

An optimal SDP algorithm for Max-Cut, and equally optimal Long Code tests

Ryan O'Donnell
Carnegie Mellon University
odonnell@cs.cmu.edu

Yi Wu
Carnegie Mellon University
yiwu@andrew.cmu.edu

November 20, 2007

Abstract

Let G be an undirected graph for which the standard Max-Cut SDP relaxation achieves at least a c fraction of the total edge weight, $\frac{1}{2} \leq c \leq 1$. If the actual optimal cut for G is at most an s fraction of the total edge weight, we say that (c, s) is an *SDP gap*. We define the *SDP gap curve* $\text{Gap}_{\text{SDP}} : [\frac{1}{2}, 1] \rightarrow [\frac{1}{2}, 1]$ by

$$\text{Gap}_{\text{SDP}}(c) = \inf\{s : (c, s) \text{ is an SDP gap}\}.$$

In this paper we complete a long line of work [DP93b, DP93a, GW95, Zwi99, FS02, FL06, CW04, KO06] by determining the entire SDP gap curve; we show $\text{Gap}_{\text{SDP}}(c) = S(c)$ for a certain explicit (but complicated to state) function S . In particular, our lower bound $\text{Gap}_{\text{SDP}}(c) \geq S(c)$ is proved via a polynomial-time ‘RPR²’ algorithm. Thus we have given an efficient, optimal SDP-rounding algorithm for Max-Cut. The fact that it is RPR² confirms a conjecture of Feige and Langberg [FL06].

We also describe and analyze the tight connection between SDP gaps and Long Code tests (and the constructions of [Kar99, AS00, ASZ02]). Using this connection, we give optimal Long Code tests for Max-Cut. Combining these with results implicit in [KKMO07, KV05] and ideas from [FS02], we derive the following conclusions:

- The Max-Cut SDP gap curve subject to triangle inequalities is also given by $S(c)$.
- No RPR² algorithm can be guaranteed to find cuts of value larger than $S(c)$ in graphs where the optimal cut is c . (Contrast this with the fact that in the graphs exhibiting the c vs. $S(c)$ SDP gap, our RPR² algorithm actually finds the optimal cut.)
- Further, no polynomial-time algorithm of *any* kind can have such a guarantee, assuming $P \neq NP$ and the Unique Games Conjecture.

1 Introduction

Given an undirected graph $G = (V, E)$, the Max-Cut problem asks for a partition of the vertices into two sets so as to maximize the number of edges connecting the two sets. It is one of the classic NP-complete problems from Karp's list of 21 [Kar72] and is arguably the simplest NP-hard problem, being the binary constraint satisfaction problem with only ' \neq ' constraints. The Max-Cut problem has applications from VLSI to statistical physics [BGJR88] and has attracted a tremendous amount of interest both in theory and in practice. For a survey, see Poljak and Tuza [PT95] and to cite just one contemporary paper on practical heuristics, see Rendl, Rinaldi, and Wiegele [RRW07].

To cope with NP-hardness and to understand hard instances, researchers have turned to approximation algorithms. The greedy algorithm (or the random-assignment algorithm) is easily shown to have an approximation ratio of $\frac{1}{2}$ (see [SG76]). In a breakthrough affecting both theory and practice, Goemans and Williamson [GW95] gave a semidefinite programming (SDP) rounding algorithm achieving a .878 approximation ratio.¹ Since the early '90s, there has been a tremendous amount of theoretical interest in the SDP relaxation, in approximation algorithms, and in hardness of approximation for Max-Cut [DP93b, DP93a, GW95, Kar99, Zwi99, AS00, FL06, FKL02, FS02, ASZ02, HLZ04, KKMO07, CW04, AZ05, KV05, KO06]. In this work, we build on the results in many of these papers and determine an essentially complete picture of the algorithms, SDP gaps, Long Code tests, and UGC-hardness for Max-Cut.

1.1 Definitions

We begin with the basic definitions. We generally work with edge-weighted, undirected graphs $G = (V, E, w)$, where $w : E \rightarrow \mathbb{R}^{\geq 0}$ gives the nonnegative edge weights. The issue of self-loops turns out to be a nuisance; our policy will be to disallow them unless otherwise specified. Without loss of generality, we will always assume the edge weights sum to 1; i.e., $\sum_{e \in E} w(e) = 1$. Thus we can think of the weights as giving a probability distribution on edges; we will therefore omit w and think of E as a (symmetric) probability distribution on edges, writing $(\mathbf{u}, \mathbf{v}) \sim E$ to denote a draw from this distribution.²

Definition 1.1 A (proper) cut in G is a partition of the vertices into two parts, $h : V \rightarrow \{-1, 1\}$. The value of the cut is

$$\text{val}_G(h) = \Pr_{(\mathbf{u}, \mathbf{v}) \sim E} [h(\mathbf{u}) \neq h(\mathbf{v})] = \mathbf{E}_{(\mathbf{u}, \mathbf{v}) \sim E} [\frac{1}{2} - \frac{1}{2}h(\mathbf{u})h(\mathbf{v})].$$

The Max-Cut problem is the following: Given G , find a proper cut h with as large a value as possible.

In general, we prefer the second definition of value given above, since it generalizes to *fractional cuts*:

Definition 1.2 A fractional cut in G is a function $h : V \rightarrow [-1, 1]$. The value of the fractional cut is

$$\text{val}_G(h) = \mathbf{E}_{(\mathbf{u}, \mathbf{v}) \sim E} [\frac{1}{2} - \frac{1}{2}h(\mathbf{u})h(\mathbf{v})].$$

Given a fractional cut h , we can randomly produce a proper cut h' by setting each value $h'(v)$ to be 1 with probability $\frac{1}{2} + \frac{1}{2}h(v)$ and -1 with probability $\frac{1}{2} - \frac{1}{2}h(v)$, independently across v 's. In this way, $\mathbf{E}[h'(v)] = h(v)$. It follows that $\mathbf{E}[\text{val}_G(h')] = \text{val}_G(h)$ (although this uses the fact that G has no self-loops). Hence there always exists a proper cut h' with value at least $\text{val}_h(G)$, and furthermore such a cut can easily be found deterministically from h using the method of conditional expectations. For these reasons, we will henceforth treat the Max-Cut problem as being about finding a *fractional* cut with as large a value as possible, and we will refer to fractional cuts simply as 'cuts'.

