2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

Making the Long Code Shorter

Boaz Barak
Microsoft Research New England, Cambridge, MA.
Email: boaz@microsoft.com

Parikshit Gopalan
Microsoft Research Silicon Valley, Mountain View, CA.
Email: parik@microsoft.com

Johan Håstad[1]
Royal Institute of Technology, Stockholm, Sweden.
Email: johanh@csc.kth.se

Raghu Meka
Institute of Advanced Study, Princeton, NJ.
Email: raghuvardhan@gmail.com

Prasad Raghavendra [2]
University of California, Berkeley, CA.
Email: prasad@cs.berkeley.edu

David Steurer
Microsoft Research New England, Cambridge, USA.
Email: dsteurer@cs.princeton.edu

***Abstract*—The *long code* is a central tool in hardness of approximation, especially in questions related to the unique games conjecture. We construct a new code that is exponentially more efficient, but can still be used in many of these applications. Using the new code we obtain exponential improvements over several known results, including the following:**

1) **For any $\varepsilon > 0$, we show the existence of an n vertex graph G where every set of $o(n)$ vertices has expansion $1-\varepsilon$, but G's adjacency matrix has more than $\exp(\log^{\delta} n)$ eigenvalues larger than $1-\varepsilon$, where δ depends only on ε. This answers an open question of Arora, Barak and Steurer (FOCS 2010) who asked whether one can improve over the noise graph on the Boolean hypercube that has $\mathrm{poly}(\log n)$ such eigenvalues.**
2) **A gadget that reduces unique games instances with linear constraints modulo K into instances with alphabet k with a blowup of $K^{\mathrm{polylog}(K)}$, improving over the previously known gadget with blowup of $2^{\Omega(K)}$.**
3) **An n variable integrality gap for Unique Games that survives $\exp(\mathrm{poly}(\log\log n))$ rounds of the SDP + Sherali Adams hierarchy, improving on the previously known bound of $\mathrm{poly}(\log\log n)$.**

We show a connection between the local testability of linear codes and small set expansion in certain related Cayley graphs, and use this connection to derandomize the noise graph on the Boolean hypercube.

***Keywords*-Unique games conjecture, Small set expansion, Long Code, Locally Testable Codes.**

I. Introduction

Khot's *Unique Games Conjecture* [13] (UGC) has been the focus of intense research effort in the last few years. The conjecture posits the hardness of approximation for a certain constraint satisfaction problem, and shows promise to settle many open questions in theory of approximation algorithms. Specifically, an instance Γ of the Unique Games problem with n variables and alphabet Σ is described by a collection of constraints of the form (x, y, π) where π is a permutation over Σ. An *assignment* to Γ is a mapping f from $[n]$ to Σ, and f's value is the fraction of constraints (x, y, π) such that $f(y) = \pi(f(x))$. The Unique Games Conjecture is that for any $\varepsilon > 0$, there is some finite Σ such that it is **NP** hard to distinguish between the case that a Unique Games instance Γ with alphabet Γ has an assignment satisfying $1-\varepsilon$ fraction of the constraints, and the case that every assignment satisfies at most ε fraction of Γ's constraint.

Many works have been devoted to studying the plausibility of the UGC, as well as exploring its implications and obtaining unconditional results motivated by this effort. Tantalizingly, at the moment we have very little evidence for the truth of this conjecture. One obvious reason to believe the UGC is that no algorithm is known to contradict it, though that of course may have more to do with our proof techniques for algorithm analysis than actual computational difficulty. Thus perhaps the strongest evidence for the conjecture comes from results showing particular instances on which certain natural algorithms will fail to solve the problem. However, even those integrality gaps are quantitatively rather weak. For example, while Arora, Barak and Steurer [2] showed a subexponential upper bound on an algorithm for the Unique Games and the related Small-Set Expansion problem, the hardest known instances for their algorithm only required quasipolynomial time [17]. Similarly (and related to this), known integrality gaps for Unique Games and related problems do not rule out their solution by an $O(\log n)$-round semidefinite hierarchy, an algorithm that can be implemented in quasipolynomial (or perhaps even polynomial [6]) time.

The *long code* has been a central tool in many of these works. This is the set of "dictator" functions mapping $\mathbb{F}_2^N$ to $\mathbb{F}_2$ that have the form $x_1 \dots x_N \mapsto x_i$ for some i. Many hardness reductions (especially from Unique

[1] Supported by ERC grant 226203
[2] Supported by NSF Career Award and Sloan Fellowship.

0272-5428/12 $26.00 © 2012 IEEE
DOI 10.1109/FOCS.2012.83

GAMES) and constructions of integrality gap instances use the long code as a tool. However, this is also the source of their inefficiency, as the long code is indeed quite long. Specifically, it has only N codewords but dimension 2^N, which leads to exponential blowup in many of these applications. In this work, we introduce a different code, which we call the "short code", that is exponentially more efficient, and can be used in the long code's place in many of these applications, leading to significant quantitative improvements. In particular, we use our code to show instances on which the [2] algorithm, as well as certain semidefinite hierarchies, require almost sub-exponential time, thus considerably strengthening the known evidence in support of the Unique Games Conjecture. Moreover, our results open up possibilities for *qualitative* improvements as well, in particular suggesting a new approach to prove the Unique Games Conjecture via an efficient alphabet reduction.

A. Our results

At the heart of the long code's applications lie its connection with the *noisy Hypercube*. This is the weighted graph $H_{N,\varepsilon}$ whose vertices are elements in $\mathbb{F}_2^N$ where a random neighbor of $x \in \mathbb{F}_2^N$ is obtained by flipping each bit of x independently with probability ε.[1] It is not too hard to show that the codewords of the long code correspond to the top eigenvectors of the noisy hypercube which also give the minimal bisections of the graph, cutting only an ε fraction of edges. In addition, several converse results are known, showing that bisections (and more general functions) cutting few edges are close to these top eigenvectors (or *dictatorships*) in some sense. (One such result is the "Majority is Stablest" Theorem of [21].) The inefficiency of the long code is manifested in the fact that the number of vertices of the noisy cube is exponential in the number N of its top eigenvectors.

The short code.: Another way to describe the long code is that it encodes $x \in \mathbb{F}_2^n$ by a binary vector v_x of length 2^{2^n} where $v_x(f) = f(x)$ for every function $f : \mathbb{F}_2^n \to \mathbb{F}_2$. This view also accounts for the name "long code", since one can see that this is the longest possible encoding of x without having repeated coordinates. For every subset $\mathcal{D}$ of functions mapping $\mathbb{F}_2^n$ to $\mathbb{F}_2$, we define the *$\mathcal{D}$-short code* to be the code that encodes x by a vector v_x of length $|\mathcal{D}|$ where $v_x(f) = f(x)$ for every $f \in \mathcal{D}$. Note that this is a very general definition that encapsulates any code without repeated coordinates. For $d \in \mathbb{N}$, we define the *d-short code* to be the the $\mathcal{D}$-short code where $\mathcal{D}$ is the set of all polynomials over $\mathbb{F}_2^n$ of degree at most d. Note that the 1-short code is the Hadamard code, while the n-short code is the long code. We use the name "short code" to denote the d short code for $d = O(1)$. Note that the short code has 2^n codewords and dimension roughly 2^{n^d}, and hence only quasipolynomial blowup, as opposed to the exponential blowup of the long code. Our main contribution is a construction of a "derandomized" noisy cube, which is a small subgraph of the noisy cube that enjoys the same relations to the short code (including a "Majority is Stablest" theorem) as the original noisy cube has to the long code. As a result, in many applications one can use the short code and the derandomized cube in place of the long code and the noisy cube, obtaining an exponential advantage. Using this approach we obtain the following results:

Small set expanders with many large eigenvalues.: Our first application, and the motivation to this work, is a question of Arora, Barak and Steurer [2]: How many eigenvectors with eigenvalue at least $1-\varepsilon$ can an n-vertex *small set expander* graph have? We say a graph is a small set expander (SSE) if all sufficiently small subsets of vertices have, say, at least 0.9 fracton of their neighbors outside the set. [2] showed an upper bound of $n^{O(\varepsilon)}$ on the number of large (i.e., greater than $1-\varepsilon$) eigenvalues of a small set expander. Arora et al. then observed that the subspace enumeration algorithm of [18], [17] for approximating small set expansion in an input graph takes time at most exponential in this number, which they then use to give an algorithm with similar running time for the UNIQUE GAMES problem. Up to this work, the best lower bound was polylog(n), with the example being the noisy cube, and hence as far as we knew the algorithm of [2] could solve the small set expansion problem in quasipolynomial time, which in turn might have had significant implications for the UNIQUE GAMES problem as well. Our derandomized noisy cube yields an example with an almost polynomial number of large eigenvalues:

Theorem 1. *For every $\varepsilon > 0$, there is an n-vertex small set expander graph with $2^{(\log n)^{\Omega(1)}}$ eigenvectors with corresponding eigenvalues at least $1-\varepsilon$.*

Theorem 1 actually follows from a more general result connecting locally testable codes to small set expanders, which we instantiate with the Reed Muller code. See Section II for details.

