Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 129 (2009)

Subsampling Mathematical Relaxations and Average-case Complexity

Boaz Barak* Moritz Hardt† Thomas Holenstein‡ David Steurer§

April 29, 2010

Abstract

We initiate a study of when the value of mathematical relaxations such as linear and semi-definite programs for constraint satisfaction problems (CSPs) is approximately preserved when restricting the instance to a sub-instance induced by a small random subsample of the variables.

Let C be a family of CSPs such as 3SAT, Max-Cut, etc.., and let Π be a mathematical program that is a *relaxation* for C, in the sense that for every instance $\mathcal{P} \in C$, $\Pi(\mathcal{P})$ is a number in $[0, 1]$ upper bounding the maximum fraction of satisfiable constraints of $\mathcal{P}$. Loosely speaking, we say that *subsampling holds* for C and Π if for every sufficiently dense instance $\mathcal{P} \in C$ and every $\varepsilon > 0$, if we let $\mathcal{P}'$ be the instance obtained by restricting $\mathcal{P}$ to a sufficiently large constant number of variables, then $\Pi(\mathcal{P}') \in (1 \pm \varepsilon)\Pi(\mathcal{P})$. We say that *weak subsampling* holds if the above guarantee is replaced with $\Pi(\mathcal{P}') = 1 - \Theta(\gamma)$ whenever $\Pi(\mathcal{P}) = 1 - \gamma$, where Θ hides only absolute constants. We obtain both positive and negative results, showing that:

1. Subsampling holds for the BasicLP and BasicSDP programs. BasicSDP is a variant of the semi-definite program considered by Raghavendra (2008), who showed it gives an optimal approximation factor for every constraint-satisfaction problem under the unique games conjecture. BasicLP is the linear programming analog of BasicSDP.
2. For tighter versions of BasicSDP obtained by adding additional constraints from the Lasserre hierarchy, *weak* subsampling holds for CSPs of unique games type.
3. There are non-unique CSPs for which even weak subsampling fails for the above tighter semi-definite programs. Also there are unique CSPs for which (even weak) subsampling fails for the Sherali-Adams *linear programming* hierarchy.

As a corollary of our weak subsampling for strong semi-definite programs, we obtain a polynomial-time algorithm to certify that *random geometric graphs* (of the type considered by Feige and Schechtman, 2002) of max-cut value $1 - \gamma$ have a cut value at most $1 - \gamma/10$. More generally, our results give an approach to obtaining average-case algorithms for CSPs using semi-definite programming hierarchies.

*Department of Computer Science and Center for Computational Intractability, Princeton University `boaz@cs.princeton.edu`.
Supported by NSF grants CNS-0627526, CCF-0426582 and CCF-0832797, and the Packard and Sloan fellowships.
†Department of Computer Science and Center for Computational Intractability, Princeton University, `mhardt@cs.princeton.edu`.
Supported by NSF grants CCF-0426582 and CCF-0832797.
‡Department of Computer Science, ETH Zurich, `thomas.holenstein@inf.ethz.ch`. Work done while at Princeton University
and supported by NSF grants CCF-0426582 and CCF-0832797.
§Department of Computer Science and Center for Computational Intractability, Princeton University `dsteurer@cs.princeton.edu`.
Supported by NSF grants 0830673, 0832797, 0528414.

ISSN 1433-8092

Contents

1 Introduction

In this paper we consider the following seemingly unrelated questions:

1. Is the Max Cut problem hard on random geometric graphs of the type considered by Feige and Schechtman [FS02]?

2. Is the value of a mathematical relaxation for a constraint-satisfaction problem (CSP) preserved when one passes from an instance P to a random induced sub-formula of P?

It turns out that (in a sense made precise below) the answer to the first question is "no" and in fact this is intimately related to the second question. The answer to the second question is much more subtle, and, in contrast to the case of the objective value[1] of the CSP, the answer strongly depends on the type of relaxation and CSP.

1.1 Max Cut on the sphere

Max Cut— the problem of finding a cut maximizing the number of cut edges— is a widely studied optimization problem, important both in its own right, and as a testbed for techniques in algorithms and hardness of approximation. The best approximation algorithm for Max Cut known today is the semi-definite program GW SDP of Goemans and Williamson [GW94], which is optimal in the worst-case under the unique games conjecture [KKMO04, MOO05]. GW SDP is a special case of the BasicSDP algorithm for CSPs considered by Raghavendra [Rag08], who showed that the latter algorithm always has an optimal approximation factor in the worst-case under the unique games conjecture.

In particular GW SDP gives a value of at most $1 - \Omega(\varepsilon^2)$ when given as input a graph whose maximum cut cuts $1-\varepsilon$ fraction of the edges.[2] In this work we study the *average-case complexity* of Max Cut— namely whether one can do better on natural distributions over the instances. Since random graphs are expanders and so obviously have a maximum cut value close to $1/2$ (and moreover this fact can be efficiently certified using the second eigenvalue), one needs to consider other distributions over the inputs. We consider *random geometric graphs*, that in light of known results, arguably constitute the most natural distribution of Max Cut instances that is not obviously easy.

Random geometric graphs. A random geometric graph is obtained by taking the vertices as random unit vectors in $\mathbb{R}^d$, and connecting two vertices $u, v \in \mathbb{R}^d$ based on their distance $\|u - v\|_2$. We consider the distribution $\mathcal{G}_{n,d,\gamma}$, where the vertices are n random unit vectors in $\mathbb{R}^d$, and we connect two vectors if $\|u - v\|_2 \geqslant 2\sqrt{1-\gamma}$. By construction, GW SDP will have value $1-\gamma$ on these graphs, but, as shown by [FS02], as long as n is not too small these graphs will have with high probability a maximum cut value of $1 - c\sqrt{\gamma}$ for some absolute constant c. Moreover, as we observe here, for a suitable choice of n, these graphs will also be hard instances for the Sherali-Adams [SA90] *linear* programming hierarchies; these are generally incomparable with GW SDP and have been shown to solve Max Cut on *dense* graphs [dlVKM07]. Nevertheless, we show here that these graphs can be certified to have small max cut in polynomial time. (A certification algorithm that the max-cut of a random graph from a distribution is at most v is an algorithm whose output always upper bounds the max-cut, and with high probability the output is at most v.)

[1] A k-CSP is a collection $\mathcal{P}$ of functions mapping n variables from some finite alphabet to $\{0, 1\}$, such that every $P \in \mathcal{P}$ depends on at most k variables. We define the *objective value* of a CSP to be the maximum of $\frac{1}{|\mathcal{P}|}\sum_{P\in\mathcal{P}} P(x)$, taken over all possible assignments x to the variables.

[2] As a relaxation for a maximization problem, the value of GW SDP is always at least as large as the integral objective value. Hence the fact that the relaxation outputs some value v for an instance G is a *certification* that the maximum cut of G is at most v.

Informal Theorem 1 (Max Cut on random geometric graphs, see Theorem 5.5)**.** *There is a polynomial-time algorithm that certifies that a random graph G from $\mathcal{G}_{n,d,\gamma}$ satisfies* Max Cut$(G) \leqslant 1 - \Omega(\sqrt{\gamma})$*, for every $\gamma \in (0,1)$, $d \in \mathbb{N}$ and $n \geqslant C(\gamma)/\mu(\gamma,d)$, where* Max Cut$(G)$ *denotes the fraction of edges cut by the maximum cut in G, $C(\gamma)$ is some constant depending only on γ, and $\mu(\gamma,d)$ denotes the normalized measure in the unit sphere of the ball of radius $\sqrt{2\gamma}$ around some unit vector.*

By a simple calculation one can show that the probability that two random unit vectors u,v in $\mathbb{R}^d$ will satisfy $\|u-v\|_2 \geqslant 2\sqrt{1-\gamma}$ is exactly $\mu(\gamma,d)$, implying that if $n \ll 1/\mu(\gamma,d)$ the graph $\mathcal{G}_{n,d,\gamma}$ will have average degree $\ll 1$ (and hence has a trivial large max cut). Thus the value of n that Theorem 1 applies to is at most a constant factor larger than the minimum possible. The algorithm A of Theorem 1 is simply a tightening of the relaxation GW SDP obtained by adding the so-called "triangle inequalities" to that program.

1.2 Subsampling mathematical relaxations

The other question we consider is whether the value of mathematical relaxations such as linear and semi-definite programming is preserved under subsampling. That is, given a CSP instance ϕ on n variables, we consider the instance ϕ' obtained by choosing at random $S \subseteq [n]$ of some specified size, and keeping only the constraints involving only variables in S. We ask in what cases the value of the relaxation of ϕ' is close to the value of ϕ.

This question is a variant of property testing [Ron00, Rub06] that we believe is interesting in its own right. It also has algorithmic applications. Subsampling gives a fast way to "sketch" a CSP in a way that preserves the the objective value but using a much smaller instance size. But since we generally cannot compute this objective value in the worst case, we'd want to make sure that if ϕ was an "easy instance" for our algorithm, then ϕ' will be such an instance as well. A subsampling theorem for mathematical relaxations guarantees this property.

Subsampling for the objective value of constraint satisfaction problem (namely the fraction of satisfied constraints) was studied before by Goldreich, Goldwasser and Ron [GGR98] who gave a subsampling theorem for Max Cut, and by Alon, de la Vega, Kannan and Karpinski [AdlVKK03] who gave a subsampling theorem for general CSPs. But, to our knowledge, subsampling for mathematical relaxations was not studied before. As we show, unlike the case of the objective value, subsampling sometimes fails for the value of relaxations, and this depends on the particular relaxation and CSP.

Another, more minor difference between prior works and ours is that while prior works focused on the *dense* case, considering k-CSPs with $\Omega(n^k)$ constraints, we consider general, possibly non dense, CSPs, and wish to optimize the trade-off between the sample size and density. We say that a 2-CSP is Δ-dense if every variable appears in at least Δ constraints, and use a suitable generalization of this notion to k-CSPs (see Section 4). We show a subsampling theorem for the objective value of Δ-dense CSPs with the optimal sample of size $O(n/\Delta)$. Namely, we show that the value of the induced instanced is equal to the value of the original instance up to $1 \pm \varepsilon$ multiplicative factor, where O notation in the sample size hides polynomial factors in $1/\varepsilon$. The only prior work to consider this trade-off was by Feige and Schechtman [FS02], who gave such a result for Max Cut with $O(n \log n/\Delta)$ sample size.

Our results for subsampling mathematical relaxations of CSPs are the following (see Section 4.1 and 7 for formal statements). In all cases we consider a Δ-dense CSP $\mathcal{P}$ and a subformula of $\mathcal{P}'$ induced on a random subset of poly$(1/\varepsilon)(n/\Delta)$ variables, and we let $\Pi(\mathcal{P})$ be the value of the relaxation Π on $\mathcal{P}$.[3]

[3] For the positive results, our sample size is as small as possible; the negative results hold also for much larger sample size and in particular show that one cannot get a constant size subset even if $\Delta = \Omega(n)$, see Section 6.

We start by showing that subsampling holds for $\mathsf{BasicSDP}$ and $\mathsf{BasicLP}$, where $\mathsf{BasicSDP}$ is the semidefinite program considered by Raghavendra [Rag08] and $\mathsf{BasicLP}$ is its linear programming analog.[4]

Informal Theorem 2 (Subsampling for $\mathsf{BasicSDP}$ and $\mathsf{BasicLP}$, see Section 4.1)**.** *In the notation above, for any CSP $\mathcal{P}$ and for Π that is either $\mathsf{BasicSDP}$ or $\mathsf{BasicLP}$,*

$$\Pi(\mathcal{P}) - \varepsilon \leqslant \Pi(\mathcal{P}') \leqslant \Pi(\mathcal{P}) + \varepsilon$$

We then show that for stronger SDPs, we still have weak subsampling if the CSP is a unique game.

Informal Theorem 3 (Subsampling for unique games, see Theorem 7.2)**.** *In the notation above, if $\mathcal{P}$ is a unique game, then for every $k \in \mathbb{N}$, letting $\gamma = 1 - \mathsf{BasicSDP}_k(\mathcal{P})$,*

$$1 - \gamma - \varepsilon \leqslant \mathsf{BasicSDP}_k(\mathcal{P}') \leqslant 1 - \gamma/9 + \varepsilon$$

where $\mathsf{BasicSDP}_k$ denotes $\mathsf{BasicSDP}$ augmented with k rounds of the Lasserre hierarchy.

Theorem 3 is the main technical contribution of this paper, and also the one used to obtain our algorithm for Max Cut on random geometric graphs. We also have negative results that complement our positive results and show that, in contrast to the case of the objective value, subsampling sometimes fails for mathematical relaxations.

Informal Theorem 4 (Negative results for subsampling, see Theorems 6.1 and 6.2)**.** *There is a (non unique) CSP $\mathcal{P}$ and absolute constant $\delta > 0$ for which $\mathsf{BasicSDP}_{O(1)}(\mathcal{P}) \leqslant 1 - \delta$ but with high probability $\mathsf{BasicSDP}_{\sqrt{n}}(\mathcal{P}') \geqslant 1 - o(1)$. There is a unique CSP $\mathcal{P}$ and absolute constant $\delta > 0$ for which $\mathsf{BasicLP}_3(\mathcal{P}) \leqslant 1 - \delta$ but with high probability $\mathsf{BasicLP}_{\omega(1)}(\mathcal{P}) \geqslant 1 - o(1)$, where $\mathsf{BasicLP}_k$ denotes $\mathsf{BasicLP}$ augmented with k rounds of the Sherali-Adams hierarchy, and $o(1)$ (resp. $\omega(1)$) denotes a function that tends to 0 (resp. ∞) with n.*

See Figure 1 for an overview of our positive and negative results on subsampling mathematical relaxations. As one can see, we cover most of the cases, with the most interesting (in our opinion) open question is whether strong semidefinite programs for unique CSPs such as Max Cut actually have *strong* subsampling, in the sense that the value of the program on a subsample approximates the value on the original instance with arbitrary accuracy. We suspect that the answer is "no", though have no proof of that.

1.3 Subsampling SDPs and average-case complexity.

As mentioned above, we use Theorem 3 (weak subsampling theorem for strong SDPs on unique CSPs) to show Theorem 1— the Max Cut algorithm for random geometric graphs. Theorem 1 is obtained from our subsampling theorem as follows: we first show that $\mathsf{BasicSDP}_3(G_{d,\gamma}) \leqslant 1 - \Omega(\sqrt{\gamma})$ where $\mathsf{BasicSDP}_3$ denotes the $\mathsf{BasicSDP}$ program augmented with the triangle inequalities, and $G_{d,\gamma}$ is the graph on the continuous d-dimensional sphere where we connect two unit vectors u, v if $\|u - v\|_2 \geqslant 2\sqrt{1-\gamma}$. (Equivalently, one can think of $G_{d,\gamma}$ as a random graph from $\mathcal{G}_{n,d,\gamma}$ where d, γ are fixed and n tends to infinity.) We show this by observing that the edges of G can be partitioned into essentially disjoint union of *odd cycles* of length $O(1/\sqrt{\gamma})$, and noting that triangle inequalities can capture the fact that one cannot cut all the edges of an odd cycle. Since random geometric graphs are simply subsamples of $G_{d,\gamma}$, our subsampling theorem implies that $\mathsf{BasicSDP}_3$ will have value in $1 - \Theta(\sqrt{\gamma})$ for these graphs.

[4]We actually use the "smoothed" version of Raghavendra's SDP considered in [RS09, Ste10]— see Section 4.1. The two programs are closely related, and [Rag08]'s result holds for the smoothed version as well.

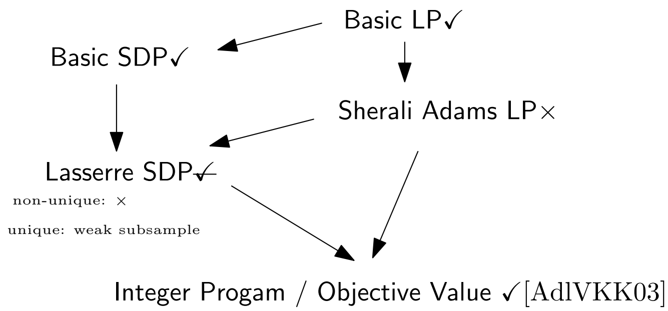

Figure 1: Overview of subsampling results. ✓denotes a subsampling theorem, while × denotes that subsampling fails. Arrows point from weaker to tighter relaxations.

This algorithm is an instance of a general recipe for using our subsampling theorem for average-case algorithms. Many natural distributions can be thought of as random subsamples of some instance (or family of instances) ϕ (e.g., random graphs are subsamples of a random dense graph, random 3SAT are subsamples of a random dense formula). In such cases, if one can give a relaxation that gives a tight value on ϕ (perhaps by exploiting its density) and the relaxation admits subsampling, then it follows that the relaxation succeeds on the distribution of subsamples as well. In our case, even though sufficiently many rounds on Sherali-Adams hierarchy give a tight value on $G_{d,\gamma}$ (since, considering γ as constant, it is dense), we cannot use those directly as they do not admit subsampling. Similarly, even though BasicSDP admits subsampling, it does not yield a tight value on dense 3SAT formulas, which is the reason our results do not refute Feige's hypothesis [Fei02] on the hardness of certifying that random 3SAT formulas are unsatisfiable.

