is a major challenge whether or not there is a better algorithm.

[4]The result is quoted here as NP-hardness result. Strictly speaking, the reduction runs in slightly super-polynomial time and hence one requires the hypothesis NP $\not\subseteq$ TIME $(n^{O(\log \log n)})$.

DEFINITION 2.11. *An instance of* Max-Cut *is a graph* $G(V, E)$. *The goal is to find a cut of maximum size (the optimum being this maximum size, but as a fraction relative to the total number of edges in the graph). A cut is a partition of vertices into sets* $(S, V \setminus S)$ *and its size is the number of edges "cut" (that is with one endpoint in* S *and the other in* $V \setminus S$*).*

For Max-Cut, there is a well-known α_{GW}-approximation by Goemans and Williamson [47] where $\alpha_{GW} \approx 0.878$ is the constant defined as below. The algorithm is based on "semi-definite programming relaxation" and was the first time semi-definite programming was used to design approximation algorithms.

$$\alpha_{GW} = \min_{\theta \in [0,\pi]} \left(\frac{2}{\pi} \frac{\theta}{1 - \cos\theta} \right). \tag{1}$$

Since the discovery of this algorithm, it has remained a challenge whether or not there is a better algorithm.

DEFINITION 2.12. *An instance of* Max Acyclic Subgraph *is a directed graph* $G(V, E)$. *The goal is to find an acyclic subgraph* $G(V, E_{\mathsf{sub}}), E_{\mathsf{sub}} \subseteq E$ *of maximum size (the optimum being this maximum size, but as a fraction relative to the total number of edges in the graph, i.e.* $\frac{|E_{\mathsf{sub}}|}{|E|}$*).*

There is a trivial $\frac{1}{2}$-approximation: simply order the vertices on a line arbitrarily and let E_{sub} be the set of all edges either going forward or going backward, whichever is larger. Clearly this yields an acyclic subgraph containing at least half of the edges whereas the maximum acyclic subgraph can (at most) have all the edges. It is a major challenge whether or not there is a better algorithm.

On all the three problems above (and many more), we neither know a better algorithm nor a matching hardness result. Until the Unique Games Conjecture was proposed in 2002 and the subsequent developments, there wasn't even a plausible argument why the answer should be one way or the other[5] and the researchers were stuck, unable to make progress in either direction.

2.6. The Unique Games Conjecture. The Unique Games Conjecture [63] was proposed to break the aforementioned impasse in study of approximability. A problem called Unique Games was proposed as a hard to approximate problem and proposed as a canonical problem to reduce to other problems establishing their hardness. The full potential of the conjecture however became apparent only in the subsequent decade. We briefly outline below the developments spawned by the research on the Unique Games Conjecture (please see surveys [97, 64, 62, 65]).

[5]As far as the author knows, several researchers believed that Max-Cut would have a better algorithm and possibly Vertex Cover too.

2.6.1. *Hardness of approximation.* The main motivation for the conjecture is to prove hardness results for problems that researchers have been unable to prove otherwise. The conjecture states that a specific computational problem called the Unique Games is hard to approximate. A reduction from this problem then implies hardness results for other NP-hard problems. Such a reduction was exhibited in [**63**] for the Min-2SAT-Deletion problem and it was soon followed by a flurry of reductions for various problems, in some cases using variants of the conjecture. In particular, it is now known that the Unique Games Conjecture implies optimal $2-\varepsilon$, $\alpha_{GW}+\varepsilon$, and $\frac{1}{2}+\varepsilon$ hardness results for Vertex Cover, Max-Cut, and Max Acyclic Subgraph problems ([**74**], [**66**], [**50**] respectively) and hence these problems do not have (modulo the conjecture) better algorithms than mentioned in the previous section!

THEOREM 2.13. *Assuming the Unique Games Conjecture, for an arbitrarily small constant $\varepsilon > 0$,*

- Gap Vertex Cover $_{\frac{1}{2}+\varepsilon,\ 1-\varepsilon}$ *is NP-hard.*
- Gap Max Cut $_{\frac{1-\cos\theta_c}{2}-\varepsilon,\ \frac{\theta_c}{\pi}+\varepsilon}$ *is NP-hard where θ_c is the critical angle that minimizes the ratio in Equation* (1).
- Gap Max Acyclic Subgraph$_{1-\varepsilon,\ \frac{1}{2}+\varepsilon}$ *is NP-hard.*

Remarkably, there is even a general result by Raghavendra [**89**] that gives a semi-definite programming based algorithm and a matching hardness result under the Unique Games Conjecture for the *entire class* of so-called Constraint Satisfaction Problems (see Section 2.9 for a definition).

2.6.2. *Discrete Fourier analysis.* The reductions from the Unique Games problem often use "gadgets" constructed from a Boolean hypercube. The gadgets can be viewed as probabilistic checking procedures to check whether a given codeword is a Long Code (or a Hadamard Code or a Grassmann Code as is the case in this article regarding the proof of the 2-to-2 Games Theorem). Fourier analytic theorems on hypercube play a crucial role in ensuring that the gadgets indeed "work". Such theorems were either already known before (e.g. the Kahn Kalai Linial (KKL) Theorem [**58**] and Friedgut's Theorem [**45**]) or invented specifically towards application to hardness of approximation. We cite an example below, namely, the Majority Is Stablest Theorem [**66, 84**], along with the minimal definitions needed for its statement. Here the intended application was to the Max Cut problem as in Theorem 2.13. More examples include Bourgain's Junta Theorem [**23**] (the intended application was to the Min-2SAT-Deletion problem) and the Grassmann Expansion Theorem in this article, Theorem 4.6, the intended application being the 2-to-2 Games problem.

Let us consider Boolean functions $f : \{-1,1\}^n \to \{-1,1\}$ with $\mathbb{E}[f] = 0$. For a coordinate $1 \leqslant i \leqslant n$, its influence on the function $I_i[f]$ is defined as the probability that on a random input $x \in \{-1,1\}^n$, changing the ith coordinate of x changes the value of the function. For a parameter $\beta \in (0, \frac{1}{2})$, let $\mathsf{NS}_\beta(f)$ denote the noise-sensitivity of f at noise-rate β, defined as the

probability that $f(x) \neq f(y)$ when $x \in \{-1,1\}^n$ is a random input and y is obtained by changing every coordinate of x independently with probability β. Finally, let $\mathsf{Majority}_n$ denote the majority function on $\{-1,1\}^n$ where the output is -1 or 1 whichever is in majority among the input bits (and say -1 in case of a tie). The Majority Is Stablest Theorem states, roughly speaking, that among the functions whose all influences are small, Majority is the most stable function under noise. Formally,

THEOREM 2.14. *For every $\beta \in (0, \frac{1}{2})$ and $\varepsilon > 0$, there exists a sufficiently small constant $\tau > 0$ such that the following holds: for any function $f : \{-1,1\}^n \to \{-1,1\}$, $\mathbb{E}[f] = 0$ and $\forall\ 1 \leqslant i \leqslant n$, $I_i[f] \leqslant \tau$, it holds that*

$$\mathsf{NS}_\beta(f) \geqslant \left(\lim_{n\to\infty} \mathsf{NS}_\beta(\mathsf{Majority}_n)\right) - \varepsilon \ = \frac{2}{\pi}\arccos(1-2\beta) - \varepsilon.$$

2.6.3. *Geometry.* The task of proving a Fourier analytic theorem on the Boolean hypercube can sometimes be reduced to an isoperimetry type question in geometry. This step relies on powerful invariance style theorems [**93**, **84**, **28**, **85**], some of which were motivated in a large part by their application to hardness of approximation. The geometric questions, in turn, are either already studied or are new, and many of these remain challenging open questions, the Propeller Conjecture [**71**, **72**] being one such example. Some "integrality gap" constructions, metric non-embeddability results, and the solution to a tiling problem are more examples of connections to geometry, described next.

2.6.4. *Integrality gaps.* For many problems, the Unique Games Conjecture rules out *every* polynomial time algorithm to compute a good approximate solution. One can investigate a less ambitious question: can we rule out algorithms based on a specific linear or semi-definite programming "relaxation"? This amounts to the so-called integrality gap constructions that are explicit combinatorial constructions, often with geometric flavor. While we do not give a formal definition here, roughly speaking, an integrality gap for a (maximization) problem $\mathcal{I}$ with respect to a specific convex (linear or semi-definite) relaxation $\mathsf{Relax}(\cdot)$, is the worst case ratio $\frac{\mathsf{Relax}(I)}{\mathsf{OPT}(I)}$ over all problem instances I. An integrality gap is taken as evidence that the problem is hard to approximate, at least via the specific convex relaxation.

It was demonstrated in [**76**] and subsequent papers that the reduction from the Unique Games problem to a target problem can in fact be used to construct an (unconditional, explicit) integrality gap instance for the target problem. This strategy was used in [**76**] to refute the Goemans-Linial Conjecture in metric geometry (the connection to approximability is via the Graph Partitioning and/or Sparsest Cut problem). We state the result below along with the minimal definitions needed for its statement.

An n-point finite metric $d(\cdot,\cdot)$ is said to be of *negative type* if the metric $\sqrt{d}$ is isometrically embeddable in L_2. Let $c_1(\mathsf{NEG}, n)$ be the least number such that every n-point negative type metric embeds into the class of L_1 metrics with *distortion* $c_1(\mathsf{NEG}, n)$, i.e. preserving all distances up to a factor

of $c_1(\mathsf{NEG}, n)$. Goemans and Linial [48, 81] conjectured that $c_1(\mathsf{NEG}, n)$ is a universal constant independent of n. If this were correct, it would have led to $O(1)$-approximation to Graph Partitioning and a host of related problems. However in [76], the authors were able to disprove the Goemans and Linial conjecture, as stated below. The lower bound is now known to be $\Omega(\sqrt{\log n})$ and is optimal, via an alternate construction based on the geometry of the Heisenberg group, proved in an amazing sequence of papers [80, 31, 30, 33, 32, 87].

THEOREM 2.15. $c_1(\mathsf{NEG}, n) \geqslant \Omega((\log \log n)^{1/8})$.

Another striking example in the context of integrality gaps is the result of Raghavendra [89] mentioned before. It shows duality between approximation algorithms and hardness reductions as far as Constraint Satisfaction Problems are concerned: a natural semi-definite programming relaxation leads to an algorithm and surprisingly, an integrality gap instance for the same relaxation leads to a hardness reduction!

2.6.5. *Algorithms and parallel repetition.* Attempts to prove or disprove the Unique Games Conjecture have led to some very nice algorithms [27, 98, 11, 9], a connection to the small set expansion problem in graphs [88, 9], a deeper understanding of the parallel repetition theorem [92], and solution to a tiling problem in Euclidean space [44, 77, 2]. We describe the last one as an example (without explaining how it was inspired by musings about the Unique Games Conjecture).

A Tiling Problem: Let $S \subseteq \mathbb{R}^d$ be a "shape" with unit volume such that its translations by $\mathbb{Z}^d$ tile $\mathbb{R}^d$. What is the least surface area of such a shape?

Clearly, the surface area of a tiling shape S must at least be that of the unit volume ball, namely $\Omega(\sqrt{d})$. But the unit volume ball is not a legal tiling shape. On the other hand, a unit volume cube is a legal tiling shape and has surface area of $2d$. The best known constructions were only a constant factor better than the cube. Based on a counter-example to *strong parallel repetition* for Unique Games [92], one now knows a tiling shape whose surface area is $O(\sqrt{d})$ [77, 2]!

2.6.6. *The statement of the Unique Games Conjecture.* We now state the Unique Games Conjecture formally. For the purposes of this article, it suffices to define Unique Games as the following computational problem. Let $\mathbb{F}_2^\ell$ denote the ℓ-dimensional vector space over the binary field $\mathbb{F}_2$, considered as an additive group with the $\oplus$ operation.

DEFINITION 2.16. *An instance $\mathcal{U}$ of the* UniqueGames$[\mathbb{F}_2^\ell]$ *problem consists of n variables $x_1, \ldots, x_n$ taking values over (the alphabet) $\mathbb{F}_2^\ell$ and m constraints $C_1, \ldots, C_m$ where each constraint C_i is a linear equation of form $x_{i_1} \oplus x_{i_2} = b_i$ and $b_i \in \mathbb{F}_2^\ell$. Let* $\mathsf{OPT}(\mathcal{U})$ *denote the maximum fraction of the constraints that can be satisfied by any assignment to the instance.*

The term "unique" refers to the specific nature of the constraints: for every assignment to the variable x_{i_1}, there is a unique assignment to the

variable x_{i_2} that satisfies the constraint and vice versa (the Unique Games problem was studied earlier by Feige and Lovász in the context of parallel repetition [43]). For constants $0 < s < c < 1$, let $\mathsf{GapUG}[\mathbb{F}_2^\ell](c, s)$ be the gap-version where the instance $\mathcal{U}$ of the $\mathsf{UniqueGames}[\mathbb{F}_2^\ell]$ problem is promised to have either $\mathsf{OPT}(\mathcal{U}) \geqslant c$ or $\mathsf{OPT}(\mathcal{U}) \leqslant s$. The Unique Games Conjecture states that.[6]

CONJECTURE 2.17. *For every constant $\varepsilon > 0$, there exists a sufficiently large integer $\ell = \ell(\varepsilon)$ such that $\mathsf{GapUG}[\mathbb{F}_2^\ell](1-\varepsilon,\ \varepsilon)$ is* NP*-hard.*

2.7. Arguments raised against the Unique Games Conjecture. In spite of the rather large body of work surrounding the Unique Games Conjecture, its correctness itself remains open. In fact, over the last decade, several arguments were put forward against the Unique Games Conjecture. We sketch these arguments here to the best of our knowledge.[7]

- Usually, for a problem that is believed to be hard, there is a distribution (often a "natural" one) over instances that is hard against all known algorithms. For instance, for the Factoring problem, known algorithms fail even on numbers that are products of two random n-bit primes. For the 3SAT-Refutation problem, where the goal is to prove unsatisfiability of a formula, known algorithms fail even on instances that are random, with say n variables and $100n$ clauses.

  However, there is no known distribution over instances of the Unique Games problem that is plausibly hard. In fact, results in [11, 78] showed that the problem is easy on "semi-random" instances (for a rather generous interpretation of the term semi-random), thus indicating otherwise.
- There is a candidate algorithm for solving the Unique Games problem for which there is no known counter-example showing that the algorithm does not work (but there is no proof that the algorithm works either). The algorithm is simply a constant number, say 10, of "rounds" of the Sum-of-Squares (a.k.a. Lasserre, Parrilo) hierarchy of semi-definite programming relaxation [16, 19]. For all we know, this algorithm could already qualify as an efficient algorithm for the problem, disproving the Unique Games Conjecture.

  Counter-examples to LP/SDP-based algorithms are known as integrality gaps as mentioned before. Stated differently, we do not know of a Unique Games integrality gap example for a constant round Sum-of-Squares hierarchy (but see [70] towards a plausible construction of such integrality gap).

[6] The original statement in [63] refers to more general constraints. However it follows from [66] that the original conjecture is equivalent to the statement here, i.e. when the constraints are linear equations over the group $\mathbb{F}_2^\ell$.

[7] Since these arguments were not made by us, we are not taking responsibility as to whether these arguments were indeed made or whether these are/were considered pressing arguments. We present these only for reader's benefit.

- Arora, Barak, Steurer [9, 96] presented an algorithm that runs in time $2^{n^{\varepsilon'}}$ and:
  - On $(1-\varepsilon)$-satisfiable instances of the Unique Games problem, finds say $\frac{3}{4}$-satisfying assignment.
  - On $\frac{1}{2}$-satisfiable instances of the Unique Games problem, finds ε-satisfying assignment.

  Here ε' depends on ε and $\varepsilon' \to 0$ as $\varepsilon \to 0$. If the running time of the algorithm is improved so that $\varepsilon' \to 0$ independently of ε, then the Unique Games problem would not be NP-hard.[8] The improvement could arguably come from a quantitative improvement in the connection between the number of large eigenvalues and expansion of small sets in graphs.
- The Arora, Barak, Steuter algorithm and the Unique Games Conjecture together imply that the Unique Games problem has "intermediate complexity" (see Section 2.9.3), a behavior one might not expect for a constraint satisfaction problem.