Efficient integrality gaps.: There is a standard semidefinite program (SDP) relaxation for the UNIQUE GAMES problem, known as the "basic SDP" [16], [24]. Several works have shown upper and lower bounds on the approximation guarantees of this relaxation, and for constant alphabet size, the relation between the alphabet size and approximation guarantee is completely understood [8]. However, for unbounded alphabet, there

[1] This graph is closely related and has similar properties to the unweighted graph where we connect x and y if their Hamming distance is at most εN.

was still a big gap in our understanding of the relation between the approximation guarantee and the number of variables. Gupta and Talwar [11] showed that if the relaxation's value is $1-\varepsilon$, there is an assignment satisfying $1 - O(\varepsilon \log n)$ fraction of constraints. On the other hand, Khot and Vishnoi [16] gave an integrality gap instance where the relaxation's value was $1 - 1/\operatorname{poly}(\log\log n)$[2] but the objective value (maximum fraction of constraints satisfied by any assignment) was $o(1)$. It was a natural question whether this could be improved, and indeed our short code allows us to obtain an almost exponential improvement:

Theorem 2. *There is an n-variable instance of* Unique Games *with objective value $o(1)$ but for which the standard semidefinite programming (SDP) relaxation has value at least $1 - 1/\operatorname{qpolylog}(n)$.*[3]

Integrality gaps for SDP hierarchies.: Our best evidence for the hardness of the Unique Games Conjecture comes from integrality gap instances for semidefinite programming *hierarchies*. These are strengthened versions of the basic SDP where one obtains tighter relaxations by augmenting them with additional constraints, we refer to [10] for a good overview of SDP hierarchies. These hierarchies are generally paramaterized by a number r (often called the *number of rounds*), where the first round corresponds to the Basic SDP, and the n^{th} round (where n is the instance size) corresponds to the exponential brute force algorithm that always computes an optimal answer. Generally, the r^{th}-round of each such hierarchy can be evaluated in $n^{O(r)}$ time (though in some cases $n^{O(1)}2^{O(r)}$ time suffices [6]). In this paper we consider two versions of these hierarchies— the SA hierarchy and the weaker LH hierarchy. Loosely speaking, the r^{th} round of the SA hierarchy adds the constraints of the r^{th} round of the Sherali-Adams linear programming hierarchy (see [25]) to the Basic SDP; the r^{th} round of the LH hierarchy augments the Basic SDP with the constraints that and subset of r vectors from the vector solutions embeds isometrically into the ℓ_1 metric. (See the full version [5] and [24] for more details.)

Barak, Raghavendra and Steurer [6] (see also [12]) showed that for every $\varepsilon > 0$, n^{ε} rounds of the SA hierarchy yields a non-trivial improvement over the basic SDP . The unique games conjecture predicts that this is optimal, in the sense that $n^{o(1)}$ rounds of any hierarchy should not improve the worst-case approximation ratio above the basic SDP.[4] However, this prediction is far from being verified, with the best lower bounds given by [24] (see also [15]) who showed instances that require $\log^{\Omega(1)} n$ rounds for the LH hierarchy, and $(\log\log n)^{\Omega(1)}$ rounds for the SA hierarchy. Moreover, these instances are *known* to be solvable in quasipolynomial time [17] and in fact via $\operatorname{polylog}(n)$ rounds of the SA hierarchy [6] . Thus prior work gave no evidence that the unique games problem cannot be solved in quasipolynomial time. In this work we obtain almost-exponentially more efficient integrality gaps, resisting $\operatorname{qpoly}(\log n)$ rounds of the SA hierarchy and $\operatorname{qqpoly}(n)$ rounds of the LH hierarchy. The latter is the first superlogarithmic SDP hierarchy lower bound for Unique Games for any SDP hierarchy considered in the literature.

Theorem 3. *For every $\varepsilon > 0$ there is some $k = k(\varepsilon)$, such that for every n there is an n variable instance Γ of* Unique Games *with alphabet size k such that the objective value of Γ is at most ε, but the value on Γ of both $\operatorname{qpoly}(\log n)$ rounds of the* SA *hierarchy and $\operatorname{qqpoly}(n)$ rounds of the* LH *hierarchy is at least $1 - \varepsilon$.*

A corollary of the above theorem is a construction of an n-point metric of negative type such that all sets of size up to some $k = \operatorname{qqpoly}(n)$ embed isometrically into ℓ_1 but the whole metric requires $\operatorname{qpolylog}(n)$ distortion to embed into ℓ_1. We remark that Theorem 3 actually yields a stronger result than stated here— as a function of k, our results (as was the case with the previous ones) obtain close to optimal gap between the objective value and the SDP value of these hierarchies; in particular we show that in the above number of rounds one cannot improve on the approximation factor of the Geomans-Williamson algorithm for Max Cut. It is a fascinating open question whether these results can be extended to the stronger *Lasserre / Sum of Squares* hierarchies [19], [26], [23], [22]. Very recent results of Barak, Brandão, Harrow, Kelner, Steurer and Zhou [4] (obtained subsequent to this work), indicate that new ideas may be needed to do this, since the Unique Games instances constructed here and in prior works are not integrality gaps for eight rounds of the Lasserre / Sum of Squares hierarchy. The underlying reason is that the proof that the original noisy cube is a small set expander is based on a sum of squares argument that can be verified efficiently via these hierarchies, and this extends also to the proof here that the derandomized noisy cube is a small set expander.

Alphabet reduction gadget.: Khot, Kindler, Mossel and O'Donnel [14] used the long code to show an "alphabet reduction" gadget for unique games. They show how to reduce a unique game instance with some large alphabet K to an instance with arbitrarily small

[2]Throughout, for any function f, $\operatorname{poly}(f(n))$ denotes a function g satisfying $g(n) = f(n)^{\Omega(1)}$.

[3]For functions $f, g : \mathbb{N} \to [0,\infty)$ we write $f = \operatorname{qpoly}(g)$ if $f = \exp(\operatorname{polylog}(g))$. That is, if there are constants $C > c > 0$ such that for all sufficiently large n, $\exp((\log g(n))^c) \leqslant f(n) \leqslant \exp((\log g(n))^C)$. (Note that we allow $c < 1$, and so $f = \operatorname{qpoly}(g)$ does not imply that $f > g$.) Similarly, we define $\operatorname{qpolylog}(g) = \operatorname{qpoly}(\log g)$ and write $f = \operatorname{qqpoly}(g)$ if $f = \exp(\exp(\operatorname{poly}(\log\log g)))$.