Definition 1.3 The optimum cut value, or *Max-Cut*, for G is denoted

$$\text{Opt}(G) = \sup_{h: V \rightarrow [-1, 1]} \text{val}_G(h).$$

Note that the optimum is always at most 1 and at least $\frac{1}{2}$ (since the fractional cut $h \equiv 0$ is always available).

¹The SDP relaxation itself was given earlier by Delorme and Poljak [DP93b], who noted it was polynomial-time computable.

²Throughout this paper we use boldface to indicate random variables.

1.2 On approximation algorithms

Let A be a polynomial-time (fractional) cut-finding algorithm, and let $\text{Alg}_A(G)$ denote the value of the cut output by A on G . We allow A to be randomized, in which case we let $\text{Alg}_A(G)$ denote the expected value of the cut output by A . (The deterministic/randomized distinction is well known to be of little importance; see Appendix C for details.)

The traditional way to measure the quality of an approximation algorithm is to look at the worst-case *ratio* $\text{Alg}_A(G)/\text{Opt}(G)$, over G . For example, the Goemans-Williamson (GW) algorithm has a guarantee that this ratio is always at least .878. However this guarantee is not very good for graphs G with only moderately large maximum cuts. For example, if $\text{Opt}(G) = .55$ then the GW algorithm may [ASZ02] only find a cut with value $.878 \cdot .55 < .49$, which is worse than the trivial fractional cut. On the other hand, Goemans and Williamson showed [GW95] that when $\text{Opt}(G) = .95$, their algorithm finds a cut with value at least .90, which is significantly *better* than $.878 \cdot .95$.

We believe it is essential to measure the quality of an approximation algorithm not with a single ratio but with a *curve*.

Definition 1.4 We say that algorithm A achieves approximation curve $\text{Apx}_A : [\frac{1}{2}, 1] \rightarrow [\frac{1}{2}, 1]$ if

$$\text{Alg}_A(G) \geq \text{Apx}_A(\text{Opt}(G)) \quad \text{for all } G.$$

For example, the GW guarantee is usually described as achieving approximation curve $c \mapsto .878c$, but in fact Goemans and Williamson achieve the guarantee³

$$\text{Apx}_{\text{GW}}(c) \geq \begin{cases} \frac{1}{\pi} \arccos(1 - 2c) & \text{if } c \geq .844, \\ .878c & \text{if } c \leq .844. \end{cases} \quad (1)$$

1.3 Semidefinite programming relaxations and gaps

All of the best approximation guarantees for Max-Cut currently known are achieved by algorithms using the semidefinite programming (SDP) relaxation [FL92, DP93b, PR95, GW95]:

Definition 1.5 The (Max-Cut) SDP value of a graph G is

$$\text{Sdp}(G) = \max_{g: V \rightarrow B_n} \mathbf{E}_{(\mathbf{u}, \mathbf{v}) \sim E} \left[\frac{1}{2} - \frac{1}{2} g(\mathbf{u}) \cdot g(\mathbf{v}) \right], \quad (2)$$

where $n = |V|$ and B_n denotes $\{x \in \mathbb{R}^n : \|x\| \leq 1\}$. Note that $\text{Sdp}(G) \geq \text{Opt}(G)$, as g can always be taken to map into $[-1, 1]$.

The utility of this relaxation is that we can actually find an essentially optimal embedding g in polynomial time (more precisely, we can find a g achieving at least $\text{Sdp}(G) - \epsilon$ in time $\text{poly}(n) \cdot \log(1/\epsilon)$; see [GW95]). We should note that for graphs without self-loops, it is easy to see that the optimal embedding maps all vertices to the boundary of the ball, $S^{n-1} = \{x \in \mathbb{R}^n : \|x\| = 1\}$.

Triangle inequalities. One can also consider strengthening the SDP by adding the ‘triangle inequalities’: i.e., enforcing

$$\begin{aligned} g(v_1) \cdot g(v_2) - g(v_2) \cdot g(v_3) - g(v_1) \cdot g(v_3) &\geq -1, \\ g(v_1) \cdot g(v_2) + g(v_2) \cdot g(v_3) + g(v_1) \cdot g(v_3) &\geq -1, \end{aligned}$$

for all $v_1, v_2, v_3 \in V$. All of our positive results (rounding algorithms) will hold without the triangle inequalities, and we focus attention in this work almost exclusively on the basic SDP (2). However, we will also show that all of our negative results (SDP gaps, algorithmic limitations) hold even *with* the triangle inequalities.

³Here and throughout, .844 and .878 are shorthand for $\frac{1}{2} - \frac{1}{2}\theta^*$ and $\frac{2}{\pi} \csc \theta^*$, where θ^* is the positive solution of $\theta = \tan(\frac{\theta}{2})$.

In evaluating approximation algorithms, we would like to compare the cut-values found by an algorithm A to the maximum cut values. However doing this directly is difficult — roughly because Max-Cut is hard, and therefore we don't analytically have access to $\text{Opt}(G)$. The approximation guarantees of SDP-based algorithms are actually based on comparing the value of the cuts found to the *SDP value*:

Definition 1.6 *We say that algorithm A achieves SDP-approximation curve $\text{SdpApx}_A : [\frac{1}{2}, 1] \rightarrow [\frac{1}{2}, 1]$ if*

$$\text{Alg}_A(G) \geq \text{SdpApx}_A(\text{Sdp}(G)) \quad \text{for all } G.$$

For example, the GW algorithm actually has *SDP-approximation curve* given by the curve in (1).