The context so far leads finally to the recent line of work: a sequence of papers [68, 39, 38, 69] proved the 2-to-2 Games Conjecture (or alternately the Unique Games Conjecture on $\frac{1}{2}$-satisfiable instances, if the reader finds it more convenient to think about). In conjunction with a missing link provided in [18], the first three papers in the sequence finally reduced the 2-to-2 Games Conjecture to a concrete combinatorial hypothesis regarding the expansion properties of the Grassmann graph (stated as Theorem 4.6). In [69] (using an insight from [67]), the authors proved this combinatorial hypothesis, thus completing the proof of the 2-to-2 Games Conjecture/Theorem. The 2-to-2 Games Theorem gives, among other things, a strong evidence towards the Unique Games Conjecture (in our opinion). All the arguments against the Unique Games Conjecture that we described applied equally well, a priori, to the 2-to-2 Games Conjecture and in spite of it, the 2-to-2 Games Conjecture, at the end of the day, does happen to be correct, somehow circumventing all these arguments!

2.8. The statement of the 2-to-2 Games Theorem. We now state the 2-to-2 Games Theorem formally and describe its significance. This is followed by the description of the Grassmann graph and its role in its proof. We note that the definition below requires constraints that are a bit more general than those described in the introductory section.

DEFINITION 2.18. *An instance $\mathcal{U}_{2\leftrightarrow 2}$ of the* 2-to-2 Games$[\mathbb{F}_2^\ell]$ *problem consists of n variables $x_1, \ldots, x_n$ taking values over (the alphabet) $\mathbb{F}_2^\ell$ and m constraints $C_1, \ldots, C_m$ where each constraint is of the form $T_{ij}x_i \oplus T'_{ij}x_j \in \{b_{ij}, b'_{ij}\}$, T_{ij}, T'_{ij} are $\ell \times \ell$ invertible matrices, and $b_{ij}, b'_{ij} \in \mathbb{F}_2^\ell$. Let*

[8] Under the rather standard hypothesis NP $\not\subseteq \cap_{\gamma>0}$ TIME (2^{n^γ}) mentioned in Section 2.1.

$\mathsf{OPT}(\mathcal{U}_{2\leftrightarrow 2})$ *denote the maximum fraction of the constraints that can be satisfied by any assignment to the instance.*

The term "2-to-2" refers to the specific nature of the constraints: for every assignment to the variable x_i, there are exactly two assignments to the variable x_j that satisfy the constraint and vice versa. For constants $0 < s < c \leqslant 1$, let $\mathsf{Gap\ 2\text{-}to\text{-}2}[\mathbb{F}_2^\ell](c,s)$ be the gap-version where the instance $\mathcal{U}_{2\leftrightarrow 2}$ of the $\mathsf{2\text{-}to\text{-}2\ Games}[\mathbb{F}_2^\ell]$ problem is promised to have either $\mathsf{OPT}(\mathcal{U}_{2\leftrightarrow 2}) \geqslant c$ or $\mathsf{OPT}(\mathcal{U}_{2\leftrightarrow 2}) \leqslant s$. The 2-to-2 Games Theorem is stated below along with an immediate corollary for the hardness of the Unique Games problem with completeness $\frac{1}{2}$. The latter is obtained by writing each 2-to-2 Games constraint as a pair of Unique Games constraints so that in the completeness case, there is a $\frac{1}{2}(1-\varepsilon)$-satisfying assignment. The completeness can be increased artificially to precisely $\frac{1}{2}$ by adding a small fraction of constraints that are always satisfied.

THEOREM 2.19. *For every constant $\varepsilon > 0$, there exists a sufficiently large integer $\ell = \ell(\varepsilon)$ such that* $\mathsf{Gap\ 2\text{-}to\text{-}2}[\mathbb{F}_2^\ell](1-\varepsilon,\ \varepsilon)$ *is NP-hard.*

THEOREM 2.20. *For every constant $\varepsilon > 0$, there exists a sufficiently large integer $\ell = \ell(\varepsilon)$ such that* $\mathsf{Gap\ UG}[\mathbb{F}_2^\ell](\frac{1}{2},\ \varepsilon)$ *is NP-hard.*

2.9. Significance of the 2-to-2 Games Theorem. We now summarize the main implications of the 2-to-2 Games Theorem (some of these implications depend on its specific proof). As before, $\varepsilon > 0$ denotes a constant that can be taken as arbitrarily small.

2.9.1. *Hardness results.* The following results were already known based on the 2-to-2 Games Conjecture. With the proof of the latter, we now know these results unconditionally. These represent a big progress, in our opinion, on flagship problems in approximability.

- **Max Cut Close to Half**

  A special case of the Max Cut problem is when the optimal cut in the graph cuts $\frac{1}{2}+\varepsilon$ fraction of the edges. In this case, the best known approximation algorithm yields a cut of size $\frac{1}{2}+\Omega\left(\frac{\varepsilon}{\log(1/\varepsilon)}\right)$. From [73], we now know that this is optimal up to the constant in the $\Omega(\cdot)$ notation, i.e. that finding a cut of size $\frac{1}{2}+\Omega\left(\frac{\varepsilon}{\log(1/\varepsilon)}\right)$ is NP-hard.
- **Vertex Cover and Independent Sets of Linear Size**

  An independent set in a graph $G(V,E)$ is a subset of vertices S such that there is no edge between any pair of vertices $u, v \in S$. Clearly a set S is an independent set if and only if the set $V \setminus S$ is a vertex cover.

  From [63], we now know that given an n-vertex graph that contains an independent set of size $\left(1-\frac{1}{\sqrt{2}}-\varepsilon\right)n$, it is NP-hard to find an independent set of size εn. As a corollary, Vertex Cover is

NP-hard to approximate within a factor strictly less than $\sqrt{2}$ (the previous best hardness result being ≈ 1.36 [36]).

- **Almost Graph Coloring**

  Coloring 3-colorable n-vertex graphs is perhaps the most notorious problem at the intersection of TCS and combinatorics. The best known polynomial time algorithms need n^c colors with the exponent c improving over the years, $c \approx 0.199$ being the current best [10, 61]. From the hardness side, we only know that it is NP-hard to color 3-colorable graphs with 5 colors, leaving a huge gap between the upper and lower bounds [26].

  From the hardness side, showing NP-hardness of coloring 3-colorable graphs with any constant number of colors is considered a holy grail. From [40], we now *almost* achieve this holy grail. A catch is that the result doesn't quite hold for 3-colorable graphs but holds for "almost 4-colorable" graphs (the concern is of second order in our mind). Specifically, it is NP-hard to distinguish whether a graph has four disjoint independent sets of size $(\frac{1}{4} - \varepsilon)n$ each (and hence is almost 4-colorable) or whether there is no independent set of size εn (and hence cannot be colored with $\frac{1}{\varepsilon}$ colors).

- **Independent Set on Degree-d Graphs**

  On graphs of degree d (thought of as a large growing constant), the best known approximation algorithm [13, 14] approximates the size of the maximum independent set within a factor $\tilde{O}(\frac{d}{\log^2 d})$, the $\tilde{O}(\cdot)$ notation hiding a factor polynomial in $\log \log d$. From [12, 21], we now know that this is essentially optimal, i.e. that it is NP-hard to find a $O(\frac{d}{\log^2 d})$-approximation. This is a remarkable example where both the approximation algorithm as well as the hardness result are highly non-trivial and the results essentially match! The algorithm is based on a hierarchy of SDP-relaxation whereas the hardness result requires the 2-to-2 Games Theorem.

2.9.2. *Integrality gaps and hard distributions.* The proof of the 2-to-2 Games Theorem gives a reduction from the 3Lin problem to the 2-to-2 Games problem and subsequently to the Unique Games problem with completeness $\frac{1}{2}$. Denoting either of these problems by $\mathcal{P}$, the reduction can be used

- To "translate" an integrality gap instance of the 3Lin problem (as in [49, 95]) to an integrality gap instance of the problem $\mathcal{P}$.
- To "translate" a distribution over 3Lin instances that is plausibly hard (e.g. random instances with appropriate parameters) to a distribution over $\mathcal{P}$ instances that is plausibly hard.

In both cases, we do not know an alternate construction, i.e. without having to go through a NP-hardness reduction (and lack of any construction so far was an argument against the Unique Games Conjecture as discussed before). On the other hand, "logically", integrality gap construction (and maybe construction of a plausibly hard distribution as well) ought to *precede* an

NP-hardness reduction, since these are lesser standards of hardness than the "gold standard" of NP-hardness. We find this phenomenon quite interesting.

2.9.3. *Intermediate complexity theorem.* The Constraint Satisfaction Problems are arguably the most well-studied family of computational problems, 3SAT being one example. We give a rough definition. For a k-ary predicate $P : [q]^k \to \{\mathsf{True}, \mathsf{False}\}$, an instance of a $\mathsf{CSP}(P)$ consists of variables $x_1, \ldots, x_n$ over an alphabet $[q] = \{0, 1, \ldots, q-1\}$ and a set of constraints $C_1, \ldots, C_m$ where each constraint C_i is of the type $P(x_{i_1}, \ldots, x_{i_k})$, i.e. the predicate P applied to some variables $x_{i_1}, \ldots, x_{i_k}$. It should be evident that 3SAT, 3Lin, Max Cut, Unique Games and 2-to-2 Games are all constraint satisfaction problems.

In the exact version, denoted $\mathsf{Exact\text{-}CSP}(P)$, the goal is to determine whether there is an assignment that satisfies all constraints. We note that there is always the brute-force algorithm that checks all possible assignments to the n variables and runs in time $2^{O(n)}$. The well-known Dichotomy Theorem established in a recent breakthrough [**25, 100**] shows that every $\mathsf{Exact\text{-}CSP}(P)$ either has a polynomial time algorithm or is NP-hard. Moreover, those that are NP-hard, cannot be solved in time $2^{\gamma n}$, a remarkable conclusion considering that the brute-force algorithm takes $2^{O(n)}$ time! The conclusion depends on the Exponential Time Hypothesis (ETH) stated in Section 2.1. We recall its statement that Exact-3SAT cannot be solved in time $2^{\gamma n}$ for some $\gamma > 0$. The Dichotomy Theorem gives, for Exact-CSPs that are NP-hard, a linear time reduction from Exact-3SAT, so the same holds for them as well.

In the approximate version, denoted $\mathsf{GapCSP}(P)_{c,s}$ for parameters $0 < s < c \leqslant 1$, the goal is to distinguish whether the instance is c-satisfiable or at most s-satisfiable. Before the proof of the 2-to-2 Games Theorem, it seemed that all such problems either have a polynomial time algorithm or[9] that the problem requires time essentially $2^{\gamma n}$. That is, all the past experience suggested that no CSP has "intermediate complexity", say between super-polynomial and $2^{n^{0.10}}$. Stated differently, all the past experience suggested that if a CSP has an algorithm that runs in time say $2^{n^{0.10}}$, it in fact should have a polynomial time algorithm. Since Arora, Barak, Steuter [**9, 96**] do give an algorithm for $\mathsf{GapUG}_{\frac{1}{2},\varepsilon}$ that runs in time 2^{n^β} ($\beta \to 0$ as $\varepsilon \to 0$), one could have argued that Unique Games should have a polynomial time algorithm, casting doubt on its correctness.

However, the 2-to-2 Games Theorem now shows that an approximation-CSP with intermediate complexity exists, namely $\mathsf{GapUG}_{\frac{1}{2},\varepsilon}$. The theorem shows that it is NP-hard and hence cannot be solved in time $2^{n^{\beta'}}$ for some $\beta' > 0$ (under the Exponential Time Hypothesis or under NP $\not\subseteq \cap_{\gamma>0}$ TIME (2^{n^γ})). Since it can be solved in time 2^{n^β} for some $\beta > 0$ by Arora, Barak,

[9]The evidence here is a near-linear time reduction from Exact-3SAT or in some cases an integrality gap on random instances.

Steurer, its true complexity is somewhere between $2^{n^{\beta'}}$ and $2^{n^{\beta}}$! Existence of such CSP with intermediate complexity is viewed as a most interesting consequence of the 2-to-2 Games Theorem by many researchers.

2.9.4. *Evidence towards the Unique Games Conjecture.* $\mathsf{GapUG}_{\frac{1}{2},\varepsilon}$ is NP-hard, i.e. a weaker form of the Unique Games Conjecture holds with completeness $\frac{1}{2}$. As far as the author knows (and we have consulted the algorithmic experts), the known algorithmic attacks on the Unique Games problem work equally well whether the completeness is ≈ 1 or whether it is $\frac{1}{2}$. Thus, the implication that $\mathsf{GapUG}_{\frac{1}{2},\varepsilon}$ is NP-hard is a compelling evidence, in our opinion, that the known algorithmic attacks are (far) short of disproving the Unique Games Conjecture. Moreover, as remarked before, all the arguments against the Unique Games Conjecture, sketched in Section 2.7, apply equally well to its weaker form with completeness $\frac{1}{2}$. In spite of all these arguments, the $\mathsf{GapUG}_{\frac{1}{2},\varepsilon}$ problem, at the end of the day, does happen to be NP-hard, circumventing all the arguments mentioned!

3. Framework for reductions, why Grassmann graphs?

As indicated earlier, the proof of the 2-to-2 Games Theorem is via a reduction from the 3Lin problem to the 2-to-2 Games problem. The reduction has two components: the first one is purely mathematical and the second one involves ideas and techniques in TCS. The mathematical component amounts to Theorem 4.6 about expansion in Grassmann graph and the reader who is mainly interested in this aspect may jump directly to Sects. 4 and 5. On the other hand, this theorem was discovered with an eye on the application, namely, the 2-to-2 Games Theorem. For the benefit of an interested reader, we provide a general framework for reductions in this section and indicate how the consideration of expansion in Grassmann graph arises naturally in this context.

3.1. Inner reduction: linearity testing. A reduction (or rather a specific kind of reduction since there are several reductions outside of this framework) consists of two "layers", an "inner layer" and an "outer layer". The inner layer amounts to an encoding scheme and a probabilistic test to check whether a given "word" is close to a valid codeword. We emphasize that the considerations here are quite different in nature than those in the traditional coding theory.[10]

A very useful encoding scheme is the Long Code [**20**]; we refer to the survey [**62**] on how Long Code testing is connected to hardness of approximation results, Boolean function analysis, and geometry. In the current

[10]E.g., to encode a k-bit string, the Long Code requires 2^{2^k} bits and Hadamard Code requires 2^k bits. In traditional coding theory, one usually wants an encoding with $O(k)$ bits. Moreover, in traditional coding theory, the coding schemes are usually linear, meaning, the sum (over $\mathbb{F}_2$) of two codewords is also a codeword. On the other hand, the Long Code is not linear in this sense whereas Hadamard Code is.

article, the encoding schemes of relevance are those that are "linear". What really concerns us is the Grassmann Code and the Grassmann Linearity Test. However, for the sake of broader context, we also describe (a) the Hadamard Code with a basic test and a more general, query-efficient test and (b) the Grassmann Code with a much easier-to-analyze Subspace-Subspace Test.

3.1.1. *Hadamard code and the basic* 3*-bit test.*

DEFINITION 3.1. *A Hadamard Code of a string* $x \in \mathbb{F}_2^k$ *is the table of values of the* $\mathbb{F}_2$*-linear function* $f_x : \mathbb{F}_2^k \to \mathbb{F}_2$ *defined as*

$$\forall a \in \mathbb{F}_2^k, \quad f_x[a] = \oplus_{i=1}^k x_i a_i.$$

The Linearity Testing problem (for the Hadamard Code) is as follows. Given a table $f : \mathbb{F}_2^k \to \mathbb{F}_2$, the goal is to "test" whether f is a Hadamard codeword. The tester is allowed only a constant number of queries (independent of k), is allowed to be probabilistic, and is required to satisfy the completeness and soundness properties as follows. The completeness property requires that any Hadamard codeword f_x is accepted with probability 1. The soundness property requires, roughly speaking, that any table f that is "far" from being a Hadamard codeword is accepted with "low" probability, say at most s. Stated in the contrapositive, the requirement is that if the test accepts with probability (at least) s, then the table f is "close to" or "is correlated with" some codeword f_x. We haven't formally defined the requirement here on purpose since the specific requirement is tailored towards intended application and the concrete examples below should clarify the issue.

The basic 3-bit linearity test for the Hadamard Code is as follows. It is based on the self-evident property that if f_x is a Hadamard codeword, then $f_x(a \oplus b) = f_x(a) \oplus f_x(b)$. The test was proposed and analyzed originally in [22].