We note that subsampling theorems have been used before for approximation algorithms for CSPs, but in a different way. Prior works used subsampling of the objective value to show worst-case approximation algorithms for dense graphs, by showing that one can first subsample to constant size and then solve the problem using brute force on the sample [AdlVKK03] (or use that argument to show that linear programming hierarchies will succeed on the original instance [dlVKM07]). In contrast we use subsampling of the relaxation value to give average-case algorithms on some specific distributions of (possibly sparse) graphs. Our result is also one of the few examples where higher order SDPs can succeed in an algorithmic task in which BasicSDP fails. As mentioned above, if the unique games conjecture is true, then BasicSDP is an optimal worst-case approximation algorithm for CSPs, though of course it can be worse than other efficient algorithms on some (distributions of) inputs.

1.4 Related work

As mentioned above, there has been many works on estimating graph parameters from random small induced subgraphs of *dense* graphs. Goldreich, Goldwasser and Ron [GGR98] show that the the Max-Cut value of a dense graph (degree $\Omega(n)$) is preserved by subsampling. (In this and other results, the constants depend on the quality of estimation.) Feige and Schechtman [FS02] showed that the result holds generally for Δ-dense graphs so long as the degree $\Delta \geqslant \Omega(\log n)$ and the subgraph is of size at least $\Omega(n \log n/\Delta)$. (As a corollary of our results, we slightly strengthen [FS02]'s bounds to hold for any $\Delta > \Omega(1)$ and subgraph size larger than $\Omega(n/\Delta)$.) Alon et al [AdlVKK03] generalize [GGR98] for k-CSP's and improve their quantitative estimates. See also [RV07] for further quantitative improvements in the case of $k = 2$.

There has also been much work on matrix and graph sparsification by means other than *uniform* sampling, see for instance [ST04, AHK06, AM07, SS08, BSS09]. Indeed, *spectral sparsifiers* are stronger than the notion we consider, in the sense that passing to a spectral sparsifier will preserve the SDP value for, say, Max Cut. Algorithmically though, if one only wants to preserve the SDP value, there are some advantages to subsampling, as it reduces not just the number of edges but also the number of vertices, hence potentially yielding *sublinear* algorithms, and can also be carried out very efficiently by just random sampling, reducing to a subgraph of constant degree. In contrast constant degree spectral sparsification [BSS09] cannot be achieved by sampling vertices (or even edges for that matter) uniformly at random, even for regular graphs.

2 Overview of proofs

In this section we give a high level overview of our proofs, focusing on our main result— Theorem 3 showing a weak subsampling theorem for strong semidefinite programs for unique games. A kCSP $\mathcal{P}$ on an alphabet $[q]$ is a collection of local functions (called "constraints") from $[q]^n \to [0,1]$, where for $x \in [q]^n$ we denote $\mathcal{P}(x) = \frac{1}{|\mathcal{P}|}\sum_{P\in\mathcal{P}} P(x)$. If U is a set of variables, then $\mathcal{P}[U]$ denotes the restriction of $\mathcal{P}$ to those constraints that depend on variables in U. We'll let $\mathcal{P}'$ denote $\mathcal{P}[U]$ where U is a random subset of size set to an appropriate parameter (that we ignore in this overview).

2.1 Subsampling for k-CSPs and BasicSDP

Alon, de la Vega, Kannan and Karpinski [AdlVKK03] proved a subsampling theorem for k-CSP. As a first step, we extend their results to hold with a better dependency between the sample size and density, and to hold for constraints that can output a real number, say in $[0,1]$, rather than just a Boolean value. The latter extension is trivial, but the former (which we need for our Max Cut application) requires some work, adapting and refining techniques of [GGR98, FS02, dlVKM07]. Our subsampling theorem for (generalized) k-CSPs is stated in Section 4 and proven in Section 8.

Subsampling for BasicSDP. BasicSDP is the semi-definite program for k-CSPs considered by Raghavendra [Rag08], who showed that it gives an optimal approximation ratio in the worst-case under the unique games conjecture. For a given kCSP $\mathcal{P}$ over alphabet $[q]$, this program assigns a vector $v_{i,a}$ for every variable x_i and alphabet symbol $a \in [q]$ of $\mathcal{P}$. It also assigns q^k numbers $\mu_{P,1^k},\ldots,\mu_{P,q^k}$ for every constraint P of $\mathcal{P}$. It makes the following consistency requirement on $\{v_{i,a}\}$ and $\{\mu_{P,x}\}$— the inner product of $v_{i,a}$ and $v_{j,b}$ should match the probability of the event "$x_i = a$ AND $x_j = b$" in any local distribution μ_P involving both variables x_i and x_j (this can be captured by linear and semi-definite conditions). The value of the CSP is simply the expectation of $P(x)$ over a random constraint P and a random partial assignment x chosen from μ_P. (To avoid the potential issue of the SDP being extremely sensitive to few of the constraints, we follow [RS09, Ste10] in allowing a bit of slackness in the consistency constraints on μ_P.)

Our subsampling theorem for BasicSDP, proven in Section 4.1, follows from the general subsampling for k-CSPs. The idea is to combine two observations: (1) because the assignment to the vectors $\{v_{i,a}\}$ determines the best choice for the local distribution, it is possible to write BasicSDP as a program that has no constraints and needs to maximize a sum of local functions over these vectors, (2) one can use dimension reduction to assume that the vectors have *constant* dimension with little loss of accuracy [RS09]. Thus by discretizing this constant dimensional space, we can think of BasicSDP as itself a CSP over some constant sized alphabet, and apply our k-CSP subsampling theorem to this CSP. A similar (even simpler) reasoning applies to the linear programming variant BasicLP, and also to quadratic programs, in particular implying a variant of property testing for positive semi-definiteness, see Section 4.

2.2 Weak subsampling for strong SDPs

We now give a high level overview of the proof of Theorem 3. Because stronger SDPs such as those from the Lasserre hierarchy actually involve constraints including several vectors, they cannot be expressed as a CSP in the same way as BasicSDP. Indeed, we have negative results showing that subsampling can fail for these SDPs (see Sections 2.3 and 6).

The result actually does not depend on the particular properties of the Lasserre hierarchy, and holds for a very general class of strong SDPs. We start by formalizing this class. Any strong SDP can be thought of as the program BasicSDP augmented with the constraint that the positive semi-definite matrix X of the inner products of all these vectors is in some convex set $\mathcal{M}$. But one needs the set $\mathcal{M}$ to satisfy some "niceness conditions" in order for it to make sense to apply the program on a subsampled CSP. The niceness conditions we consider are rather mild, and require that solutions remain valid under renaming and identifying of vertices (see Section 7). In particular they apply to any SDP obtained by adding a number of Lasserre rounds to BasicSDP.

If Π is any strong SDP, $\mathcal{P}$ is a CSP, and $\mathcal{P}'$ is a subsample of $\mathcal{P}$, then it's not hard to show that with high probability $\Pi(\mathcal{P}') \geqslant \Pi(\mathcal{P}) - \varepsilon$, since this only needs the argument that the value of one solution (the optimal one for $\mathcal{P}$) will be approximately preserved. The challenging task is to show that $\Pi(\mathcal{P}')$ is not much larger than $\Pi(\mathcal{P})$, and because subsampling does not always hold for SDPs, we know that the proof for subsampling of k-CSPs does not generalize to this case.

The crucial notion we use is of that of a *proxy CSP*. Let $\mathcal{G}$ and $\mathcal{H}$ be two unique games on the same alphabet and number of variables, we say that $\mathcal{H}$ is *proxy* for $\mathcal{G}$ (with respect to the program Π), if for *every* assignment X (even possibly outside $\mathcal{M}$) to the vectors of Π, $1 - \Pi(\mathcal{G})[X] \leqslant 1 - \Pi(\mathcal{H})[X]/10$, where $\Pi(\mathcal{P})[X]$ denotes the value of the program Π on the CSP $\mathcal{P}$ with assignment X to the vectors of Π.[5] That is, one can think of $\mathcal{H}$ as pointwise dominating $\mathcal{G}$ with respect to the program Π. We then show that this domination condition is somewhat preserved under subsampling, at least for the optimal solutions. That is, we show that with high probability $1 - \Pi(\mathcal{G}') \leqslant 1 - \Pi(\mathcal{H}')/10 + \varepsilon$, where $\mathcal{G}'$ and $\mathcal{H}'$ are the subsampled versions of $\mathcal{G}$ and $\mathcal{H}'$. The idea here is to use our subsampling theorem for SDP looking at the SDP $\max_X \Pi(\mathcal{H})[X]/10 - \Pi(\mathcal{G})$. This is a basic SDP since it places no constraints on X, and so since we know its optimum is at most 0, this should be approximately preserved under subsampling.

The above discussion shows that to prove Theorem 3 it suffices to find some unique game $\mathcal{H}$ such that (*) $\mathcal{H}$ is a proxy for $\mathcal{G}$ and (**) with high probability $1 - \Pi(\mathcal{H}') \leqslant 1 - \Pi(\mathcal{G}) + \varepsilon$. This is what we do. The proxy game $\mathcal{H}$ is simply the game $\mathcal{G}^3$ obtained by taking all length-3 paths in the constraint graph of $\mathcal{G}$ and composing the corresponding permutations. Condition (*) is not that hard to show. Intuitively, an assignment that satisfies $1 - \gamma$ fraction of the constraints of $\mathcal{G}$ should satisfy about $1 - 3\gamma$ fraction of the constraints of $\mathcal{H}$, (since each one is just three constraints of $\mathcal{G}$) and this reasoning carries over to SDP assignments as well.

Condition (**) looks suspiciously close to what we're trying to prove in the first place (preservation of value under subsampling), but note the asymmetry— we need to show that a subsample of $\mathcal{H}$ will have roughly the same value as the original graph $\mathcal{G}$. It turns out this will actually help us. What we need to show is a way to decode an assignment for the SDP of the subsampled game $\mathcal{H}'$ into an assignment of roughly the same value for the SDP of the original game $\mathcal{G}$. For simplicity, assume that the alphabet of the CSP is $\{0, 1\}$ in which case the vector assignment is just one vector per variable.[6] Suppose that $\mathcal{G}$ has n variables, each

[5] The actual domination condition we use will restrict the possible vector assignments based on the norms of the vectors, but because we restrict the vectors to a product set, it does not make a difference in our arguments.

[6] Although in the phrasing above it seems that one would need two vectors per variable for alphabet of size 2, it is known how to transform the SDP into an equivalent program needing only one vector per variable in this case.

participating in Δ constraints, and we subsample to a set S of size $n' = O(n/\Delta)$ variables.[7] We are given a vector assignment $\{v'_{i'}\}_{i'\in S}$ for each of the n' variables in the sample that gives value τ for $\Pi(\mathcal{H}')$, and need to "decode" it into an assignment $\{v_i\}_{i\in[n]}$ that gives value roughly τ for $\Pi(\mathcal{G})$. We will use a randomized decoding, assigning for every variable i of $\mathcal{G}$ the vector $v_{i'}$ where i' is a random neighbor of i in $\mathcal{G}$ that is contained in the sample S.[8] Let (i', i, j, j') be the length-3 path corresponding to a random constraint of $\mathcal{H}'$ that survived the subsampling. That is, $i', j' \in S$. If the subsampled graph is (approximately) regular, we can choose (i', i, j, j') in the following way: first let (i, j) be variables corresponding to a random constraint in $\mathcal{G}$, then take i' to be a random neighbor of i that is also in S, and take j' to be a random neighbor of j that is also in S. We know that on average the vectors $v'_{i'}$ and $v'_{j'}$ contribute τ to the value of $\Pi(\mathcal{H}')$. But then on expectation the contribution to $\Pi(\mathcal{G})$ of the decoded vectors v_i and v_j is also τ, since v_i is exactly obtained by taking $v'_{i'}$ for a random neighbor $i' \in S$ of i, and v_j is obtained by taking $v'_{j'}$ for a random neighbor $j' \in S$ of j. This concludes the proof. We remark that this reasoning is somewhat reminiscent of Dinur's analysis of her gap amplification lemma for PCP's [Din07].

2.3 Negative results for subsampling

We now briefly sketch why, unlike the case for kCSPs, subsampling sometimes *fails* for strong semidefinite and linear programs—- see Section 6 for more details. The idea is simple: many integrality gaps examples, for both LP hierarchy and SDP's, are actually obtained from random instances. Examples include Schoenebeck's result [Sch08] showing random 3SAT is an integrality gap example for the Lasserre SDP hierarchy, and results showing that random graphs (and more generally good expanders) are integrality gap examples for linear programming hierarchies for Max Cut [dlVKM07, CMM09]. Such random instances can be thought of as subsampling of sufficiently dense instance. But sufficiently strong SDP or LP programs will succeed in certifying that a dense instance has small value. Thus these integrality gaps give example of a CSP $\mathcal{P}$ where $\Pi(\mathcal{P})$ is small, where Π is a sufficiently strong linear or semidefinite program, but $\Pi(\mathcal{P}')$ is close to 1 for a random induced sub-instance $\mathcal{P}'$ of $\mathcal{P}$. Note that indeed for unique games random graphs are actually easy for semi-definite programs [AKK+08], explaining perhaps why subsampling for unique games is possible for semi-definite programs but not for linear programs.

3 Preliminaries

Let G be a Δ-regular graph with vertex set $V = [n]$ and edge set E (no parallel edges or self-loops). We give weight $2/\Delta n$ to each edge of G so that every vertex of G has (weighted) degree $2/n$ and G has total edge weight 1. We say a graph is *normalized* if it has total edge weight 1. (We choose this normalization, because we will often think of a graph as a probability distribution over unordered vertex pairs.) For a graph G as above and a vertex subset $U \subseteq V$, let $G[U]$ denote the *induced subgraph* on U. To preserve our normalization, we scale the weights of the edges of $G[U]$ such that the total edge weight in $G[U]$ remains 1. We denote by V_δ a random subset of a δ fraction of the vertices in V, and hence $G[V_\delta]$ denotes a random induced subgraph of G of size $\delta|V|$. With our normalization, the typical weight of an edge in $G[V_\delta]$ is $2/\delta^2\Delta n$.

[7] Note that, ignoring constant factors, $\mathcal{G}$ has roughly $n\Delta$ constraints, $\mathcal{G}'$ has n/Δ constraints, $\mathcal{H}$ has $n\Delta^3$ constraints, and $\mathcal{H}'$ has $n\Delta$ constraints— the latter fact is some indication why one may hope to decode an assignment to $\mathcal{H}'$ into an assignment to $\mathcal{G}$.

[8] Our "niceness conditions" will ensure that if the inner-product matrix of the original assignment was in $\mathcal{M}$ then the same will hold for the decoded assignment. Also we will flip the vector if the corresponding permutation on $\{0, 1\}$ was $a \mapsto -a$, but in the discussion below as assume that all permutations involved are the identity— this simplifies notation and is immaterial to this argument.

Max k-CSPs. A k-CSP instance $\mathcal{P}$ is a set of predicates (or pay-off functions) of the form $P\colon [q]^n \to \mathbb{R}$, where every $P = P(x_{i_1}, x_{i_2}, \ldots, x_{i_k})$ is a k-junta, meaning it depends only on k of the n variables in x. We'll think of $\mathrm{Var}(P) = (i_1, \ldots, i_k)$ as an ordered set and denote the r-th variable by $\mathrm{Var}_r(P) = i_r$. Without loss of generality we may assume that in each predicate $P \in \mathcal{P}$, all k variables are distinct. The norm of a pay-off function is defined as $|P| \stackrel{\text{def}}{=} \max_{x\in[q]^n} |P(x)|$, and we put $|\mathcal{P}| = \sum_{P\in\mathcal{P}} |P|$.

We think of $\mathcal{P}$ itself as a mapping $\mathcal{P}\colon [q]^n \to \mathbb{R}$ defined as $\mathcal{P}(x) \stackrel{\text{def}}{=} \frac{1}{|\mathcal{P}|}\sum_{P\in\mathcal{P}} P(x)$. The optimum is denoted $\mathrm{opt}(\mathcal{P}) = \max_{x\in[q]^n} \mathcal{P}(x)$. We will typically assume that $|P| \leqslant 1$ for all $P \in \mathcal{P}$ in which case $\mathrm{opt}(\mathcal{P}) \leqslant 1$. For a subset $U \subseteq [n]$, with $|U| = \delta n$, we let $\mathcal{P}_U$ denote the k-CSP $\mathcal{P}_U = \{\delta^{-k}P\colon P \in \mathcal{P}, \mathrm{Var}(P) \subseteq U\}$. In this case, we define $\mathcal{P}_U(x) = \frac{1}{|\mathcal{P}|}\sum_{P\in\mathcal{P}_U} P(x)$.

Unique Games. A unique game $\mathcal{G}$ is given by a constraint graph $G = (V, E)$, an alphabet $[R]$ and constraints $\pi_{v\leftarrow u}$ for each edge $e = (u, v) \in E$. An assignment $x \in [R]^n$ satisfies the edge e if $\pi_{v\leftarrow u}(x_u) = x_v$. It will be convenient for us to define unique games as a *minimization* problem in which the objective is to minimize the number of unsatisfied constraints. Note that throughout the introduction Unique Games was a maximization problem, but these two views are equivalent. As a minimization problem unique game has the following SDP relaxation (which is closely related to BasicSDP program mentioned before):

$$\mathrm{sdp}(\mathcal{G}) \stackrel{\text{def}}{=} \min \mathop{\mathbb{E}}_{(u,v)\in E}\ \mathop{\mathbb{E}}_{a\in[R]} \|u_a - v_{\pi_{v\leftarrow u}(a)}\|^2 \tag{1}$$

subject to the constraints that $\sum_{a\in[R]} \|u_a\|^2 = 1$ for every $u \in V$ and $\langle u_a, u_b\rangle = 0$ for all $u \in V$ and $a \neq b$. An SDP solution is a positive semidefinite $(V \times [R]) \times (V \times [R])$ matrix written as $(X_{ua,vb})_{u,v\in V,a,b\in[R]}$ so that $X_{ua,vb} = \langle u_a, v_b\rangle$. We will denote by $\mathcal{M}_2$ the set of such matrices that satisfy the three constraints above. We will write $\mathrm{sdp}(\mathcal{G})[X]$ to denote the value of $\mathrm{sdp}(\mathcal{G})$ under the particular solution X. We denote by $\mathcal{G}[U]$ the unique game $\mathcal{G}$ restricted to the constraint graph $G[U]$.