There is an obvious barrier to how good SDP-approximation guarantees can be: If there exists a graph G with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$ then of course no algorithm could have an SDP-approximation curve SdpApx with $\text{SdpApx}(c) > s$. This leads us to the notion of the ‘SDP gap curve’, generalizing the usual SDP gap ratio:

Definition 1.7 *For $\frac{1}{2} \leq s \leq c \leq 1$, we call the pair (c, s) an SDP gap if there exists a graph G with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$. We define the SDP gap curve by*

$$\text{Gap}_{\text{SDP}}(c) = \inf\{s : (c, s) \text{ is an SDP gap}\}.$$

We analogously define the curve $\text{Gap}_{\Delta\text{SDP}}$ for the SDP with the triangle inequalities. Of course, we have $\text{Gap}_{\Delta\text{SDP}}(c) \geq \text{Gap}_{\text{SDP}}(c)$ for all c .

In Appendix A we show that Gap_{SDP} must be a continuous, strictly increasing function.

1.4 RPR² algorithms

Generalizing the GW algorithm, Feige and Langberg [FL06] introduced the ‘RPR²’ (Randomized Projection, Randomized Rounding) framework for rounding the solutions of semidefinite programming relaxations:

Definition 1.8 *An RPR² algorithm for Max-Cut is defined by a rounding function, $r : \mathbb{R} \rightarrow [-1, 1]$. Given a graph G , the steps of the algorithm are as follows:*

1. *Use semidefinite programming to find an optimal embedding $g : V \rightarrow S^{n-1}$ for the SDP (2).*
2. *Choose a random vector $\vec{Z} \in \mathbb{R}^n$ according to the n -dimensional Gaussian distribution.*
3. *Output the (fractional) cut $\mathbf{h} : V \rightarrow [-1, 1]$ defined by $\mathbf{h}(v) = r(g(v) \cdot \vec{Z})$.*

(Certain implementation details of the RPR² method are discussed in Appendix C.)

All of the known lower bounds for $\text{Gap}_{\text{SDP}}(c)$ fall into the RPR² framework. For example, the GW algorithm is RPR² with rounding function $r(x) = \text{sgn}(x)$; the random-assignment algorithm is RPR² with rounding function $r(x) \equiv 0$. Zwick’s algorithm [Zwi99] is not obviously RPR², but it is shown to be so by Feige and Langberg [FL06]. In that paper, the authors suggest using ‘ s -linear’ rounding functions: i.e., functions of the form $r(t) = t/s$ if $-s \leq t \leq s$, $r(t) = 1$ if $t \geq s$, $r(t) = -1$ if $t \leq -s$. Charikar and Wirth’s analysis [CW04] of $\text{Gap}_{\text{SDP}}(c)$ near $c = \frac{1}{2}$ indeed uses RPR² with s -linear rounding functions.

We conclude the discussion of RPR² algorithms by mentioning that, given an input graph G , it can be advantageous to try several different rounding functions r . It is well known (as discussed in Appendix C) that given a collection \mathcal{R} of rounding functions, one can achieve the performance of the best of them with running time slowdown only $O(|\mathcal{R}| \log |\mathcal{R}|)$. Indeed, Feige and Langberg even suggested the idea of trying ‘all’ possible rounding functions, up to some ϵ -discretization. Whether or not this achieves the performance of the ‘optimal’ rounding function up to an additive ϵ is a tricky issue which we discuss further in Section 3.2.

1.5 Long Code tests

‘Long Code tests’ are certain kinds of Property Testing algorithms, operating on boolean functions $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$. Specifically, they test for the property of being a ‘Dictator’ function; i.e., one of the n functions defined by $f(x) = x_i$, for $i \in [n]$. Long Code tests were first studied by Bellare, Goldreich, and Sudan [BGS98] for the purposes of proving inapproximability results; the connection was significantly extended by Håstad [Hås01]. The general idea is that to show an inapproximability result for the constraint satisfaction problem with constraints of type Φ , one tries to construct a Long Code test whose acceptance predicate is of type Φ . In particular, for Max-Cut one needs a test making only 2 queries and testing $f(x) \neq f(y)$. The rule of thumb is that giving a such a test with ‘completeness’ c and ‘soundness’ s may allow one to derive a c vs. s inapproximability result. (We give concrete theorems along these lines later in this section.) Thus it is natural to investigate, for each c , what the minimum achievable value of s is.

Let us briefly introduce some of the relevant definitions:

Definition 1.9 *A 2-query, \neq -based Long Code test for functions $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$ is a randomized procedure for choosing two strings $\mathbf{x}, \mathbf{y} \in \{-1, 1\}^n$. We think of the test as querying $f(\mathbf{x})$ and $f(\mathbf{y})$, and then accepting when $f(\mathbf{x}) \neq f(\mathbf{y})$, and rejecting otherwise.*

Definition 1.10 *The completeness of a Long Code test T for n -bit functions is*

$$\text{Completeness}(T) = \min_{i \in [n]} \{\Pr[T \text{ accepts } \chi_i]\},$$

where $\chi_i : \{-1, 1\}^n \rightarrow \{-1, 1\}$ is the i th ‘Dictator’ function, $\chi_i(x) = x_i$.

As for ‘soundness’, Property Testing definitions would normally require a Long Code test to reject with large probability any function ϵ -far from being a Dictator. However since our Long Code tests T only make 2 queries, we could never achieve soundness smaller than $\text{Completeness}(T) - 2\epsilon$ under this definition. Fortunately, key applications of Long Code tests only require certain relaxed soundness conditions. A useful such relaxation was introduced by Khot, Kindler, Mossel, and O’Donnell [KKMO07]. It only requires the test to reject functions that have sufficiently small ‘low-degree influences’ — or, essentially equivalently, functions that are sufficiently ‘Gaussianic’. We defer the formal explanation to Section 8; for now, suffice it to say we make a definition along the following lines:

Definition 1.11 (informal) *The soundness of a Long Code test T for functions $f : \{-1, 1\}^n \rightarrow [-1, 1]$ is*

$$\text{Soundness}(T) = \max\{\Pr[T \text{ accepts } f] : f \text{ is ‘Gaussianic’}\}.$$

In addition to the unspecified notion ‘Gaussianic’, the reader will notice that we have generalized to testing functions whose range is $[-1, 1]$ rather than $\{-1, 1\}$. The reason for doing this is that all the applications we present require this generalized setting. The distinction is similar to the one between proper and fractional cuts. Again, formal definitions appear in Section 8.