The Test $\mathcal{T}_{\mathsf{Had,basic}}$

- **Given:** $f : \mathbb{F}_2^k \to \mathbb{F}_2$.
- Pick $a, b \in \mathbb{F}_2^k$ uniformly and independently.
- Accept if and only if $f(a \oplus b) = f(a) \oplus f(b)$.

The completeness property is self-evident. The soundness property can be summarized in the following theorem. We provide a quick proof for reader's benefit and then make additional remarks regarding its application. We note that the "agreement" between tables f and g, $\Pr_a[f(a) = g(a)]$, is a measure of how close the two tables are and herein, agreement strictly above $\frac{1}{2}$ is considered non-trivial.

THEOREM 3.2. *If the test* $\mathcal{T}_{\mathsf{Had,basic}}$ *accepts a table* $f : \mathbb{F}_2^k \to \mathbb{F}_2$ *with probability* $\frac{1}{2} + \varepsilon$*, then there exists a codeword* f_w *such that*

$$\Pr_a[f(a) = f_w(a)] \geqslant \frac{1}{2} + \varepsilon.$$

PROOF. The proof uses elementary Fourier analysis. We note the standard fact that any function $h : \mathbb{F}_2^k \to \mathbb{R}$ can be uniquely expressed as

$$h = \sum_{\mathbf{w} \in \mathbb{F}_2^k} \hat{h}(\mathbf{w})\ \chi_{\mathbf{w}},$$

where $\hat{h}(\mathbf{w})$ are real numbers called the Fourier (or Walsh) coefficients and $\chi_{\mathbf{w}}(a) = (-1)^{\oplus_{i=1}^k \mathbf{w}_i a_i}$ are the Fourier (or Walsh) characters. By the orthogonality of the characters, $\hat{h}(\mathbf{w}) = \langle h, \chi_{\mathbf{w}} \rangle = \mathbb{E}_a [h(a)\chi_{\mathbf{w}}(a)]$.

For a function $f : \mathbb{F}_2^k \to \mathbb{F}_2$, let $h(a) = (-1)^{f(a)}$ be a real, $\{-1, 1\}$-valued function. This is really the same function as f, but written in the "multiplicative" notation. Clearly, $\chi_{\mathbf{w}}$ are same as the corresponding linear functions f_w, but written in the multiplicative notation (here $\mathbf{w}$ and w denote the same vector in the "Fourier domain" and the "original domain"). The test, in the multiplicative notation, is same as testing $h(a \oplus b) = h(a)h(b)$. Further, the expression $\frac{1+h(a)h(b)h(a\oplus b)}{2}$ equals 1 if the test accepts and 0 otherwise. Hence,

$$\begin{aligned} \frac{1}{2} + \varepsilon &= \Pr[\text{ Test Accepts }] = \mathop{\mathbb{E}}_{a,b}\left[\frac{1 + h(a)h(b)h(a \oplus b)}{2}\right] \\ &= \frac{1}{2} + \frac{1}{2} \mathop{\mathbb{E}}_{a,b} [h(a)h(b)h(a \oplus b)]. \end{aligned}$$

This gives, after writing h in its Fourier expression,

$$\begin{aligned} 2\varepsilon &= \mathop{\mathbb{E}}_{a,b}\left[\sum_{\mathbf{u},\mathbf{v},\mathbf{w}} \hat{h}(\mathbf{u})\hat{h}(\mathbf{v})\hat{h}(\mathbf{w}) \cdot \chi_{\mathbf{u}}(a)\chi_{\mathbf{v}}(b)\chi_{\mathbf{w}}(a \oplus b)\right] \\ &= \sum_{\mathbf{u},\mathbf{v},\mathbf{w}} \hat{h}(\mathbf{u})\hat{h}(\mathbf{v})\hat{h}(\mathbf{w}) \cdot \mathop{\mathbb{E}}_{a}[\chi_{\mathbf{u}\oplus\mathbf{w}}(a)] \cdot \mathop{\mathbb{E}}_{b}[\chi_{\mathbf{v}\oplus\mathbf{w}}(b)] \\ &= \sum_{\mathbf{w}} \hat{h}(\mathbf{w})^3. \end{aligned}$$

We note that the expectations over a, b vanish unless $\mathbf{u} = \mathbf{w}$ and $\mathbf{v} = \mathbf{w}$ respectively. Since, by Parseval, $\sum_{\mathbf{w}} \hat{h}(\mathbf{w})^2 = \|h\|_2^2 = 1$, it follows that there exists $\mathbf{w}$ such that $\hat{h}(\mathbf{w}) \geqslant 2\varepsilon$. This is easily seen to be equivalent to the statement that h agrees with the character $\chi_{\mathbf{w}}$ on $\frac{1}{2} + \varepsilon$ fraction of the inputs. Switching to the "additive" notation, this is same as saying that f agrees with the linear function f_w on $\frac{1}{2} + \varepsilon$ fraction of the inputs, completing the proof. □

We make some remarks regarding the application of Theorem 3.2. Similar remarks hold for the tests and their applications described in subsequent sections as well.

- Along with the framework described next in Sects. 3.2 and 3.3, Theorem 3.2 implies the hardness of approximation for the 3Lin problem, namely, Theorem 2.4. Specifically, using Theorem 3.2, one

can reduce $\mathsf{Gap3Lin}_{1-\varepsilon',\frac{3}{4}}$ to $\mathsf{Gap3Lin}_{1-\varepsilon,\frac{1}{2}+\varepsilon}$, amplifying the hardness.[11]

- The acceptance predicate of the test is a linear predicate on three bits and this naturally leads to hardness of approximation for 3Lin. In general, if the acceptance predicate is P, it leads to hardness of approximation of the constraint satisfaction problem with the same predicate P. Rather, if one desires a hardness result for CSP with predicate P, one better design an encoding scheme and a test with the predicate P. This consideration leads naturally to the Grassmann Code and the 2-to-2 Test described in Sects. 3.1.4 and 4 with an eye towards proving the 2-to-2 Games Theorem.
- While applying Theorem 3.2, one needs to take into consideration *every* $w \in \mathbb{F}_2^k$ such that f_w has $(\frac{1}{2}+\varepsilon)$-agreement with f. In coding theoretic language, f_w is referred to as a *decoding* of f. If $\frac{1}{2}+\varepsilon$ is close to 1, then the decoding is unique because any two distinct codewords have agreement exactly $\frac{1}{2}$. On the other hand, if ε is small, then the decoding need not be unique and one decodes every such w, outputting their list. This is referred to as the list-decoding. Naturally, one wishes that the list is of bounded size, its size depending only on ε and not on the dimension k. This indeed turns out to be the case here: for every potential decoding w, we observed that the corresponding Fourier coefficient is at least 2ε and since the squared Fourier coefficients sum up to 1, there are at most $O(\frac{1}{\varepsilon^2})$ of them.

3.1.2. *Hadamard code and the query-efficient test.* We now describe a test on Hadamard Code that is more general and is "query-efficient". Let $r \geqslant 2$ be a fixed integer. The test is as follows.

The Test $\mathcal{T}_{\mathsf{Had,qe}}$

- **Given:** $f : \mathbb{F}_2^k \to \mathbb{F}_2$.
- Pick $a_1, \ldots, a_r \in \mathbb{F}_2^k$ uniformly and independently.
- Accept if and only if $\forall 1 \leqslant i < j \leqslant r$, $f(a_i \oplus a_j) = f(a_i) \oplus f(a_j)$.

This test amounts to $\binom{r}{2}$ invocations of the basic test, but in a highly dependent manner. What is remarkable is that somehow the $\binom{r}{2}$ tests behave as if they were independent; the soundness conclusion is summarized below.

THEOREM 3.3. *If the test $\mathcal{T}_{\mathsf{Had,qe}}$ accepts a table $f : \mathbb{F}_2^k \to \mathbb{F}_2$ with probability $2^{-\binom{r}{2}} + \varepsilon$, then there exists a Fourier coefficient $\hat{h}(\mathbf{w})$ for the function $h(x) = (-1)^{f(x)}$ such that $\hat{h}(\mathbf{w}) \geqslant \varepsilon$.*

[11] This however amounts to circular reasoning since the framework needs the hardness of 3Lin to begin with. Still, in our opinion, this serves as a good illustration of the framework. The actual hardness of approximation for 3Lin is proved using the Long Code based reduction [52], which directly yields the $1-\varepsilon$ versus $\frac{1}{2}+\varepsilon$ gap.

As before, the list-decoding gives a bounded list of large Fourier coefficients. The test and its proof appear in [94]. One of its intended applications is the hardness result for Clique, namely, Theorem 2.7. This gives a different and much easier proof than the original proof for the hardness of Clique in [51].

3.1.3. *Grassmann code and the subspace-subspace test.* We now define the Grassmann Code that is of relevance in this article. To encode a string $x \in \mathbb{F}_2^k$, one writes down the restriction of the linear function $f_x : \mathbb{F}_2^k \to \mathbb{F}_2$ to all subspaces of $\mathbb{F}_2^k$ of dimension ℓ. Here ℓ is an integer parameter thought of as $\ll k$. We note that the restriction of f_x to a subspace $L, \mathsf{dim}(L) = \ell$, denoted as $f_x|_L$, is itself a linear function and since there are 2^ℓ linear functions on L, the alphabet for the encoding has size 2^ℓ. Formally:

DEFINITION 3.4. *The Grassmann Encoding F_x of a string $x \in \mathbb{F}_2^k$ is a table with one entry for every subspace $L \subseteq \mathbb{F}_2^k$, $\mathsf{dim}(L) = \ell$, and*

$$F_x[L] = f_x|_L.$$

Here $f_x[a] = \oplus_{i=1}^k x_i a_i$ is the linear function (as before) on $\mathbb{F}_2^k$.

There is a natural test for the Grassmann Code. Let $1 \leqslant b \leqslant \frac{\ell}{2}$ be a parameter. The test picks two random subspaces L, L' such that $\mathsf{dim}(L \cap L') = b$ and tests consistency on $L \cap L'$. Clearly, for a valid encoding of some string x, the entries in the table at L, L' are $f_x|_L$ and $f_x|_{L'}$ respectively and these are consistent on $L \cap L'$, both being the restrictions of the same global linear function f_x.

The Test $\mathcal{T}_{\mathsf{Gr,subspace}}$

- **Given:** $F[L]$, a linear function on L, for every subspace $L \subseteq \mathbb{F}_2^k$, $\mathsf{dim}(L) = \ell$.
- Pick subspaces $L, L' \subseteq \mathbb{F}_2^k$ uniformly at random so that $\mathsf{dim}(L \cap L') = b$.
- Accept if and only if $F[L]|_{L\cap L'} = F[L']|_{L\cap L'}$.

We note that if the linear functions on L, L' were random, then they would agree on $L \cap L'$ with probability $\frac{1}{2^b}$. It is possible to show that if a table $F[\cdot]$ is accepted with probability significantly larger than this and for "reasonable" b, say $b \leqslant \frac{\ell}{4}$, then $F[\cdot]$ has non-trivial agreement with some codeword (or equivalently some global linear function).

THEOREM 3.5. *If the test $\mathcal{T}_{\mathsf{Gr,subspace}}$ accepts a table $F[\cdot]$ with probability δ, then there exists a global linear function $f : \mathbb{F}_2^k \to \mathbb{F}_2$ such that*

$$\Pr_L [F[L] = f|_L] \geqslant \Omega(\delta^3),$$

and this holds for $1 \leqslant b \leqslant \frac{\ell}{4}$ and $\delta \geqslant 6 \cdot 2^{-\frac{b}{4}}$.

We are not aware of a direct, compelling application of this theorem by itself. It is used however in the overall proof of the 2-to-2 Games Theorem and an analogous theorem over large field $\mathbb{F}_q$ has a nice application to

hardness of approximation [75]. We note that coding theoretic arguments show easily that the list-size, that is the number of codewords F_x that have $\Omega(\delta^3)$ agreement with the given table $F[\cdot]$, is at most polynomial in $\frac{1}{\delta}$. The reason is that the code has very good distance; the fraction of subspaces L for which $F_x[L] \neq F_y[L]$ for distinct codewords F_x and F_y is $1 - \frac{1}{2^\ell}$.

3.1.4. *Grassmann code and the Grassmann linearity test.* We finally describe the Grassmann Linearity Test that is most relevant for this article. The Grassmann Code is same as in Definition 3.4. The test is also the same, except that it picks a pair L, L' with $\mathsf{dim}(L \cap L') = \ell - 1$.

The Test $\mathcal{T}_{\mathsf{Gr},\ \mathsf{Glt}}$

- **Given:** $F[L]$, a linear function on L, for every subspace $L \subseteq \mathbb{F}_2^k$, $\mathsf{dim}(L) = \ell$.
- Pick subspaces $L, L' \subseteq \mathbb{F}_2^k$ uniformly at random so that $\mathsf{dim}(L \cap L') = \ell - 1$.
- Accept if and only if $F[L]|_{L\cap L'} = F[L']|_{L\cap L'}$.

It is observed immediately that the test is a "2-to-2 test" in the sense that for every assignment/answer $F[L]$ there are exactly two answers to $F[L']$ so that the test accepts (and vice versa). This is because a linear function on $L \cap L'$ can be extended to L (and similarly to L') in exactly two ways (and this is how the test eventually leads to hardness of 2-to-2 Games). By design, the test has perfect completeness: if $F[\cdot]$ is a codeword, then the test passes with probability 1 since $F[L]$, $F[L']$ are then restrictions of the same global linear function. However it turns out that a conclusion similar to Theorem 3.5 does not hold even for constant δ (say 0.01) and there is no list-decoding (with bounded list size) either. Still, it is possible to derive a more refined conclusion and a more refined way of list-decoding, namely Theorem 4.3, and use it to prove the 2-to-2 Games Theorem! The details appear in Section 4.

3.2. Outer reduction: 3Lin to 3Lin-blocks. We now describe the "outer layer" of the reduction. Here one starts with an instance of 3Lin and constructs an instance of a constraint satisfaction problem where every constraint depends on two variables, but whose variables are "large"; specifically, its variables are tuples of equations and variables of the 3Lin instance.[12]

Let (X, Eq) be an instance of the NP-hard problem $\mathsf{Gap3Lin}(1-\varepsilon, \frac{3}{4})$ as in Theorem 2.4. Here $X = \{x_1, \ldots, x_n\}$ is the set of variables over $\mathbb{F}_2$ and $\mathsf{Eq} = \{e_1, \ldots, e_m\}$ is the set of equations. Let k be a sufficiently large integer parameter. The parameter ε is chosen to be sufficiently small so that $\varepsilon \ll \frac{1}{k}$. The parameters ε, k are thought of as constants and n, m represent the growing input size. We construct a CSP instance $\mathcal{I}$ as below. It is a weighted instance in the sense that there is a probability distribution on the

[12]In the hardness of approximation literature, this is a canonical construction that is referred to as Label Cover or 2-Prover-1-Round Game.

constraints and for any assignment, the fraction of constraints satisfied is measured with respect to this distribution. The set of variables, constraints, and the distribution on constraints is defined as follows. The variables will be referred to as blocks to distinguish them from variables of the 3Lin instance.

- **Blocks:** The set of blocks is partitioned as $\mathcal{U} \cup \mathcal{V}$ and each constraint depends on one block $U \in \mathcal{U}$ and another block $V \in \mathcal{V}$. The set $\mathcal{U} = \mathsf{Eq}^k$ is simply the set of all k-tuples $U = (e_{i_1}, \ldots, e_{i_k})$ of equations in Eq. The set $\mathcal{V} = (X \cup \mathsf{Eq})^k$ is the set of all "mixed" k-tuples $V = (a_1, \ldots, a_k)$ where a_j is either a variable in X or an equation in Eq.
- **Distribution on constraints:** A random constraint (U, V) is chosen as below:
  - Pick a uniformly random tuple of k equations $U = \{e_{i_1}, \ldots, e_{i_k}\}$.
  - Independently for $1 \leqslant j \leqslant k$, with probability $1 - \beta$, let a_j be same as the equation e_{i_j} and with probability β, let a_j be one of the three variables, picked at random, appearing in the equation e_{i_j}.
  - Let $V = (a_1, \ldots, a_k)$.
- **Constraint:** Let U also denote the set of all $3k$ variables that appear in its k equations and V also denote the set of all $3k - 2t$ variables that appear in $k - t$ equations and as t variables. Here t is w.h.p. close to it expected size βk and clearly $V \subseteq U$.