4 Subsampling theorem for Max-kCSPs

We will now state our subsampling theorem for k-CSPs and, as direct application, obtain subsampling theorems for basic semidefinite relaxations of k-CSPs. To state the theorem we need a notion of *density* of a k-CSP. For 2-CSPs we will use the standard notion of density in a graph. Specifically, we will say a 2-CSP is Δ-*dense* if every vertex has $\Theta(\Delta)$ neighbors. For k-CSPs when $k > 2$ a natural generalization is to demand that after assigning $k-1$ out of k coordinates in each constraint, there are still $\Theta(\Delta)$ constraints remaining. In this case we say that the k-CSP is Δ-dense.

Theorem 4.1. *Let $\varepsilon > 0, \Delta > 1$. Let $\mathcal{P}$ be a Δ-dense k-CSP in n variables over an alphabet of size q so that $|P| \leqslant 1$ for all $P \in \mathcal{P}$. Put $\delta \geqslant \varepsilon^{-C}\log(q)/\Delta$ for some absolute constant C. Suppose $U \subseteq [n]$ is chosen uniformly at random so that $|U| = \delta n$. Then,*

$$\left|\mathbb{E}\,\mathrm{opt}(\mathcal{P}_U) - \mathrm{opt}(\mathcal{P})\right| \leqslant \varepsilon\,. \tag{2}$$

The formal density condition and the proof of this theorem are given in Section 8. We instead proceed to discuss the applications of this theorem.

4.1 Subsampling basic semidefinite programs

The above subsampling theorem for k-CSPs can actually be used to give a general subsampling theorem for *basic* semidefinite programs. A semidefinite program is called *basic* if it can be written as a 2-CSP Q in n variables (over infinite alphabet) of the following form:

$$\mathrm{opt}(Q) = \max \mathop{\mathbb{E}}_{i,j\in[n]} P_{ij}(\{v_{i,a}\}_{a\in[R]}, \{v_{j,b}\}_{b\in[R]}) \tag{3}$$

where $Q = \{P_{ij}\}_{i,j\in[n]}$ so that each P_{ij} is a continuous function satisfying a Lipschitz condition in the inner products $\langle v_{i,a}, v_{j,b}\rangle$. Here the maximum is taken over a bundle of R vectors $\{v_{i,a}\}$ per variable $i \in [n]$. We further require that each constraint on the vectors involves only vectors from the same bundle $\{v_{i,a}\}_{a\in[R]}$ (such as $\|v_{i,a}\|^2 \leqslant 1$ or $\sum_{a\in[R]} \|v_{i,a}\|^2 = 1$). We also assume that $|P_{ij}| \leqslant 1$.

It is crucial here that the maximization is over a *product* space of n coordinates. Each coordinate corresponds to one vector bundle $\{v_{i,a}\}_{a\in[R]}$. Still we cannot yet apply our subsampling theorem, because each coordinate is maximized over a continuous space, i.e., $(B_2^d)^R$. However, using dimension reduction as in [RS09], the dimension of the vectors can be assumed to be $\mathrm{poly}(1/\varepsilon)$ without changing the objective value by more than an $\varepsilon/2$. Once the dimension is small we can discretize the space by an ε'-net (for small enough ε') changing the inner products again only by $\varepsilon/2$. Hence we have the following lemma.

Lemma 4.2. *Let Q be a Δ-dense 2-CSP of the form (3). Then there is a 2-CSP Q' with alphabet size at most $2^{\mathrm{poly}(1/\varepsilon)}$ such that $|\mathrm{opt}(Q) - \mathrm{opt}(Q')| \leqslant \varepsilon$.*

This shows that we do have a strong subsampling theorem for any *basic* semidefinite program:

Corollary 4.3. *Let Q denote a basic semidefinite program. Assume Q is Δ-dense and let $\varepsilon > 0$. Then,*

$$\left| \mathbb{E}\,\mathrm{opt}(Q_U) - \mathrm{opt}(Q) \right| \leqslant \varepsilon\,, \tag{4}$$

where $U \subseteq [n]$ is a randomly chosen set of size $\varepsilon^{-C} n/\Delta$ for sufficiently large $C > 0$.

Proof. After applying Lemma 4.2, we can use Theorem 4.1 to conclude the claim. Note that the alphabet size of $2^{\mathrm{poly}(1/\varepsilon)}$ translates into a factor $\mathrm{poly}(1/\varepsilon)$ in sample size. ∎

We will next demonstrate that both BasicSDP for k-CSPs and the Unique Games SDP are in fact basic relaxation of the above form and therefore have a strong subsampling theorem.

For the Unique Games SDP this is immediate after changing it from a minimization problem to a maximization problem. (We can simply multiply the objective by -1.) Note that the SDP relaxation for unique games corresponds to a dense 2-CSP if this is the case for the constraint graph. We remark that the same is true for the difference of two dense Unique Games relaxations and this is the case that will be used in the proof of our main theorem later (Section 7).

More generally, the same can be done for the BasicSDP relaxation of any k-CSP. Raghavendra [Rag08] defined BasicSDP for a k-CSP $\mathcal{P} = \{P_1, \ldots, P_m\}$ with $|P_t| \leqslant 1$ over the alphabet $[R]$ as the program

$$\max \mathop{\mathbb{E}}_{t\in[m]} \mathop{\mathbb{E}}_{x\sim\mu_t} P_t(x)$$

subject to the constraint that $\mathbb{P}\mathrm{r}_{x\sim\mu_t}\{x_i = a, x_j = b\} = \langle v_{i,a}, v_{j,b}\rangle$ for all $t \in [m]$, $i, j \in \mathrm{Var}(P_t)$, and $a, b \in [R]$. The maximum is taken over all ensembles $\{v_{i,a}\}$ of unit vectors and R^k tuples of variables μ_t, each of which is required to be a probability distribution on $\mathrm{Var}(P_t)$. Let $\mathsf{violate}(t)$ denote the sum of $|\,\mathbb{P}\mathrm{r}_{x\sim\mu_t}\{x_i = a, x_j = b\} - \langle v_{i,a}, v_{j,b}\rangle|$ over all $i, j \in \mathrm{Var}(P_t)$ and $a, b \in R$. While the constraints of [Rag08] is that $\mathsf{violate}(t) = 0$ for all $t \in [m]$, we follow [RS09, Ste10] that replaced this with the constraint $\mathbb{E}_{t\in[m]}\,\mathsf{violate}(t) \leqslant \varepsilon$ and showed the two programs are approximately equivalent up to $\operatorname{poly}(\varepsilon)$ perturbation of the instance. As shown in [Ste10], because there are only a few (R^2k^2) constraints per pay-off function P_t, we can introduce this penalty function into the objective function, adding the term $-\,\mathbb{E}_{t\in[m]}\,\mathsf{violate}(t)/\varepsilon$ into the expression we maximize. Hence, for our purposes we may assume that BasicSDP has the form in (3) so that our subsampling theorem applies. We stress that this approach can only work since there are a few constraints for each pay-off functions of $\mathcal{P}$. The approach breaks down in the presence of constraints that involve arbitrary combinations of variables, such as ℓ_2^2 triangle inequalities. In this case it is no longer possible to assign a meaningful penalty to each constraint.

In the case of BasicLP similar arguments apply. BasicLP is the same as BasicSDP except that we don't require the probability distributions to be realized as inner products of vectors. Two distributions μ_P and $\mu_{P'}$ are however required to be consistent whenever they share a variable. These constraints can be written in the objective function and this results in a 2-CSP to which our subsampling theorem applies.

Application to property testing positive definite matrices. Our subsampling theorem also applies to quadratic forms and this can be very useful. We illustrate one application in the context of property testing. Specifically, we will get a property testing algorithm for the class of positive semidefinite matrices. Let us say that a matrix B is *ε-far* from positive semidefinite definite if there exists a vector x with $\|x\|_\infty \leqslant 1$ such that $-\varepsilon \geqslant \langle x, Bx\rangle = \sum_{ij} b_{ij}x_ix_j$. Recall that B is positive semidefinite if and only if $\langle x, Bx\rangle \geqslant 0$ for all x. Notice we could have defined distance in terms of the operator norm which is to say that there exists an x with $\|x\|_2 \leqslant 1$ such that $\langle x, Bx\rangle \leqslant -\varepsilon$. However, since every vector x of Euclidean norm 1 also satisfies $\|x\|_\infty \leqslant 1$, this would only be a stronger notion of "ε-far" thus applying to fewer matrices. Note that the expression $\max_{x:\|x\|_\infty\leqslant 1}\langle x, Bx\rangle$ is a 2-CSP to which we can apply our subsampling theorem (after discretization of the domain.) This lets us distinguish between matrices that are positive semidefinite and those that are ε-far from a small subsample. Formally, we get the following corollary. The simple proof is omitted.

Corollary 4.4. *Let B by a matrix with $\|B\|_\infty \leqslant D/n^2$. Then there is a property testing algorithm $\mathcal{A}$ such that: If B is ε-far from being positive semidefinite, then $\mathcal{A}$ rejects B with probability greater than $2/3$. If B is positive semidefinite, then $\mathcal{A}$ rejects B with probability less than $1/3$. Furthermore, $\mathcal{A}$ reads only* $\operatorname{poly}(D, \varepsilon^{-1})$ *many entries of B and runs in time* $\operatorname{poly}(D, \varepsilon^{-1})$

5 Max Cut in random geometric graphs

In this section we discuss the application of our theorem to solving Max Cut in random geometric graphs. Let us first recall some basic facts. The value of the maximum cut of a graph G is given by $\operatorname{opt}(G) := \max_{x\in\{-1,1\}^n}\langle x, {}^1\!/\!_4 L(G)x\rangle$. Here $L(G)$ denotes the combinatorial Laplacian of G. The Goemans-Williamson [GW94] relaxation for Max Cut is $\operatorname{sdp}(G) = \max\left\{{}^1\!/\!_4 L(G) \bullet X \mid X \succeq 0, \forall i\colon X_{ii} = 1\right\}$. Note that $\operatorname{opt}(G)$ and $\operatorname{sdp}(G)$ range between 0 and 1, the total edge weight of a normalized graph. We will consider relaxations obtained by adding valid constraints to the above program. A specific set of constraints we'll be interested in are the *ℓ_2^2 triangle inequalities* which can be expressed by adding the constraint $X_{ij}+X_{jk}-X_{ik} \leqslant 1$ and $X_{ij} + X_{jk} + X_{ik} \geqslant -1$. for every $i, j, k \in V$. The relaxation including triangle inequalities will be denoted $\operatorname{sdp}_3(G)$.

Sphere graphs. We denote by G_γ the graph on the vertex set $V = \mathbb{S}^{d-1}$ with edge set $E = \{(u, v) \in V \times V \mid \frac{1}{4}\|u - v\|^2 \geqslant 1 - \gamma\}$. The integral value of G_γ, denoted $\operatorname{opt}(G_\gamma)$, is defined as the maximum of $\mu(A, \bar{A}) \stackrel{\text{def}}{=} \mu^2(\{(x, y) \in E\colon x \in A, y \notin A\})$ taken over all measurable subsets $A \subseteq \mathbb{S}^{d-1}$ Here, μ denotes the uniform surface measure of the sphere $\mathbb{S}^{d-1}$ and $\mu^2 = \mu \times \mu$. A theorem of Feige and Schechtman shows that the maximum is attained for any hemisphere.

Theorem 5.1 (Feige-Schechtman [FS02])**.** *Fix $\gamma \in [0, 1]$ and consider the graph G_γ. Then, the maximum of $\mu(A, \bar{A})$ over all measurable subsets $A \subseteq \mathbb{S}^{d-1}$ is attained for any hemisphere $H \subseteq \mathbb{S}^{d-1}$.*

Recall, if A is a hemisphere, $\mu(A, \bar{A}) = 1 - \Theta(\sqrt{\gamma})$. Hence $\mathrm{opt}(G_\gamma) = 1 - \Theta(\sqrt{\gamma})$. At this point we mention that the SDP relaxation for Max Cut is well-defined on infinite graphs though we omit the formal details. In this case it is easiest to think of E as a distribution over edges so that the SDP maximizes the quantity $\mathbb{E}_{(u,v)\sim E} \frac{1}{4}\|f(u) - f(v)\|^2$ over all embeddings $f \colon V \to B$ satisfying the usual additional constraints. Here B can be taken to be the unit ball of the infinite dimensional Euclidean space.

The sphere graph itself can then be interpreted as an SDP solution, hence the following fact.

Fact 5.2 (Basic SDP)**.** $\mathrm{sdp}(G_\gamma) \geqslant 1 - \gamma$.

Proof. The graph itself gives an embedding (the identity embedding) such that for each edge $(u, v) \in E$, $\frac{1}{4}\|u - v\|^2 \geqslant 1 - \gamma$. Since the SDP averages this quantity over all edges in the graph, the claim follows. ■

We will show next that triangle inequalities change the value of the SDP from $1 - \gamma$ to $1 - \Omega(\sqrt{\gamma})$ thus capturing the integral value up to constant factors in front of γ.

Lemma 5.3. $\mathrm{sdp}_3(G_\gamma) \leqslant 1 - \Omega(\sqrt{\gamma})$.

This lemma was quite possibly previously known, but we will give a proof in Section B for lack of a reference. Using standard discretization arguments all previous lemmas can be transferred to a sufficiently dense discretization of the continuous sphere. Similarly, it is not difficult to show that sufficiently many random points from the sphere will give a good discretization.

Lemma 5.4. *Fix $\gamma \in [0, 1], d \in \mathbb{N}$. Then, there exists an $n_0(d, \gamma) \in \mathbb{N}$ so that if we pick $V \subseteq S^{d-1}$ uniformly at random with $|V| \geqslant n_0$, then the induced subgraph $G_\gamma[V]$ satisfies (1)* $\mathrm{opt}(G_\gamma[V]) = 1 - \Theta(\sqrt{\gamma})$, *and (2)* $\mathrm{sdp}_3(G_\gamma[V]) = 1 - \Theta(\sqrt{\gamma})$.

The proof is given in Section B. It is worth noting that the proof of the previous lemma gives a very weak bound on the number of vertices that we are required to subsample. In particular, it is not difficult to see that the average degree of the graph will be $n^{1-o(1)}$. A priori, it could therefore be the case that the SDP value changes when considering a subsample of the sphere with average degree $\log(n)$ or even $O(1)$. Indeed, [FS02] show that for some fixed γ, a random subsample of the sphere of expected degree $O(\log n)$ will satisfy most triangle inequality constraints with high probability thus exhibiting some integrality gap for sdp_3.[9] However, our main theorem in this section implies that asymptotically sdp_3 behaves like $1 - \sqrt{\gamma}$ rather than $1 - \gamma$.

To argue this, we'd like to use our subsampling theorem for unique games. It is well known how to express the max-cut problem on a graph G as an instance $\mathcal{G}$ of Unique Games where the constraint graph is exactly G. Since we defined unique games to be minimization problems, this corresponds to minimizing the number of uncut edges. We therefore have that $\mathrm{opt}(G) = 1 - \mathrm{opt}(\mathcal{G})$ and furthermore it is well known that $\mathrm{sdp}(G) = 1 - \mathrm{sdp}(\mathcal{G})$ for the basic SDP relaxation and also $\mathrm{sdp}_3(G) = 1 - \mathrm{sdp}_3(\mathcal{G})$ where the latter refers to an SDP relaxation for Unique Games that includes triangle inequalities, yielding the following theorem:

[9] In their example the angle between two neighboring vertices is chosen to be more than 60 degrees corresponding to very large γ to which our theorem does not apply due to the constant factor loss in γ.

Theorem 5.5. *Fix $\gamma \in [0,1]$ and let $\Delta > \mathrm{poly}(1/\gamma)$. Fix d and choose n such that for n uniformly random points $V \subseteq \mathbb{S}^{d-1}$ the induced graph $G_\gamma[V]$ has expected degree Δ. Then,*

$$\mathrm{sdp}_3(G_\gamma[V]) = 1 - \Theta(\sqrt{\gamma}).$$

Proof. We think of $G_\gamma[V]$ as a uniform vertex subsample of a random dense discretization $G_\gamma[W]$ in d dimension. Note that by Lemma 5.4 we have $\mathrm{sdp}_3(G_\gamma[W]) = 1 - \Theta(\sqrt{\gamma})$. We can reduce $G_\gamma[W]$ to a unique game $\mathcal{G}$ so that $\mathrm{sdp}_3(\mathcal{G}) = \Theta(\sqrt{\gamma})$. Now $G_\gamma[V]$ corresponds to the unique game $\mathcal{G}[V]$, since the constraint graph of $\mathcal{G}[V]$ is precisely $G_\gamma[V]$. By Theorem 7.2 (subsampling theorem for Unique Games), we know that $\mathrm{sdp}_3(\mathcal{G}[V]) = \Theta(\sqrt{\gamma})$. Note that triangle inequalities correspond to a reasonable relaxation. But then it follows that $\mathrm{sdp}_3(G_\gamma[V]) = \Theta(\sqrt{\gamma})$. ∎

Theorem 1 is a corollary of this theorem, since sdp_3 can now be used to certify that random geometric graphs have small max-cut value.