[4]This is under the widely believed assumption that **NP** $\not\subseteq$ **Dtime**$(\exp(n^{o(1)}))$.

alphabet. (In particular, they showed how one can reduce arbitrary unique games instances into binary alphabet instances, which turns out to be equivalent to the *Max Cut* problem.) However, quantitatively their result was rather inefficient, incurring an exponential in K blowup of the instance. By replacing the long code with our "short code", we obtain a more efficient gadget, incurring only a *qusipolynomial* blowup. One caveat is that, because the short code doesn't support arbitrary permutations, this reduction only works for unique games instances whose constraints are affine functions over $\mathbb{F}_2^k$ where $k = \log K$; however this class of unique games seems sufficiently rich for many applications.[5]

Theorem 4. *For every ε there are k, δ, and a reduction that for every ℓ maps any n-variable* Unique Games *instance Γ whose constraints are affine permutations over alphabet $\mathbb{F}_2^\ell$ into an $n \cdot \exp(\mathrm{poly}(\ell, k))$-variable* Unique Games *instance Γ' of alphabet k, such that if the objective value of Γ is larger than $1 - \delta$, then the objective value of Γ' is larger than $1 - \varepsilon$, and if the objective value of Γ is smaller than δ, then the objective value of Γ' is smaller than ε.*

Once again, our quantitative results are stronger than stated here, and as in [14], we obtain nearly optimal relation between the alphabet size k and the soundness and completeness thresholds. In particular for $k = 2$ our results match the parameters of the Max Cut algorithm of Geomans and Williamson. Our alphabet reduction gadget suggests a new approach to proving the unique games conjecture by using it as an "inner PCP". For example, one could first show hardness of unique games with very large alphabet (polynomial or even subexponential in the number of variables) and then applying alphabet reduction. At the very least, coming up with plausible hard instances for unique games should be easier with a large alphabet.

Remark I.1. The long code is also used as a tool in applications that do not involve the unique games conjecture. On a high level, there are two properties that make the long code useful in hardness of approximation: **(i)** it has a 2 query test obtained from the noisy hypercube and **(ii)** it has many symmetries, and in particular one can read off any function of x from the x^{th} codeword. Our short code preserves property **(i)** but (as is necessary for a more efficient code) does not preserve property **(ii)**, as one can only read off low degree polynomials of x (also it is only symmetric under affine transformations). We note that if one does not care about property **(i)** and is happy with a 3 query test, then it's often possible to use the Hadamard code which is more efficient than the short code (indeed it's essentially equal to the d-short code for $d = 1$). Thus, at least in the context of hardness of approximation, it seems that the applications the short code will be most useful are those where property **(i)** is the crucial one.

Despite the name "short code", our code is not the shortest possible code. While in our applications, dimension linear in the number of codewords is necessary (e.g., one can't have a graph with more eigenvalues than vertices), it's not clear that the dimension needs to be polynomial. It is a very interesting open question to find shorter codes that can still be used in the above applications.

II. Our techniques

To explain our techniques we focus on our first application— the construction of a small set expander with many eigenvalues close to 1. The best way to view this construction is as a derandomization of the noisy hypercube, and so it will be useful to recall why the noisy hypercube itself is a small set expander.

Recall that the ε-noisy hypercube is the graph $H_{N,\varepsilon}$ whose vertex set is $\{\pm 1\}^N$ where we sample a neighbor of x by flipping each bit independently with probability ε. The eigenvectors in $H_{N,\varepsilon}$ are given by the parity functions $\chi_\alpha(x) = \prod_{i \in \alpha} x_i$ for subsets $\alpha \subseteq [N]$ and the corresponding eigenvalues are $\lambda_\alpha = (1 - 2\varepsilon)^{|\alpha|}$. Thus λ_α only depends on the degree $|\alpha|$ of χ_α. In particular, the "dictator" functions $\chi_{\{i\}}(x) = x_i$ have eigenvalue $1 - 2\varepsilon$ and they correspond to balanced cuts (where vertices are partitioned based on the value of x_i) with edge expansion ε. As α increases, λ_α decreases, becoming a constant around $|\alpha| = O(1/\varepsilon)$.

Given $f : \{\pm 1\}^N \to \{0, 1\}$ which is the indicator of a set S, its Fourier expansion $f(x) = \sum_\alpha \hat{f}(\alpha)\chi_\alpha(x)$ can be viewed as expressing the vector f in the eigenvector basis. The edge expansion of S is determined by the distribution of its Fourier mass; sets where most of the Fourier mass is on large sets will expand well. Given this connection, small-set expansion follows from the fact that the indicator functions of small sets have most of their mass concentrated on large Fourier coefficients. More precisely a set S of measure μ has most of its Fourier mass on coefficients of degree $\Omega(\log(1/\mu))$. This follows from the so-called (2,4)-hypercontractive inequality for low-degree polynomials— that for every degree d polynomial f,

$$\mathop{\mathbb{E}}_{x \in \{\pm 1\}^N}[f(x)^4] \leqslant C \mathop{\mathbb{E}}_{x \in \{\pm 1\}^N}[f(x)^2]^2 \qquad \text{(II.1)}$$

for some C depending only on d. (See full version [5] for the proof, though some intuition can be obtained by noting that if f is a characteristic function of a set S of measure $\mu = o(1)$ then $\mathbb{E}[f^2]^2 = \mu^2$ and $\mathbb{E}[f^4] = \mu$

[5] For example, because the multiplicative group of the field $\mathbb{F}_{2^n}$ is cyclic, one can represent constraints of the form $x_i - x_j = c_{i,j} \pmod{2^n - 1}$ as linear constraints over $\mathbb{F}_2^n$ (i.e., constraints of the form $x_i = C_{i,j}x_j$ where $C_{i,j}$ is an invertible linear map over $\mathbb{F}_2^n$).

and hence Equation (II.1) shows that f cannot be an $O(1)$-degree polynomial.)

By a "derandomized hypercube" we mean a graph on much fewer vertices that still (approximately) preserves the above properties of the noisy hypercube. Specifically we want to find a very small subset $\mathcal{D}$ of $\{\pm 1\}^N$ and a subgraph G of $H_{N,\varepsilon}$ whose vertex set is $\mathcal{D}$ such that **(i)** G will have similar eigenvalue profile to $H_{N,\varepsilon}$, and in particular have N eigenvalues close to 1 and **(ii)** G will be a a small set expander. To get the parameters we are looking for, we'll need to have the size of $\mathcal{D}$ be at most qpoly(N).

A natural candidate is to take $\mathcal{D}$ to be a random set, but it is not hard to show that this will not work. A better candidate might be a linear subspace $\mathcal{D} \subseteq \mathbb{F}_2^N$ that looks suitably pseudorandom. We show that in fact it suffices to choose a subspace $\mathcal{D}$ whose dual $C = \mathcal{D}^\perp$ is a sufficiently good locally testable code. (We identify $\mathbb{F}_2^N$ with $\{\pm 1\}^N$ via the usual map $(b_1, \ldots, b_N) \mapsto ((-1)^{b_1}, \ldots, (-1)^{b_N})$.)

Our construction requires an asymptotic family of $[N, K, D]_2$ linear codes $C \subseteq \mathbb{F}_2^N$ where the distance D tends to infinity. The code should have a εN-query local tester which when given a received word $\alpha \in \mathbb{F}_2^N$ samples a codeword q of weight at most εN from a distribution $\mathcal{T}$ on $C^\perp$ and accepts if $\langle \alpha, q \rangle = 1$. The test clearly accepts codewords in C, we also require it to reject words that are distance at least $D/10$ from every codeword in C with probability 0.49. Given such a locally testable code C, we consider the Cayley graph[6] G whose vertices are the codewords of the dual code $\mathcal{D} = C^\perp$ while the (appropriately weighted) edges correspond to the distribution $\mathcal{T}$. That is, a vertex of G is a codeword $x \in \mathcal{D}$, while a random neighbor of x is obtained by picking a random q from $\mathcal{T}$ and moving to $x + q$.

Because $\mathcal{D}$ is a subspace, it is easy to show that the eigenvectors of G are linear functions of of the form $\chi_\alpha(x)$ for $x, \alpha \in \mathbb{F}_2^N$ (where if $\alpha \oplus \alpha' \in C$ then χ_α and $\chi_{\alpha'}$ are identical on G's vertices). Moreover, from the way we designed the graph, for every $\alpha \in \mathbb{F}_2^n$, the corresponding eigenvalue λ_α is equal to $\mathbb{E}_{q \in \mathcal{T}}[(-1)^{\langle \alpha, q \rangle}] = 1 - 2\,\mathbb{P}_{\mathcal{T}}[\text{Test rejects } \alpha]$. This connection between the spectrum of G and the local testability of C allows us to invoke machinery from coding theory in our analysis.