  The assignments to the blocks U and V are supposed to be strings $s_U \in \mathbb{F}_2^U = \mathbb{F}_2^{3k}$ and $s_V \in \mathbb{F}_2^V = \mathbb{F}_2^{3k-2t}$ respectively, (supposedly) giving values to all variables appearing in them. The constraint is satisfied if s_U agrees with s_V on the variables in V and moreover if s_U satisfies the k equations in U.

(Completeness): It is clear that if the instance (X, Eq) has a $(1 - \varepsilon)$-satisfying assignment σ, then one can take this assignment σ and define the assignments $s_U = \sigma(U)$, $s_V = \sigma(V)$ to all $U \in \mathcal{U}$, $V \in \mathcal{V}$ according to σ, and satisfy $1 - k\varepsilon$ fraction of the constraints of the instance $\mathcal{I}$. A constraint (U, V) is satisfied if all equations in U are satisfied by σ and this happens with probability at least $1 - k\varepsilon$.

(Soundness): On the other hand, it follows from the Parallel Repetition Theorem [**91**, **54**, **90**] that if every assignment to the instance (X, Eq) is at most $\frac{3}{4}$-satisfying, then *any* assignment $\{U \to s_U\}_{U \in \mathcal{U}}$, $\{V \to s_V\}_{V \in \mathcal{V}}$ satisfies at most $2^{-\Omega(\beta k)}$ fraction of the constraints. It is emphasized here that the assignment here need not be consistent with a (global) assignment σ to X (and hence the conclusion is highly non-trivial).

3.3. Inner/outer composition. We finally describe, at a high level, the overall reduction that combines the inner and outer layers of the reduction. Let $\mathcal{I}$ be the CSP instance as in the outer reduction. We choose an encoding scheme along with a test, e.g. any of the four schemes described in

Section 3.1. For the sake of concreteness, let the scheme be the Grassmann Code with the Grassmann Linearity Test.

We replace every block $U \in \mathcal{U}$ by the Grassmann Code of the (intended) assignment $s_U \in \mathbb{F}_2^U$. More precisely, there is a table F^U, with one entry for each ℓ-dimensional subspace $L \subseteq \mathbb{F}_2^U$, and $F^U[L]$ is intended to be the restriction of the linear function $f_{s_U} : \mathbb{F}_2^U \to \mathbb{F}_2$, for some assignment s_U to the block U as in the outer reduction. We stress that while this is the intention and while this will indeed be the case in the "completeness" part of the reduction, the table F^U can be entirely arbitrary in the "soundness" part of the reduction.

Similarly, we replace every block $V \in \mathcal{V}$ by the Grassmann Code of the (intended) assignment $s_V \in \mathbb{F}_2^V$. More precisely, there is a table F^V, with one entry for each ℓ-dimensional subspace $L' \subseteq \mathbb{F}_2^V$, and $F^V[L']$ is intended to be the restriction of the linear function $f_{s_V} : \mathbb{F}_2^V \to \mathbb{F}_2$, for some assignment s_V to the block V as in the outer reduction.

The entries $F^U[L]$ and $F^V[L']$ (over all (U, L), (V, L')) are the variables of the 2-to-2 Games instance. The constraints are now defined as follows. The constraints are weighted, so we indicate the distribution from which a random constraint is picked:

- Pick a constraint (U, V) in the outer reduction according to the distribution therein.
- Pick a random pair of ℓ-dimensional subspaces $L, L' \subseteq \mathbb{F}_2^V$ such that $\mathsf{dim}(L \cap L') = \ell - 1$.
- Since $V \subseteq U$, one naturally views $\mathbb{F}_2^V \subseteq \mathbb{F}_2^U$ (by setting coordinates in $U \setminus V$ to zero). Thus L is viewed as a subspace of $\mathbb{F}_2^U$.
- The constraint is now on $F^U[L]$ and $F^V[L']$ and it tests that (this is a "2-to-2" test)

$$\mathbb{F}^U[L]|_{L \cap L'} = \mathbb{F}^V[L']|_{L \cap L'}. \tag{2}$$

This completes the reduction from 3Lin to 2-to-2 Games. We have left out many crucial details and we will skip the analysis of the correctness of the reduction. The analysis (in the "soundness" part) needs to show that any assignment $\{F^U\}_{U \in \mathcal{U}}$, $\{F^V\}_{V \in \mathcal{V}}$, for which a δ fraction of the constraints as in Equation (2) are satisfied, can be "decoded" into an assignment $\{s_U\}_{U \in \mathcal{U}}$, $\{s_V\}_{V \in \mathcal{V}}$ to the instance $\mathcal{I}$ as in the outer reduction that satisfies $\alpha = \alpha(\delta)$ fraction of the constraints therein. This is a contradiction if the parameters of the instance $\mathcal{I}$ were chosen so that its "soundness" $2^{-\Omega(\beta k)}$ was lower than α. As we remarked, the Grassmann Linearity Test on the Grassmann Code does not lead to a (list-) decoding in the usual/standard sense and new ideas are needed towards the correct "interface" between the inner reduction (= the Grassmann Linearity Test on the Grassmann Code and a refined version of decoding) and the outer reduction (= the blocked instance $\mathcal{I}$ and assignments to it). The details are skipped from this article.

4. Grassmann expansion theorem and linearity testing theorem

We now present the main mathematical results, Theorem 4.3, referred to as the Grassmann Linearity Testing Theorem and Theorem 4.6, referred to as the Grassmann Expansion Theorem. We stress how these two theorems are related and their relevance to the proof of the 2-to-2 Games Theorem.

- The Grassmann Linearity Testing Theorem is viewed as the "soundness" statement for the Grassmann Code and the Grassmann Linearity Test presented in Section 3.1.4. As shown in [**68, 39**], the theorem implies the 2-to-2 Games Theorem via the reduction described in Section 3.3.
- The Grassmann Expansion Theorem implies, almost immediately, the Grassmann Linearity Testing Theorem as shown in [**18**]. A proof appears in Section 4.1.
- The Grassmann Expansion Theorem gives a characterization of sets in the Grassmann graph (defined below) that have expansion strictly below 1. It is proved in [**38, 69**]. In Section 5, we present the basic approach and a (very) partial proof.

We recall some of the definitions again for reader's benefit. In the following, one thinks of the parameter ℓ as a sufficiently large integer and (after fixing it) the parameter k as a sufficiently large integer.

DEFINITION 4.1. *The Grassmann graph* $\mathsf{Gr}_{k,\ell}$ *is defined as follows. Its vertex set consists of all* ℓ*-dimensional subspaces* L *of* $\mathbb{F}_2^k$ *and* (L, L') *is an edge if and only if* $\mathsf{dim}(L \cap L') = \ell - 1$.

As in Section 3.1.4, associated with the Grassmann graph is the Grassmann Code that encodes linear functions $f : \mathbb{F}_2^k \to \mathbb{F}_2$. The encoding of a linear function f is given by a table $F[\cdot]$ that assigns to each vertex L of the graph, the restriction of f to L, i.e. $F[L] = f|_L$. Since there are 2^ℓ linear functions on an ℓ-dimensional space, the alphabet for the encoding has size 2^ℓ. The Grassmann Code is equipped with a natural testing primitive that we called the Grassmann Linearity Test: given a word $F[\cdot]$ (not necessarily a codeword), the test picks an edge (L, L') uniformly at random from the graph and checks that $F[L]|_{L\cap L'} = F[L']_{L\cap L'}$, i.e. that the linear functions $F[L]$ and $F[L']$ are consistent on the common intersection of L, L'.

It is observed immediately that the test is a "2-to-2 test" in the sense that for every assignment/answer $F[L]$ there are exactly two answers to $F[L']$ so that the test accepts (and vice versa). This is because a linear function on $L \cap L'$ can be extended to L (and similarly to L') in exactly two ways. By design, the test has perfect completeness: if $F[\cdot]$ is a codeword, then the test passes with probability 1 since $F[L]$, $F[L']$ are then restrictions of the same global linear function.

The question of interest is what about the soundness of the test? I.e. if a given table $F[\cdot]$ passes the test with (small) probability $\geqslant \delta$, what "decoding" could we infer? Could we infer that the given table $F[\cdot]$ necessarily has good

consistency with some codeword (and if so, list-decode)? One is tempted to speculate that the answer is positive, formally stated below.[13] Here δ, ε are thought of as constants independent of the parameters k, ℓ.

SPECULATION 4.1. *For every $\delta > 0$, there exists $\varepsilon > 0$ such that if a table $F[\cdot]$ passes the Grassmann Linearity Test with probability δ, then there exists a global linear function $f : \mathbb{F}_2^k \to \mathbb{F}_2$ such that*

$$\Pr_L [F[L] = f|_L] \geqslant \varepsilon.$$

It turns out however that the speculation is false, the key reason being that the Grassmann graph has small sets whose expansion is strictly bounded away from 1. We present a counter-example to the speculation now.

DEFINITION 4.2. *Let $G = (V, E)$ be an n-vertex, d-regular graph. For a non-empty set of vertices $S \subseteq V$ with $|S| \leqslant \frac{n}{2}$, its (edge-)expansion is defined as*

$$\Phi(S) = \frac{\left|E(S, \overline{S})\right|}{d \cdot |S|},$$

where $E(S, \overline{S})$ denotes the set of edges with one endpoint in S and the other in $\overline{S} = V \setminus S$.

Alternately, $\Phi(S)$ is the probability that selecting a uniformly random vertex in S and moving along a uniformly random edge incident on that vertex, one lands outside S. We will be interested in whether a set S has expansion very close to 1 (near-perfect expansion) or has expansion strictly bounded away from 1.

Counter-example to Speculation 4.1: Consider the following construction (it will be clear soon what the sets S_i would be):

(1) Let $S_1, \ldots, S_m$ be disjoint subsets of vertices of the Grassmann graph $\mathsf{Gr}_{k,\ell}$, all of equal size, such that their union constitutes a constant α fraction of vertices of the graph.
(2) The sets S_i are very small. Specifically, $m = m(k, \ell) \to \infty$ as $k, \ell \to \infty$.
(3) Suppose that $\Phi(S_i) \leqslant 1 - \beta$ for every $1 \leqslant i \leqslant m$ for a constant β.
(4) For each $1 \leqslant i \leqslant m$, select a global linear function $f_i : \mathbb{F}_2^k \to \mathbb{F}_2$ at random.
(5) Define $F[L] = f_i|_L$ for every $L \in S_i$. For $L \notin \cup_{i=1}^m S_i$, $F[L]$ is defined at random.

We show that the table $F[\cdot]$ passes the Grassmann Linearity Test with probability $\alpha\beta$, but has negligible consistency with any global linear function. Firstly, since S_i cover α fraction of vertices and each S_i has expansion at most $1-\beta$, the fraction of edges of the Grassmann graph that are inside some S_i is at least $\alpha\beta$. Since on each S_i, the table $F[\cdot]$ is consistent with the global

[13] Moreover, a positive answer would have led to a very straightforward analysis of the reduction to 2-to-2 Games, avoiding most of the complications in [68, 39].

function f_i, the table passes the test for all edges (L, L') that are inside some S_i. Secondly, since the functions f_i on different pieces S_i are random and unrelated to each other, no single global function has non-negligible consistency with $F[\cdot]$. This completes the description of the counter-example.

How does one get around this counter-example, i.e. reformulate Speculation 4.1 so that it is *correct* as well as *sufficient* towards analysis of the reduction to 2-to-2 Games? With regards to the specific counter-example above, here is a vacuous statement: if we restrict our attention to *only* the subset of vertices in say S_1, then $F[\cdot]$ indeed has full consistency with a global linear function, namely the function f_1. Moreover, as we will see, a canonical example of a small set with expansion strictly bounded away from 1 is $S = \mathsf{Gr}_{k,\ell}[A, B]$ where $A \subseteq B \subseteq \mathbb{F}_2^k$ are subspaces with $\mathsf{dim}(A) + \mathsf{codim}(B) \leqslant r$ and

$$S = \mathsf{Gr}_{k,\ell}[A, B] = \{L \mid A \subseteq L \subseteq B\}.$$

In this case, $\Phi(S) = 1 - 2^{-r}$ which is strictly bounded away from 1 for small integer r (say $r = 4$).[14] These observations motivated the following Linearity Testing Hypothesis (now a theorem) in [39].

THEOREM 4.3 (Grassmann Linearity Testing Theorem). *For every constant $\delta > 0$, there exists a constant $\varepsilon > 0$ and an integer r such that for all sufficiently large integers ℓ and (after fixing it) for all sufficiently large integer k, the following holds. If a table $F[\cdot]$ passes the Grassmann Linearity Test with probability δ, then there exist subspaces $A \subseteq B \subseteq \mathbb{F}_2^k$ with $\mathsf{dim}(A) + \mathsf{codim}(B) \leqslant r$ and a linear function $f : B \to \mathbb{F}_2$, such that*

$$\Pr_{A \subseteq L \subseteq B} [F[L] = f|_L] \geqslant \varepsilon.$$

In words, while $F[\cdot]$ need not have good consistency with a global linear function on the *entire* graph $\mathsf{Gr}_{k,\ell}$, there must be a *structured subgraph* $\mathsf{Gr}_{k,\ell}[A, B]$ on which it does have good consistency with a global linear function and moreover this subgraph is of constant "co-order", defined as $\mathsf{dim}(A) + \mathsf{codim}(B)$. As remarked before, the Grassmann Linearity Testing Theorem is sufficient towards analysis of the reduction to 2-to-2 Games [68, 39]. The theorem is implied immediately by the Grassmann Expansion Theorem, proposed and proved in [38, 69], that we describe next. It states, roughly, that a set in the Grassmann graph whose expansion is strictly bounded away from 1 "resembles" a canonical set $\mathsf{Gr}_{k,\ell}[A, B]$ as described earlier.

DEFINITION 4.4. *Suppose $A \subseteq B \subseteq \mathbb{F}_2^k$ are subspaces. Let $\mathsf{dim}(A) = a$, $\mathsf{codim}(B) = b$ and think of a, b as small constants (say $a = b = 2$). Then (as*

[14] The sets S_i in the counter-example are essentially the canonical sets of this kind. We can arrange them to be disjoint by carefully selecting $S_1, \ldots, S_m$ successively as follows. Let $\tilde{S}_i$ be a canonical set such that at most α fraction of it is covered by the sets $S_1, \ldots, S_{i-1}$ already selected. Let $S_i = \tilde{S}_i \setminus (S_1 \cup \cdots \cup S_{i-1})$. The process continues until an α fraction of the whole graph is covered.

introduced before) the subgraph $\mathsf{Gr}_{k,\ell}[A,B]$ *is an induced subgraph of* $\mathsf{Gr}_{k,\ell}$ *induced on precisely the set of vertices* L *such that* $A \subseteq L \subseteq B$*. It is easily seen that* $\mathsf{Gr}_{k,\ell}[A,B]$ *is an isomorphic copy of a lower order Grassmann graph* $\mathsf{Gr}_{k-a-b,\ell-a}$*. We call* $a+b$ *as the co-order of* $\mathsf{Gr}_{k,\ell}[A,B]$ *with respect to* $\mathsf{Gr}_{k,\ell}$*.*

The sets $\mathsf{Gr}_{k,\ell}[A,B]$ are natural examples of sets in $\mathsf{Gr}_{k,\ell}$ that have expansion strictly bounded away from 1 (when a, b are small constants). Indeed, the expansion of $\mathsf{Gr}_{k,\ell}[A,B]$, when seen as a subset of $\mathsf{Gr}_{k,\ell}$, has expansion precisely $1-2^{-(a+b)}$ (up to an error $O(2^{-\ell})$ which is thought of as negligible and ignored for the ease of presentation). The reasoning is as follows. For a vertex $L \in \mathsf{Gr}_{k,\ell}[A,B]$, its random neighbor L' is obtained by picking a random subspace $T \subseteq L$, $\mathsf{dim}(T) = \ell - 1$ and a random point $x \in \mathbb{F}_2^k \setminus L$ and letting $L' = T \oplus \mathsf{Span}(x)$. Now $L' \in \mathsf{Gr}_{k,\ell}[A,B]$ if and only if $A \subseteq T$ and $x \in B$ and these events happen independently with probabilities 2^{-a} and 2^{-b} respectively (up to an error $O(2^{-\ell})$). Thus a random neighbor of a random vertex in $\mathsf{Gr}_{k,\ell}[A,B]$ is also inside it with probability $2^{-(a+b)}$ and hence its expansion is $1-2^{-(a+b)}$. Furthermore, we observe that if $S \subseteq \mathsf{Gr}_{k,\ell}[A,B] \subseteq \mathsf{Gr}_{k,\ell}$ is such that

$$\frac{|S|}{|\mathsf{Gr}_{k,\ell}[A,B]|} = \varepsilon,$$

then $\Phi(S) \leqslant 1-\varepsilon\cdot 2^{-(a+b)}$. This is because (we skip the easy proof) any set of density ε inside a Grassmann graph has at least ε^2 fraction of the edges inside it (and hence has expansion at most $1-\varepsilon$). Therefore, a random neighbor of a random vertex in $S \subseteq \mathsf{Gr}_{k,\ell}[A,B]$ lies inside $\mathsf{Gr}_{k,\ell}[A,B]$ with probability $2^{-(a+b)}$ as seen above and then inside S with probability at least ε, justifying the observation. We summarize the overall observation as:

Fact 4.5. *(Informal): A subset of constant density inside a constant co-order copy of Grassmann graph inside a Grassmann graph has expansion strictly bounded away from* 1*.*
(Formal): Let $S \subseteq \mathsf{Gr}_{k,\ell}[A,B] \subseteq \mathsf{Gr}_{k,\ell}$ *be such that* $\mathsf{dim}(A) = a$, $\mathsf{codim}(B) = b$ *and the density of* S *inside* $\mathsf{Gr}_{k,\ell}[A,B]$ *is* ε*. Then* $\Phi(S) \leqslant 1-\varepsilon\cdot 2^{-(a+b)}$*.*

The converse of the above fact is essentially correct. As the theorem below states, any set S in the Grassmann graph $\mathsf{Gr}_{k,\ell}$ whose expansion is strictly bounded away from 1 has constant density inside *some* copy of Grassmann graph of constant co-order. From this statement, it is easy to derive a structural result that there is a subset $T \subseteq S$ such that (1) T has constant density inside S (2) T can be written as $T = T_1 \cup \cdots \cup T_m$ where T_i are disjoint and (3) each T_i has constant density inside some copy of Grassmann graph of constant co-order. We skip the easy proof.