To emphasize the definitions of completeness and soundness, we will henceforth refer to our Long Code tests as *Dictator-vs.-Gaussianic tests*. It is a natural question in Property Testing to ask how far apart completeness and soundness can be for Dictator-vs.-Gaussianic tests. To formalize the question, we can introduce the notion of the Dictator-vs.-Gaussianic gap curve:

Definition 1.12 (informal) *We call the pair (c, s) a Dictator-vs.-Gaussianic gap if for all $\eta > 0$, for sufficiently large n there is a Dictator-vs.-Gaussianic test $T^{(n)}$ for functions $f : \{-1, 1\}^n \rightarrow [-1, 1]$ with $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}(T^{(n)}) \leq s + \eta$. We define the Dictator-vs.-Gaussianic gap curve by*

$$\text{Gap}_{\text{Test}}(c) = \inf\{s : (c, s) \text{ is a Dictator-vs.-Gaussianic gap}\}.$$

As mentioned, our interest in Dictator-vs.-Gaussianic tests comes from their application to algorithmic hardness results. We give three such applications here. The first is the original application, implicitly proved in [KKMO07]:

Theorem 1.13 ([KKMO07]) *Suppose (c, s) is a Dictator-vs.-Gaussian gap, and $\eta > 0$. Then the Unique Games Conjecture (UGC) implies that it is NP-hard to distinguish Max-Cut instances with value at least $c - \eta$ from instances with value at most $s + \eta$. I.e., assuming the UGC and $P \neq NP$ we essentially have $\text{Apx}_A(c) \leq \text{Gap}_{\text{Test}}(c)$ for all efficient algorithms A and all c .*

(The ‘essentially’ here refers to the fact that we really only have $\text{Apx}_A(c - \eta) \leq \text{Gap}_{\text{Test}}(c)$ for all $\eta > 0$. Ultimately we will show that Gap_{Test} is continuous, so this distinction is irrelevant.)

Combining the reduction used to prove Theorem 1.13 with an SDP integrality gap instance for the Unique Games problem with several special properties, Khot and Vishnoi implicitly showed the following application:

Theorem 1.14 ([KV05]) *If (c, s) is a Dictator-vs.-Gaussian gap, then $(c - \eta, s)$ is also an SDP gap for all $\eta > 0$ — even for the SDP with triangle inequalities. I.e., essentially $\text{Gap}_{\Delta\text{SDP}}(c) \leq \text{Gap}_{\text{Test}}(c)$ for all c .*

Finally, in Section 11 of this paper we show how to extend a result of Feige and Schechtman [FS02], itself based on work of Karloff [Kar99], Alon and Sudakov [AS00], and Alon, Sudakov, and Zwick [ASZ02], as follows:

Theorem 1.15 *Suppose (c, s) is a Dictator-vs.-Gaussian gap. Fix any rounding function r , and let A be the RPR² algorithm which solves the SDP with triangle inequalities and randomly rounds using $\mathbf{h}(v) = r(g(v) \cdot \vec{Z})$. Then $\text{Apx}_A(c) \leq s$. Further, this holds even if A is not required to choose \vec{Z} to be a random n -dimensional Gaussian, but rather is allowed to deterministically select the best \vec{Z} satisfying $\|\vec{Z}\|_2 = \Theta(\sqrt{n})$.*

2 Our results, and prior work

2.1 Statement of main results

Our first result, from which the remaining results derive, is a complete determination of the SDP gap curve. We introduce an explicit function $S : [\frac{1}{2}, 1] \rightarrow [\frac{1}{2}, 1]$, and show that $\text{Gap}_{\text{SDP}}(c) = S(c)$ for all c . In particular, the proof of the lower bound, $\text{Gap}_{\text{SDP}}(c) \geq S(c)$, is achieved via a poly(n)-time RPR² algorithm. Thus we have an efficient algorithm for Max-Cut which has optimal SDP-approximation curve. The fact that an RPR² algorithm achieves the SDP gap confirms a conjecture suggested by Feige and Langberg [FL06].

Next, we show how to transform the SDP results into Dictator-vs.-Gaussian testing results. Specifically, we are able to show that the Dictator-vs.-Gaussian gap curve is identical to the SDP gap curve; i.e., $\text{Gap}_{\text{Test}}(c) = S(c)$ for all $c \in [\frac{1}{2}, 1]$. This result gives us optimal Dictator-vs.-Gaussian tests. Substituting these into Theorems 1.14, 1.15, and 1.13 yields the following conclusions:

- The SDP gap curve with triangle inequalities, $\text{Gap}_{\Delta\text{SDP}}$, is also identical to the curve S .
- If A is any RPR² algorithm then $\text{Apx}_A(c) \leq S(c)$ for all c , even assuming both of the following: (i) A uses the SDP with triangle inequalities; (ii) A is not required to choose \vec{Z} to be a random n -dimensional Gaussian, but rather is allowed to deterministically select the best \vec{Z} satisfying $\|\vec{Z}\| = \Theta(\sqrt{n})$. (Contrast this with the fact that in graphs exhibiting the c vs. $S(c)$ SDP gap, our RPR² algorithm actually finds an essentially optimal cut.)
- If A is any polynomial-time algorithm then $\text{Apx}_A(c) \leq S(c)$ for all c , assuming $P \neq NP$ and the UGC.