From this one can deduce that the eigenvalue spectrum of G does indeed resemble the hypercube in the range close to 1. In particular each $\chi_{\{i\}}(x) = x_i$ is a distinct eigenvector with eigenvalue $1 - 2\varepsilon$, and gives a bad cut in G (where vertices are partitioned based on the value of x_i). On the other hand for any eigenvector χ of G, choose α of minimal weight such that $\chi = \chi_\alpha$. Now if $|\alpha| > D/10$ this means that the distance of α from C is at least $D/10$, which using the testing property implies that $\lambda_\alpha \leqslant 1 - 2 \cdot 0.49 = 0.02$.

If we can show that indicator functions of small sets have most of their Fourier mass on such eigenvectors (with small eigenvalue), that will imply that small sets have good expansion. For small subsets of the hypercube, recall that this is proved using (2,4)-hpercontractivity for low-degree polynomials. The key observation is that the inequality

$$\mathop{\mathbb{E}}_{x \in \mathcal{D}}[f(x)^4] \leqslant C \mathop{\mathbb{E}}_{x \in \mathcal{D}}[f(x)^2]^2 \tag{II.2}$$

still holds for all polynomials f of degree $d < D/4$. This is because the distance of C is D, hence the distribution of a random x in $\mathcal{D}$ is D-wise independent, which means that the expectation of any polynomial of degree at most D is equal over such x and over a uniform x in $\{\pm 1\}^N$. Thus (II.2) follows from (II.1), completing our proof.

We instantiate this approach with using for C the Reed Muller code consisting of polynomials in n variables over $\mathbb{F}_2$ of degree $n - d - 1$. This is a code of distance $D = 2^{d-1}$. We note that the degree $n - d - 1$ and hence the rate of the code C are very high. The graph is over the codewords of $\mathcal{D} = C^\perp$ that is itself the Reed Muller code of polynomials over $\mathbb{F}_2^n$ of degree d. Our basic tester consists of selecting a random minimum weight codeword of $\mathcal{D}$.[7] Thus our graph $\mathcal{G}$ has as its vertices the d degree polynomials over $\mathbb{F}_2^n$ with an edge between every polynomials p, q such that $p - q$ is a product of d linearly-independent affine functions (as those are the minimal weight codewords in the Reed Muller code). We use the optimal analysis of Bhattacharyya et al. [7] to argue about the local testability of C which is a high degree Reed Muller code. We should note that this test is very closely related to the Gowers uniformity test that was first analyzed in the work of Kaufman et al. [1], but our application requires the stronger result from [7].

A. Other applications

We now briefly outline how we use the above tools to obtain more efficient versions of several other constructions such as alphabet reduction gadgets and integrality gaps for unique games and other problems.

Efficient integrality gaps for Unique Games.: To beign with, the graph we construct can be used to prove Theorem 2. That is, a construction of an M variable instance Γ of unique games where every assignment can satisfy at most a very small (say $1/100$) fraction of the constraints, but for which the standard semidefinite programming (SDP) relaxation has value of at least $1 - 1/\operatorname{qpoly}(\log M)$. The basic idea is to simply take the graph $\mathcal{G}$ we constructed above, and turn it into an

[6] Cayley graph are usually defined to be unweighted graph. However, the definition can be generalized straightforwardly to weighted graphs.

[7] For many applications we amplify the success of this tester by selecting a sum of t random such words, this corresponds to taking some power of the basic graph $\mathcal{G}$ described.

instance of unique games by considering it to be the *label extended graph* of some unique games instance. We now elaborate a bit below, leaving the full details to Section V. Recall that a Unique Games instance Γ with M variables and alphabet Σ is described by a collection of constraints of the form (x, y, π) where π is a permutation over Σ. An *assignment* to Γ is a mapping f from $[M]$ to Σ, and f's value is the fraction of constraints (x, y, π) such that $f(y) = \pi(f(x))$. The *label extended graph* corresponding to Γ is the graph G_Γ over vertices $[M] \times \Sigma$ where for every constraint of the form (x, y, π) and $\sigma \in \Sigma$ we add an edge between (x, σ) and $(y, \pi(\sigma))$. It is not hard to see that an assignment of value $1 - \varepsilon$ corresponds to a subset S containing exactly M of G_Γ's vertices with small expansion (i.e., ε fraction of the edges from S leave the set). Thus if G_Γ is an expander for sets of measure $1/|\Sigma|$ in G_Γ then there is no nearly satisfying assignment for the unique games instance Γ. In our case, our graph $\mathcal{G}$ has the degree d polynomials over $\mathbb{F}_2^n$ as its vertices, and we transform it into a unique game instance whose variables correspond to degree d polynomials *without linear terms*. The alphabet Σ consists of all linear functions over $\mathbb{F}_2^n$. We ensure that the graph $\mathcal{G}$ is the label extended graph of Γ by setting the permutations accordingly: given a polynomial p without a linear term, and a function q that is a product of d affine functions,[8] if we write $q = q' + q''$ where q'' is the linear part of q, then we add a constraint of the form $(p, p + q', \pi)$ where π is the permutation that maps a linear function r into $r + q''$. Some not too difficult calculations show that the top eigenvectors of our graph $\mathcal{G}$ yield a solution for the semidefinite program for Γ (if the top eigenvectors are $f^1, \ldots, f^K$, our vector solution will associate with each vertex x the vector $(f^1(x), \ldots, f^K(x))$. By choosing carefully the parameters of the graph $\mathcal{G}$, the instance Γ will have SDP value $1 - 1/\operatorname{qpoly}(\log M)$ where M is the number of variables.

Derandomized Invariance Principle.: While hypercontractivity of low degree polynomials suffices for some applications of the long code, other applications require other theorems, and in particular the *invariance principle*, shown for the hypercube by Mossel, O'Donnel and Oleszkiewicz [21]. Roughly speaking their invariance principle says that for "nice" functions f on the vertices of the N-dimensional noisy hypercube, the distribution of $f(x)$ where x is a random vertex is close to the distribution of $f(y)$ where y consists of N independent standard Gaussian random variables (appropriately extending f to act on $\mathbb{R}^N$). To obtain more efficient version of these applications, we first show that the same holds even when x is a random vertex in our smaller subset of N-dimensional strings – the Reed–Muller codewords. Our central tool is a recent result by Meka and Zuckerman [20] which derandomizes the invariance principle of Mossel et al. Our key insight is that taking a random Reed–Muller codeword can in fact be viewed as an instantiation of the Meka-Zuckerman generator, which involves splitting the input into blocks via a pairwise independent hash function, and using independent k-wise independent distributions in each block. This allows us to obtain a version of the "Majority is Stablest" theorem for our graph, which is the main corollary of the invariance principle that is used in applications of the longcode. See full version [5] for more details.

Efficient alphabet reduction .: With the "Majority of Stablest" theorem in hand, proving Theorem 4 (efficient alphabet reduction for unique games), is fairly straightforward. The idea is to simply replace the noisy hypercube gadget used by [14] with our derandomized hypercube. This is essentially immediate in the case of alphabet reduction to binary alphabet (i.e., reduction to Max Cut) but requires a bit more work when reducing to a larger alphabet. See full version [5] for more details.

Efficient hierarchy integrality gaps.: Our proof Theorem 3 again works by plugging in our short code / derandomized noisy hypercube in place of the long code in the previous integrality gap constructions [16], [15], [24]. Specifically, these constructions worked by starting with an integrality gap for unique games where the basic SDP yields $1 - 1/r$, and then composing it with an alphabet reduction gadget to obtain a new instance; Raghavendra and Steurer [24] showed that the composed instances resist $\operatorname{poly}(r)$ rounds of the SA hierarchy and $\exp(\operatorname{poly}(r))$ rounds of the LH hierarchy. These constructions used the noisy cube twice— both to obtain the basic unique games gap instance, and to obtain the alphabet reduction gadget. We simply plug in our short code in both usages— using for the basic unique games instance the efficient version obtained in Theorem 2, and for the alphabet reduction gadget the efficient version obtained in Theorem 4. (Luckily, our unique games instance has affine constraints and so is compatible with our alphabet reduction gadget.) The result essentially follows in a blackbox way from the analysis of [24]. See full version [5] for details.