6 Negative results for subsampling

In this section we first observe that its is impossible to obtain even a weak subsampling result for the semidefinite programming relaxation of k-CSPs with $k \geqslant 3$. This results follows from Schoenebeck's integrality gap [Sch08]. We also argue that even in the case of 2-CSPs subsampling is impossible when the constraints are *not* unique.

Second, we give a separation between semidefinite programming and linear programming by showing that a subsampling result for linear programming is impossible even in the case of Max Cut and Unique Games. Here, our results are based on the integrality gap construction of [CMM09].

6.1 No subsampling for SDP relaxations of k-CSPs with $k \geqslant 3$.

Theorem 6.1. *There is a k-CSP $\mathcal{P}$ with $\Omega(n^k)$ constraints in the variables $[n]$ so that $\mathrm{sdp}_{O(1)}(\mathcal{P}) \leqslant 0.51$, but with high probability $\mathrm{sdp}_{\Omega(n)}(\mathcal{P}[U]) \geqslant 0.99$ where $U \subseteq [n]$ is a random set of size δn with $\delta > c/n^{1-1/k}$ for some constant c.*

Proof sketch. We may take $\mathcal{P}$ to be a random dense instance of k-XOR. It is known that an SDP with a constant number of rounds of Lasserre captures the integral value of the CSP. Now $\mathcal{P}[U]$ is a k-XOR instance with $\Omega(\delta^k n^k) = Cn$ constraints for some constant C. For large enough C , the result of [Sch08] then implies the claim. ∎

6.2 No subsampling for SDP relaxations of non-unique 2-CSPs

The above result also shows that we cannot hope for a subsampling theorem for semidefinite relaxations of *non-unique* 2-CSPs. Indeed, we can take a dense instance $\mathcal{P}$ of 3-SAT and express it as a 2-CSP $\mathcal{P}'$ as follows: Every constraint $P \in \mathcal{P}$ gets mapped to a new variable x_P over the alphabet [8]. Each label represents an assignment to the original constraint. Every two constraints sharing one variable in $\mathcal{P}$ contribute one constraint $P' \in \mathcal{P}'$ which enforces that the assignment to the shared variable is consistent.

Subsampling variables in $\mathcal{P}'$ corresponds to subsampling constraints in $\mathcal{P}$. Using [Sch08], the subsample of $\mathcal{P}$ will be a gap instance for the Lasserre hierarchy. Since our reduction is local, ideas of [Tul09] show that also the subsample of $\mathcal{P}'$ will be a gap instance. This rules out the possibility of a subsampling theorem for non-unique 2-CSPs of alphabet size 8.

6.3 No subsampling for LP relaxations of 2-CSPs

In this section we rule out subsampling theorems for strong linear programming relaxations even in the case of Max Cut for which strong semidefinite relaxations do admit a subsampling theorem. Specifically, we consider the Sherali-Adams LP relaxation for Max Cut: $\mathrm{lp}_r(G) = \max \sum_{(u,v)\in E} x_{uv}$ over (u, v) s.t. the vector $(x_{uv})_{u,v\in V}$ lies in the Sherali-Adams relaxation of the cut polytope.

The Sherali-Adams relaxation of the cut polytope is obtained by applying r rounds of lift-and-project operations to the base set of linear inequalities that define the metric polytope, i.e., $\{x_{ij} + x_{jk} \geqslant x_{ik}, x_{ij} + x_{jk} + x_{ik} \leqslant 2, x_{ij} = x_{ji}, 1 \geqslant x_{ij} \geqslant 0\}$. For a formal definition see, for instance, [CMM09].

The next theorem shows that there are graphs which have Sherali-Adams value bounded away from 1 for a constant number of rounds. But after subsampling the value comes arbitrarily close to 1 even when considering a huge number of rounds.

Theorem 6.2. *For every function $\varepsilon = \varepsilon(n)$ that tends to 0 with n, there exists a function $r = r(n)$ that tends to ∞ with n and family of graphs $\{G_n\}$ of degree $D = D(n)$ such that*

1. *For every n,* $\mathrm{lp}_3(G_n) \leqslant 0.8$
2. *If G' is a random subgraph of G of size $(n/D)^{1+\varepsilon(n)}$ then* $\mathbb{E}[\mathrm{lp}_{r(n)}(G')] \geqslant 1 - \frac{1}{r(n)}$.

where $\mathrm{lp}_k(H)$ *denotes the value of k levels of the Sherali-Adams linear program for Max-Cut on the graph H.*

Proof sketch. Let $G_n = G_{n,p}$ for some $p \leqslant \frac{1}{2}$. It is not difficult to argue that three rounds of Sherali-Adams have value at most 0.7 on $G = G_n$ with high probability over G_n itself. This follows by considering triangles in G and arguing that every edge in G occurs in the same number of triangles up to negligible deviation. But 3 rounds of Sherali-Adams have value at most 2/3 on a triangle. Hence, $\mathrm{lp}_3(G) \leqslant 2/3 + o(1)$.

On the other hand let $\delta = \frac{n^{\varepsilon}}{D}$ where $D = pn$ is the expected degree of G. We observe that $G' = G[V_\delta]$ is exactly distributed like $G' = G_{m,\lambda/m}$ for $m = (n/D)^{1+\varepsilon}$ and $\lambda = m^{\varepsilon}$. Using arguments similar to [ABLT06], one can check that such graphs have girth going to infinity, and for some $M \in \omega(1)$, all subsets size M are $(1 + \eta)$-sparse, where $\eta \in o(1)$. Hence, we can follow the proof as above and use [CMM09] to argue that $G[V_\delta]$ has Sherali-Adams value larger than $1 - o(1)$ for $\omega(1)$ rounds, and therefore picking $r(n)$ sufficiently small concludes the proof sketch. ■

Remark 6.3. We remark that such expansion based arguments can be used to give similar results for subsamples of any Δ-regular graph and in particular for subsamples of the Feige-Schechtman graph.

7 Proof of the main theorem for Unique Games

We now come to the proof of our main theorem —a weak subsampling theorem for strong SDP relaxations of Unique Games. Let us first formalize the notion of a "reasonable" SDP relaxation.

Definition 7.1 (Reasonable SDP relaxation for Unique Games)**.** Let V be a set of n vertices and let $\mathcal{M}$ be a convex subset of the set $\mathcal{M}_1$ defined as

$$\mathcal{M}_1 \overset{\text{def}}{=} \left\{X \in \mathbb{R}^{(V\times[R])\times(V\times[R])} \mid X \succeq 0, \quad \forall i \in V, a \in [R].\ X_{ia,ia} \leqslant 1\right\}.$$

For a unique game on a graph G with vertex set V, we define $\mathrm{sdp}_{\mathcal{M}}(\mathcal{G})$ by

$$\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}) \overset{\text{def}}{=} \min_{X\in\mathcal{M}} \mathop{\mathbb{E}}_{(u,v)\in E} \mathop{\mathbb{E}}_{a\in[R]} \|u_a - v_{\pi_{v\leftarrow u}(a)}\|^2 .$$

We say that $\mathrm{sdp}_{\mathcal{M}}$ is a *reasonable* relaxation for Unique Games if $\mathcal{M}$ is closed under renaming of coordinates and permutation of labels in the sense that

$$\forall F\colon V \to V. \quad \forall \pi_1, \dots, \pi_n\colon [R] \to [R]. \quad \forall X \in \mathcal{M}. \quad (X_{F(i)\pi_i(a),F(j)\pi_j(b)})_{i,j\in V,a,b\in[R]} \in \mathcal{M}\,.$$

Here, the function F is not required to be bijective, but for every $u, v \in V$, $\pi_{v\leftarrow u}$ is a permutation of $[R]$. We also say that $\mathcal{M}$ is *reasonable* if it satisfies the condition above.

For an SDP to be reasonable it is only needed that any set of vectors used for one vertex of the unique game can also be used in any other vertex, even after a permutation of the labels.

The next theorem gives a subsampling result for any reasonable relaxation of Unique Games.

Theorem 7.2 (Main)**.** *Let $\varepsilon > 0$ and let $\mathcal{G}$ be a unique game on a Δ-regular constraint graph. Then, for $\delta = \Delta^{-1} \cdot \mathrm{poly}(1/\varepsilon)$,*

$$\tfrac{1}{9}\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}) - \varepsilon \leqslant \mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) \leqslant \mathrm{sdp}_{\mathcal{M}}(\mathcal{G}) + \varepsilon\,,$$

where $\mathrm{sdp}_{\mathcal{M}}$ is any reasonable relaxation.

The theorem is proven in the next two steps.

7.1 First step: proxy graph theorem via subsampling theorem

For our first step we'll need a special case of our subsampling theorem for semidefinite programs. It shows that under certain regularity conditions subsampling is possible for semidefinite programs that correspond roughly to the SDP of a unique game on a regular graph.

Lemma 7.3. *Let $\varepsilon > 0$ and let $\mathcal{P}$ be a 2-CSP over n variables of the form $\mathcal{P}(x) = \sum_{i,j\in V} b_{ij}P(x_i, x_j)$ where we interpret each variable x_i as a collection of vectors $x_i = (v_{i,a})_{a\in[R]}$ and each pay off function is bounded and of the form $P(x_i, x_j) = \sum_{a,b} d_{a,b}\langle x_{i,a}, x_{j,b}\rangle$. Assume that each $b_{ij} \leqslant 1/\Delta n$ and $\sum_{i=1}^{n} b_{ij} = \Theta(1/n)$ for every j, Then, for $\delta \geqslant \mathrm{poly}(1/\varepsilon)/\Delta$,*

$$\mathbb{E}\,\mathrm{opt}(\mathcal{P}[V_\delta]) = \mathrm{opt}(\mathcal{P}) \pm \varepsilon\,.$$

As shown in Section 4 this lemma can be derived easily from Theorem 4.1. We'll proceed to state and prove our proxy theorem.

Theorem 7.4 (Proxy Theorem)**.** *Let $\mathcal{G}, \mathcal{H}$ be unique games on Δ-dense constraint graphs and suppose*

$$\mathrm{sdp}(\mathcal{G})[X] \geqslant c \cdot \mathrm{sdp}(\mathcal{H})[X] \tag{5}$$

for every SDP solution $X \in \mathcal{M}_1$. Then for $\delta \geqslant \Delta^{-1}\mathrm{poly}(1/\varepsilon)$, we have

$$\mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) \geqslant c \cdot \mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{H}[V_\delta]) - \varepsilon\,.$$

Proof of Theorem 7.4. Consider the 2-CSP instance,

$$\max_{X\in\mathcal{M}_1} \mathcal{P}(X) = c \cdot \mathrm{sdp}(\mathcal{H})[X] - \mathrm{sdp}(\mathcal{G})[X].$$

Note that by our assumption

$$\mathrm{opt}(\mathcal{P}) = \max_{X\in\mathcal{M}_1} \mathcal{P}(X) \leqslant 0\,.$$

Let $A(G)$ and $A(H)$ denote the adjacency matrices of G and H respectively. Since G is Δ-regular and H has degree at least Δ, we know that each entry of $B = cA(H) - A(G)$ is bounded by $O(1/\Delta n)$, whereas each row/column in B sums up to $\Theta(1)$. Hence, the matrix B satisfies the assumption of Lemma 7.3. It remains to check that in $\mathcal{P}$ each pay off function is bounded. This follows from the fact that in both $\mathrm{sdp}(\mathcal{H})$ and $\mathrm{sdp}(\mathcal{G})$ each pay-off function is of the form $\mathbb{E}_{a\in[R]} \|u_a - v_{\pi(v)}\|^2$ and this expression is bounded since each vector has norm at most 1 so that each payoff function is bounded by $O(R)$.

Therefore, by Lemma 7.3,

$$\begin{aligned}
\varepsilon \geqslant \mathbb{E}\,\mathrm{opt}(\mathcal{P}[V_\delta]) &= \mathbb{E} \max_{X\in\mathcal{M}_1} c\,\mathrm{sdp}(\mathcal{H}[V_\delta])[X] - \mathrm{sdp}(\mathcal{G}[V_\delta])[X] \\
&\geqslant \mathbb{E} \max_{X\in\mathcal{M}} c\,\mathrm{sdp}(\mathcal{H}[V_\delta])[X] - \mathrm{sdp}(\mathcal{G}[V_\delta])[X] \qquad (\text{since } \mathcal{M}\subseteq\mathcal{M}_1) \\
&\geqslant \mathbb{E} \max_{X\in\mathcal{M}} c\,\mathrm{sdp}(\mathcal{H}[V_\delta])[X] - \max_{X\in\mathcal{M}} \mathrm{sdp}(\mathcal{G}[V_\delta])[X]\,.
\end{aligned}$$

Hence,

$$\mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) \geqslant c\cdot \mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{H}[V_\delta]) - \varepsilon\,.$$

■

7.2 Second step: proxy graphs for unique games

In this section, we show that taking the "third power" of a unique game results in a useful proxy graph.

Definition 7.5 (Third power of a unique game)**.** For a unique game G we define G^3 to be the unique game defined on the third graph power of the constraint graph. An edge $e = (u, v)$ therefore corresponds to a path (u, u', v', v) in G. The constraint $\pi_{v\leftarrow u}$ on the edge (u, v) is defined as the composition of the three constraints along the path in G, that is

$$\pi_{v\leftarrow u} = \pi_{v\leftarrow v'} \circ \pi_{v'\leftarrow u'} \circ \pi_{u'\leftarrow u}\,. \tag{6}$$

Lemma 7.6. *Let G denote a Δ-regular unique game. Then, for every SDP solution $X \in \mathcal{M}_1$,*

$$\mathrm{sdp}(G)[X] \geqslant 1/9 \cdot \mathrm{sdp}(G^3)[X]\,.$$

Proof. Let $X \in \mathcal{M}_1$ and let (u, v) be an edge in G^3 corresponding to a path (u, u', v', v) in G. Let $a \in [R]$ and put $a' = \pi_{u'\leftarrow u}(a)$, $b' = \pi_{v'\leftarrow u'}(a')$, and $b = \pi_{v\leftarrow v'}(b')$. Note that by definition of G^3, we have $\pi_{v\leftarrow u}(a) = b$. By triangle inequalities

$$\|u_a - v_b\| \leqslant \|u_a - u'_{a'}\| + \|u'_{a'} - v'_{b'}\| + \|v'_{b'} - v_b\|$$

Squaring both sides and taking expectation over $a \in [R]$, we get

$$\mathop{\mathbb{E}}_{a\in[k]} \|u_a - v_b\|^2 \leqslant 3 \mathop{\mathbb{E}}_{a\in[k]} \|u_a - u'_{a'}\|^2 + 3 \mathop{\mathbb{E}}_{a\in[k]} \|u'_{a'} - v'_{b'}\|^2 + 3 \mathop{\mathbb{E}}_{a\in[k]} \|v'_{b'} - v_b\|^2\,.$$

Averaging over edges in G^3, we get

$$\mathop{\mathbb{E}}_{(u,v)\in G^3} \mathop{\mathbb{E}}_{a\in[k]} \|u_a - v_{\pi_{v\leftarrow u}(a)}\|^2 \leqslant 9 \mathop{\mathbb{E}}_{(u,v)\in E} \mathop{\mathbb{E}}_{a\in[k]} \|u_a - v_{\pi_{v\leftarrow u}(a)}\|^2\,.$$

■

Lemma 7.7. *Let $\mathcal{G}$ be a Δ-regular unique game on a graph $G = (V, E)$ and let $\widetilde{\mathcal{G}}$ be the unique game on a graph $\widetilde{G} = (V_\delta, \widetilde{E})$ defined by the edge distribution*

– *sample a random edge (i, j) from G,*

– *choose u and v to be random neighbors of i and j in V_δ (if i or j have no neighbor in V_δ, choose a random vertex in V_δ),*

– *output (u, v) as an edge in $\widetilde{E}$. The constraint on the edge (u, v) is taken to be the composition of the constraints on $(u, i), (i, j), (j, v)$ the same way as in Definition 7.5.*

Then for $\delta > \Delta^{-1}\mathrm{poly}(1/\varepsilon)$,

$$\mathbb{E}\|G^3[V_\delta] - \widetilde{G}\|_{\mathrm{TV}} \leqslant \varepsilon\,.$$

Here, $\|\cdot\|_{\mathrm{TV}}$ denotes statistical distance.

Proof Sketch. If every vertex of G has the same number of neighbors in V_δ, then the two graphs $G^3[V_\delta]$ and G' are identical. For $\delta > \Delta^{-1}\mathrm{poly}(1/\varepsilon)$, the following event happens with probability $1 - \varepsilon$: Most vertices of G (all but an ε fraction) have up to a multiplicative $(1 \pm \varepsilon)$ error the same number of neighbors in V_δ. Conditioned on this event, it is possible to bound $\|G^3[V_\delta] - G'\|_{\mathrm{TV}}$ by $O(\varepsilon)$. Assuming this fact, the lemma follows. The details can be found in Section A. ∎

Lemma 7.8. *Let $\mathcal{G}$ be unique game on a Δ-regular constraint graph. Then for $\delta > \Delta^{-1} \cdot \mathrm{poly}(1/\varepsilon)$ and for any reasonable relaxation $\mathrm{sdp}_{\mathcal{M}}$,*

$$\mathbb{E}\,\mathrm{sdp}_{\mathcal{M}}(\mathcal{G}^3[V_\delta]) \geqslant \mathrm{sdp}_{\mathcal{M}}(\mathcal{G}) - \varepsilon\,.$$

Proof. Let $\widetilde{\mathcal{G}}$ be as in Lemma 7.7 and let $\widetilde{X}$ be an optimal solution for $\mathrm{sdp}_{\mathcal{M}}(\widetilde{\mathcal{G}})$.