2.2 The critical curve, S

At this point the reader might wish to know the identity of this critical curve $S(c)$. Unfortunately, there is no ‘nice’ formula for it. Rather, it is defined as follows:

$$S(c) = \inf_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{G_P}(r). \quad (3)$$

Not all of the expressions above have even been defined yet — in particular ‘ $(1, \rho_0)$ -distribution’ (a certain simple kind of probability distribution on $[-1, 1]$) and ‘ \mathcal{G}_P ’ (a certain infinite graph). Further, on the face of it this definition does not look very ‘explicit’, especially since the inf and sup are both over infinite sets. Nevertheless, in Section 6 we prove the following:

Theorem 2.1 *There is an algorithm that, on input $c \in [\frac{1}{2}, 1]$ and $\epsilon > 0$, runs in time $\text{poly}(1/\epsilon)$ and computes $S(c)$ to within $\pm\epsilon$.*

We believe this justifies our claim that S is ‘explicitly given’. A brief discussion of this point appears in Section 7.1.

In fact, as we will describe in the next section, significant portions of $S(c)$ can be described or estimated more simply. For $c \geq .844$, $S(c)$ agrees with the Goemans-Williamson SDP-approximation curve, $\frac{1}{\pi} \arccos(1 - 2c)$. For $c = \frac{1}{2} + \epsilon$, $S(c) \approx \frac{1}{2} + \frac{1}{2} \cdot \epsilon / \ln(1/\epsilon)$ up to lower-order terms (this is proved in Appendix D, tightening the asymptotics of [CW04, KO06]). A plot of $S(c)$ versus c appears in Appendix E.

2.3 Prior work

Surveying the entirety of the previous work on approximation algorithms, SDP gaps, and hardness results for Max-Cut would take several pages, so we restrict ourselves to briefly summarizing the best results known prior to this work.

SDP and Long Code testing gaps. Combining prior work of many authors yields the following:

1. For $c \geq .844$: $\text{Gap}_{\text{SDP}}(c) = \text{Gap}_{\Delta\text{SDP}}(c) = \text{Gap}_{\text{Test}}(c) = \frac{1}{\pi} \arccos(1 - 2c)$.
2. For $c = \frac{1}{2} + \epsilon$: $\text{Gap}_{\text{SDP}}(c)$, $\text{Gap}_{\Delta\text{SDP}}(c)$, and $\text{Gap}_{\text{Test}}(c)$ all have asymptotics $\frac{1}{2} + \Theta(\epsilon / \ln(1/\epsilon))$.

As can be seen, this already pins down substantial portions of these curves fairly well. In the next section we will argue the merits of pinning them down precisely.

The lower bound $\text{Gap}_{\text{SDP}}(c) \geq \frac{1}{\pi} \arccos(1 - 2c)$ for $c \geq .844$ is, as mentioned, due to Goemans and Williamson [GW95], using RPR² with the rounding function sgn . The matching upper bound is due to Feige and Schechtman [FS02], using infinite graphs with vertex set S^{n-1} and edge set connecting all vectors with inner product at most $1 - 2c$. The lower bound $\text{Gap}_{\text{SDP}}(c) \geq \frac{1}{2} + \Omega(\epsilon / \ln(1/\epsilon))$ is due to Charikar and Wirth [CW04], using RPR² with s -linear rounding functions, as suggested by Feige and Langberg [FL06]. The upper bound $\text{Gap}_{\text{SDP}}(c) \leq \frac{1}{2} + O(\epsilon / \ln(1/\epsilon))$ is due to Khot and O’Donnell [KO06], using mixtures of correlated Gaussian graphs (described in Section 3.2). As mentioned, we tighten the asymptotics of the previous two results in Appendix D. Finally, Feige and Langberg showed some additional numerical lower bounds for $\text{Gap}_{\text{SDP}}(c)$, via RPR² with s -linear rounding functions; e.g., $\text{Gap}_{\text{SDP}}(.6) \geq .5477$.

The upper bound $\text{Gap}_{\text{Test}}(c) \leq \frac{1}{\pi} \arccos(1 - 2c)$ actually holds for all $c \in [\frac{1}{2}, 1]$; this was conjectured by Khot, Kindler, Mossel, and O’Donnell [KKMO07] and proved by Mossel, O’Donnell, and Oleszkiewicz [MOO05]. The ‘noise sensitivity’ test from [KKMO07] involves choosing $\mathbf{x} \in \{-1, 1\}^n$ uniformly at random and choosing \mathbf{y} by flipping each coordinate of \mathbf{x} with probability c . (As we will discuss in Section 11, this construction is quite similar to one introduced by Karloff [Kar99] and analyzed further in [AS00, ASZ02].) The upper bound $\text{Gap}_{\text{Test}}(\frac{1}{2} + \epsilon) \leq \frac{1}{2} + O(\epsilon / \ln(1/\epsilon))$ was proved by Khot and O’Donnell [KO06], by mixing together two tests of the type in [KKMO07]. The remaining parts of the above statements implicitly follow from Khot and Vishnoi [KV05], specifically, from Theorem 1.14’s statement that $\text{Gap}_{\text{Test}}(c) \geq \text{Gap}_{\Delta\text{SDP}}(c) \geq \text{Gap}_{\text{SDP}}(c)$. Interestingly, although proving lower bounds for $\text{Gap}_{\text{Test}}(c)$ is a very natural problem from the point of view of Property Testing, it doesn’t seem to have been explicitly considered in the literature. Indeed, using the Khot-Vishnoi result is a very circuitous way to prove Long Code testing lower bounds. We discuss this point further in Section 10.

Algorithmic hardness. Early results on algorithmic hardness involved showing upper bounds on the approximation curve of specific algorithms. In particular, work of Karloff [Kar99], Alon and Sudakov [AS00], and Alon, Sudakov, and Zwick [ASZ02] showed that for the GW algorithm, $\text{Apx}_{GW}(c) \leq \frac{1}{\pi} \arccos(1 - 2c)$, where $\text{Apx}_{GW}(c)$ denotes the *expected* performance, over \vec{Z} , of the GW algorithm. Further, this result holds even if one adds all ‘valid’ constraints to the SDP. As we describe in Section 11, these results can be seen as very weak forms of Dictator-vs.-Gaussian tests. Feige and Schechtman [FS02] extended these results in the manner of Theorem 1.15, to the case where the algorithm can pick *any* halfspace cut (although only under the triangle inequalities, not any valid constraints). Assuming the Unique Games Conjecture, [KKMO07]’s Theorem 1.13 implies NP-hardness of achieving approximation curve exceeding $\text{Gap}_{\text{Test}}(c)$. The best unconditional NP-hardness result is much weaker: Håstad [Hås01] together with Trevisan, Sorkin, Sudan, and Williamson [TSSW00] showed that achieving $\text{Apx}(\frac{17}{21}) > \frac{16}{21}$ is NP-hard; it is easy to translate this into hardness of $\text{Apx}(\frac{1}{2} + \epsilon) > \frac{1}{2} + \frac{11}{13}\epsilon$ for $\epsilon \leq \frac{13}{42}$ and hardness of $\text{Apx}(1 - \epsilon) > 1 - \frac{5}{4}\epsilon$ for $\epsilon \leq \frac{4}{21}$.