III. Preliminaries

Let G be a regular graph with vertex set V. For a subset $S \subseteq V$ we define the *volume* of S, denoted $\mu(S)$, to be $|S|/|V|$. We define the *expansion* of S, denoted $\Phi(S)$, to be the probability over a random edge (u, v), conditioned on $u \in S$ that $v \notin S$. Equivalently (since G is regular), $\Phi(S) = G(S, V \setminus S)/(\deg_G |S|)$ where $\deg_G$ is the degree of the graph G and $G(S, V \setminus S)$ is the

[8] Actually, to get better parameters, we take some power t of $\mathcal{G}$, meaning that we consider q that is a sum of t functions that are products of d affine functions.

number of edges going from S to $V \setminus S$. Throughout, we denote the normalized adjacency matrix of a graph G also by G, and refer to the spectrum of the adjacency matrix as the spectrum of the graph G. Note that by definition, every regular graph has maximum eigenvalue 1. In this paper, we use *expectation norms* for real-valued functions. That is, for a function $f: S \to \mathbb{R}$ and $p \geqslant 1$, we let $\|f\|_p := (\mathbb{E}_{x \in S} |f(x)|^p)^{1/p}$.

Many of the unique games instances that appear in this work belong to a special subclass of unique games, namely $\mathbb{F}_2^n$-Max-2Lin instances defined below.

Definition III.1. Given a group $\mathcal{H}$, an $\mathcal{H}$-Max-2Lin instance consists of a system of linear equations over the group $\mathcal{H}$ where each equation is of the form $x_i - x_j = c_{ij}$ for some $c_{ij} \in \mathcal{H}$.

Locally Testable Codes.: Let C be an $[N, K, D]_2$ code, that is, C is a K-dimensional linear subspace of $\mathbb{F}_2^N$ with minimum distance D $(= \min\{\mathsf{wt}(x) : x \in C\})$. (In this paper, we are mostly interested in the extremely high rate regime when $H = N - K$ is very small compared to N and are happy with D being some large constant.) Let $\Delta(x, y) \in \{0, \ldots, N\}$ denote Hamming distance between $x, y \in \mathbb{F}_2^N$. For $\alpha \in \mathbb{F}_2^N$ and a code C we define $\Delta(\alpha, C) = \min_{c \in C} \Delta(\alpha, c)$.

Definition III.2. We say a distribution $\mathcal{T}$ over $\mathbb{F}_2^N$ is a *canonical tester* for C if every vector in the support of the distribution $\mathcal{T}$ is a codeword $q \in C^\perp$. The *query complexity* of $\mathcal{T}$ is the maximum weight of a vector in its support. The tester's *soundness curve* $s_{\mathcal{T}}: \mathbb{N} \to [0, 1]$ is defined as

$$s_{\mathcal{T}}(k) \stackrel{\text{def}}{=} \min_{\alpha \in \mathbb{F}_2^N,\, \Delta(\alpha, C) \geqslant k} \mathbb{P}_{q \sim \mathcal{T}} \{\langle \alpha, q \rangle = 1\} .$$

Similarly, we denote the *rejection probability* of $\mathcal{T}$ for a vector $\alpha \in \mathbb{F}_2^N$ by $s_{\mathcal{T}}(\alpha) = \mathbb{P}_{q \sim \mathcal{T}} \{\langle \alpha, q \rangle = 1\}$. We let the *query probability* $\tau \in [0, 1]$ of a tester be the expected fraction of queried coordinates, that is, $\tau = \mathbb{E}_{q \sim \mathcal{T}} \mathsf{wt}(q)/N$. We say that a tester $\mathcal{T}$ with query probability τ is *smooth* if for any coordinate $i \in [N]$, $\mathbb{P}_{q \sim \mathcal{T}} \{q_i = 1\} = \tau$ and we say it is 2-*smooth* if in addition, for any two distinct coordinates $i \neq j$, $\mathbb{P}_{q \sim \mathcal{T}} \left\{q_i = q_j = 1\right\} = \tau^2$.

If the tester $\mathcal{T}$ is clear from the context, we will sometimes drop the subscript of the soundness curve / rejection probability $s_{\mathcal{T}}$. In the setting of this paper, we will consider testers with query probability slowly going to 0 (with N). Further, given a canonical tester $\mathcal{T}$, it is easy to amplify the probability of rejection by repeating the test and taking the XOR of the results.

Finally, the following simple lemma gives some estimates for rejection probabilities of vectors for smooth testers. The proof can be found in full version [5]

Lemma III.3. *If $\mathcal{T}$ is a smooth canonical tester with query probability τ, then $s_{\mathcal{T}}(\alpha) \leqslant \Delta(\alpha, C) \cdot \tau$ for every vector $\alpha \in \mathbb{F}_2^N$. Furthermore, if $\mathcal{T}$ is 2-smooth, then $s_{\mathcal{T}}(\alpha) \geqslant (1 - \gamma) \cdot \Delta(\alpha, C) \cdot \tau$ for every vector $\alpha \in \mathbb{F}_2^N$ with $\Delta(\alpha, C)\tau \leqslant \gamma$.*

We review the prerequisites for Majority is Stablest and Unique Games related results in the corresponding sections.

IV. Small Set Expanders from Locally Testable Codes

In this section we first use some known properties of hypercontractive norms to give a sufficient condition for graphs to be small set expanders. We then describe a generic way to construct graphs satisfying this condition from locally testable codes, proving Theorem 1.

Let $\mathcal{V}$ be a subspace of the set of functions from V to $\mathbb{R}$ for some finite set V. We denote by $P_{\mathcal{V}}$ the projection operator to the space $\mathcal{V}$. For $p, q \geqslant 1$, we define

$$\|\mathcal{V}\|_{p \to q} \stackrel{\text{def}}{=} \max_{f: V \to \mathbb{R}} \frac{\|P_{\mathcal{V}} f\|_q}{\|f\|_p} .$$

The next lemma can be viewed qualitatively as a generalization of one direction of the classical Cheeger's inequality relating combinatorial expansion to eigenvalue gap [9]. We defer the proof to full version [5].

Lemma IV.1. *Let $G = (V, E)$ be regular graph, and $\mathcal{V}$ be the span of the eigenvectors of G with eigenvalue larger than λ. Then, for every $S \subseteq V$, $\Phi(S) \geqslant 1 - \lambda - \|\mathcal{V}\|_{2 \to 4}^2 \sqrt{\mu(S)}$.*

A. Cayley graphs on codes

Motivated by the above lemma, we now construct a graph for which the projection operator on to the top eigenspace is hypercontractive, i.e., has small $2 \to 4$ norm, while also having high rank.

Let $C \subseteq \mathbb{F}_2^N$ be an $[N, K, D]_2$ code. The graph we construct will be a Cayley graph with vertices indexed by $C^\perp$ and edges drawn according to a canonical local tester $\mathcal{T}$ for C. Let $\mathrm{Cay}(C^\perp, \mathcal{T})$ denote the (weighted) Cayley graph with vertex set $C^\perp$ and edges generated by $\mathcal{T}$. We describe the graph more precisely by specifying the neighbor distribution for a random walk on the graph. For a vertex $p \in C^\perp$, a random neighbor has the form $p + q$ with q sampled from the tester $\mathcal{T}$. (Since the group $C^\perp$ has characteristic 2, the graph $\mathrm{Cay}(C^\perp, \mathcal{T})$ is symmetric for every tester $\mathcal{T}$.)

We argue that if the tester $\mathcal{T}$ has small query complexity and good soundness, then the graph $\mathrm{Cay}(C^\perp, \mathcal{T})$ has many large eigenvalues while being a small-set expander.