Let $\mathcal{F}(V_\delta)$ denote the distribution over mappings $F\colon V \to V_\delta$, where for every vertex $i \in V$, we choose $F(i)$ to be a random neighbor of i in V_δ (and if i has no neighbor in V_δ, we choose $F(i)$ to be a random vertex in V_δ). For convenience, we introduce the notation $N(i, V_\delta)$ for the set of neighbors of i in V_δ (if i has no neighbor in V_δ, we put $N(i, V_\delta) = V_\delta$). For each $F \sim \mathcal{F}(V_\delta)$ we define a *decoded* SDP solution $\mathcal{A}_F(\widetilde{X})$ for $\mathcal{G}$. Specifically, the entry corresponding to $i, j \in V$ and labels $a', b' \in [R]$

1. Let $F(i) = u$ and $F(j) = v$. Assigning the label a' to i forces j to have label $b' = \pi_{j\leftarrow i}(a')$ and hence u and v must have labels $a = \pi_{u\leftarrow i}(a')$ and $b = \pi_{v\leftarrow j}(b')$.

2. Define

$$\mathcal{A}_F(\widetilde{X})_{ia',jb'} := \widetilde{X}_{ua,vb} = \widetilde{X}_{F(i)\pi_{u\leftarrow i}(a),F'(j)\pi_{v\leftarrow j}(\pi_{j\leftarrow i}(a))}\,.$$

Since $\mathcal{M}$ is reasonable (see Definition 7.1), we have $\mathcal{A}_F(\widetilde{X}) \in \mathcal{M}$ for any mapping $F\colon V \to V_\delta$.

We define $\mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X}) := \mathbb{E}_{F\sim\mathcal{F}(V_\delta)}\,\mathcal{A}_F(\widetilde{X})$. Since $\mathcal{M}$ is convex, we also have

$$\mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X}) = \mathop{\mathbb{E}}_{F\sim\mathcal{F}(V_\delta)} \mathcal{A}_F(\widetilde{X}) \in \mathcal{M}\,.$$

We claim that

$$\mathrm{sdp}(\mathcal{G})[\mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X})] = \mathrm{sdp}(\widetilde{\mathcal{G}})[\widetilde{X}]\,.$$

Indeed,

$$
\begin{aligned}
\operatorname{sdp}(\mathcal{P})[\mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X})] &= 2 \mathop{\mathbb{E}}_{ij\sim G} \mathop{\mathbb{E}}_{a'\in[R]} \mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X})_{ia',j\pi_{j\leftarrow i}(a')} \\
&= 2 \mathop{\mathbb{E}}_{ij\sim G} \mathop{\mathbb{E}}_{a'\in[R]} \mathop{\mathbb{E}}_{F\sim\mathcal{F}(V_\delta)} \widetilde{X}_{F(i)\pi_{F(i)\leftarrow i}(a'),F'(j)\pi_{F(j)\leftarrow j}(\pi_{j\leftarrow i}(a'))} \\
&= 2 \mathop{\mathbb{E}}_{u'v'\sim G} \mathop{\mathbb{E}}_{a\in[R]} \mathop{\mathbb{E}}_{u\in N(i,V_\delta)} \mathop{\mathbb{E}}_{v\in N(j,V_\delta)} \widetilde{X}_{u\pi_{u\leftarrow i}(a'),v\pi_{v\leftarrow j}(\pi_{j\leftarrow i}(a'))} \\
&= 2 \mathop{\mathbb{E}}_{u'v'\sim G} \mathop{\mathbb{E}}_{a\in[R]} \mathop{\mathbb{E}}_{u\in N(i,V_\delta)} \mathop{\mathbb{E}}_{v\in N(j,V_\delta)} \widetilde{X}_{ua,v\pi_{v\leftarrow u}(a)} && \text{(using (6))} \\
&= 2 \mathop{\mathbb{E}}_{uv\sim\widetilde{G}} \mathop{\mathbb{E}}_{a\in[R]} \widetilde{X}_{ua,v\pi_{v\leftarrow u}(a)} \\
&= \operatorname{sdp}(\widetilde{\mathcal{G}})[\widetilde{X}]\,. && (7)
\end{aligned}
$$

It follows that

$$
\begin{aligned}
\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}) &\leqslant \operatorname{sdp}_{\mathcal{M}}(\mathcal{G})[\mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X})] && \text{(using } \mathcal{A}_{\mathcal{F}(V_\delta)}(\widetilde{X}) \in \mathcal{M}\text{)} \\
&= \operatorname{sdp}_{\mathcal{M}}(\widetilde{\mathcal{G}})[\widetilde{X}] \qquad \text{(using (7))} \\
&= \operatorname{sdp}_{\mathcal{M}}(\mathcal{G}')\,. && (8)
\end{aligned}
$$

We can now finish the proof of the lemma,

$$
\begin{aligned}
\mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}^3[V_\delta]) &\geqslant \mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\widetilde{\mathcal{G}}) - O(1)\,\mathbb{E}\|G^3[V_\delta] - G'\|_{\mathrm{TV}} \\
&\geqslant \operatorname{sdp}_{\mathcal{M}}(\mathcal{G}) - \varepsilon \qquad \text{(using (8) and Lemma 7.7)}\,.
\end{aligned}
$$

∎

7.3 Putting things together

By combining the previous two steps we can prove Theorem 7.2.

Proof of Theorem 7.2. We need to show that

$$\tfrac{1}{9}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}) - O(\varepsilon) \leqslant \mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) \leqslant \operatorname{sdp}_{\mathcal{M}}(\mathcal{G})\,.$$

The upper bound on $\mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta])$ is easy to show. We consider an optimal solution $X \in \mathcal{M}$ for $\mathcal{G}$. Note that the value of X is preserved for $\mathcal{G}[V_\delta]$ in expectation, i.e.,

$$\mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) \leqslant \mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta])[X] = \operatorname{sdp}_{\mathcal{M}}(\mathcal{G})[X]\,.$$

We combine the lemmas in this section to prove the lower bound. Notice that by Lemma 7.6, we can choose $\mathcal{H} = \mathcal{G}^3$ (for $c = 1/9$) in Lemma 7.4. With this choice of $\mathcal{H}$, we can finish the proof of the theorem,

$$
\begin{aligned}
\mathbb{E}\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}[V_\delta]) &\geqslant \tfrac{1}{9}\cdot\mathbb{E}\operatorname{sdp}_{\mathcal{M}}((\mathcal{G}^3)[V_\delta]) - \varepsilon \qquad \text{(using Lemma 7.4)} \\
&\geqslant \tfrac{1}{9}\cdot\operatorname{sdp}_{\mathcal{M}}(\mathcal{G}) - \tfrac{10}{9}\varepsilon \qquad \text{(using Lemma 7.8)}\,.
\end{aligned}
$$

∎

8 Proof of subsampling theorem

In this section prove our main subsampling theorem for k-CSPs. We will work with the following notion of density.

Definition 8.1 (density)**.** We say that a k-CSP $\mathcal{P}$ is Δ-dense if $|P| \leqslant 1$ for every $P \in \mathcal{P}$ and furthermore for every $r \in [k]$ and fixing of $k-1$ variables $I = (i_1, \ldots, i_{r-1}, *, i_{r+1}, \ldots, i_k)$, we have

$$\frac{\Delta}{c} \leqslant \sum_{P \in \mathcal{P}\colon \operatorname{Var}(P)=I} |P| \leqslant c\Delta \tag{9}$$

for some absolute constant $c > 0$.

Here, Δ is a parameter in $[1, n]$. The larger Δ the denser the instance. In a 2-CSP Δ corresponds to the degree of each variable. In a *dense* k-CSP [AdlVKK03] we have $\Delta = \Theta(n)$.

Theorem 8.2. *Let $\varepsilon > 0, \Delta > 1$. Let $\mathcal{P}$ be a β-dense k-CSP in n variables over an alphabet of size q. Put $\delta \geqslant \varepsilon^{-C}\log(q)/\Delta$ for some absolute constant C. Suppose $U \subseteq [n]$ is chosen uniformly at random so that $|U| = \delta n$. Then,*

$$\left|\mathbb{E}\operatorname{opt}(\mathcal{P}_U) - \operatorname{opt}(\mathcal{P})\right| \leqslant \varepsilon\,. \tag{10}$$

Remark 8.3. In the case $k = 2$, our notion of density reduces to the usual notion of density in a graph. We get the optimal trade-off between density and sample size in that case. When $k > 3$ there are k-CSPs with n^{k-1} constraints that do not allow sparsification. For instance, consider a dense k-CSP in which all constraints share the same variable. We cannot subsample here, since we would likely lose that variable and hence all constraints.

Throughout the proof we will think of k as an absolute constant and consider any function of k as $O(1)$. We will also assume that coordinates in $[n]$ are sampled *with* replacement.

One direction of the theorem is immediate.

Lemma 8.4.

$$\mathbb{E}\left[\max_{x\in[q]^n} \mathcal{P}_U(x)\right] \geqslant \max_{x\in[q]^n} \mathcal{P}(x)\,. \tag{11}$$

Proof. Suppose $x^* \in [q]^n$ maximizes $\mathcal{P}(x)$. Note that $\mathbb{E}\left[\max_{x\in[q]^n} \mathcal{P}_U(x)\right] \geqslant \mathbb{E}\left[\mathcal{P}_U(x^*)\right]$. On the other hand, $\mathbb{E}\,\mathcal{P}_U(x) = \mathcal{P}(x)$. ∎

The other direction requires all the work. We will split it up into two main lemmas. The first lemma shows that the subsampling step is random enough to give a concentration bound for large subsets of $[q]^n$.

Lemma 8.5 (Concentration)**.** *There are constants c_0, c_1 so that for $\delta_0 = \varepsilon^{-c_0}\log(q)/\Delta$ and randomly chosen $U \subseteq [n]$ of size $|U| \geqslant \delta_0 n$ we have that for every subset $\Psi \subseteq [q]^n$ of size $|\Psi| \leqslant \exp(\varepsilon^{c_1}|U|)$,*

$$\left|\mathbb{E}\left[\max_{x\in\Psi} \mathcal{P}_U(x)\right] - \max_{x\in\Psi} \mathcal{P}(x)\right| \leqslant \varepsilon\,. \tag{12}$$

We think of $\delta_0 n$ as the smallest sample size for which we can expect concentration. The previous lemma shows that the maximum value of any fixed set of $\exp(\operatorname{poly}(\varepsilon)|U|)$ assignments is preserved when sampling U of size larger than $\delta_0 n$.

The second main lemma shows that this concentration bound is actually good enough for us. Indeed, the maximum of the subsample turns out to have enough redundancy so that we can find a suitably small set of assignments in $[q]^n$ that captures the optimal value of the subsample up to a small error.

Lemma 8.6 (Structure)**.** *For every constant c there is a constant C and a set of assignments $\Psi \subseteq [q]^n$ of size $|\Psi| \leqslant \exp(\varepsilon^c \delta n)$ where $\delta = \varepsilon^{-C} \log(q)/\Delta$ such that for randomly chosen U of size $|U| = \delta n$, we have*

$$\mathbb{E}\left[\max_{x\in[q]^n} \mathcal{P}_U(x)\right] \leqslant \mathbb{E}\left[\max_{x\in\Psi} \mathcal{P}_U(x)\right] + \varepsilon\,. \tag{13}$$

Together these two lemmas direcly imply the main subsampling theorem as shown next.

Proof of Theorem 8.2. In one direction, let Ψ be the set from Lemma 8.6 which we obtain for $c = c_1$ where c_1 is the constant from Lemma 8.5. Let C be the constant given by Lemma 8.6 for the given choice of c. Then with $\delta = \varepsilon^{-C} \log(q)/\Delta$, we have

$$\begin{aligned}
\mathbb{E}\left[\max_{x\in[q]^n} \mathcal{P}_U(x)\right] &\leqslant \mathbb{E}\left[\max_{x\in\Psi} \mathcal{P}_U(x)\right] + \varepsilon/2 && \text{(by Lemma 8.6)}\\
&\leqslant \mathbb{E}\left[\max_{x\in\Psi} \mathcal{P}(x)\right] + \varepsilon && \text{(by Lemma 8.5)}\\
&\leqslant \mathbb{E}\left[\max_{x\in[q]^n} \mathcal{P}(x)\right] + \varepsilon\,.
\end{aligned}$$

The other direction follows from Lemma 8.4. ∎

8.1 Proof of Concentration Lemma

Fix a vector $x \in [q]^n$. We will first analyze the case where we sample sets $U_1, U_2, \ldots, U_k \subseteq [n]$ independently at random of size $\delta_1 n$ (δ_1 is some parameter that we'll instantiate later) and we keep all constraints whose rth variable is contained in U_r. Later we will be able to conclude the case where $U_1 = U_2 = \cdots = U_k$.

The argument proceeds in k steps. At each step i we restrict the $\mathcal{P}$ to those constraints whose r-th variable is contained in U_r. After each step we perform a pruning operation in which we remove variables whose *influence* has become too large. We then argue that the pruned CSP has the desired concentration properties and moreover that pruning doesn't remove to many constraints in expectation.

Denote by $\mathcal{P}_1$ the CSP obtained from $\mathcal{P}$ by throwing away all predicates whose first variable is not in U_1. Then normalize by a factor δ_1^{-1}, since we expect to remove a δ_1 fraction of the predicates. Formally,

$$\mathcal{P}_1 = \{\delta_1^{-1} P \mid \mathrm{Var}_1(P) \in U_1, P \in \mathcal{P}\}\,.$$

Now, let $\mathrm{Inf}_i(\mathcal{P}_1) = \frac{1}{|\mathcal{P}|} \sum_{P\in\mathcal{P}_1\colon i\in\mathrm{Var}(P)} |P|$ denote the *influence* of variable i in $\mathcal{P}_1$. Let $N = \mathbb{E}\,\mathrm{Inf}_i(\mathcal{P}_1) = O(1/n)$. As mentioned before, we will throw away all predicates that contain a variable whose influence in $\mathcal{P}_1$ has become too large, say, larger than $2N$,

$$\mathcal{P}_1^{\text{prune}} = \{P \in \mathcal{P}_1 \mid \forall i \in \mathrm{Var}(P)\colon \mathrm{Inf}_i(\mathcal{P}_1) \leqslant 2N\}\,.$$

We think of this as the pruning of $\mathcal{P}_1$. Continue this process, inductively, by putting

$$\mathcal{P}_r = \{\delta_1^{-1} P \mid \mathrm{Var}_r(P) \in U_r, P \in \mathcal{P}_{r-1}^{\text{prune}}\}\,,$$

and

$$\mathcal{P}_r^{\text{prune}} = \{P \in \mathcal{P}_r \mid \forall i \in \mathrm{Var} P\colon \mathrm{Inf}_i(\mathcal{P}_r) \leqslant 2^r N\}\,.$$

Here $\mathrm{Inf}_i(\mathcal{P}_r) = \frac{1}{|\mathcal{P}|} \sum_{P\in\mathcal{P}_r\colon i\in\mathrm{Var}(P)} |P|$. Note that $\mathcal{P}_{k-1}^{\text{prune}}$ will still have maximum influence at most $O(1/n)$.

In the following, when we write $\mathcal{P}_r(x)$ we think of it as normalized in the same way we normalize $\mathcal{P}_U(x)$, i.e., by a factor $1/|\mathcal{P}|$.

Lemma 8.7. *For every $x \in [q]^n$,*

$$\Pr\{|\mathcal{P}_k(x) - \mathcal{P}(x)| + t\varepsilon\} \leqslant O(\exp(-t^2\varepsilon^2\delta_1 n)) \tag{14}$$

Proof. The proof proceeds in k steps. At each step we will apply a variant of Azuma's inequality (sometimes referred to as McDiarmid's inequality) given by Lemma D.3.

Specifically, we claim that

$$\Pr\left\{\mathcal{P}_r(x) > \mathcal{P}_{r-1}^{\text{prune}}(x) + t\varepsilon\right\} \leqslant O(\exp(-t^2\varepsilon^2\delta_1 n)), \tag{15}$$

for every $0 < r \leqslant k$. We define the mapping

$$f_r(U_r) = \sum_{P \in \mathcal{P}_{r-1}^{\text{prune}}, \mathrm{Var}_r(P) \in U_r} \delta_1^{-1} P(x) = \mathcal{P}_r$$

where we think of U_r as a tuple $(i_1, \ldots, i_{\delta_1 n})$ each coordinate being an index in $[n]$. Note that

$$\mathbb{E} f_r = \sum_{P \in \mathcal{P}_{r-1}^{\text{prune}}} \delta_1 \delta_1^{-1} P(x) = \mathcal{P}_{r-1}^{\text{prune}}(x).$$

We claim that f_r has Lipschitz constant $O(1/\delta_1 n)$ in the sense that replacing any coordinate $i \in U_r$ by a $i' \in [n]$ can change the function value by at most $O(1/\delta_1 n)$. This is because the influence of each variable in $\mathcal{P}_{r-1}^{\text{prune}}$ is at most $O(1/n)$.