2.4 Motivation and discussion

In this section we discuss the motivation and merits of pinning down the approximability of Max-Cut precisely for all values of c .

First, Max-Cut is a fundamental algorithmic problem; indeed, it is arguably the simplest NP optimization problem. For the reasons discussed in Section 1.2, we feel that understanding its approximability for the entire range of c is important. We are hardly alone in this regard; for example, in 2001 Feige and Langberg [FL06] wrote that they were “trying to extend the techniques of [FS02] in order to prove [that RPR² algorithms can match the SDP gap curve for values of $c < .844$]”. Besides the algorithmic work on the Max-Cut curve we’ve already described [GW95, Zwi99, FL06, CW04], there has also been a great deal of work recently on the very related problem of the Max-2Lin [BBC04, HV04, AN06, AMMN06, ABH⁺05]. For example the Grothendieck/Quadratic Programming results of [AN06, AMMN06, CW04] are nothing more than analysis of the Max-2Lin approximability curve at $\frac{1}{2} + \epsilon$ — with the underlying graph structure fixed to be bipartite, in the Grothendieck case. Further, analyzing the Max-Cut/Max-2Lin approximability curves at $1 - \epsilon$ for *subconstant* ϵ is very strongly related to analyzing Sparsest-Cut approximability.

Further, the fundamental nature of the Max-Cut problem makes our inability to understand its computational complexity all the more galling. Recall that every value of c for which we don’t know the largest efficiently achievable value of $\text{Apx}_A(c)$ yields a basic, natural problem not known to be in P and not known to be NP-hard: e.g., “Given a graph with a cut of size 60%, find a cut of size 55%”. Without the Unique Games Conjecture, it seems we have no idea how to prove sharp inapproximability results, although in this paper we did the best we could by ruling out RPR² algorithms from achieving $\text{Apx}(c) > S(c)$. Assuming the Unique Games Conjecture, though, the present work completely closes the Max-Cut problem. Even if one does not believe the UGC, there are several takeaways: First, we’ve shown that the UGC cannot be disproved by giving good Max-Cut SDP rounding algorithms, for any value of c . Second, our work gives an improved approximation *algorithm* inspired by UGC/Dictator-vs.-Gaussian test considerations.

Next, the present work develops a framework for studying SDP gaps, algorithms, and Dictator-vs.-Gaussian tests, which we believe explains away some of the seeming coincidences in previous work. For example, in the next section we explain why the worst SDP gaps for Max-Cut can *always* be based on symmetric infinite graphs on the surface of the sphere, as they are in [FS02]; we further explain how one naturally derives the mixtures of correlated Gaussian graphs used for SDP gaps in [KO06]. Next, we essentially explain why the SDP gap curve and the Dictator-vs.-Gaussian curve are identical: why SDP gaps translate into Dictator-vs.-Gaussian tests and why RPR² algorithms can be viewed as Dictator-vs.-Gaussian testing lower bounds. Finally, we explain the connection between Dictator-vs.-Gaussian tests and the constructions of Karloff, Alon-Sudakov, and Alon-Sudakov-Zwick [Kar99, AS00, ASZ02], showing that these constructions can be viewed as Dictator-vs.-Gaussian tests with extremely weak soundness guarantees.

Finally, we hope that the methods developed in this paper — specifically, the use of Hermite analysis, von Neumann’s Minimax Theorem, Borell’s rearrangement inequality [Bor85], and the Karush-Kuhn-Tucker

conditions — can be used to make progress on understanding SDP gaps and approximability of other fundamental problems. Specifically, we believe our methods should be useful for attacking Max-2Sat and other 2-CSPs (some indication of this is given already in the recent work of Austrin [Aus07a, Aus07b]), 3-CSPs, and perhaps even for determining the Grothendieck constant [Gro53].

3 Proof ideas

In this section we describe the ideas and intuition underlying the determination of Gap_{SDP} . By the end of the section we will also have defined all the terms necessary for the definition (3) of the curve $S(c)$.

3.1 Embedded graphs

The first idea is to slightly shift the way one looks at SDP gaps for Max-Cut. Usually one thinks of first finding a graph G , then showing $\text{Sdp}(G)$ is large and $\text{Opt}(G)$ is small. But suppose one determines that $\text{Sdp}(G)$ is large for some graph G ; then one *may as well identify G with its optimal SDP embedding on the sphere*.