Theorem 4.6 (Grassmann Expansion Theorem). *For every constant* $0 < \alpha < 1$*, there exists a constant* $\varepsilon > 0$ *and an integer* $r \geqslant 0$ *such that for*

all sufficiently large integers ℓ and (after fixing it) for all sufficiently large integers k, the following holds. Let $S \subseteq \mathsf{Gr}_{k,\ell}$ be such that $\Phi(S) \leqslant \alpha$. Then there exist subspaces $A \subseteq B \subseteq \mathbb{F}_2^k$ such that $\mathsf{dim}(A) = a$, $\mathsf{codim}(B) = b$, $a + b \leqslant r$ and

$$\frac{|S \cap \mathsf{Gr}_{k,\ell}[A,B]|}{|\mathsf{Gr}_{k,\ell}[A,B]|} \geqslant \varepsilon.$$

The theorem is proved by spectral analysis of the Grassmann graph (introduced in [**38, 68**]; the eigenvalues and eigenspaces of the Grassmann graph were known before). Roughly speaking, given a set S with expansion at most $\alpha < 1 - 2^{-(s+1)}$, it is easily observed that the indicator vector of the set $\mathbf{1}_S$ must have a significant projection onto the eigenspace at "level" at most s (s is a constant when α is strictly bounded away from 1). The spectral analysis then attempts to use this projection to deduce the desired structure of S. In Section 5 we sketch the approach in more detail and present a proof when $\mathbf{1}_S$ has a significant projection onto the eigenspace at level 1.

We make a few remarks on Theorem 4.6. Firstly, the subspaces A and B therein are referred to as "zoom-in" and "zoom-out" spaces respectively [**68, 39, 38**]. This makes sense if one imagines searching for the appropriate subgraph $\mathsf{Gr}_{k,\ell}[A,B]$ where the set S happens to have significant density. Secondly, we note that if S has density $\geqslant \varepsilon$, then the conclusion of the theorem is vacuously true without any need for a zoom-in or a zoom-out (i.e. $a = b = 0$, $A = \{0\}$, $B = \mathbb{F}_2^k$), so the theorem is really about "small" sets. Thirdly, the proof gives correct dependence of the required zoom-in-out dimension r on the upper bound on expansion α. For $\alpha < 1 - 2^{-(s+1)}$, one gets a significant projection onto the eigenspace at level at most s and then in the proof, a combined zoom-in-out dimension of at most $r = s$ is needed. This is tight (i.e. a lesser zoom-in-out dimension is not sufficient) since we know that subgraphs $\mathsf{Gr}_{k,\ell}[A,B]$ have expansion $1 - 2^{-(a+b)}$ and the combined zoom-in-out dimension (obviously) $a+b$. Finally, we note that towards proving the theorem, it is easier to work with the contra-positive: a set S that has very small density inside every copy of the Grassmann graph with constant co-order (such a set will be called pseudorandom) has near-perfect expansion (i.e. very near 1).

The phenomenon as in Theorem 4.6 occurs also in the Johnson graph and has been analyzed in [**67**]. In the Johnson graph, the vertices are ℓ-subsets of a k-set and the edges are t-wise intersecting pairs (we are concerned with the case when $t = \lfloor \frac{\ell}{2} \rfloor$). The Johnson case can informally be seen as a special case of the Grassmann case and the analysis of the former in [**67**] has been insightful in the analysis of the latter (there are no zoom-outs in the Johnson case, only the zoom-ins).

4.1. Grassmann expansion theorem implies linearity testing theorem. We present a quick proof that Theorem 4.6 implies Theorem 4.3 [**18**]. Consider a table $F[\cdot]$ that passes the Grassmann Linearity Test with probability δ. Consider a random linear function on the whole space

$g : \mathbb{F}_2^k \to \mathbb{F}_2$ and consider the set $S = S_g \subseteq \mathsf{Gr}_{k,\ell}$ on which g agrees with $F[\cdot]$, i.e.

$$S_g = \{L \mid F[L] = g|_L\}.$$

Since g is a random linear function and each L is ℓ-dimensional, the (fractional) size of S_g, in expectation, is $\theta = 2^{-\ell}$. On the other hand, consider any edge (L, L') for which the Linearity Test passes, i.e. for which $F[L]$ and $F[L']$ agree on $L \cap L'$. For such an edge, $L \oplus L'$ is $(\ell+1)$-dimensional and with probability $2^{-(\ell+1)}$ over the choice of g, both the events $g|_L = F[L]$ and $g|_{L'} = F[L']$ happen and then both $L, L' \in S_g$. The fraction of such edges is δ, the probability with which the Linearity Test passes.

Hence we conclude that, the size of S_g is θ in expectation and the fraction of edges inside S_g is at least $\frac{\delta}{2}\theta$ in expectation. Clearly, there exists a choice of g, say g^*, for which the fraction of edges inside S_{g^*} is at least $\frac{\delta}{2}$ times the size of S_{g^*}, or in other words $\Phi(S_{g^*}) \leqslant 1 - \frac{\delta}{2}$. Now we apply Theorem 4.6 (the choice of ε, r are as therein) and conclude that there exist subspaces $A \subseteq B \subseteq \mathbb{F}_2^k$ such that $\mathsf{dim}(A) = a$, $\mathsf{codim}(B) = b$, $a + b \leqslant r$ and

$$\frac{|S_{g^*} \cap \mathsf{Gr}_{k,\ell}[A, B]|}{|\mathsf{Gr}_{k,\ell}[A, B]|} \geqslant \varepsilon.$$

This is same as saying that for $f = g^*|_B : B \to \mathbb{F}_2$,

$$\Pr_{A \subseteq L \subseteq B}[F[L] = f|_L] \geqslant \varepsilon.$$

5. Outline of proof of Grassmann expansion theorem

In this section, we give an outline of the proof of Theorem 4.6 and as an illustration, give essentially a full proof when expansion of a set is below $\frac{3}{4}$ (which in turn implies that the indicator of the set has significant projection onto the eigenspace of level 1). It will be convenient to restate Theorem 4.6 in the contra-positive and in terms of "pseudo-random sets".

Definition 5.1. *A subset of vertices $S \subseteq \mathsf{Gr}_{k,\ell}$ is called (r, ε)-pseudo-random if for any subspaces $A \subseteq B \subseteq \mathbb{F}_2^k$ such that $\mathsf{dim}(A) = a$, $\mathsf{codim}(B) = b$, $a + b \leqslant r$, we have*

$$\mu_{\mathsf{in}(A),\mathsf{out}(B)}(S) \stackrel{def}{=} \frac{|S \cap \mathsf{Gr}_{k,\ell}[A, B]|}{|\mathsf{Gr}_{k,\ell}[A, B]|} \leqslant \varepsilon.$$

Theorem 5.2 (Theorem 4.6 restated). *For every constant $\zeta > 0$, there exists a constant $\varepsilon > 0$ and an integer $r \geqslant 0$ such that for all sufficiently large integers ℓ and (after fixing it) for all sufficiently large integers k, the following holds. If $S \subseteq \mathsf{Gr}_{k,\ell}$ is (r, ε)-pseudorandom, then $\Phi(S) \geqslant 1 - \zeta$.*

5.1. Spectral decomposition of $\mathsf{Gr}_{k,\ell}$. To prove Theorem 5.2, we use spectral analysis of the graph $\mathsf{Gr}_{k,\ell}$. This is a distance-regular graph and its spectrum is well-known. We only state here facts that are relevant in the current context. Let J denote the linear space of functions $F : \mathsf{Gr}_{k,\ell} \to \mathbb{R}$ (on the vertex set of the graph). The standard inner product on this space is

$\langle F, F' \rangle = \mathbb{E}_L [F[L]F[L']]$. Let M_{adj} denote the normalized adjacency matrix of the graph (i.e. entries corresponding to non-edges are zero and corresponding to edges are $\frac{1}{\mathsf{Degree}}$). It is well-known that the space J decomposes into mutually orthogonal eigenspaces

$$J = J_{=0} \oplus J_{=1} \oplus \cdots \oplus J_{=\ell}.$$

The eigenvalue corresponding to the eigen-space $J_{=i}$ is very close to 2^{-i}. For clarity of exposition, we will assume henceforth that this eigenvalue is exactly 2^{-i}. Thus for any $F \in J_{=i}$, $M_{\mathsf{adj}}F = 2^{-i}F$. The space $J_{=0}$ is single-dimensional, consisting of all constant functions. In general, the dimension of $J_{=i}$ equals $\begin{bmatrix} k \\ i \end{bmatrix} - \begin{bmatrix} k \\ i-1 \end{bmatrix}$ where $\begin{bmatrix} k \\ i \end{bmatrix}$ denotes the number of i-dimensional subspaces of $\mathbb{F}_2^k$. We will explain in more detail the structure of the space $J_{=1}$ in Section 5.4 when we need it.

It follows that any function $F : \mathsf{Gr}_{k,\ell} \to \mathbb{R}$ has a decomposition $F = \sum_{i=0}^{\ell} F_{=i}$ where $F_{=i}$ is the component in (projection onto) the space $J_{=i}$. Henceforth, let $S \subseteq \mathsf{Gr}_{k,\ell}$ denote a set as in the statement of Theorem 5.2 and $F : \mathsf{Gr}_{k,\ell} \to \{0,1\}$ denote its indicator function. Let $\delta = \mu(S) = \|F\|_2^2$ denote the (fractional) size of the set S relative to the size of the whole graph $\mathsf{Gr}_{k,\ell}$. Using Parseval's identity (since F is Boolean and $F_{=0} = \mathbb{E}_L [F[L]] = \delta$),

$$\delta = \|F\|_2^2 = \delta^2 + \sum_{i=1}^{\ell} \|F_{=i}\|_2^2.$$

The squared norm $\|F_{=i}\|_2^2$ will be referred to as the Fourier weight at the ith level. Roughly speaking, Theorem 5.2 is proved by showing that the (r, ε)-pseudorandomness condition on the set S implies that F has low weight on low (that is first r) levels (this is the difficult step). This in turn is used to show that S has near-perfect expansion (this is easy). We sketch the overall reasoning and then as an illustration, give essentially a full proof that $(1, \varepsilon)$-pseudorandomness condition on the set S implies that F has low weight on the first Fourier level, which in turn implies that expansion of S is at least $\frac{3}{4} - o(1)$.

5.2. Pseudorandomness implies low weight at low levels implies near-perfect expansion. As noted, $F : \mathsf{Gr}_{k\ell} \to \{0,1\}$ denotes the indicator function of a pseudorandom set S and $\delta = \mu(S) = \|F\|_2^2$ denotes its density. Theorem 5.2 requires us to show that if S is (r, ε)-pseudorandom, then it has near-prefect expansion, that is at least $1 - \zeta$. At a high-level, this is accomplished in two steps below (here $\gamma = \gamma(r, \varepsilon) \to 0$ as $\varepsilon \to 0$ for fixed r; a precise dependence is stated in Theorem 5.3).

- One shows that a (r, ε)-pseudorandom set must have low (that is $\leqslant \gamma\delta$ for some constant γ) weight at all low (that is up to r) levels.
- One shows that if there is low weight at all low levels, then the set must have near-perfect expansion (that is $\geqslant 1 - \gamma(r+1) - 2^{-(r+1)} \stackrel{def}{=} 1 - \zeta$).

We stress that as $\varepsilon \to 0$ for a fixed r, the lower bound on the expansion $\to 1 - 2^{-(r+1)}$. We include a quick proof of the second step below for the sake of completeness. The main task remains thereafter to prove the first step. Assume therefore that F has weight at most $\gamma\delta$ at each level up to r. Below, a random neighbor of vertex $L \in \mathsf{Gr}_{k,\ell}$ is denoted as $L' \sim L$, and as before, the normalized adjacency matrix is M_{adj} and the inner product is $\langle F, F' \rangle = \mathbb{E}_L [F[L]F[L']]$. We have

$$\begin{aligned} 1 - \Phi(S) &= \Pr_{L \in S, L' \sim L} \left[L' \in S \right] = (1/\delta) \cdot \Pr_{L, L' \sim L} \left[L \in S \wedge L' \in S \right] \\ &= (1/\delta) \cdot \langle F,\ M_{\mathsf{adj}}\ F \rangle. \end{aligned}$$

Using the decomposition $F = \sum_{i=0}^{\ell} F_{=i}$ into mutually orthogonal components $F_{=i}$ of eigenvalues 2^{-i}, and that $\delta = \sum_{i=0}^{\ell} \|F_{=i}\|_2^2$, we get that

$$\begin{aligned} \delta(1 - \Phi(S)) &= \sum_{i=0}^{\ell} 2^{-i} \|F_{=i}\|_2^2 \leqslant \sum_{i=0}^{r} \|F_{=i}\|^2 + 2^{-(r+1)} \sum_{i=r+1}^{\ell} \|F_{=i}\|_2^2 \\ &\leqslant \gamma\delta(r+1) + \delta\ 2^{-(r+1)}. \end{aligned}$$

Dividing by δ gives us $\Phi(S) \geqslant 1 - \gamma(r+1) - 2^{-(r+1)} \stackrel{def}{=} 1 - \zeta$ as claimed. To summarize, to prove Theorem 5.2, it suffices to prove (hence this can ve viewed as the main result):

THEOREM 5.3. *Let S be a set of vertices in $\mathsf{Gr}_{k,\ell}$ that has density δ and is (r, ε) pseudo-random. Let $F \colon \mathsf{Gr}_{k,\ell} \to \{0,1\}$ be the indicator function of S. Then for any $i = 0, 1, \ldots, r$,*

$$\eta = \|F_{=i}\|_2^2 \leqslant 2^{7r^3+3} \varepsilon^{\frac{1}{4}} \delta.$$

We now summarize the high-level plan to prove Theorem 5.3 as in [**38, 69**]. The idea is to consider the fourth moment of $F_{=i}$ and prove both a lower bound and an upper bound on it. Specifically, let S be a set that has density δ and is (r, ε) pseudo-random as in the statement of the theorem. Let $0 \leqslant i \leqslant r$ and let $\eta = \|F_{=i}\|_2^2$. The theorem follows by showing that (one cancels η from both sides, moves $2^9\delta^4$ on the right and then takes a fourth root)

$$\frac{\eta^5}{2^9 \cdot \delta^4} \leqslant \mathbb{E} \left[F_{=i}^4 \right] \leqslant 2^{25r^3} \eta\varepsilon. \tag{3}$$

5.3. Lower-bounding the fourth moment of $F_{=i}$.

LEMMA 5.4. *Under the condition and notation of Theorem 5.3,* $\mathbb{E} \left[F_{=i}^4 \right] \geqslant \frac{\eta^5}{2^9 \cdot \delta^4}$.