Lemma D.3 then implies

$$\Pr\{f_r > \mathbb{E} f_r + t\varepsilon\} \leqslant \exp\left(-\frac{\Omega(\delta_1^2 n^2 t^2 \varepsilon^2)}{\delta_1 n}\right) = \exp\left(-\Omega(t^2\varepsilon^2\delta_1 n)\right).$$

This is what we claimed in (15). By a union bound, (15) holds for all $r \in [k]$. Hence, we can chain these inequalities together and get

$$\Pr\{|\mathcal{P}_k(x) - \mathcal{P}_1| > t\varepsilon\} \leqslant O(\exp(-t^2\varepsilon^2\delta_1 n)).$$

■

We'd like to argue that at every pruning step only a few predicates get removed and hence $\mathcal{P}_r^{\text{prune}}$ and $\mathcal{P}_r$ are close. Specifically we'd like to show that the influence of i has enough concentration so that it is larger than twice its expectation only with small probability. This directly gives us a bound on the expected amount of pruning. The key observation is the next lemma which shows that the degree of each fixing I of $k-1$ variables is concentrated.

Lemma 8.8. *Assume $\delta_1 \geqslant 1/\varepsilon^2\Delta$, fix $I = (i_1, \ldots, i_{r-1}, *, i_{r+1}, \ldots, i_k)$ and let $Q = \{P \in \mathcal{P} \mid \mathrm{Var}(P) = I\}$. Then,*

$$\mathbb{E}\left|\delta_1\Delta - \sum_{P \in Q:\ \mathrm{Var}_r(P) \in U_r} |P|\right| \leqslant \varepsilon\delta_1\Delta. \tag{16}$$

Proof. By the density condition on $\mathcal{P}$, we have $\sum_{P\in Q}|P| \geqslant \Omega(\Delta)$ and $|P| \leqslant 1$. (In particular, $|Q| \geqslant \Omega(\Delta)$.)

Consider the random variable

$$Z = \sum_{P\in Q, \mathrm{Var}_r(P)\in U_r} |P|\,,$$

which sums the norm of all predicates in Q that are selected by U_r. Let $\mu = \mathbb{E}\, Z = \delta_1\Delta$. We can express Z as a sum of independent variables $Z = \sum_{i=1}^{|U|} Z_i$, where Z_i is the outcome of the i-th sample in U. Since we sampled with replacement, the Z_i's are independent and identically distributed. Every Z_i assumes each value $|P|$ for $P \in Q$ with probability $1/n$. We note that $\mathbb{E}\, Z_i = \frac{1}{n}\sum_{P\in Q}|P| = \Theta(\frac{\Delta}{n})$. Let us compute the fourth moment of $Z - \mathbb{E}\, Z$. First observe that $\mathbb{E}(Z_i - \mathbb{E}\, Z_i)^4 \leqslant O(\mathbb{E}\,|Z_i|^4)$ and

$$\mathbb{E}\,|Z_i|^4 \leqslant \frac{1}{n}\sum_{P\in Q}|P|^4 \leqslant \frac{O(\Delta)}{n}\,.$$

Similarly, for any $i \neq k$:

$$\mathbb{E}(Z_i - \mathbb{E}\, Z_i)^2(Z_k - \mathbb{E}\, Z_k)^2 \leqslant \frac{O(1)}{n^2}\sum_{P,P'\in Q}|P|^2|P'|^2 \leqslant \frac{O(\Delta^2)}{n^2}\,.$$

By independence and the fact that $\mathbb{E}(Z_i - \mathbb{E}\, Z_i) = 0$, we therefore have

$$\begin{aligned}
\mathbb{E}(Z - \mathbb{E}\, Z)^4 &= \sum_i \mathbb{E}(Z_i - \mathbb{E}\, Z_i)^4 + 6\sum_{i\neq k}\mathbb{E}(Z_i - \mathbb{E}\, Z_i)^2\,\mathbb{E}(Z_k - \mathbb{E}\, Z_k)^2 \\
&\leqslant \delta_1 n\cdot\frac{O(\Delta)}{n} + (\delta_1 n)^2\cdot\frac{O(\Delta^2)}{n^2} \\
&= O(\mu^2)\,.
\end{aligned}$$

Thus, by Markov's inequality,

$$\Pr(|Z - \mathbb{E}\, Z| > t) \leqslant \frac{E(Z - \mathbb{E}\, Z)^2}{t^4} \leqslant \frac{O(\mu^2)}{t^4}\,. \tag{17}$$

Therefore we can bound $\mathbb{E}\,|Z - \mathbb{E}\, Z|$ in expectation by integrating (17) over $t \geqslant 1$,

$$\int_{t\geqslant 1} t\cdot\Pr(|Z - \mathbb{E}\, Z| > t\varepsilon\mu)\mathrm{d}t \leqslant \int_{t\geqslant 1} t\cdot\frac{O(\mu^2)}{(t\varepsilon\mu)^4}\mathrm{d}t \leqslant \frac{O(1)}{\varepsilon^4\mu^2}\int_{t\geqslant 1}\frac{1}{t^3}\mathrm{d}t \leqslant \varepsilon \tag{18}$$

for μ larger than c/ε^3, i.e., $\delta_1 \geqslant c/\varepsilon^3\Delta$. ∎

Lemma 8.9. *For every* $0 < r < k$,

$$\frac{1}{|\mathcal{P}|}\,\mathbb{E}\sum_{P\in\mathcal{P}_r\setminus\mathcal{P}_r^{\text{prune}}}|P| \leqslant \varepsilon\,. \tag{19}$$

Proof. We'd like to bound

$$\frac{1}{|\mathcal{P}|}\sum_{P\in\mathcal{P}_r\setminus\mathcal{P}_r^{\text{prune}}}|P| \leqslant 2\sum_{i=1}|\mathrm{Inf}_i(\mathcal{P}_r) - \mathbb{E}\,\mathrm{Inf}_i(\mathcal{P}_r)| \tag{20}$$

in expectation over U_r by $O(\varepsilon)$. We will bound $\mathbb{E}\,|\mathrm{Inf}_i(\mathcal{P}_r) - \mathbb{E}\,\mathrm{Inf}_i(\mathcal{P}_r)|$ for every fixing I that includes variable i and fixes all but the variable in position r. Indeed fix $I = (i_1,\ldots,i_{r-1},*,i_{r+1},\ldots,i_k)$. Let $Q = \{P \in$

$\mathcal{P}_r \mid \mathrm{Var}(P) = I\}$. We will bound the expected gain of influence of variable i in Q. By linearity of expectation this will give us a bound on (20). Let $Z = \sum_{P \in Q | \mathrm{Var}_r(P) \in U_r} |P|$. By Lemma 8.8,

$$|\mathbb{E}\, Z - Z| \leqslant \varepsilon\, \mathbb{E}\, Z\,.$$

Since we have this bound for every fixing and these fixings form a partition of $\mathcal{P}_r$ we find that after renormalization, we have

$$\mathbb{E}\,|\mathrm{Inf}_i(\mathcal{P}_r) - \mathbb{E}\,\mathrm{Inf}_i(\mathcal{P}_r)| \leqslant \frac{\varepsilon}{n}\,.$$

■

Let us denote by $\mathcal{P}'_r$ the CSP that is obtained in the exact same way as $\mathcal{P}_r$ except without the pruning step. In particular, $\mathcal{P}'_k$ is simply the CSP $\mathcal{P}$ in which the r-th variable is restricted to U_r for each $r \in [k]$.

The next corollary summarizes what we have shown so far.

Corollary 8.10. *Let* $\Psi \subseteq [q]^n$ *of size* $\exp(\Omega(\varepsilon^2 \delta_1 n))$. *Then,*

$$\left|\mathbb{E}\left[\max_{x \in \Psi} \mathcal{P}'_k(x)\right] - \max_{x \in \Psi} \mathcal{P}(x)\right| \leqslant \varepsilon. \tag{21}$$

Proof. We first note that $\mathcal{P}_k \subseteq \mathcal{P}'_k$ and we can get

$$\mathbb{E}\, \frac{1}{|\mathcal{P}|} \sum_{P \in \mathcal{P}'_k \setminus \mathcal{P}_k} |P| \leqslant \frac{\varepsilon}{2}\,.$$

This follows from repeated application of Lemma 8.9 (with sufficiently small value of ε) for each $r \in [k]$. In particular this shows that

$$\left|\mathbb{E} \max_{x \in [q]^n} \mathcal{P}'_k(x) - \mathbb{E} \max_{x \in [q]^n} \mathcal{P}_k(x)\right| \leqslant \varepsilon/2\,. \tag{22}$$

On the other hand, by Lemma 8.7 and the union bound over $x \in \Psi$, we get that

$$\left|\mathbb{E}\left[\max_{x \in \Psi} \mathcal{P}_k(x)\right] - \max_{x \in \Psi} \mathcal{P}(x)\right| \leqslant \varepsilon/2\,. \tag{23}$$

Here we used the fact that the probability that the maximum deviates by $t \cdot \varepsilon$ drops of exponentially in t so that we can integrate over $t > 1$ to get a bound on the expectation. Thus,

$$\left|\mathbb{E}\left[\max_{x \in \Psi} \mathcal{P}'_k(x)\right] - \max_{x \in \Psi} \mathcal{P}(x)\right| \leqslant \varepsilon\,,$$

which is what we wanted to show. ■

We are now ready to prove the first main lemma. The proof reduces the general case to the case where each coordinate is subsampled independently as previously dealt with. The idea is to partition the set of variables into m bins and only consider predicates whose variables fall into k distinct bins. The total weight of the remaining predicates can be neglected for large enough m.

Proof of Lemma 8.5 (Concentration). Partition $[n]$ randomly into m bins, i.e., $[n] = S_1 \cup S_2 \cup \cdots \cup S_m$, with $m = \Theta(1/\varepsilon^2)$ (say). Furthermore, let $U_\ell = U \cap S_\ell$ for $\ell \in [m]$. One can show that with probability $1 - o(1)$, for all $r \in [k]$ we have $|U_r| \in [\frac{1}{2m}|U|, \frac{2}{m}|U|]$.

For a given $P \in \mathcal{P}$ the probability that there are $i, j \in \mathrm{Var}(P)$ and $j \in [m]$ so that $i \in S_\ell$ and $j \in S_\ell$ is at most $O(\varepsilon^2)$. Hence, we can throw away all such $P \in \mathcal{P}$ and lose only an $O(\varepsilon^2)$ fraction in expectation.

On the other hand, for every $u \in [m]^k$ with pairwise distinct coordinates, we let $\mathcal{P}^u = \{P \in \mathcal{P} \mid \forall r : \mathrm{Var}_r \in S_{u_r}\}$. For every $\mathcal{P}^u$ we may then apply Corollary 8.10, since $U_{u_1}, U_{u_2}, \ldots, U_{u_r}$ are independently chosen. We apply the corollary with $\varepsilon' = \varepsilon/m^k = \mathrm{poly}(1/\varepsilon)$. This requires us to choose δ_0 large enough as a function of ε so that the previous lemmas (in particular Lemma 8.9) apply to subsets of size $|U_r|$. This allows us to sum the error over all applications of the Corollary for a total error of ε. The Corollary applies to sets Ψ of size $\exp(\Omega(\varepsilon^2|U_r|)) = \exp(\mathrm{poly}(\varepsilon)|U|)$ which is what we needed. ■

8.2 Proof of Structure Lemma

Proof Idea. The main idea is the following. We have a subsample U of size δn. Hence, $\max_{x \in [q]^n} \mathcal{P}_U(x)$ is a maximization problem in δn variables. In particular the maximum is achieved by one of roughly $2^{\delta \log(q) n}$ assignments to these variables. The whole problem is that we need a set of assignments Ψ of size $2^{\mathrm{poly}(\varepsilon)\delta n} \ll 2^{\delta n}$ with the property that one of the assignments in Ψ is near optimal with respect to $\mathcal{P}_U$.

The proof strategy is to design a deterministic algorithm $D(y)$ that is given a *seed* $y \in [q]^S$ where $S \subseteq U$. The algorithm returns an assignment $x = D(y)$ to the variables in U with the guarantee that for some seed $y \in [q]^S$, the induced assignment $x = D(y)$ is near optimal in $\mathcal{P}_U$. An important parameter is the seed length of D, i.e., the size of S. It is also crucial that the algorithm does not know U but only S and $\mathcal{P}_S$. (Otherwise the algorithm could trivially return an optimal assignment for $\mathcal{P}_U$.) Specifically, we want to achieve seed length $|S| \leqslant \mathrm{poly}(\varepsilon)\delta_1 n/\log(q)$. This will suffice for the purpose of our proof, since then we can put $\Psi = \{D(y) : y \in [q]^S\}$. In this case Ψ will be sufficiently small.

The key point in the proof is to choose U so large that for every $x \in [q]^n$ both $\mathcal{P}_U(x)$ and $\mathcal{P}_S(x)$ are a good approximation of $\mathcal{P}(x)$. This fact will be the main reason why we can hope to obtain a near optimal assignment for $\mathcal{P}_U$ by just looking at $\mathcal{P}_S$. We remark that this proof strategy is due to [GGR98]. Formally we will prove the next lemma.

Lemma 8.11. *For every constant c, there is another constant C and a deterministic algorithm $D : [q]^S \to [q]^U$ which extends an assignment to the coordinates S to an assignment to the coordinates in U so that*

$$\mathbb{E} \max_{x \in [q]^S} \mathcal{P}_U(D(x)) \geqslant \mathbb{E} \max_{x \in [q]^U} \mathcal{P}_U(x) - \varepsilon\,.$$

Here the expectation is taken over random $U \subseteq [n]$ of size $\delta n = \varepsilon^{-C}\delta_0 n$ and random subset $S \subseteq U$ of size $|S| \leqslant \varepsilon^{-c}\delta n/\log(q)$.

Once we have this lemma it will be easy to conclude the Structure Lemma. We will next describe our algorithm and then prove Lemma 8.11.

Deterministic greedy algorithm. Let $\alpha = \varepsilon^{c_1}/\log(q)$, the factor by which S needs to be smaller than U. Assume a fixed partition of U into m pairwise disjoint sets $U = U_1 \cup U_2 \cup \cdots \cup U_m$ of equal size. Here m is some parameter that we'll need and determine later. Choose S_ℓ uniformly at random from $U \backslash U_\ell$ of size $|S_\ell| = \frac{\alpha}{m}|U|$ for some parameter α. Let $S = S_1 \cup \cdots \cup S_m$. Note that $|S| = \alpha|U|$. We want $|S_\ell| \geqslant \delta_0 n$ so that the concentration lemma will apply even to the sets S_ℓ. We take $m = \mathrm{poly}(1/\varepsilon)$, e.g., $m = \varepsilon^{-2k}$ will be sufficient. Hence, $\frac{\alpha}{m}$ is some fixed polynomial in ε and this determines the size of U.

The algorithm D works as follows.

Input: $x \in [q]^S$
Output: $z \in [q]^U$
Algorithm:

- For every $i \in S$, we put $z_i = x_i$.
- For every $\ell \in [m]$, do the following: Let $y^* \in [q]^{U_\ell \backslash S}$ denote the partial assignment that maximizes the function $f(y) = \mathcal{P}_{S_\ell}(x[y])$ where $x[y]$ denote the assignment which is equal to y for all coordinates $i \in U_\ell \backslash S$ and equal to x elsewhere. Formally, let

$$y^* = \arg \max_{y \in [q]^{U_\ell \backslash S}} \mathcal{P}_{S_\ell}(x[y]) \tag{24}$$

  and put $z_i = y_i^*$ for every $i \in U_\ell \backslash S$.

This defines an assignment z to all coordinates in U.

Analysis. To analyze our algorithm, let $x^* \in [q]^n$ denote the assignment that maximizes $\max_{x \in [q]^n} \mathcal{P}_U(x)$. Our goal is to define a sequence of "hybrid" assignments $x^0, x^1, \ldots, x^m$ where $x^0 = x^*$ and $x^m = D(y)$ for some $y \in [q]^S$ so that in expectation over U and S, we have

$$\mathcal{P}_U(x^m) \geqslant \mathcal{P}_U(x^0) - \varepsilon \,.$$

The sequence is defined as follows. Let $x^0 = x^*$. Inductively, let x^ℓ for $0 < \ell \leqslant m$ be equal to $x^{\ell-1}$ in all coordinates except U_ℓ. The coordinates U_ℓ are induced from S_ℓ exactly as in our algorithm in equation (24), i.e., for all $i \in U_\ell \backslash S$, we let $x_i^\ell = y_i^*$ where y^* is defined as

$$y^* = \arg \max_{y \in [q]^{U_\ell \backslash S}} \mathcal{P}_{S_\ell}(x^{\ell-1}[y]) \,.$$

We observe that indeed x^m is the generated by D for some $x \in [q]^S$ since all coordinates in x^m are induced by looking only at coordinates in S (though it need not be the case that $x^m = D(x^*)$). Now, denote the error at step ℓ by

$$\mathrm{err}(\ell) = \mathcal{P}_U(x^{\ell-1}) - \mathcal{P}_U(x^\ell) \,.$$

Note that $\mathrm{err}(\ell)$ is a random variable depending on both U and S. The claim now reduces to showing

$$\mathbb{E} \sum_{\ell \in [m]} \mathrm{err}(\ell) \leqslant \varepsilon \,, \tag{25}$$

since by definition $\mathcal{P}_U(x^*) - \mathcal{P}_U(x^m) \leqslant \sum_{\ell \in [m]} \mathrm{err}(\ell)$.