Definition 3.1 *An (n -dimensional) embedded graph G is one whose vertex set V is a subset of S^{n-1} . For embedded graphs, we explicitly allow self-loops.⁴ The ρ -distribution of the embedded graph, denoted $P = P(G)$, is the discrete probability distribution on $[-1, 1]$ given by the distribution of $\mathbf{u} \cdot \mathbf{v}$ when $(\mathbf{u}, \mathbf{v}) \sim E$. We define the spread of G (which we also call the spread of P) to be*

$$\text{Spread}(G) = \text{Spread}(P) = \mathbf{E}_{\rho \sim P} \left[\frac{1}{2} - \frac{1}{2}\rho \right] \in [0, 1].$$

Thinking about embedded graphs leads to some important observations. The first is that we can *symmetrize* any SDP gap instance. Specifically, let G be an embedded graph with $\text{Spread}(G) = c$ and $\text{Opt}(G) \leq s$. Suppose \mathcal{O} is any rotation of space; then it is clear that the rotated embedded graph $\mathcal{O}G$ also satisfies $\text{Spread}(\mathcal{O}G) = c$ and $\text{Opt}(\mathcal{O}G) \leq s$, and is thus an equally good gap instance. Further, if one takes a mixture $H = \lambda G + (1 - \lambda)G'$ of any two embedded graphs G and G' with $\text{Spread}(G) = \text{Spread}(G') = c$ and $\text{Opt}(G), \text{Opt}(G') \leq s$, then $\text{Spread}(H)$ is again c , and also $\text{Opt}(H) \leq s$ by a simple averaging argument. Hence we can average an SDP gap instance G over *all* rotations of space, and preserve the gap. When we do this we get an ‘infinite embedded graph’ whose vertex set is all of S^{n-1} and whose edge distribution is ‘symmetric’, in the sense that the density on the pair (u, v) depends only on the inner product $u \cdot v$. In fact, the ‘ ρ -distribution’ of the symmetrized graph is precisely the original ρ -distribution $P(G)$.

Definition 3.2 *Let P denote any discrete probability distribution on $[-1, 1]$. We define the d -dimensional symmetric embedded graph $\mathcal{S}_P^{(d)}$ to be the embedded graph with vertex set S^{d-1} and edge distribution over $S^{d-1} \times S^{d-1}$ given by drawing a random pair of unit vectors with inner product ρ , where ρ itself is drawn from P .*

Thus we have reduced the search for graphs with large SDP gap to the search for ρ -distributions P such that $\text{Spread}(P) = c$ (i.e., the mean of P is $1 - 2c$) but $\text{Opt}(\mathcal{S}_P^{(d)})$ is small. Indeed, Feige and Schechtman’s SDP gap instance [FS02] is precisely of this form; roughly speaking, they take P to be the distribution with all of its mass concentrated on $1 - 2c$.

Unfortunately, analyzing $\text{Opt}(\mathcal{S}_P^{(d)})$ is not so easy; we will come back to the problem later. For now let us move to the algorithmic side of things. We have seen that we can reduce the problem of finding large SDP gaps to studying symmetric embedded graphs. Can we similarly reduce the problem of finding large cuts in arbitrary graphs to studying symmetric embedded graphs? The observation here is that, in some sense, this is just what the RPR² algorithm is doing. Consider the steps of the algorithm from Definition 1.8. RPR² algorithms do not use the fact that the SDP solution they operate on is optimal; hence we can mentally dispense with Step 1 (semidefinite programming) and view RPR² algorithms as simply taking an embedded

⁴Although we disallow self-loops in Max-Cut inputs, we allow them in embedded graphs. One reason for this is that there is no guarantee that every optimal SDP embedding $g : V \rightarrow S^{n-1}$ is injective.

graph G as input and trying to find a large cut in it. Next, recalling that the d -dimensional Gaussian distribution is spherically symmetric, we see that the RPR² algorithm can, at a rough level, be thought of as: (i) implicitly constructing the symmetrized version of G ; and then, (ii) outputting the ‘one-dimensional’ fractional cut r . We will make this idea more precise in the next section. For now, we note that if RPR² algorithms are to achieve the SDP gap, it must in some sense be the case that optimal cuts in symmetric embedded graphs $\mathcal{S}_P^{(d)}$ are ‘one-dimensional’. The key to our determination of $\text{Gap}_{\text{SDP}}(c)$ is showing that this statement is sufficiently true.

3.2 Gaussian mixture graphs

By now our analysis is heavily dependent on understanding $\text{Opt}(\mathcal{S}_P^{(d)})$, where P is a distribution with mean $1 - 2c$. I.e., we want to determine

$$\sup_{h: S^{d-1} \rightarrow [-1,1]} \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{u}, \mathbf{v}) \sim S^{d-1} \times S^{d-1} \\ \text{with } \langle \mathbf{u}, \mathbf{v} \rangle = \rho}} \left[\frac{1}{2} - \frac{1}{2} h(\mathbf{u}) \cdot h(\mathbf{v}) \right].$$

This is somewhat complicated by the fact the distribution on vertices — i.e., the uniform distribution on the surface of the sphere — is not a product distribution, and depends in a nontrivial way on the dimension d . It is possible to at once avoid this difficulty *and* hew much more closely to the RPR² framework by replacing the uniform distribution on S^{d-1} by the d -dimensional *Gaussian* distribution.

Definition 3.3 *Let P denote any discrete probability distribution on $[-1, 1]$. We define the d -dimensional Gaussian mixture graph $\mathcal{G}_P^{(d)}$ to be the probability measure on $\mathbb{R}^d \times \mathbb{R}^d$ given by drawing a pair of ‘ ρ -correlated d -dimensional Gaussians’, where ρ itself is drawn from P . In the case $d = 1$, we simply write \mathcal{G}_P . By ρ -correlated d -dimensional Gaussians we mean a pair $(\vec{\mathbf{x}}, \vec{\mathbf{y}})$, where $\vec{\mathbf{x}}$ is a standard d -dimensional Gaussian and $\vec{\mathbf{y}} \sim \rho \vec{\mathbf{x}} + \sqrt{1 - \rho^2} \vec{\mathbf{Z}}$, with $\vec{\mathbf{Z}}$ being another d -dimensional Gaussian independent of $\vec{\mathbf{x}}$. Note that this distribution is symmetric in $\vec{\mathbf{x}}$ and $\vec{\mathbf{y}}$.*

Gaussian mixture graphs, with P concentrated on 1 and $-\frac{1}{2}$, were introduced in [KO06] to show SDP gaps for c near $\frac{1}{2}$.