PROOF. We note the decomposition $F = \sum_{j=0}^{\ell} F_{=j}$ into mutually orthogonal components and that $\|F\|_2^2 = \delta$, $\|F_{=i}\|_2^2 = \eta$. Hence $\mathbb{E} \left[(F - F_{=i})^2 \right] = \delta - \eta$. By Markov's inequality,

$$\Pr \left[(F - F_{=i})^2 \geqslant 1 - \frac{\eta}{2\delta} \right] \leqslant \delta - \frac{\eta}{2}.$$

On the other hand, F is Boolean and $\Pr[F=1]=\delta$. Thus with probability at least $\frac{\eta}{2}$, both the events below occur:

$$F=1, \qquad (F-F_{=i})^2 \leqslant 1-\frac{\eta}{2\delta},$$

in which case it holds that $(1-F_{=i})^2 \leqslant 1-\frac{\eta}{2\delta}$ and in turn that $F_{=i} \geqslant \frac{\eta}{4\delta}$. Hence as claimed,

$$\mathbb{E}\left[F_{=i}^4\right] \geqslant \frac{\eta}{2}\cdot\left(\frac{\eta}{4\delta}\right)^4. \qquad \square$$

5.4. Upper-bounding the fourth moment of $F_{=1}$. In general, upper bounding the fourth moment of $F_{=i}$ requires a rather involved analysis and appears in [38, 69]. However the special case $i=1$ is easy and quite illustrative, so we present (essentially) a full proof in this case. Specifically, we prove the statement below and its immediate corollary.

THEOREM 5.5. *Let S be a set of vertices in $\mathsf{Gr}_{k,\ell}$ that has density δ and is $(1,\varepsilon)$ pseudo-random. Let $F\colon \mathsf{Gr}_{k,\ell}\to\{0,1\}$ be the indicator function of S and $\eta=\|F_{=1}\|_2^2$. Then*

$$\mathbb{E}\left[F_{=1}^4\right] \leqslant O(\eta\varepsilon).$$

COROLLARY 5.6. *Let S be a set of vertices in $\mathsf{Gr}_{k,\ell}$ that has density δ and is $(1,\varepsilon)$-pseudorandom. Let $F\colon \mathsf{Gr}_{k,\ell}\to\{0,1\}$ be the indicator function of S. Then the Fourier weight of F at the first level is at most $O(\varepsilon^{\frac{1}{4}}\delta)$ and hence S has expansion at least $\frac{3}{4}-O(\varepsilon^{\frac{1}{4}})$.*

Towards proving Theorem 5.5, we need some understanding of the first level component $F_{=1}$ in the decomposition $F=\sum_{i=0}^{\ell}F_{=i}$. We recall that the space J of all functions on $\mathsf{Gr}_{k,\ell}$ has a decomposition into eigenspaces $J=\oplus_{i=0}^{\ell}J_{=i}$ and for any function F, $F_{=i}$ is its projection onto the eigenspace J_i. The eigen-space $J_{=0}$ is single dimensional, consisting of all constant functions.

For a point $x\in\mathbb{F}_2^k$, $x\neq 0$, let $1_x:\mathsf{Gr}_{k,\ell}\to\{0,1\}$ denote the indicator of the set $\{L|x\in L\}$ and let $\Theta=\mathbb{E}_L[1_x(L)]$. In the calculations below, Θ would (merely) serve as a normalizing factor. Similarly, for a subspace $W\subseteq\mathbb{F}^k$ of dimension $k-1$, let $1_W:\mathsf{Gr}_{k,\ell}\to\{0,1\}$ denote the indicator of the set $\{L|L\subseteq W\}$ so that $\mathbb{E}_L[1_W(L)]\simeq 2^{-\ell}$. Henceforth, for the sake of clarity, we use the notation "$\simeq$" to denote an approximation that is correct up to a negligible additive or multiplicative term, which does not affect the analysis in any significant manner, and if desired, the correcting terms can be accounted for at the expense of clarity.

The main observation we need is that the eigen-space $J_{=0}\bigoplus J_{=1}$ is spanned precisely by the indicators 1_x or alternately by the indicators 1_W.

LEMMA 5.7. *J_0 consists of the constant functions on $\mathsf{Gr}_{k,\ell}$ and $J_{=0}\bigoplus J_{=1}$ consists of the linear span of all functions $\{1_x|x\neq 0\}$. Moreover*

$$J_{=0}\oplus J_{=1}=\mathsf{Span}(\{1_x|x\neq 0\})=\mathsf{Span}(\{1_W|\mathsf{dim}(W)=k-1\}).$$

PROOF. While we refer the reader to [39] for a proof, it is instructive to see that the function 1_W lies in the span of functions 1_x. Indeed,

$$2^{\ell-1}(2^\ell-1)\cdot 1_W = 2^{\ell-1}\cdot \sum_{x\in W, x\neq 0} 1_x - (2^{\ell-1}-1)\cdot \sum_{x\notin W} 1_x.$$

To confirm, any $L \subseteq W$ is counted exactly $2^{\ell-1}(2^\ell-1)$ times on both sides. Specifically, on the right hand side, it is counted for exactly $2^\ell - 1$ of $x \in L$, $x \neq 0$. On the other hand, any $L \nsubseteq W$ is not counted on either side. Specifically, on the right hand side, it is counted positively for $2^{\ell-1}-1$ of $x \in L \cap W$, $x \neq 0$ and counted negatively for $2^{\ell-1}$ of $x \in L \setminus W$. □

LEMMA 5.8. *A function F lies in the space $J_{=1}$ if and only if there exists a function $f : \mathbb{F}_2^k \to \mathbb{R}$, $f(0) = 0$, $\mathbb{E}_x[f(x)] = 0$ and*

$$\forall L,\ F[L] = \sum_{x\in L} f(x).$$

PROOF. For the forward direction, we note that since $J_{=1}$ is contained in the span of $\{1_x | x \neq 0\}$, any $F \in J_{=1}$ can be written as

$$F[L] = \sum_{x\in\mathbb{F}_2^k,\ x\neq 0} f(x)1_x(L) = \sum_{x\in L, x\neq 0} f(x),$$

for some coefficients $f(x)$. Since $F \in J_{=1}$, $\mathbb{E}_L[F[L]] = 0$. Taking expectation (over L) on both sides, $0 = \left(\sum_{x\neq 0} f(x)\right)\Theta$. By defining in addition $f(0) = 0$ proves the claim. For the reverse direction, we note similarly that for f as therein, we have

$$F[L] = \sum_{x\in L} f(x) = \sum_{x\in\mathbb{F}_2^k,\ x\neq 0} f(x)1_x(L),$$

and hence F belongs to the span of functions $\{1_x | x \neq 0\}$. Moreover $\mathbb{E}_L[F[L]] = 0$ since $\sum_{x\neq 0} f(x) = 0$ and hence $F \in J_{=1}$. □

As usual, we will be concerned with a function $F : \mathsf{Gr}_{k,\ell} \to \{0,1\}$ that is an indicator of a set $S \subseteq \mathsf{Gr}_{k,\ell}$ and $\delta = \mu(S) = \mathbb{E}[F]$ is its relative size. The level one component $F_{=1}$ is defined, according to above Lemma 5.8, in terms of a function $f_{=1} : \mathbb{F}_2^k \to \mathbb{R}$, $\mathbb{E}_x[f_{=1}(x)] = 0$, $f_{=1}(0) = 0$. The main point we need is that the function $f_{=1}$ and its Fourier coefficients capture how the density of the set S changes when restricted to those L for which $x \in L$ or $L \subseteq W$ respectively (for $x \in \mathbb{F}_2^k$, $x \neq 0$ and $\mathsf{dim}(W) = k-1$ as before). To see this, let us define

$$S_x = \{L | x \in L, L \in S\}, \quad S_W = \{L | L \subseteq W, L \in S\},$$

and denote by $\mu(S_x)$ and $\mu(S_W)$ their relative conditional sizes, i.e.

$$\mu(S_x) = \Pr_{L: x\in L}[L \in S], \quad \mu(S_W) = \Pr_{L: L\subseteq W}[L \in S].$$

Let us define the function $h : \mathbb{F}_2^k \to \mathbb{R}$, $h(0) = 0$, and for $x \neq 0$,

$$h(x) = \mu(S_x) - \mu(S).$$

That is, $h(x)$ records the change in the density of S when restricted to those L for which $x \in L$. The main point is that $f_{=1}$ is essentially same as h and that the Fourier coefficients of h essentially record (up to a normalization factor $2^{-\ell}$) the change in the density of S when restricted to those L for which $L \subseteq W$. For a vector $w \in \mathbb{F}_2^k$, $w \neq 0$, let $\hat{h}(\mathbf{w})$ denote the corresponding Fourier coefficient and $W = w^{\perp}$ denote its orthogonal subspace of dimension $k-1$ (with respect to standard inner product in $\mathbb{F}_2^k$).

LEMMA 5.9. *We have*

- $\|F_{=1}\|_2^2 \simeq 2^{\ell} \|f_{=1}\|_2^2$.
- $f_{=1}(x) \simeq h(x)$.
- $\hat{h}(\mathbf{w}) \simeq 2^{-\ell}(\mu(S_W) - \mu(S))$.

PROOF. For the first item in the lemma, we first note that $\mathbb{E}_x [f_{=1}(x)] = 0$ and hence

$$\mathop{\mathbb{E}}_{x \neq y} [f_{=1}(x) f_{=1}(y)] = \mathop{\mathbb{E}}_{x} [f_{=1}(x)]^2 - \frac{1}{2^k} \cdot \mathop{\mathbb{E}}_{x} \left[f_{=1}(x)^2\right] \simeq 0,$$

where the first term is zero and the second term is negligible due to the factor $\frac{1}{2^k}$. Thus,

$$\|F_{=1}\|_2^2 = \mathop{\mathbb{E}}_{L} \left[\left(\sum_{x \in L} f_{=1}(x) \right)^2 \right] \simeq 2^{\ell} \mathop{\mathbb{E}}_{x} \left[f_{=1}(x)^2\right] + 2^{2\ell} \mathop{\mathbb{E}}_{x \neq y} [f_{=1}(x) f_{=1}(y)].$$

The second term is $\simeq 0$ as seen and hence $\|F_{=1}\|_2^2 \simeq 2^{\ell} \mathbb{E}_x \left[f_{=1}(x)^2\right] = 2^{\ell} \|f_{=1}\|_2^2$ as desired. For the second item in the lemma, noting that $\Theta = \mathbb{E}[1_x]$ and treating it as a normalization factor,

$$\Theta \cdot \mu(S_x) = \langle F, 1_x \rangle = \langle F_{=0}, 1_x \rangle + \langle F_{=1}, 1_x \rangle = \Theta \cdot \mu(S) + \langle F_{=1}, 1_x \rangle.$$

Further,

$$\Theta^{-1} \langle F_{=1}, 1_x \rangle = \mathop{\mathbb{E}}_{L : x \in L} [F_{=1}[L]] = \mathop{\mathbb{E}}_{L : x \in L} \left[\sum_{y \in L} f_{=1}(y) \right].$$

We note that over the choice of L such that $x \in L$, in the inner summation, the term for $y = x$ always appears whereas the terms for $y \neq x$ appear with probability roughly $2^{-k+\ell}$. Hence

$$\Theta^{-1} \langle F_{=1}, 1_x \rangle \simeq f_{=1}(x) + 2^{-k+\ell} \mathop{\mathbb{E}}_{y \neq x} [f_{=1}(y)] \simeq f_{=1}(x).$$

We remark that the second term is negligible, both due to the factor $2^{-k+\ell}$ and due to $\mathbb{E}_y [f_{=1}(y)] = 0$. Thus we have,

$$f_{=1}(x) \simeq \mu(S_x) - \mu(S) \stackrel{def}{=} h(x),$$

as desired. For the third item in the lemma, we denote by 1_W and $1_{\overline{W}}$ the indicators of the subspace W and its complement $\overline{W}$ respectively. We note that $h(0) = 0$. The Fourier coefficient is by definition,

$$\hat{h}(\mathbf{w}) = \mathop{\mathbb{E}}_{x}\left[h(x)(1_W(x) - 1_{\overline{W}}(x))\right] \simeq \mathop{\mathbb{E}}_{x\neq 0}\left[(\mu(S_x) - \mu(S))(1_W(x) - 1_{\overline{W}}(x))\right].$$

We note that $\mu(S)$ is a constant and $\mathbb{E}_{x\neq 0}\left[1_W(x) - 1_{\overline{W}}(x)\right] \simeq -2^{-k}$ (which is negligble and is ignored). Hence

$$\hat{h}(\mathbf{w}) \simeq \mathop{\mathbb{E}}_{x\neq 0}\left[\mu(S_x)\cdot(1_W(x) - 1_{\overline{W}}(x))\right].$$

Denoting by $\mathbf{1}_S$ the indicator of set S, the above is same as

$$\begin{aligned}\mathop{\mathbb{E}}_{x\neq 0, L: x\in L}\left[\mathbf{1}_S(L)(1_W(x) - 1_{\overline{W}}(x))\right] &= \mathop{\mathbb{E}}_{L, x: x\in L, x\neq 0}\left[1_S(L)(1_W(x) - 1_{\overline{W}}(x))\right] \\ &\simeq 2^{-\ell}(\mu(S_W) - \mu(S)),\end{aligned}$$

as desired. In the last step, we observed that for $L \in S$, $L \subseteq W$, we get a contribution of 1, and for $L \in S$, $L \nsubseteq W$, we get a contribution of $-2^{-\ell}$. □

5.5. Proof of Theorem 5.5: the main argument. We now have all the ingredients needed to prove Theorem 5.5. We note the crucial fact that since S is $(1, \varepsilon)$-pseudorandom, by definition of pseudorandomness, we have

- $\|F\|_2^2 = \mu(S) \leqslant \varepsilon$.
- For all $x \neq 0$, $|h(x)| \stackrel{def}{=} |\mu(S_x) - \mu(S)| \leqslant \max\{\mu(S_x), \mu(S)\} \leqslant \varepsilon$.
- For all $w \neq 0$, $|\hat{h}(\mathbf{w})| \simeq 2^{-\ell}|\mu(S_W) - \mu(S)| \leqslant 2^{-\ell}\max\{\mu(S_W), \mu(S)\} \leqslant 2^{-\ell}\varepsilon$.

In short, $\|h\|_\infty \leqslant \varepsilon$ and $\|\hat{h}\|_\infty \leqslant 2^{-\ell}\varepsilon$. We start by writing $\mathbb{E}\left[F_{=1}^4\right]$ as

$$\begin{aligned}\mathbb{E}_L\left[F_{=1}[L]^4\right] &= \mathbb{E}_L\left[\left(\sum_{x\in L} f_{=1}(x)\right)^4\right] \simeq \mathbb{E}_L\left[\left(\sum_{x\in L} h(x)\right)^4\right] \\ &= \mathbb{E}_L\left[\sum_{x,y,z,w\in L} h(x)h(y)h(z)h(w)\right].\end{aligned}$$

We note that in the expectation above, first picking L and then picking $x, y, z, w \in L$ is equivalent to picking $x, y, z, w \in \mathbb{F}_2^k$ while preserving the linear dependencies among x, y, z, w. We consider different cases depending on these linear dependencies and in particular depending on the dimension of the space spanned by them. We note that if $d = \mathsf{dim}(\mathsf{Span}(x, y, z, w))$, $1 \leqslant d \leqslant 4$, then the corresponding term appears with a coefficient of (at most) $2^{\ell d}$, i.e. the number of tuples $(x, y, z, w) \in L$ with d-dimensional span. We now show how to bound each term by $O(\eta\varepsilon)$ where $\eta = \|F_{=1}\|_2^2 \simeq 2^\ell\|f_{=1}\|_2^2 \simeq 2^\ell\|h\|_2^2$.