Theorem IV.2. *Let C be an $[N, K, D]_2$ linear code that has a canonical tester $\mathcal{T}$ with query complexity εN and soundness curve $s()$ and let $k < D/5$. The graph $\mathrm{Cay}(C^\perp, \mathcal{T})$ has $2^{N-K} = 2^H$ vertices with at least $N/2$*

eigenvalues larger than $1-4\varepsilon$. *All subsets* S *of* $C^\perp$ *have expansion at least* $\Phi(S) \geqslant 2s(k) - 3^k\sqrt{\mu(S)}$.

By Xoring the results of mulitple tests, one can let the soundness $s(k)$ tend to $1/2$. Hence, if $s(k)$ is significantly larger than ε (for appropriate k), one can obtain a graph with many large eigenvalues such that small enough sets have near-perfect expansion.

Eigenfunctions and Eigenvalues.: We identify the graph $G = \mathrm{Cay}(C^\perp, \mathcal{T})$ by its normalized adjacency matrix. For every vector $\alpha \in \mathbb{F}_2^N$, the character $\chi_\alpha : C^\perp \to \{\pm 1\}$ with $\chi_\alpha(p) = (-1)^{\langle \alpha, p\rangle}$ is an eigenfunction of G. If two vectors $\alpha, \beta \in \mathbb{F}_2^N$ belong to the same coset of C, they define the same character over $C^\perp$ since $\langle \alpha + \beta, p\rangle = 0$ for all $p \in C^\perp$, while if $\alpha + \beta \notin C$ then $\langle \chi_\alpha, \chi_\beta\rangle = 0$. Thus, the set of characters of $C^\perp$ corresponds canonically to the quotient space $\mathbb{F}_2^N/C$. If we fix a single representative α for every coset in $\mathbb{F}_2^N/C$, we have exactly $2^{N-K} = 2^H$ distinct, mutually orthogonal characters. We define the degree of a character as follows:

$$\deg(\chi_\alpha) = \min_{c \in C} \mathsf{wt}(\alpha + c) = \Delta(\alpha, C)\,. \tag{IV.1}$$

Note that if $\deg(\chi_\alpha) < D/2$, then the minimum weight representative in $\alpha + C$ is unique. (This uniqueness will allow us later to define low-degree influences of functions, see full version [5].)

We let λ_α denote the eigenvalue corresponding to character χ_α. The following observation connects the soundness of the canonical tester to the spectrum of G:

Lemma IV.3. *For any* $\alpha \in \mathbb{F}_2^N$, $\lambda_\alpha = 1 - 2s(\alpha)$.

Proof: From standard facts about Cayley graphs, it follows that $\lambda_\alpha = \mathbb{E}_{q\in\mathcal{T}}[\chi_\alpha(q)] = \mathbb{E}_{q\in\mathcal{T}}[(-1)^{\alpha\cdot q}] = 1 - 2\,\mathbb{P}_{q\in\mathcal{T}}[\alpha \cdot q = 1] = 1 - 2s(\alpha)$. ∎

We use this to show that many *dictator cuts* in G which correspond to characters with degree 1 have eigenvalues close to 1. We let λ_i, χ_i denote $\lambda_{\{i\}}, \chi_{\{i\}}$. As noted before, for $D > 2$ these are distinct characters.

Corollary IV.4. *We have* $\lambda_i \geqslant 1 - 4\varepsilon$ *for at least* $N/2$ *coordinates* $[i] \in N$.

Proof: We have $\lambda_i = 1 - 2\,\mathbb{P}_{q\in\mathcal{T}}[q_i = 1]$. Since $\mathsf{wt}(q) \leqslant \varepsilon N$ for every $q \in \mathcal{T}$, $\sum_{i=1}^N \mathbb{P}_{q\in\mathcal{T}}[q_i = 1] \leqslant \varepsilon N$. So we can have $\mathbb{P}_{q\in\mathcal{T}}[q_i = 1] \geqslant 2\varepsilon$ for at most $N/2$ coordinates. ∎

Another immediate conseuqence of Lemma IV.3 is that large degree characters have small eigenvalues.

Corollary IV.5. *If* $\deg(\chi_\alpha) \geqslant k$, *then* $\lambda_\alpha \leqslant 1 - 2s(k)$.

Subspace Hypercontractivity.: Given a function $f : C^\perp \to \mathbb{R}$ we can write it (uniquely) as a linear combination of the characters $\{\chi_\alpha\}_{\alpha\in\mathbb{F}_2^N/C}$

$$f(p) = \sum_{\alpha\in\mathbb{F}_2^N/C} \hat{f}(\alpha)\chi_\alpha(p)\,,$$

where $\hat{f}(\alpha) = \langle \chi_\alpha, f\rangle$ is the *Fourier transform* of f (over the abelian group $C^\perp$).

We define the *degree* of f, denoted $\deg(f)$ to be $\max_{\alpha:\hat{f}(\alpha)\neq 0} \deg(\chi_\alpha)$. Note that $\deg(f+g) \leqslant \max\{\deg(f), \deg(g)\}$ and $\deg(fg) \leqslant \deg(f) + \deg(g)$. The following crucial observation follows immediately from the fact that C has minimum distance D.

Fact IV.6. *The uniform distribution on* $C^\perp$ *is* $(D-1)$ *wise independent. That is, for any* $\alpha \in \mathbb{F}_2^N$ *such that* $1 \leqslant \mathsf{wt}(\alpha) < D$ *we have* $\mathbb{E}_{p\in C^\perp}[\chi_\alpha(p)] = 0$.

This fact has the following corollary:

Lemma IV.7. *Let* $\ell < (D-1)/4$ *and let* $\mathcal{V}$ *be the subspace of functions with degree at most* ℓ. *Then* $\|\mathcal{V}\|_{2\to 4} \leqslant 3^{\ell/2}$.

Proof: The proof follows from the following two facts:

1) This bound on the $2 \to 4$ norm is known to hold for true low degree polynomials under the uniform distribution on the hypercube by the Bonami-Beckner-Gross inequality.
2) The expectation of polynomials of degree up to $4\ell < D-1$ are the same under the uniform distribution and a $D-1$-wise independent distribution.

Given $f : \mathbb{R}^n \to \mathbb{R}$, let f^ℓ denote its projection onto the space $\mathcal{V}$ spanned by characters where $\deg(\chi_\alpha) \leqslant \ell$. We have

$$\|f^\ell\|_4^4 = \mathop{\mathbb{E}}_{p\in C^\perp}[f^\ell(p)^4] = \mathop{\mathbb{E}}_{p\in\{0,1\}^N}[f^\ell(p)^4]\,,$$

$$\|f\|_2^2 \geqslant \|f^\ell\|_2^2 = \mathop{\mathbb{E}}_{p\in C^\perp}[f^\ell(p)^2] = \mathop{\mathbb{E}}_{p\in\{0,1\}^N}[f^\ell(p)^2]\,.$$

By the $2 \to 4$ hypercontractivity for degree ℓ polynomials over $\{0,1\}^N$,

$$\mathop{\mathbb{E}}_{p\in\{0,1\}^N}[f^\ell(p)^4] \leqslant 9^\ell \mathop{\mathbb{E}}_{p\in\{0,1\}^N}[f^\ell(p)^2]^2\,.$$

So we conclude that

$$\mathop{\mathbb{E}}_{p\in C^\perp}[f^\ell(p)^4] \leqslant 9^\ell \mathop{\mathbb{E}}_{p\in C^\perp}[f^\ell(p)^2]^2 \leqslant 9^\ell \mathop{\mathbb{E}}_{p\in C^\perp}[f(p)^2]^2\,,$$

which implies that $\|\mathcal{V}\|_{2\to 4} \leqslant 3^{\ell/2}$. ∎

Combining the above bound with Lemma IV.1 we get that, if the local tester rejects sufficiently far codewords with high probability, then the resulting graph is a small set expander:

Corollary IV.8. *For every vertex subset* S *in the graph* $\mathrm{Cay}(C^\perp, \mathcal{T})$ *and every* $k < D/5$, *we have* $\Phi(S) \geqslant 2s(k) - 3^k\mu(S)^{\frac{1}{2}}$.

In particular, as $s(k)$ tends to $1/2$, the expansion of small sets tends to 1. This corollary together with Corollary IV.4 completes the proof of Theorem IV.2.