In order to argue (25), it will be convenient to consider

$$\widetilde{\mathrm{err}}(\ell) = \mathcal{P}_{U \backslash U_\ell}(x^{\ell-1}) - \mathcal{P}_{U \backslash U_\ell}(x^\ell) \,.$$

Since we chose m large enough and hence U_ℓ is a sufficiently small fraction of U, it follows that for all $\ell \in [m]$,

$$\sum_{\ell \in [m]} \mathrm{err}(\ell) \leqslant \sum_{\ell \in [m]} \widetilde{\mathrm{err}}(\ell) + \frac{\varepsilon}{2} \,. \tag{26}$$

Let

$$z^* = \arg \max_{y \in [q]^{U_\ell \setminus S}} \mathcal{P}_{U \setminus U_\ell}(x^{\ell-1}[y]) .$$

Note that here we are maximizing over $\mathcal{P}_{U\setminus U_\ell}$ rather than $\mathcal{P}_{S_\ell}$. The following lemma gives us a concrete way of bounding $\widetilde{\mathrm{err}}(\ell)$.

Lemma 8.12.

$$\widetilde{\mathrm{err}}(\ell) \leqslant \left|\mathcal{P}_{U\setminus U_\ell}(x^{\ell-1}[z^*]) - \mathcal{P}_{S_\ell}(x^{\ell-1}[z^*])\right| \tag{27}$$

Proof. If (27) were false, then we would have

$$\mathcal{P}_{S_\ell}(x^{\ell-1}[z^*]) > \mathcal{P}_{S_\ell}(x^{\ell-1}[y^*]) .$$

But this is a contradiction, since we chose y^* as the maximum with respect to $\mathcal{P}_{S_\ell}$. ∎

The next lemma shows that the RHS above is small in expectation. The reason is that S_ℓ is chosen uniformly at random inside $U \setminus U_\ell$. Let $\tilde{x} = x^{\ell-1}[z^*]$. We have $\mathbb{E}\,\mathcal{P}_{S_\ell}(\tilde{x}) = \mathcal{P}_{U\setminus U_\ell}(\tilde{x})$. We only need to argue that the average deviation of $\mathcal{P}_{S_\ell}(\tilde{x})$ from its mean is small. This can be argued directly but it also follows from Lemma 8.5 applied to $\mathcal{P}_{U\setminus U_\ell}$ and $\Psi = \{\tilde{x}\}$. To apply this lemma, we actually need that $\mathcal{P}_{U\setminus U_\ell}$ is sufficiently close to being sufficiently dense. This is true in expectation over U.

Lemma 8.13.

$$\mathbb{E}\left|\mathcal{P}_{U\setminus U_\ell}(\tilde{x}) - \mathcal{P}_{S_\ell}(\tilde{x})\right| \leqslant \frac{\varepsilon}{2m} \tag{28}$$

Proof. As mentioned before we think of $\mathcal{P}_{S_\ell}$ as a subsample of $\mathcal{P}_{U\setminus U_\ell}$. We would like to apply Lemma 8.5 (Concentration) to conclude the claim. However $\mathcal{P}_{U\setminus U_\ell}$ need not satisfy the density condition. However, by Lemma 8.8, $\mathcal{P}_{U\setminus U_\ell}$ does satisfy, for every fixing I of $k-1$ variables,

$$\mathop{\mathbb{E}}_{U}\left|\sum_{P\in\mathcal{P}_{U\setminus U_\ell}, \mathrm{Var}(P)=I} |P| - \delta\Delta\right| \leqslant \varepsilon'\delta\Delta . \tag{29}$$

In other words, every fixing I satisfies the density requirement in expectation. We can therefore treat $\mathcal{P}_{U\setminus U_\ell}$ as a $\delta\Delta$-dense CSP and subsample $S_\ell \subseteq U\setminus U_\ell$ from it. Note that we can take $\delta\Delta = \mathrm{poly}(1/\varepsilon)$ arbitrarily large so that we may subsample an α fraction of the variables of $U\setminus U_\ell$ and expect error $\varepsilon/4m$ in the application of Lemma 8.5. The fact that $\mathcal{P}_{U\setminus U_\ell}$ satisfies only (29) leads to additional approximation errors in the application of Lemma 8.5. By summing (29) over all possible I, we can bound these errors by ε'. Again taking $\delta\Delta$ large enough we can assure $\varepsilon' \leqslant \varepsilon/4m$. Hence, we get a total expected error of $\varepsilon/2m$ which is what we wanted to show. ∎

Combining Lemma 8.12 with Lemma 8.13, we conclude that for every $\ell \in [m]$,

$$\mathbb{E}\,\widetilde{\mathrm{err}}(\ell) \leqslant \frac{\varepsilon}{2m} . \tag{30}$$

Finally,

$$\begin{aligned}
\mathbb{E}\sum_{\ell\in[m]} \mathrm{err}(\ell) &\leqslant \mathbb{E}\sum_{\ell\in[m]} \widetilde{\mathrm{err}}(\ell) + \frac{\varepsilon}{2} && \text{(by (26))}\\
&= \sum_{\ell\in[m]} \mathbb{E}\,\widetilde{\mathrm{err}}(\ell) + \frac{\varepsilon}{2}\\
&\leqslant m\cdot\frac{\varepsilon}{2m} + \frac{\varepsilon}{2} && \text{(using (30))}\\
&= \varepsilon
\end{aligned}$$

We can now complete the proof of the Structure Lemma. We would like to put $\Psi(S) = \{D(x) \mid x \in [q]^S\}$. Then, by Lemma 8.11,

$$\mathbb{E} \max_{x \in \Psi(S)} \mathcal{P}_U(x) \geqslant \mathbb{E} \max_{x \in [q]^U} \mathcal{P}_U(x) - \varepsilon\,. \tag{31}$$

We are not quite done, since the Structure Lemma requires a single fixed set $\Psi(S)$. So far we are choosing S randomly as a subset of U. Hence, the set $\Psi(S)$ that we constructed above depends on the choice of U. To finish the proof we need a single set $\Psi \subseteq [q]^n$ that is independent of the choice of U. This is easy to accomplish from what we have. Simply pick S and U independently and consider $U' = U \cup S$. Since $|S| \leqslant \mathrm{poly}(\varepsilon)|U|$, we can make the difference between $\mathcal{P}_U(x)$ and $\mathcal{P}_{U'}(x)$ negligible for any $x \in [q]^n$. Therefore, we may exchange U' for U in the previous argument so that the choice of S and U is independent. Since (31) is then true in expectation taken over independent S and U, there must also exist a fixed choice of S for which (31) is true in expectation taken over U. But now we may take $\Psi = \Psi(S)$ in order to conclude the proof of the Structure Lemma.

References

[ABLT06] S. Arora, B. Bollobás, L. Lovász, and I. Tourlakis. Proving integrality gaps without knowing the linear program. *Theory of Computing*, 2(1):19–51, 2006.

[AdlVKK03] N. Alon, W. F. de la Vega, R. Kannan, and M. Karpinski. Random sampling and approximation of MAX-CSPs. *J. Comput. Syst. Sci.*, 67(2):212–243, 2003.

[AHK06] S. Arora, E. Hazan, and S. Kale. A fast random sampling algorithm for sparsifying matrices. In *Proc.* 10*th APPROX*, pages 272–279. Springer, 2006.

[AKK+08] S. Arora, S. Khot, A. Kolla, D. Steurer, M. Tulsiani, and N. Vishnoi. Unique games on expanding constraints graphs are easy. In *Proc.* 40*th STOC*. ACM, 2008.

[AM07] D. Achlioptas and F. McSherry. Fast computation of low-rank matrix approximations. *J. ACM*, 54(2), 2007.

[BSS09] J. D. Batson, D. A. Spielman, and N. Srivastava. Twice-Ramanujan sparsifiers. In *Proc.* 41*st STOC*, pages 255–262. ACM, 2009.

[CMM09] M. Charikar, K. Makarychev, and Y. Makarychev. Integrality gaps for sherali-adams relaxations. In *Proc.* 41*st STOC*. ACM, 2009.

[Din07] I. Dinur. The PCP theorem by gap amplification. *J. ACM*, 54(3):12, 2007.

[dlVKM07] W. F. de la Vega and C. Kenyon-Mathieu. Linear programming relaxations of maxcut. In N. Bansal, K. Pruhs, and C. Stein, editors, *SODA*, pages 53–61. SIAM, 2007.

[Fei02] U. Feige. Relations between average case complexity and approximation complexity. In *STOC*, pages 534–543, 2002.

[FS02] U. Feige and G. Schechtman. On the optimality of the random hyperplane rounding technique for max cut. *Random Struct. Algorithms*, 20(3):403–440, 2002.

[GGR98] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998.

[GW94] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995. Preliminary version in STOC'94.

[KKMO04] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM J. Comput*, 37(1):319–357, 2007. Preliminary version in FOCS' 04.

[MOO05] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low in.uences invariance and optimality. In *Proc.* 46*th FOCS*, pages 21–30. IEEE, 2005.

[Rag08] P. Raghavendra. Optimal algorithms and inapproximability results for every csp? In *Proc.* 40*th STOC*, pages 245–254. ACM, 2008.

[Ron00] D. Ron. Property testing. In *Handbook of Randomized Computing, Vol. II*, pages 597–649. Kluwer Academic Publishers, 2000.

[RS09] P. Raghavendra and D. Steurer. How to round any csp. In *Proc.* 50*th FOCS*, pages 586–594. IEEE, 2009.

[Rub06] R. Rubinfeld. Sublinear time algorithms. In *Proc. International Congress of Mathematicians*, volume 3, pages 1095–1110. European Math. Soc., 2006.

[RV07] M. Rudelson and R. Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *J. ACM*, 54(4), 2007.

[SA90] 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.*, 3(3):411–430, 1990.

[Sch08] G. Schoenebeck. Linear level Lasserre lower bounds for certain k-CSPs. In *FOCS*, pages 593–602. IEEE Computer Society, 2008.

[SS08] D. A. Spielman and N. Srivastava. Graph sparsification by effective resistances. In *Proc.* 40*th STOC*. ACM, 2008.

[ST04] D. A. Spielman and S.-H. Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proc.* 36*th STOC*, pages 81–90. ACM, 2004.

[Ste10] D. Steurer. Fast sdp algorithms for constraint satisfaction problems. In *Proc.* 21*st ACM-SIAM SODA*, 2010.

[Tul09] M. Tulsiani. CSP gaps and reductions in the Lasserre hierarchy. In *Proc.* 41*st STOC 2009*, pages 303–312. ACM, 2009.

A Edge distribution of subsample of the third power

In this section we compare the edge distribution of the subsample of G^3 to a somewhat nicer distribution. This step was needed in Lemma 7.7. In the following let $G = (V, E)$ be a Δ-regular graph and $\delta \geqslant \mathrm{poly}(\varepsilon^{-1})\Delta^{-1}$. Further denote $W = V_\delta$.

Lemma A.1. *Let D_1 denote the uniform distribution over edges in $G^3[W]$. Let D_2 denote the distribution obtained as follows:*

1. *Pick a random edge $(v, v') \in E$.*
2. *Choose uniformly at random $w \in N_W(v)$ and $w' \in N_W(v')$.*
3. *Output (w, w').*

Then,

$$\mathop{\mathbb{E}}_{W} [\mathrm{TV}(D_1, D_2)] \leqslant \varepsilon\,.$$

Here and in the following $\mathrm{TV}(D_1, D_2)$ *denote the total variation distance between the two distributions D_1 and D_2.*

Proof. Let us compare the following two distributions:

P_1 : Pick a uniformly random path $p = (w, v, v', w')$ from the set of all paths of length 3 in G which have $w, w' \in W$.

P_2 : Pick a random edge $v, v' \in E$ and random neighbors $w \in N_W(v), w' \in N_W(v')$ and consider the path (w, v, v', w').

Notice that it suffices to bound the statistical distance between P_1 and P_2. This is because D_1 is just the marginal distribution of P_1 on the endpoints of the path (w, w'). Likewise D_2 is the marginal distribution of P_2 on (w, w').

Now, let $p = (w, v, v', w')$ denote any path of length 3 in G so that $w, w' \in W$. Let N denote the number of such paths. Note that $\mathbb{E}\, N = \delta^2\Delta^3 n$. Let us now compare the probability of this path under the two distributions. For P_1 we get

$$P_1(p) = \frac{1}{N}\,.$$

On the other hand, under P_2,

$$P_2(p) = \frac{1}{|N_W(v)|} \cdot \frac{1}{\Delta n} \cdot \frac{1}{|N_W(v')|}.$$

Note that for every $v \in V$, we have $\mathbb{E}\,|N_W(v)| = \delta\Delta$. It now suffices to argue the bound

$$\mathbb{E}\,[\mathrm{TV}(P_1, P_2)] = \mathbb{E}\,\frac{1}{2}\sum_p \left|\frac{1}{N} - \frac{1}{|N_W(v)||N_W(v')|\Delta n}\right| \leqslant \varepsilon. \tag{32}$$

Let us call a path $p = (w, v, v', w')$ *good* if

$$\frac{1}{|N_W(v)| \cdot |N_W(v')|} = \frac{1 \pm \varepsilon'}{\delta^2\Delta^2}.$$

Later we will choose $\varepsilon' = \Omega(\varepsilon)$ to be sufficiently small, say, $\varepsilon' = \varepsilon/100$. We need the following simple concentration bounds.

Claim A.2. *With probability $1 - \varepsilon'$ over the choice of W, we have*

1. $N^{-1} = (1 \pm \varepsilon')/(\delta^2\Delta^3 n)\,.$

2. *The fraction of bad paths is less than* $1/O(\varepsilon'^5(\delta\Delta)^3)$.

Proof. The first claim follows from Lemma D.1. Regarding the second claim, it is not hard to show for every v, v' that

$$\Pr\left\{\frac{1}{|N_W(v)||N_W(v')|} \notin \frac{1 \pm \varepsilon'}{\delta^2\Delta^2}\right\} \leqslant \frac{1}{O(\varepsilon'^4(\delta\Delta)^3)} .$$

This can be shown by computing the fourth moment $\mathbb{E}(|N_W(v)| - \delta\Delta)^4$ and bounding the probability of a factor $1 + \alpha$ deviation of $|N_W(v)|$ from its mean for small enough $\alpha = \Omega(\varepsilon')$. This argument shows that the expected number of bad paths is at most $1/O(\varepsilon'^4(\delta\Delta)^3)$ and the claim is completed by applying Markov's inequality. ■

Given this claim, we can finish the proof of the lemma. Indeed letting Q denote the set of good paths, we have with probability $1 - \varepsilon'$,

$$\begin{aligned}
\sum_p \left|\frac{1}{N} - \frac{1}{|N_W(v)||N_W(v')|\Delta n}\right| &\leqslant \sum_{p\in Q} \frac{2\varepsilon'}{\delta^2\Delta^3 n} + \sum_{p\notin Q} \frac{1}{\Delta n} \\
&\leqslant 2\varepsilon' + \frac{N}{O(\varepsilon'^5(\delta\Delta)^3)} \cdot \frac{1}{\Delta n} \\
&= 2\varepsilon' + \frac{1}{O(\varepsilon'^5\delta\Delta)} \cdot \frac{N}{\delta^2\Delta^3 n} \\
&\leqslant O(\varepsilon') .
\end{aligned}$$

In the first inequality we used the fact that $|N_W(v)| \geqslant 1$ for any existing path and hence the term $1/|N_W(v)||N_W(v')|\Delta n$ is never larger than $1/\Delta n$. In the last step we used that we may choose $\delta\Delta \geqslant C\varepsilon'^{-5}$ for sufficiently large constant $C > 0$, and that $N \leqslant (1 + \varepsilon')\delta^2\Delta^3 n$. Hence,

$$\mathbb{E}\,\mathrm{TV}(P_1, P_2) \leqslant (1 - \varepsilon')O(\varepsilon') + \varepsilon' \leqslant \varepsilon .$$ ■

B Details on random geometric graphs 5

In this section we will in the details that were left out in Section 5. We start with the proof of Lemma 5.3.

Lemma 5.3 (Restated)**.** $\mathrm{sdp}_3(G_\gamma) \leqslant 1 - \Omega(\sqrt{\gamma})$.

The proof works as follows. First, triangle inequalities are known to imply the odd cycle constraints which means that an SDP with triangle inequalities on an odd cycle of length k has value at most (and, in fact, equal to) $1 - 1/k$.

Lemma B.1. *Let C be an odd cycle of length k. Then,* $\mathrm{sdp}_3(C) \leqslant 1 - 1/k$.

Second, it follows that if a graph G can be covered uniformly by odd cycles of length k, then its sdp_3-value can be at most $1 - 1/k$.

Lemma B.2. *Let $G = (V, E)$ be a (possibly infinite) graph. Suppose there exists a distribution $\mathcal{C}$ over odd cycles of length k for some fixed number k such that the marginal distribution on each edge of a random cycle from $\mathcal{C}$ has statistical distance ε to the uniform distribution over edges in G. Then,* $\mathrm{sdp}_3(G) \leqslant 1 - 1/k + \varepsilon$.