Case $\dim(\mathsf{Span}(x,y,z,w)) = 1$**:**

In this case, we have $x = y = z = w$. In the notation below, the choice of x is switched from $x \in L$ to $x \in \mathbb{F}_2^k$ as indicated earlier (incurring a factor 2^ℓ in this case). The term is bounded as

$$\mathop{\mathbb{E}}_{L}\left[\sum_{x\in L} h(x)^4\right] = 2^\ell \mathop{\mathbb{E}}_{x}\left[h(x)^4\right] \leqslant 2^\ell \|h\|_\infty^2 \|h\|_2^2 \leqslant \eta\varepsilon^2 \leqslant \eta\varepsilon.$$

Case $\dim(\mathsf{Span}(x,y,z,w)) = 4$**:**

In this case, x, y, z, w are independent. Noting that $\mathbb{E}_x[h(x)] = 0$, the term is bounded as

$$\mathop{\mathbb{E}}_{L}\left[\sum_{x,y,z,w\in L} h(x)h(y)h(z)h(w)\right] \simeq 2^{4\ell} \mathop{\mathbb{E}}_{x}[h(x)]^4 = 0.$$

Case $\dim(\mathsf{Span}(x,y,z,w)) = 3$**:**

In this case, let us consider x, y, z as being independent. Then we have three subcases (up to symmetry): either $w = x$ or $w = x+y$ or $w = x+y+z$. The terms are bounded as (the third one being the interesting one)

$$\mathop{\mathbb{E}}_{L}\left[\sum_{x,y,z\in L} h(x)^2h(y)h(z)\right] \simeq 2^{3\ell} \mathop{\mathbb{E}}_{x}\left[h(x)^2\right] \mathop{\mathbb{E}}_{y}[h(y)] \mathop{\mathbb{E}}_{z}[h(z)] = 0.$$

$$\begin{aligned}\mathop{\mathbb{E}}_{L}\left[\sum_{x,y,z\in L} h(x)h(y)h(z)h(x+y)\right] &\simeq 2^{3\ell} \mathop{\mathbb{E}}_{x,y}[h(x)h(y)h(x+y)] \mathop{\mathbb{E}}_{z}[h(z)] \\ &= 0.\end{aligned}$$

$$\begin{aligned}\mathop{\mathbb{E}}_{L}\left[\sum_{x,y,z\in L} h(x)h(y)h(z)h(x+y+z)\right] &\simeq 2^{3\ell} \mathop{\mathbb{E}}_{x,y,z}[h(x)h(y)h(z)h(x+y+z)] \\ &= 2^{3\ell}\sum_{\mathbf{w}} \hat{h}(\mathbf{w})^4,\end{aligned}$$

which is bounded in turn as

$$2^{3\ell}\|\hat{h}\|_\infty^2 \left(\sum_{\mathbf{w}} \hat{h}(\mathbf{w})^2\right) \leqslant 2^{3\ell}(2^{-\ell}\varepsilon)^2\|h\|_2^2 \simeq \eta\varepsilon^2 \leqslant \eta\varepsilon.$$

Case $\dim(\mathsf{Span}(x,y,z,w)) = 2$**:**

In this case, let us consider x, y as being independent. Then we have three subcases (up to symmetry): either $z = w = x$ or $z = x$ and $w = y$ or

$z = w = x + y$. These terms are bounded as (using Cauchy-Schwartz for the third and the fourth terms and using $\eta = \|F_{=1}\|_2^2 \leqslant \|F\|_2^2 \leqslant \varepsilon$)

$$\mathop{\mathbb{E}}_{L}\left[\sum_{x,y\in L} h(x)^3 h(y)\right] \simeq 2^{2\ell} \mathop{\mathbb{E}}_{x}\left[h(x)^3\right] \mathop{\mathbb{E}}_{y}[h(y)] = 0.$$

$$\mathop{\mathbb{E}}_{L}\left[\sum_{x,y\in L} h(x)^2 h(y)^2\right] \simeq 2^{2\ell} \mathop{\mathbb{E}}_{x}\left[h(x)^2\right]^2 = (2^\ell \|h\|_2^2)^2 \simeq \eta^2 \leqslant \eta\varepsilon.$$

$$\begin{aligned}\mathop{\mathbb{E}}_{L}\left[\sum_{x,y\in L} h(x)h(y)h(x+y)^2\right] &\simeq 2^{2\ell} \mathop{\mathbb{E}}_{x,y}[h(x)h(x+y)h(y)h(x+y)] \\ &\leqslant 2^{2\ell} \sqrt{\mathop{\mathbb{E}}_{x,y}[h(x)^2 h(x+y)^2]} \sqrt{\mathop{\mathbb{E}}_{x,y}[h(y)^2 h(x+y)^2]} \\ &= 2^{2\ell}\|h\|_2^4 = (2^\ell \|h\|_2^2)^2 \simeq \eta^2 \leqslant \eta\varepsilon.\end{aligned}$$

This completes the proof of Theorem 5.5 (modulo all the approximations involved that are not incorporated for the sake of clarity).

6. Open problems

We conclude this article by pointing out some open problems at the interface of approximation algorithms and hardness, analysis, and geometry. We keep our descriptions brief, pointing to the respective sources for more detailed descriptions.

6.1. The Unique Games Conjecture. Even though the 2-to-2 Games Theorem now gives strong evidence towards its correctness, the Unique Games Conjecture remains open and baffling. As far as we know, it is possible (though not likely) that a constant number of rounds of the Lasserre hierarchy already gives an efficient algorithm for the Unique Games problem, disproving the Unique Games Conjecture. In other words, we do not know of a counter-example (= integrality gap) showing that this algorithm does not work. The proof of the 2-to-2 Games Theorem is not likely to extend to that of the Unique Games Conjecture: here the issue, at a technical level, is that the relevant mathematical structure is over a field $\mathbb{F}_2$, limiting the "completeness" to $\frac{1}{2}$. To prove the Unique Games Conjecture, one might wish to work over a field with $1 + \varepsilon$ elements!

At present, the best hope seems to be an approach suggested in [70]. The authors suggest a candidate reduction (or alternately a candidate integrality gap instance) where the relevant mathematical structure is over the real, Gaussian space. However the authors are unable to provide a full soundness analysis. Though it seems, roughly speaking, that it would help to understand surfaces with low Gaussian surface area, the authors are unable to isolate a concrete mathematical question whose solution would imply the soundness of their reduction (see [86] for some progress).

6.2. The small set expansion conjecture. Raghavendra and Steurer [88] propose the Small Set Expansion Conjecture and show that it implies the Unique Games Conjecture. Their conjecture states, roughly speaking, that it is NP-hard to find small, non-expanding sets in graphs. More formally, let $\Phi(S)$ denotes the expansion of a set S in an n-vertex graph G. The Small Set Expansion problem is to determine the expansion of G at "scale" δ, that is,

$$\Phi_\delta(G) = \min_{\frac{\delta}{2}n \leqslant |S| \leqslant \delta n} \Phi(S),$$

for a small, given constant $\delta > 0$. The conjecture states that for every constant $\varepsilon > 0$, there is a sufficiently small constant $\delta > 0$ such that is NP-hard to distinguish whether $\Phi_\delta(G) \leqslant \varepsilon$ or whether $\Phi_\delta(G) \geqslant 1 - \varepsilon$.

While it would be great if this conjecture were correct (it would have several interesting consequences on top of those already implied by the Unique Games Conjecture), we note that there is a reason to doubt it. The conjecture implies the following combinatorial conjecture.

CONJECTURE 6.1. *For every positive constant $\varepsilon > 0$, there is a positive constant $\delta > 0$ and a family of n-vertex graphs $\{G_n\}_{n\to\infty}$ such that: (1) every subset of size between $\frac{\delta}{2}n$ and δn in G_n has expansion at least (say) $\frac{1}{2}$ and (2) the number of eigenvalues of the normalized adjacency matrix of G_n that exceed $1 - \varepsilon$ is at least n^δ.*

We refer the reader to [17, 65] regarding this implication of the Small Set Expansion Conjecture. Contra-positively, if there were no such family of graphs for some $\varepsilon > 0$, the Small Set Expansion Conjecture would be incorrect.

6.3. The symmetric parallel repetition. Given a Unique Games instance $\mathcal{U}$, k-wise "parallel repetition" is a natural product operation that yields a new Unique Games instance $\mathcal{U}^{\otimes k}$. Roughly speaking, the variables of the instance $\mathcal{U}^{\otimes k}$ are k-tuples of variables $(u_1, \ldots, u_k)$ of the instance $\mathcal{U}$ and constraints are between tuples $(u_1, \ldots, u_k)$ and $(v_1, \ldots, v_k)$ whenever there are constraints between (u_i, v_i) in the instance $\mathcal{U}$. A well-studied question is how $\mathsf{OPT}(\mathcal{U})$ and $\mathsf{OPT}(\mathcal{U}^{\otimes k})$ are related. Clearly, if $\mathsf{OPT}(\mathcal{U}) = 1 - \varepsilon$, then $\mathsf{OPT}(\mathcal{U}^{\otimes k}) \geqslant (1 - \varepsilon)^k$ since $\mathcal{U}^{\otimes k}$ always has an assignment that is derived faithfully from an assignment to $\mathcal{U}$ (but there could be other assignments and that's what makes the question challenging). It is known that $\mathsf{OPT}(\mathcal{U}^{\otimes k}) \leqslant (1 - \varepsilon^2)^{\Omega(k)}$ and moreover that this upper bound cannot be improved [91, 54, 90, 92].

The open question is whether the upper bound can be improved to $\mathsf{OPT}(\mathcal{U}^{\otimes k}) \leqslant (1 - \varepsilon)^{\Omega(k)}$ provided (1) there is a (expansion-like) structural condition on the graph of constraints of the instance $\mathcal{U}$ and (2) the assignments to the instance $\mathcal{U}^{\otimes k}$ are required to be symmetric, meaning, the assignments to a tuple $(u_1, \ldots, u_k)$ and its permuted copy $(u_{\pi(1)}, \ldots, u_{\pi(k)})$ are permuted as well. The question is relevant towards "gap-amplification",

whereby, one might hope to first prove hardness of Unique Games with a weaker gap and then amplify the gap to prove the Unique Games Conjecture in full. The strength of the (expansion-like) structural condition would need to depend on this intended application.

6.4. Approximating CSPs on satisfiable instances. As mentioned before, Constraint Satisfaction Problems (CSPs) are some of the most widely studied NP-hard problems. Let $P : \{0,1\}^k \to \{\mathsf{True}, \mathsf{False}\}$ be a k-ary Boolean predicate. An instance of the CSP(P) problem consists of n Boolean variables $x_1, \ldots, x_n$ and m constraints $C_1, \ldots, C_m$ where each constraint is the predicate P applied to an ordered subset of k variables, possibly in negated form.

A CSP(P) instance is said to be satisfiable if there is an assignment that satisfies all constraints. The instance is said to be near-satisfiable if there is an assignment that satisfies almost all the constraints. A most fascinating question is: given a predicate P, what is the optimal approximation algorithm for CSP(P) where the goal is to efficiently find an assignment that satisfies a maximum number of the constraints?

A remarkable result of Raghavendra [89] cited before completely characterizes such "optimal algorithm" on near-satisfiable instances; moreover, the algorithm is necessarily based on a semi-definite programming relaxation. However a complete characterization on satisfiable instances remains wide open. This case is qualitatively different and much more delicate (as demonstrated by CSPs with linear constraints; these are easy on satisfiable instances but hard on near-satisfiable instances). The optimal algorithm here is, at the very least, a clever combination of SDPs and Gaussian elimination. The well-known "Dichotomy Conjecture" needed to be resolved first before attempting this question; thankfully, it has now been resolved (independently) by Bulatov and Zhuk [25, 100]! While the proofs are *algebraic*, it could be very productive to seek an *analytic* proof of the Dichotomy Conjecture (as suggested in [79, 24]), which in turn, might hold the key to resolving the question of approximability of CSPs on satisfiable instances.

6.5. Uniform sparsest cut. In the Uniform Sparsest Cut problem, given an n-vertex graph G, one seeks to determine its expansion $\Phi(G) = \min_{1 \leqslant |S| \leqslant \frac{n}{2}} \Phi(S)$. There is a natural semi-definite programming relaxation to estimate the graph expansion, but we do not know its precise approximation guarantee (= the integrality gap). The known upper and lower bounds are $\tilde{O}(\sqrt{\log n})$ [7] and $2^{\Omega(\sqrt{\log \log n})}$ [34, 59] respectively. For the more general version of this problem known as the Non-uniform Sparsest Cut problem, the sharp bound of $\tilde{\Theta}(\sqrt{\log n})$ is known [4, 87]. These problems have influenced a tremendous amount of research (e.g. [80, 32, 87]) including connections with the Unique Games Conjecture (e.g. [76]) as referred to in Section 2.6.4.

6.6. ETH versus Gap-ETH. We recall the Exponential Time Hypothesis (ETH) stating that solving Exact-3SAT with n variables, i.e. dis-

tinguishing whether a Exact-3SAT instance is satisfiable or not, takes truly exponential time, i.e. time $2^{\gamma n}$ for some constant $\gamma > 0$. The Gap-ETH [37, 83] is a stronger hypothesis that solving Gap3SAT, i.e. distinguishing whether a 3SAT instance is satisfiable or at most $(1-\beta)$-satisfiable, takes time $2^{\beta n}$ for some constant $\beta > 0$. Both these hypotheses are very useful towards hardness results where the standard techniques do not suffice. Recently, researchers have been quite interested in knowing whether Gap-ETH follows from ETH (if so, both would be equivalent).

6.7. Analysis of boolean functions. Analysis of Boolean functions $f : \{-1,1\}^n \to \{-1,1\}$ has played an influential role in hardness of approximation (and there has been influence in the reverse direction as well). Some prominent examples of this interaction, as cited before, are: (1) Sparsest Cut: the Kahn, Kalai, Linial Theorem and Bourgain's Junta Theorem [76, 29, 58, 23] (2) Vertex Cover: Friedgut's Junta Theorem and It Ain't Over Till It's Over Theorem [74, 15, 45, 84] (3) Max Cut: Majority Is Stablest Theorem [66, 84] (4) 2-to-2 Games: Structure of non-expanding sets in Grassmann graph [68, 39, 38, 69].

While we do not know their direct application to hardness of approximation results, we do mention a couple of open questions in Boolean function analysis, posed in [46, 1] respectively.

- Entropy-Influence Conjecture: There is an absolute constant $C \geqslant 1$ such that for any $f : \{-1,1\}^n \to \{-1,1\}$,

$$\sum_{\mathbf{w} \in \mathbb{F}_2^n} \hat{f}(\mathbf{w})^2 \log\left(\frac{1}{|\hat{f}(\mathbf{w})|}\right) \leqslant C \cdot I[f],$$

  i.e. the Fourier entropy of a Boolean function is at most a constant times its total influence. The total influence $I[f] = \sum_{i=1}^n I_i[f]$ is the sum of all coordinate-wise influences.
- There is an absolute constant K such that for any $g : \{-1,1\}^n \to [-1,1]$, $\mathbb{E}[g] = 0, \mathbb{E}[g^2] \geqslant \frac{1}{50}$ (say), there exists a co-ordinate $1 \leqslant i \leqslant n$ such that

$$I_i[g] \geqslant \frac{1}{K \cdot \deg(g)^K},$$

  i.e. when the variance is a constant, there exists a co-ordinate with influence at least inverse polynomial in the degree (this is known to hold if g were Boolean). The influence for a real-valued function is defined as $I_i[g] = \sum_{\mathbf{w}:\mathbf{w}_i \neq 0} \hat{g}(\mathbf{w})^2$.

Acknowledgement

Many thanks to Amey Bhangale, Euiwoong Lee, and Dor Minzer for proof-reading and commenting on an earlier draft of the article. Thanks to the organizers of the *Current Developments in Mathematics 2019* conference

at Harvard University, for inviting the author and for insisting on writing this article!

References

[1] Scott Aaronson and Andris Ambainis. The need for structure in quantum speedups. *Theory of Computing*, 10:133–166, 2014. MR 3249097

[2] N. Alon and B. Klartag. Economical toric spines via Cheeger's inequality. *Journal of Topology and Analysis*, 1:101–111, 2009. MR 2541756

[3] S. Arora, L. Babai, J. Stern, and E. Z. Sweedyk. The hardness of approximate optima in lattices, codes and systems of linear equations. *Journal of Computer and Systems Sciences*, 54:317–331, 1997. MR 1462727

[4] S. Arora, J. Lee, and A. Naor. Euclidean distortion and the sparsest cut. In *Proc. 37th ACM Symposium on Theory of Computing*, pages 553–562, 2005. MR 2181659

[5] S. Arora and C. Lund. *Approximation Algorithms for NP-hard Problems, editor: D. Hochbaum*. PWS Publishing, 1996.