B. A Canonical Tester for Reed Muller codes

We instantiate the construction from the previous section for the Reed Muller code. Let $C = \mathsf{RM}(n, n-d-1)$ be the Reed Muller code on n variables of degree $n-d-1$, which has $N = 2^n$, $H = \sum_{j \leqslant d} \binom{n}{j}$ and $D = 2^{d+1}$. Bhattacharyya et al. [7] analyze the canonical tester $\mathcal{T}_{\mathsf{RM}}$ which samples a random minimum weight codeword from $C^\perp$. It is well known that the dual of $\mathsf{RM}(n, n-d-1)$ is exactly $\mathsf{RM}(n,d)$ and that the minimum weight codewords in $\mathsf{RM}(n,d)$ are products of d linearly independent affine forms. They have weight $2^{n-d} = \varepsilon N$ where $\varepsilon = 2^{-d}$. Thus, our graph $\mathrm{Cay}_{\mathsf{RM}} = \mathrm{Cay}(\mathsf{RM}_{n,d}, \mathcal{T}_{\mathsf{RM}})$ has as its vertices the d-degree polynomials over $\mathbb{F}_2^n$ with an edge between every pair of polynomials P, Q such that $P - Q$ is equal to a minimum weight codeword, which are known to be products of d linearly independent affine forms.

Theorem IV.9 ([7])**.** *There exists a constant $\eta_0 > 0$ such that for all n, d, and $k < \eta_0 2^d$ the tester $\mathcal{T}_{\mathsf{RM}}$ described above has soundness $s(k) \geqslant (k/2) \cdot 2^{-d}$.*

Theorem IV.9 allows us to estimate the eigenvalue profile of $\mathrm{Cay}_{\mathsf{RM}}$ and shows that small sets have expansion close to $O(\eta_0)$. From here, we can get near perfect expansion by taking short random walks. We defer the proof of the theorem to the full version [5].

Theorem IV.10. *There exists a constant $c_1 > 0$ such that for any $\varepsilon, \eta > 0$, there exists a graph G with $2^{(\log |G|)^{\frac{1}{d}}}$ eigenvalues larger than $1-\varepsilon$ for $d = \log(1/\varepsilon) + \log\log(1/\eta) + O(1)$ and where every set $S \subseteq G$ has expansion $\Phi(S) \geqslant 1 - \eta - 3^{\frac{c_1}{\varepsilon}\log(1/\eta)}\sqrt{\mu(S)}$.*

V. Efficient integrality gaps for unique games

In this section, we present constructions of SDP integrality gap instances starting from a code C along with a local tester. To this end, we make an additional assumption on the code C. Specifically, let us suppose there exists a subcode $\mathcal{H}$ of $\mathcal{D} = C^\perp$ with distance $\frac{1}{2}$. Formally, we show the following result.

Theorem V.1. *Let C be an $[N, K, D]_2$ linear code with a canonical tester $\mathcal{T}$ as described in Definition III.2. Furthermore, let $\mathcal{H}$ be a subcode of $\mathcal{D} = C^\perp$ with distance $\frac{1}{2}$. Then, there exists an instance of unique games, more specifically a $\mathcal{H}$-Max-2Lin instance, whose vertices are $\mathcal{D}$ ($|\mathcal{D}| = 2^{N-K}$) and alphabet $\mathcal{H}$ such that:*

- *The optimum value of the natural SDP relaxation for unique games is at least $\left(1 - \frac{2t}{N}\right)^2$ where t is the number of queries made by the canonical tester $\mathcal{T}$.*
- *No labelling satisfies more than $\min_{k \in [0, D/5]} \left(1 - 2s(k) + \frac{3^k}{|\mathcal{H}|^{\frac{1}{2}}}\right)$ fraction of constraints.*

Instantiating the above theorem with the Reed–Muller code and its canonical tester we obtain the following explicit SDP integrality gap instance.

Corollary V.2. *For every integer n, $\delta > 0$ there exists a $\mathbb{F}_2^n$-Max-2Lin instance Γ on $M = 2^{2^{\log^2 n}}$ vertices such that the optimum value of the SDP relaxation on Γ is $1 - O(\frac{\log(1/\delta)}{n}) = 1 - O\left(\frac{\log(1/\delta)}{2^{(\log\log M)^{1/2}}}\right)$ while every labelling of Γ satisfies at most $O(\delta)$ fraction of edges.*

Starting from a code C, we construct an SDP integrality gap instance $\Gamma(C, \mathcal{T})$ for unique games as follows.

> The vertices of Γ_C are the codewords of $\mathcal{D}$. The alphabet of the unique games instance $\Gamma(C, \mathcal{T})$ are the codewords in $\mathcal{H}$. The constraints of unique games instance $\Gamma(C, \mathcal{T})$ are given by the tests of the following verifier.
> The input to the verifier is a labeling $\ell : \mathcal{D} \to \mathcal{H}$. Let us denote by $R = |\mathcal{H}|$. The verifier proceeds as follows:
> - Sample codewords $c \in \mathcal{D}$ and $h, h' \in \mathcal{H}$ uniformly at random.
> - Sample a codeword $q \in \mathcal{D}$ from the tester $\mathcal{T}$.
> - Test if $\ell(c+q+h) - \ell(c+h') = h - h'$

SDP Solution.: Here we construct a feasible solution to a natural SDP relaxation of unique games [16].

$$\begin{aligned}
&\text{Maximize} \mathop{\mathbb{E}}_{c \in \mathcal{D}, h, h' \in \mathcal{H}} \mathop{\mathbb{E}}_{q \in \mathcal{T}} \left[\frac{1}{R} \sum_{\ell \in \mathcal{H}} \langle \boldsymbol{b}_{c+h', \ell+h'}, \boldsymbol{b}_{c+q+h, \ell+h} \rangle \right] \\
&\text{Subject to} \langle \boldsymbol{b}_{c,h}, \boldsymbol{b}_{c,h'} \rangle = 0 \qquad \forall c \in \mathcal{D}, h \neq h' \in \mathcal{H} \\
&\qquad \langle \boldsymbol{b}_{c,h}, \boldsymbol{b}_{c',h'} \rangle \geqslant 0 \qquad \forall c, c' \in \mathcal{D}, h, h' \in \mathcal{H}. \\
&\qquad \sum_{\ell \in \mathcal{H}} \langle \boldsymbol{b}_{c,\ell}, \boldsymbol{b}_{c,\ell} \rangle = R \qquad \forall c \in \mathcal{D}
\end{aligned}$$

For a vector $c \in \mathbb{F}_2^m$, we will use $(-1)^c \in \mathbb{R}^m$ to denote the vector whose coordinates are given by $(-1)^c_i = (-1)^{c_i}$. For a pair of vectors c, c', we have $\langle (-1)^c, (-1)^{c'} \rangle = 1 - 2\Delta(c, c')$. For each vertex $c \in \mathcal{D}$ associate vectors $\{\boldsymbol{b}_{c,h} = (-1)^{c+h} \otimes (-1)^{c+h} | h \in \mathcal{H}\}$. Notice that for a pair of vectors $\boldsymbol{b}_{c,h}, \boldsymbol{b}_{c',h'}$ we have,

$$\langle \boldsymbol{b}_{c,h}, \boldsymbol{b}_{c',h'} \rangle = \langle (-1)^{c+h}, (-1)^{c'+h'} \rangle^2 = (1 - 2\Delta(c+h, c'+h'))^2 .$$

Since the distance of the code $\mathcal{H}$ is $\frac{1}{2}$, we have

$$\langle \boldsymbol{b}_{c,h}, \boldsymbol{b}_{c,h'} \rangle = (1 - 2\Delta(h, h'))^2 = \begin{cases} 1 & \text{if } h = h' \\ 0 & \text{if } h \neq h' \end{cases}$$

In other words, for every vertex c, the corresponding SDP vectors are orthonormal. The objective value of the SDP solution is given by,

$$\mathsf{OBJ} = \mathop{\mathbb{E}}_{c \in \mathcal{D}, h, h' \in \mathcal{H}} \mathop{\mathbb{E}}_{q \in \mathcal{T}} \frac{1}{R} \sum_{\ell \in \mathcal{H}} \langle \boldsymbol{b}_{c+h', \ell+h'}, \boldsymbol{b}_{c+q+h, \ell+h} \rangle$$

$$= \frac{1}{R} \mathop{\mathbb{E}}_{\substack{c\in\mathcal{D},h\in\mathcal{H}\\ q\in\mathcal{T}}} \sum_{\ell\in\mathcal{H}} (1-2\Delta(c+h'+\ell+h', c+q+h+\ell+h))^2$$

$$= \mathop{\mathbb{E}}_{c\in\mathcal{D},h\in\mathcal{H}} \mathop{\mathbb{E}}_{q\in\mathcal{T}} \left[(1-2\Delta(0,q))^2\right] \geqslant \left(1-\frac{2t}{N}\right)^2$$

where t is the number of queries made by the canonical tester $\mathcal{T}$ for C.