Proof. By our assumption we have that for every embedding $f\colon V \to B$,

$$\mathop{\mathbb{E}}_{(u,v)\sim E} \tfrac{1}{4}\|f(u) - f(v)\|^2 \leqslant \mathop{\mathbb{E}}_{C\sim \mathcal{C}} \mathop{\mathbb{E}}_{(u,v)\sim C} \tfrac{1}{4}\|f(u) - f(v)\|^2 + \varepsilon\,.$$

But we know, by Lemma B.1, that for every $f\colon V \to B$, satisfying the triangle inequalities,

$$\mathop{\mathbb{E}}_{(u,v)\sim C} \frac{1}{4}\|f(u) - f(v)\|^2 \leqslant 1 - \tfrac{1}{k}\,.$$

Hence,

$$\mathop{\mathbb{E}}_{(u,v)\sim E} \frac{1}{4}\|f(u) - f(v)\|^2 \leqslant 1 - \tfrac{1}{k} + \varepsilon\,.$$

■

We will next see that the sphere graph can by uniformly covered by odd cycles of length $O(1/\sqrt{\gamma})$. We begin with the following simple observation.

Lemma B.3. *For every $l \in [1-\gamma, 1-\gamma/2]$, there exists an odd cycle, denoted $C_l = (v_1, \ldots, v_k)$, in G_γ of length $k = O(\sqrt{\gamma})$ such that $\frac{1}{4}\|v_i - v_{i+1}\|^2 = l$ for all $i \in \{1, \ldots, k-1\}$.*

Proof sketch. Pick an arbitrary great circle around the sphere and place the vertices $v_1, \ldots, v_k$ equally spaced along this circle. For $k = O(\sqrt{\gamma})$ vertices, we can accomplish the Euclidean distance between two consecutive vertices is less than, say, $\sqrt{\gamma}/10$. Now connect each vertex v on the circle to the unique vertex w which maximized $\|v-w\|^2$. This creates an odd cycle and, by our previous observation, it follows that $\frac{1}{4}\|v-w\|^2 \geqslant 1-\gamma$. Now we can make $\frac{1}{4}\|v-w\|^2 = l$ be walking along the cycle and moving vertices in a direction orthogonal to the plane defined by the circle until all edges have length l. ■

Lemma B.4. *Let $\gamma > 0$ and let S^{d-1} be the sphere. There exists a distribution $\mathcal{C}$ over odd cycles $C = (v_1, \ldots, v_k)$ for some $k \leqslant \frac{10\pi}{\sqrt{\gamma}}$ such that for all i, the marginal distribution of (v_i, v_{i+1}) has statistical distance $o(1)$ to the uniform distribution over edges in G_γ (as $d \to \infty$).*

Proof. We will describe the distribution $\mathcal{C}$ as follows:

1. Pick a random edge $e = (u, v) \in E$ from G_γ.

2. Let $l = \frac{1}{4}\|u - v\|^2$. If $l \leqslant 1 - \gamma/2$, let $C_l = (v_1, v_2, \ldots, v_k)$ denote the odd cycle given by Lemma B.3. If $l \geqslant 1 - \gamma/2$, declare "failure".

3. If the previous step succeeded, pick a random rotation R and output $RC = (Rv_1, Rv_2, \ldots, Rv_k)$.

We claim that if the second step succeeds, then indeed every marginal (Rv_i, Rv_{i+1}) is distributed like a uniformly random edge. This is (1) because (u, v) was chosen to be a uniformly random edge and (2) (Rv_i, Rv_{i+1}) is a random rotation of (u, v) and hence, by spherical symmetry, is equally likely to be any edge in E that has the same length as (u, v).

On the other hand, by measure concentration, with probability $1 - \exp(-\Omega(d))$, we have that $\frac{1}{4}\|u - v\|^2 \in [1-\gamma, 1-\gamma/2]$. This completes the claim since the probability of failure only introduces $o(1)$ statistical distance. ■

In this section we give some details on how to obtain a dense discretization of the Feige-Schechtman graph.

Lemma 5.4 (Restated)**.** *Fix $\gamma \in [0,1], d \in \mathbb{N}$. Then, there exists an $n_0(d,\gamma) \in \mathbb{N}$ so that if we pick $V \subseteq S^{d-1}$ uniformly at random with $|V| \geqslant n_0$, then the induced subgraph $G_\gamma[V]$ satisfies (1)* $\mathrm{opt}(G_\gamma[V]) = 1 - \Theta(\sqrt{\gamma})$, *and (2)*

$\mathrm{sdp}_3(G_\gamma[V]) = 1 - \Theta(\sqrt{\gamma})$.

Proof sketch. The first claim is shown in [FS02]. For the second claim, let us decompose $\mathbb{S}^{d-1}$ into equal volume cells of diameter at most ε. Here, ε is a parameter that we will later take to be very small, say, $\varepsilon \leqslant d/100$. Now pick enough vectors $V \subseteq \mathbb{S}^{d-1}$ uniformly at random such that with probability at least $1-\varepsilon$ every two cells have the same number of vectors up to a factor of $1 \pm \varepsilon$ in it.

We need to show that $\mathrm{sdp}_3(G_\gamma[V]) \leqslant 1 - \Omega(\sqrt{\gamma})$. To this end we first consider a related graph G', which has the same vertex set as G but different edges. A random edge in G' is defined by the following process: Pick first a random edge on the continuous sphere, then for each endpoint pick a random vertex in the equal volume cell containing the endpoint. Finally, normalize the edges such that the total edge weight is the same as in G.

We can use the distribution over odd cycles given by Lemma B.4 in order to get a distribution for the graph G' as follows: Pick the cycle and map each point to a vertex in the corresponding cell. The resulting marginal distributions will be uniform in G'. Thus, $\mathrm{sdp}_3(G') = 1 - \Theta(\sqrt{\gamma})$.

Finally, we will show that $\mathbb{E}\|L(G') - L(G_\gamma[V])\|_{\mathrm{TV}}$ tends to zero with ε. That is, the two distributions have statistical distance tending to zero. This also shows that for sufficiently small ε, the semidefinite programs also have approximately the same value. Now to argue the above point, consider the process of picking a random edge. Consider first the case that in G', the two cells containing the chosen points have exactly the expected number of vectors in them, and furthermore, suppose that the two cells are *good* in the sense that either none of the vertices in them share an edge or all pairs of vertices between the two cells share an edge in G. In this case, the edges in G going between these two cells have exactly the same probability as under G'.

The first assumption is close enough to the truth, since the number of vertices in different cells differ by at most a factor of $1 \pm \varepsilon$, For the second assumption it suffices to pick ε small enough so that a cap of radius r has the same volume as a cap of radius $r \pm \varepsilon$ up to a factor of $1 \pm o(1)$. This happens for, say, $\varepsilon \ll 1/d$. This will guarantee that the number of bad pairs of cells is small. This argument can be found in [FS02]. ■

C Subsampling edges

In this section, we will briefly discuss the analogue of our main theorem in the setting where we sample a fraction of the edges in G at random so that the expected degree in G is constant. Here, $G = (V, E)$ will always denote a Δ-regular graph on n vertices. Our proof in the case of edge subsampling is much simpler. As it turns out it suffices to bound the cut norm between the original graph and its subsample and to argue that the SDP value is a Lipschitz function of the cut norm. The latter fact is a consequence of Grothendieck's inequality.

We let $E_\delta \subseteq E$ denote a random subset of E of size $\delta|E|$. We'll overload notation slightly by using $G[E_\delta]$ for the graph G restricted to the edge set E_δ.

Definition C.1. The cut norm of a real valued $n \times n$ matrix A is defined as

$$\|A\|_C = \max_{U,V\subseteq[n]} \left| \sum_{i\in U, j\in V} a_{ij} \right| . \tag{33}$$

It is known that the cut norm is within constant factors of the norm

$$\|A\|_{\infty\mapsto 1} = \max_{x_i, y_j \in \{-1,1\}} \sum_{i,j\in[n]} a_{ij} x_i y_j \,. \tag{34}$$

A natural semidefinite relaxation of (34) replaces every pair x_i, y_j by two unit vectors u_i, v_j, i.e.,

$$\mathrm{sdp}_C(A) = \max_{\|u_i\|=\|v_i\|=1} a_{ij}\langle u_i, v_j\rangle. \tag{35}$$

A theorem of Grothendieck bounds the gap between the cut norm and its relaxation by a multiplicative constant (the Grothendieck constant).

Theorem C.2. *There is a constant K_G (known to be less than 1.8) such that $\mathrm{sdp}_C(A) \leqslant K_G\|A\|_{\infty\mapsto 1}$.*

The next lemma shows that the cut norm between a graph and its subsample is small.

Lemma C.3. *Let $\delta \geqslant c\varepsilon^{-2}\Delta^{-1}$. Then, $\mathbb{E}\left\|A(G) - \delta^{-1}A(G[E_\delta])\right\|_{\infty\mapsto 1} \leqslant \varepsilon$.*

Proof. We can show that with probability $1 - e^{\Omega(n)}$, $|\langle x, Ay\rangle - \delta^{-1}\langle x, A'y\rangle| \leqslant \varepsilon$ simultaneously for all $x, y \in \{-1,1\}^n$. The proof follows from Hoeffding's bound and the union bound. The details are straightforward and therefore omitted from this paper. ∎

Similarly the following lemma can be shown.

Lemma C.4. *Let $\delta \geqslant c\varepsilon^{-2}\Delta^{-1}$. Then, $\mathbb{E}\left\|D(G) - \delta^{-1}D(G[E_\delta])\right\|_{\infty\to 1} \leqslant \varepsilon$.*

The previous two lemmas showed that the expected difference in cut norm between the graph G and its edge subsample $G[E_\delta]$ is small.

Corollary C.5. *For $\delta \geqslant c\varepsilon^{-2}\Delta^{-1}$, we have $\mathbb{E}\,\|L(G) - \delta^{-1}L(G[E_\delta])\|_C \leqslant \varepsilon$.*

It turns out that bounding the difference in cut norm is sufficient for bounding the difference in SDP values.

Lemma C.6. *Let G and G' be any two graphs on n vertices. Let $\mathcal{M} \subseteq \mathcal{M}_2$ (see Definition 7.1) be any set of positive semidefinite $n \times n$ matrices. Suppose $\|L(G) - L(G')\|_C \leqslant t$. Then,*

$$|\mathrm{sdp}_{\mathcal{M}}(G) - \mathrm{sdp}_{\mathcal{M}}(G')| \leqslant O(t)\,.$$

Proof.

$$\begin{aligned}
\left|\mathrm{sdp}_{\mathcal{M}}(G) - \mathrm{sdp}_{\mathcal{M}}(G')\right| &\leqslant \left|\max_{X\in\mathcal{M}_2} (L(G) - L(G')) \bullet X\right| \\
&\leqslant O(1)\cdot\|L(G) - L(G')\|_C && \text{(by Theorem C.2)} \\
&\leqslant O(t)\,.
\end{aligned}$$

∎

Corollary C.7. *Let G denote a Δ-regular graph and let $\delta \geqslant \mathrm{poly}(1/\varepsilon)\Delta^{-1}$. Then,*

$$\mathbb{E}\left|\mathrm{sdp}_{\mathcal{M}}(G) - \mathrm{sdp}_{\mathcal{M}}(G[E_\delta])\right| \leqslant \varepsilon \tag{36}$$

for any $\mathcal{M} \subseteq \mathcal{M}_2$.

Negative results for linear programs. We remark that using the approach of [CMM09] one can obtain strong and general results ruling out subsampling for linear programs.

Theorem C.8. *Let $\varepsilon, \lambda > 0$. Suppose G is a Δ-regular graph with $\Delta > n^{\theta}$. Then with high probability over $G' = G[E_{\lambda/\Delta}]$, after removing $o(n)$ vertices, $\mathrm{lp}_r(G') \geqslant 1 - \varepsilon$ for $r = n^{\alpha}$ where $\alpha(1/\varepsilon, 1/\theta, \lambda)$ tends to zero as any of its arguments grows.*

The proof follows by arguing that $G[E_{\lambda/\Delta}]$ has sufficient small set expansion so that [CMM09] applies. Details are omitted.

D Deviation bounds

Deviation bounds for submatrices. The following general lemma is useful in bounding the deviation of expressions $\sum_{i,j\in S} |a_{ij}|$ when S denotes a random subset of $[n]$ and A is a $n \times n$ matrix.

Lemma D.1. *Let A denote a symmetric $n \times n$ matrix such that $a_{ii} = 0$ for all $i \in [n]$. Suppose there is some $\beta > 0$ such that $|a_{ij}| \leqslant \beta$ for all $i, j \in [n]$ and $\sum_j |a_{ij}| \leqslant 1$ for all i. Now, let $S \subseteq [n]$ denote a random subset of $[n]$ of size δn for some $\delta > \beta$. Then, for all $\varepsilon > 0$,*

$$\mathbb{P}\mathrm{r}\left(\left|\delta^{-2} \sum_{i,j\in S} a_{ij} - \sum_{i,j\in[n]} a_{ij}\right| > \varepsilon n\right) \leqslant \frac{O(1)}{\varepsilon^2 \delta n}\,. \tag{37}$$

Proof. Denote by X_{ij} the random variable which is equal to a_{ij} when both $i \in S$ and $j \in S$ and is zero otherwise. Let $\mu_{ij} = \mathbb{E}\, X_{ij} = \delta^2 a_{ij}$. Putting $X = \sum_{i,j\in[n]} X_{ij}$ and $\mu = \mathbb{E}\, X$ we will compute the variance of X. The key fact that we will use is that the selection of i, j and k, l is independent unless either $i = k$ or $j = l$. Pairs where neither is the case will not contribute to the variance. More precisely,

$$\begin{aligned}
\mathbb{E}\,(X-\mu)^2 &= \mathbb{E}\left(\sum_{ij} X_{ij} - \mu_{ij}\right)^2 \\
&= \mathbb{E}\left[\sum_{i,j,k,l} (X_{ij} - \mu_{ij})(X_{kl} - \mu_{kl})\right] \\
&= \sum_{ij} \mathbb{E}\left(X_{ij} - \mu_{ij}\right)^2 + \sum_{ijk} \mathbb{E}\left(X_{ij} - \mu_{ij}\right)\left(X_{kj} - \mu_{kj}\right) \\
&= \sum_{ij} O(\delta^2) a_{ij}^2 + \sum_{ijk} O(\delta^3) a_{ij} a_{kj}\,.
\end{aligned}$$

At this point notice that $\sum_{ij} a_{ij}^2$ is maximized when in every row we have $1/\beta$ entries of magnitude β in which case the expression evaluates to $\frac{1}{\beta}\beta^2 n = \beta n$. Likewise the second expression $\sum_{ijk} a_{ij} a_{kj}$ is maximized when in every column $j \in [n]$ we have $1/\beta$ nonzero entries of magnitude β. In this case the expression is $(1/\beta)^2 \beta^2 n = n$. Hence,

$$\sigma^2 = \mathbb{E}\,(X-\mu)^2 \leqslant O(\delta^2 \beta n) + O(\delta^3 n) \leqslant O(\delta^3 n)\,, \tag{38}$$

where we used that $\delta > \beta$. Hence by Chebyshev's inequality,

$$\mathbb{P}\mathrm{r}(|X - \mu| \geqslant \varepsilon \delta^2 n) \leqslant \frac{\sigma^2}{\varepsilon^2 \delta^4 n^2} = \frac{O(1)}{\varepsilon^2 \delta n}\,.$$

This is what we claimed up to scaling. ∎

In the proof of the proxy graph theorem we used the following simple observation relating the Laplacian of a subsample $L(G[V_\delta])$ to the corresponding principal submatrix of the Laplacian $L(G)_{V_\delta}$.

Lemma D.2. *Let G be a Δ-regular graph and let H be a graph of degree at least Δ. Let $\delta \geqslant \mathrm{poly}(1/\varepsilon)\Delta^{-1}$. Then,*

$$\mathbb{E}\|L(G[V_\delta]) - L(G)_{V_\delta}\|_{\mathrm{TV}} \leqslant \varepsilon\,.$$

Proof. By inspection of the two matrices we see that the difference in the entries of the matrix is due to irregularities in the degrees of $G[V_\delta]$. Specifically, the matrix $L(G)_{V_\delta}$ has diagonal entries equal to $1/\delta n$. On the other hand, the i-th diagonal entry of $L(G[V_\delta])$, call it d_i, is equal to $\delta^{-1}\sum_{j\in V_\delta} a_{ij}$. We have that $\mathbb{E}\, d_i = \frac{1}{\delta n}$ and we claim,

$$\sum_{i\in V_\delta} \mathbb{E}\left|d_i - \frac{1}{\delta n}\right| \leqslant \varepsilon\,.$$

This can be derived from Lemma D.1. ■

McDiarmid's inequality. We also needed McDiarmid's large deviation bound (sometimes called Azuma's inequality).

Lemma D.3. *Let $X_1, \ldots, X_m$ be independent random variables all taking values in the set $\mathcal{X}$. Further, let $f\colon \mathcal{X}^m \to \mathbb{R}$ be a function of $X_1, \ldots, X_m$ that satisfies for all $i, x_1, x_2, \ldots, x_m, x_i' \in \mathcal{X}$,*

$$|f(x_1, \ldots, x_i, \ldots, x_m) - f(x_1, \ldots, x_i', \ldots, x_m)| \leqslant c_i.$$

Then, for all $t > 0$,

$$\mathbb{P}\mathrm{r}\{|f - \mathbb{E}[f]| \geqslant t\} \leqslant 2\exp\left(\frac{-2t^2}{\sum_{i=1}^m c_i^2}\right).$$

ECCC ISSN 1433-8092

http://eccc.hpi-web.de