[6] S. Arora, C. Lund, R. Motawani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. MR 1639346

[7] S. Arora, S. Rao, and U. Vazirani. Expander flows, geometric embeddings and graph partitioning. In *Proc. 36th ACM Symposium on Theory of Computing*, pages 222–231, 2004. MR 2121604

[8] S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. MR 1614328

[9] Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. *J. ACM*, 62(5):42:1–42:25, 2015. MR 3424199

[10] Sanjeev Arora, Eden Chlamtac, and Moses Charikar. New approximation guarantee for chromatic number. In *Proc. ACM Symposium on the Theory of Computing*, pages 215–224, 2006. MR 2277147

[11] Sanjeev Arora, Subhash Khot, Alexandra Kolla, David Steurer, Madhur Tulsiani, and Nisheeth K. Vishnoi. Unique games on expanding constraint graphs are easy. In *Proc. ACM Symposium on the Theory of Computing*, pages 21–28, 2008. MR 2582928

[12] Per Austrin, Subhash Khot, and Muli Safra. Inapproximability of vertex cover and independent set in bounded degree graphs. *Theory of Computing*, 7(1):27–43, 2011. MR 2804389

[13] Nikhil Bansal. Approximating independent sets in sparse graphs. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4–6, 2015*, pages 1–8, 2015. MR 3451025

[14] Nikhil Bansal, Anupam Gupta, and Guru Guruganesh. On the Lovász theta function for independent sets in sparse graphs. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, June 14–17, 2015, Portland, OR, USA*, pages 193–200, 2015. MR 3388197

[15] Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25–27, 2009, Atlanta, Georgia, USA*, pages 453–462, 2009. MR 2648426

[16] Boaz Barak, Fernando G. S. L. Brandão, Aram Wettroth Harrow, Jonathan A. Kelner, David Steurer, and Yuan Zhou. Hypercontractivity, sum-of-squares proofs, and their applications. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, May 19–22, 2012, New York, NY, USA*, pages 307–326, 2012. MR 2961513

[17] Boaz Barak, Parikshit Gopalan, Johan Håstad, Raghu Meka, Prasad Raghavendra, and David Steurer. Making the long code shorter. *SIAM J. Comput.*, 44(5):1287–1324, 2015. MR 3416138

[18] Boaz Barak, Pravesh K. Kothari, and David Steurer. Small-set expansion in shortcode graph and the 2-to-2 conjecture. In *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10–12, 2019, San Diego, California, USA*, pages 9:1–9:12, 2019. MR 3899803

[19] Boaz Barak, Prasad Raghavendra, and David Steurer. Rounding semidefinite programming hierarchies via global correlation. In *FOCS*, pages 472–481, 2011. MR 2932723

[20] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free bits, pcps, and nonapproximability-towards tight results. *SIAM J. Comput.*, 27(3):804–915, 1998. MR 1612644

[21] Amey Bhangale and Subhash Khot. UG-hardness to NP-hardness by losing half. In *34th Computational Complexity Conference, CCC 2019, July 18–20, 2019, New Brunswick, NJ, USA*, pages 3:1–3:20, 2019. MR 3984608

[22] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *J. Comput. Syst. Sci.*, 47(3):549–595, 1993. MR 1248868

[23] J. Bourgain. On the distribution of the Fourier spectrum of Boolean functions. *Israel J. of Math.*, (131):269–276, 2002. MR 1942312

[24] Jonah Brown-Cohen and Prasad Raghavendra. Correlation decay and tractability of CSPs. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11–15, 2016, Rome, Italy*, pages 79:1–79:13, 2016. MR 3577140

[25] Andrei Bulatov. A Dichotomy theorem for nonuniform CSPs. In *IEEE Annual Symposium on Foundations of Computer Science, FOCS*, 2017. MR 3734240

[26] Jakub Bulín, Andrei A. Krokhin, and Jakub Oprsal. Algebraic approach to promise constraint satisfaction. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, June 23–26, 2019, Phoenix, AZ, USA*, pages 602–613, 2019. MR 4003368

[27] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Near-optimal algorithms for unique games. In *Proc. ACM Symposium on the Theory of Computing*, pages 205–214, 2006. MR 2277146

[28] S. Chatterjee. A simple invariance theorem. *arXiv:math/0508213v1*, 2005.

[29] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. In *Proc. 20th IEEE Conference on Computational Complexity*, pages 144–153, 2005. MR 2243123

[30] J. Cheeger and B. Kleiner. On the differentiation of Lipschitz maps from metric measure spaces to Banach spaces. *Inspired by S.S. Chern, Volume 11 of Nankai Tracts. Math.*, pages 129–152, 2006. MR 2313333

[31] Jeff Cheeger and Bruce Kleiner. Differentiating maps into L^1, and the geometry of BV functions. *Ann. of Math. (2)*, 171(2):1347–1385, 2010. MR 2630066

[32] Jeff Cheeger, Bruce Kleiner, and Assaf Naor. A $(\log n)^{\Omega(1)}$ integrality gap for the sparsest cut SDP. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009*, pages 555–564. IEEE Computer Soc., Los Alamitos, CA, 2009. MR 2648435

[33] Jeff Cheeger, Bruce Kleiner, and Assaf Naor. Compression bounds for Lipschitz maps from the Heisenberg group to L_1. *Acta Math.*, 207(2):291–373, 2011. MR 2892612

[34] N. Devanur, S. Khot, R. Saket, and N. Vishnoi. Integrality gaps for sparsest cut and minimum linear arrangement problems. In *Proc. 38th ACM Symposium on Theory of Computing*, 2006. MR 2277179

[35] I. Dinur. The PCP theorem by gap amplification. In *Proc. 38th ACM Symposium on Theory of Computing*, 2006. MR 2314254

[36] I. Dinur and S. Safra. The importance of being biased. In *Proc. 34th Annual ACM Symposium on Theory of Computing*, 2002. MR 2121123

[37] Irit Dinur. Mildly exponential reduction from gap 3sat to polynomial-gap label-cover. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:128, 2016.

[38] Irit Dinur, Subhash Khot, Guy Kindler, Dor Minzer, and Muli Safra. On non-optimally expanding sets in grassmann graphs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, June 25–29, 2018, Los Angeles, CA, USA*, pages 940–951, 2018. MR 3826307

[39] Irit Dinur, Subhash Khot, Guy Kindler, Dor Minzer, and Muli Safra. Towards a proof of the 2-to-1 games conjecture? In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, June 25–29, 2018, Los Angeles, CA, USA*, pages 376–389, 2018. MR 3826261

[40] Irit Dinur, Elchanan Mossel, and Oded Regev. Conditional hardness for approximate coloring. *SIAM J. Comput.*, 39(3):843–873, 2009. MR 2538841

[41] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998. MR 1675095

[42] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, 1996. MR 1408323

[43] U. Feige and L. Lovász. Two-prover one-round proof systems, their power and their problems. In *Proc. 24th Annual ACM Symposium on Theory of Computing*, pages 733–744, 1992.

[44] Uriel Feige, Guy Kindler, and Ryan O'Donnell. Understanding parallel repetition requires understanding foams. In *Proc. Annual IEEE Conference on Computational Complexity*, pages 179–192, 2007.

[45] E. Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, 1998. MR 1645642

[46] Ehud Friedgut and Gil Kalai. Every monotone graph property has a sharp threshold. *Proc. Amer. Math. Soc.*, 124(10):2993–3002, 1996. MR 1371123

[47] M. Goemans and D. Williamson. 0.878 approximation algorithms for MAX-CUT and MAX-2SAT. In *Proc. 26th ACM Symposium on Theory of Computing*, pages 422–431, 1994.

[48] Michel X. Goemans. Semidefinite programming in combinatorial optimization. *Math. Program.*, 79:143–161, 1997. MR 1464765

[49] Dima Grigoriev. Linear lower bound on degrees of positivstellensatz calculus proofs for the parity. *Theor. Comput. Sci.*, 259(1-2):613–622, 2001. MR 1832812

[50] Venkatesan Guruswami, Rajsekar Manokaran, and Prasad Raghavendra. Beating the random ordering is hard: Inapproximability of maximum acyclic subgraph. In *Proc. Annual IEEE Symposium on Foundations of Computer Science*, pages 573–582, 2008.

[51] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182:105–142, 1999. MR 1687331

[52] J. Håstad. Some optimal inapproximability results. *Journal of ACM*, 48:798–859, 2001. MR 2144931

[53] Johan Håstad. On the Efficient Approximability of Constraint Satisfaction Problems. In *Surveys in Combinatorics*, volume 346, pages 201–222. Cambridge University Press, 2007. MR 2252794

[54] Thomas Holenstein. Parallel repetition: simplifications and the no-signaling case. In *Proc. ACM Symposium on the Theory of Computing*, pages 411–419, 2007. MR 2402466

[55] Russell Impagliazzo. Hardness as randomness: a survey of universal derandomization. In *Proceedings of the International Congress of Mathematicians, Vol. III (Beijing, 2002)*, pages 659–672. Higher Ed. Press, Beijing, 2002. MR 1957568

[56] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. MR 1820597

[57] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001. MR 1894519

[58] J. Kahn, G. Kalai, and N. Linial. The influence of variables on Boolean functions. In *Proc. 29th Symposium on the Foundations of Computer Science*, pages 68–80, 1988.

[59] Daniel M. Kane and Raghu Meka. A PRG for Lipschitz functions of polynomials with applications to sparsest cut. In *Symposium on Theory of Computing Conference, STOC'13, June 1–4, 2013, Palo Alto, CA, USA*, pages 1–10, 2013. MR 3210761

[60] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Proceedings of a symposium on the Complexity of Computer Computations, March 20–22, 1972, the IBM Thomas J. Watson Research Center, Yorktown Heights, New York*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972. MR 0378476

[61] Ken-Ichi Kawarabayashi and Mikkel Thorup. Coloring 3-colorable graphs with less than n1/5 colors. *J. ACM*, 64(1):4:1–4:23, 2017. MR 3634492

[62] S. Khot. Inapproximability of NP-complete problems, discrete Fourier analysis, and geometry. In *Proc. the International Congress of Mathematicians*, 2010. MR 2827989

[63] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proceedings of 34th Annual ACM Symposium on Theory of Computing, May 19–21, 2002, Montréal, Québec, Canada*, pages 767–775, 2002. MR 2121525

[64] Subhash Khot. On the unique games conjecture (invited survey). In *IEEE Conference on Computational Complexity*, pages 99–121, 2010. MR 2932348

[65] Subhash Khot. Hardness of approximation.In *Proc. of the International Congress of Mathematicians*, 2014. MR 3728489

[66] 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. MR 2306295

[67] Subhash Khot, Dor Minzer, Dana Moshkovitz, and Muli Safra. Pseudorandom sets in Johnson graph have near-perfect expansion. *ECCC Report TR18-078*.

[68] Subhash Khot, Dor Minzer, and Muli Safra. On independent sets, 2-to-2 games, and Grassmann graphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, June 19–23, 2017, Montreal, QC, Canada*, pages 576–589, 2017. MR 3678212

[69] Subhash Khot, Dor Minzer, and Muli Safra. Pseudorandom sets in Grassmann graph have near-perfect expansion. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, October 7–9, 2018, Paris, France*, pages 592–601, 2018. MR 3899624

[70] Subhash Khot and Dana Moshkovitz. Candidate hard unique game. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, June 18–21, 2016, Cambridge, MA, USA*, pages 63–76, 2016. MR 3536555

[71] Subhash Khot and Assaf Naor. Approximate kernel clustering. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25–28, 2008, Philadelphia, PA, USA*, pages 561–570, 2008. MR 2573605

[72] Subhash Khot and Assaf Naor. Sharp kernel clustering algorithms and their associated grothendieck inequalities. *Random Struct. Algorithms*, 42(3):269–300, 2013. MR 3039681

[73] Subhash Khot and Ryan O'Donnell. SDP gaps and UGC-hardness for Max-Cut-Gain. *Theory of Computing*, 5(1):83–117, 2009. MR 2521349

[74] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within $2-\epsilon$. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008. MR 2384079

[75] Subhash Khot and Muli Safra. A two-prover one-round game with strong soundness. *Theory of Computing*, 9:863–887, 2013. MR 3142437

[76] Subhash Khot and Nisheeth K. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative-type metrics into l_1. *J. ACM*, 62(1):8:1–8:39, 2015. MR 3323774

[77] Guy Kindler, Ryan O'Donnell, Anup Rao, and Avi Wigderson. Spherical cubes and rounding in high dimensions. In *Proc. Annual IEEE Symposium on Foundations of Computer Science*, pages 189–198, 2008.

[78] Alexandra Kolla, Konstantin Makarychev, and Yury Makarychev. How to play unique games against a semi-random adversary: Study of semi-random models of unique games. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, October 22–25, 2011, Palm Springs, CA, USA*, pages 443–452, 2011. MR 2932720

[79] Gábor Kun and Mario Szegedy. A new line of attack on the dichotomy conjecture. *Eur. J. Comb.*, 52:338–367, 2016. MR 3425984

[80] J. R. Lee and A. Naor. l_p metrics on the Heisenberg group and the Goemans-Linial conjecture. In *Proc. 47th IEEE Symposium on Foundations of Computer Science*, pages 99–108, 2006.

[81] N. Linial. Finite metric spaces-combinatorics, geometry and algorithms. In *Proc. International Congress of Mathematicians*, volume 3, pages 573–586, 2002. MR 1957562

[82] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. *Journal of the ACM*, 41:960–981, 1999. MR 1371491

[83] Pasin Manurangsi and Prasad Raghavendra. A birthday repetition theorem and complexity of approximating dense CSPs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10–14, 2017, Warsaw, Poland*, pages 78:1–78:15, 2017. MR 3685818

[84] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. In *Proc. 46th IEEE Symposium on Foundations of Computer Science*, pages 21–30, 2005.

[85] Elchanan Mossel. Gaussian bounds for noise correlation of functions and tight analysis of long codes. In *Proc. Annual IEEE Symposium on Foundations of Computer Science*, pages 156–165, 2008. MR 2594620

[86] Elchanan Mossel and Joe Neeman. Noise stability and correlation with half spaces. *Electron. J. Probab.*, 23:Paper No. 16, 17, 2018. MR 3771753

[87] Assaf Naor and Robert Young. The integrality gap of the Goemans-Linial SDP relaxation for sparsest cut is at least a constant multiple of $\sqrt{\log n}$. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, June 19–23, 2017, Montreal, QC, Canada*, pages 564–575, 2017. MR 3678211

[88] P. Raghavendra and D. Steurer. Graph expansion and the unique games conjecture. In *Proc. 42nd ACM Symposium on Theory of Computing*, 2010. MR 2743325

[89] Prasad Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, May 17–20, 2008, British Columbia, Canada*, pages 245–254, 2008. MR 2582901

[90] Anup Rao. Parallel repetition in projection games and a concentration bound. In *Proc. ACM Symposium on the Theory of Computing*, pages 1–10, 2008. MR 2582654

[91] R. Raz. A parallel repetition theorem. *SIAM J. of Computing*, 27(3):763–803, 1998. MR 1612640

[92] Ran Raz. A counterexample to strong parallel repetition. In *Proc. Annual IEEE Symposium on Foundations of Computer Science*, pages 369–373, 2008.

[93] V.I. Rotar'. Limit theorems for polylinear forms. *J. Multivariate Anal.*, 9(4):511–530, 1979. MR 0556909

[94] A. Samorodnitsky and L. Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proc. 32nd ACM Symposium on Theory of Computing*, pages 191–199, 2000. MR 2114532

[95] Grant Schoenebeck. Linear level Lasserre lower bounds for certain k-CSPs. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25–28, 2008, Philadelphia, PA, USA*, pages 593–602, 2008.

[96] D. Steurer. Subexponential algorithms for d-to-1 two-prover games and for certifying almost perfect expansion. *Unpublished manuscript*, 2011.

[97] L. Trevisan. Inapproximability of combinatorial optimization problems. *Optimisation Combinatiore 2, (Vangelis Paschos, Editor), Hermes*, 2005.

[98] Luca Trevisan. Approximation algorithms for unique games. *Theory of Computing*, 4(1):111–128, 2008. MR 2448925

[99] V. V. Vazirani. *Approximation Algorithms*. Springer, 2001. MR 1851303

[100] Dmitriy Zhuk. The proof of CSP Dichotomy conjecture. In *IEEE Annual Symposium on Foundations of Computer Science, FOCS*, 2017. MR 3734241

Department of Computer Science, Courant Institute of Mathematical Sciences, New York University, USA

E-mail address: khot@cs.nyu.edu