Soundness.: Let $\ell : \mathcal{D} \to \mathcal{H}$ be an arbitrary labelling of the Unique Games instance $\Gamma(C,\mathcal{T})$. For each $p \in \mathcal{H}$, define a function $f_p \colon \mathcal{D} \to [0,1]$ as follows,

$$f_p(c) = \mathop{\mathbb{E}}_{h\in\mathcal{H}} \left[\mathbb{I}[\ell(c+h) = p+h]\right].$$

The fraction of constraints satisfied by the labelling ℓ is given by,

$$\mathop{\mathbb{E}}_{\substack{c\in\mathcal{D},h,h'\in\mathcal{H}\\ q\in\mathcal{T}}} \sum_{p\in\mathcal{H}} \mathbb{I}[\ell(c+h') = p+h'] \cdot \mathbb{I}[\ell(c+q+h) = p+h]$$

$$= \mathop{\mathbb{E}}_{\substack{c\in\mathcal{D}\\ q\in\mathcal{T}}} \sum_{p\in\mathcal{H}} \mathop{\mathbb{E}}_{h'\in\mathcal{H}} \mathbb{I}[\ell(c+h') = p+h'] \cdot \mathop{\mathbb{E}}_{h\in\mathcal{H}} \mathbb{I}[\ell(c+q+h) = p+h]$$

$$= \mathop{\mathbb{E}}_{c\in\mathcal{D}} \mathop{\mathbb{E}}_{q\in\mathcal{T}} \sum_{p\in\mathcal{H}} f_p(c) f_p(c+q) = \sum_{p\in\mathcal{H}} \langle f_p, G f_p\rangle \qquad \text{(V.1)}$$

where $G = \mathrm{Cay}(C^\perp, \mathcal{T})$ is the graph associated with the code $C^\perp$ and tester $\mathcal{T}$. The expectation of the function f_p is given by,

$$\mathop{\mathbb{E}}_{c\in\mathcal{D}} f_p(c) = \mathop{\mathbb{P}}_{c\in\mathcal{D},h\in\mathcal{H}} \left[\ell(c+h) = p+h\right]$$

$$= \mathop{\mathbb{P}}_{c\in\mathcal{D},h\in\mathcal{H}} \left[\ell(c) = p+h\right] = \frac{1}{|\mathcal{H}|} = \frac{1}{R}.$$

where we used the fact that $c+h, h) \sim (c,h)$. Since f_p is bounded in the range $[0,1]$ we have,

$$\langle f_p, f_p\rangle = \mathop{\mathbb{E}}_{c\in\mathcal{D}} [f_p(c)^2] \leqslant \mathop{\mathbb{E}}_{c\in\mathcal{D}} [f_p(c)] = \frac{1}{R}.$$

Applying Corollary IV.8, we get that for each p,

$$\langle f_p, G f_p\rangle \leqslant \frac{1}{R} \cdot \min_{k\in[0,\frac{D}{5}]} \left(1 - 2s(k) + \frac{3^k}{R^{1/2}}\right).$$

Substituting the previous equation in to (V.1), we get the desired conclusion.

References

[1] N. Alon, T. Kaufman, M. Krivelevich, S. Litsyn, and D. Ron, "Testing reed-muller codes," *IEEE Transactions on Information Theory*, vol. 51, no. 11, pp. 4032–4039, 2005.

[2] S. Arora, B. Barak, and D. Steurer, "Subexponential algorithms for unique games and related problems," in *FOCS*, 2010, pp. 563–572.

[3] S. Arora, S. Khot, A. Kolla, D. Steurer, M. Tulsiani, and N. K. Vishnoi, "Unique games on expanding constraint graphs are easy," in *STOC*, 2008, pp. 21–28.

[4] B. Barak, F. G. B. ao, A. Harrow, J. Kelner, D. Steurer, and Y. Zhou, "Hypercontractive inequalities, sums of squares proofs, and their applications," in *STOC*, 2011.

[5] B. Barak, P. Gopalan, J. Håstad, R. Meka, P. Raghavendra, and D. Steurer, "Making the long code shorter, with applications to the unique games conjecture," *CoRR*, vol. abs/1111.0405, 2011.

[6] B. Barak, P. Raghavendra, and D. Steurer, "Rounding semidefinite programming hierarchies via global correlation," in *FOCS*, 2011, pp. 472–481.

[7] A. Bhattacharyya, S. Kopparty, G. Schoenebeck, M. Sudan, and D. Zuckerman, "Optimal testing of reed-muller codes," in *FOCS*, 2010, pp. 488–497.

[8] M. Charikar, K. Makarychev, and Y. Makarychev, "Near-optimal algorithms for unique games," in *STOC*, 2006, pp. 205–214.

[9] J. Cheeger, "A lower bound for the smallest eigenvalue of the Laplacian," in *Problems in analysis (Papers dedicated to Salomon Bochner, 1969)*. Princeton, N. J.: Princeton Univ. Press, 1970, pp. 195–199.

[10] E. Chlamtac and M. Tulsiani, "Convex relaxations and integrality gaps," 2010, chapter in Handbook on Semidefinite, Cone and Polynomial Optimization.

[11] A. Gupta and K. Talwar, "Approximating unique games," in *SODA*, 2006, pp. 99–106.

[12] V. Guruswami and A. K. Sinop, "The complexity of finding independent sets in bounded degree (hyper)graphs of low chromatic number," in *SODA*, 2011, to appear.

[13] S. Khot, "On the power of unique 2-prover 1-round games," in *STOC*, 2002, pp. 767–775.

[14] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell, "Optimal inapproximability results for Max-Cut and other 2-variable CSPs?" in *FOCS*, 2004, pp. 146–154.

[15] S. Khot and R. Saket, "SDP integrality gaps with local ℓ_1-embeddability," in *FOCS*, 2009, pp. 565–574.

[16] S. Khot and N. K. Vishnoi, "The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1," in *FOCS*, 2005, pp. 53–62.

[17] A. Kolla, "Spectral algorithms for unique games," in *IEEE Conference on Computational Complexity*, 2010, pp. 122–130.

[18] A. Kolla and M. Tulsiani, "Playing random and expanding unique games," 2007, to appear in journal version of [3].

[19] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 796–817, 2001.

[20] R. Meka and D. Zuckerman, "Pseudorandom generators for polynomial threshold functions," in *STOC*, 2010, pp. 427–436.

[21] E. Mossel, R. O'Donnell, and K. Oleszkiewicz, "Noise stability of functions with low influences invariance and optimality," in *FOCS*, 2005, pp. 21–30.

[22] Y. Nesterov, "Squared functional systems and optimization problems," *High performance optimization*, vol. 13, pp. 405–440, 2000.

[23] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," MIT, Tech. Rep., 2000, ph.D thesis.

[24] P. Raghavendra and D. Steurer, "Integrality gaps for strong SDP relaxations of unique games," in *FOCS*, 2009, pp. 575–585.

[25] H. D. Sherali and W. P. Adams, "A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems," *SIAM J. Discrete Math.*, vol. 3, no. 3, pp. 411–430, 1990.

[26] N. Shor, "An approach to obtaining global extremums in polynomial mathematical programming problems," *Cybernetics and Systems Analysis*, vol. 23, no. 5, pp. 695–700, 1987.