Regarding the effect of switching from $\mathcal{S}_P^{(d)}$ to $\mathcal{G}_P^{(d)}$, recall that the Gaussian distribution in a high dimension d is very similar to the uniform distribution on the sphere of radius \sqrt{d} . Using this fact, it is not too hard to show that when $\text{Spread}(P) = c$ we have $\text{Sdp}(\mathcal{G}_P^{(d)}) \geq c - o_d(1)$, via the embedding $x \mapsto x/\|x\|$. Thus we can equally well search for SDP gaps based on Gaussian mixture graphs. As for algorithms, the RPR² framework now has a very simple interpretation: Given an embedded graph G with ρ -distribution P , the RPR² algorithm implicitly converts it to \mathcal{G}_P and cuts it with the rounding function r . More specifically, the expected value of the cut produced by RPR² on graph G is:

$$\begin{aligned} \text{Alg}_{\text{RPR}^2}(G) &= \mathbf{E}_{\vec{\mathbf{Z}}} \left[\mathbf{E}_{(\mathbf{u}, \mathbf{v}) \sim E} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{u} \cdot \vec{\mathbf{Z}}) r(\mathbf{v} \cdot \vec{\mathbf{Z}}) \right] \right] \\ &= \mathbf{E}_{\rho \sim P(G)} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \text{ } \rho\text{-corr'd} \\ \text{1-dim Gaussians}}} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{x}) r(\mathbf{y}) \right] = \text{val}_{\mathcal{G}_P^{(1)}}(r). \end{aligned} \quad (4)$$

The reader can now see that given G , an RPR² algorithm should strive to take r to be the optimal cut $r: \mathbb{R} \rightarrow [-1, 1]$ for \mathcal{G}_P (i.e., $\mathcal{G}_P^{(1)}$). This leads us to two questions:

1. Can we *algorithmically* determine an r which gives a near-optimal cut for \mathcal{G}_P ?
2. Whether or not we can, would this be enough to match the SDP gap? In other words, is it true that for all ρ -distributions P with spread $c \in [\frac{1}{2}, 1]$,

$$\text{Opt}(\mathcal{G}_P) \geq \inf_{P' \text{ with mean } 1 - 2c} \text{Opt}(\mathcal{G}_{P'}^{(d)})? \quad (5)$$

Here the left-hand side represents what we hope to achieve algorithmically with RPR², and the right-hand side represents the upper-bound on $\text{Gap}_{\text{SDP}}(c)$ we can achieve using Gaussian mixture graphs.

Question 2 above is the heart of the matter; we describe its affirmative answer in the next section. For now, let us discuss Question 1. Although analytically we don't know the optimal cut for \mathcal{G}_P , there is a feeling that one could algorithmically find an r coming within ϵ of the optimum by using the Feige-Langberg idea of trying 'all' possible r , suitably discretized. Indeed, Feige and Langberg wrote that if one only considers 'well-behaved' rounding functions r (suggesting piecewise differentiable functions with bounded derivatives) then one can construct a collection of $2^{\text{poly}(1/\epsilon)}$ many discretized rounding functions such that one of them achieves a cut in \mathcal{G}_P that is within ϵ of that achieved by the best *well-behaved* rounding function.

Unfortunately, there is no guarantee that the optimal cut for \mathcal{G}_P is 'well-behaved'. Even if it were guaranteed to be piecewise differentiable, we have no way of proving that its derivatives don't depend on ' n '; i.e., the number of points in P 's support. Thus we do not know of any way of efficiently (in n) discretizing the search space for the optimal rounding function of a given \mathcal{G}_P . But luckily, in the next section we will see that for the 'worst' P , there is a relatively well-behaved optimal cut r ; specifically, there is an *increasing* optimal cut. The fact that increasing functions are $O(1/\epsilon)$ -Lipschitz except on a set of measure ϵ means it will be sufficient to discretize the set of rounding functions r in a way depending only on ϵ and not on n . Indeed, our actual algorithm for finding cuts of size at least $S(c) - \epsilon$ in graphs G with $\text{Sdp}(G) \geq c$ is:

Algorithm 3.4 *Perform the RPR² algorithm, trying out all $2^{\tilde{O}(1/\epsilon^2)}$ possible ' ϵ -discretized' rounding functions r .*

The definition of ' ϵ -discretized' is given in Section 5. A discussion of the running time, $\text{poly}(|V|) \cdot 2^{\tilde{O}(1/\epsilon^2)}$, appears in Section 7.2.

3.3 Hermite analysis, Minimax, and Borell's Gaussian rearrangement

We now come to the main conceptual part of the determination of Gap_{SDP} , namely proving (5). Suppose we could show that for every P' , there was an optimal cut f for $\mathcal{G}_{P'}^{(d)}$ that was 'one-dimensional' — i.e., of the form $f(\vec{x}) = r(\vec{u} \cdot \vec{x})$, where $r : \mathbb{R} \rightarrow [-1, 1]$ and \vec{u} is any unit vector. It's easy to see that the value of f in $\mathcal{G}_{P'}^{(d)}$ is just $\text{val}_{\mathcal{G}_{P'}}(r)$; hence we would show $\text{Opt}(\mathcal{G}_{P'}^{(d)}) = \text{Opt}(\mathcal{G}_{P'})$, proving (5). Unfortunately, we do not know whether this is the case. What we *will* show, though, is that when P' is the 'worst' distribution, $\mathcal{G}_{P'}^{(d)}$ has an optimal one-dimensional (and increasing, as promised) cut.

To start, we take advantage of our switch to Gaussian graphs; this allows us to express the value of cuts $f : \mathbb{R}^d \rightarrow [-1, 1]$ using 'Hermite analysis' (akin to Fourier analysis over $\{-1, 1\}^n$). Specifically, given a cut f one has

$$\text{val}_{\mathcal{G}_P^{(d)}}(f) = \frac{1}{2} - \frac{1}{2} \mathbf{E}_{\rho \sim P} \left[\sum_{S \in \mathbb{N}^d} \hat{f}(S)^2 \rho^{|S|} \right], \quad (6)$$

where each $\hat{f}(S) \in \mathbb{R}$ is a 'Hermite coefficient', and $|S|$ denotes $\sum_{i=1}^d S_i$. Using this formula one can easily show that any optimal cut f may as well be odd; i.e., satisfy $f(-\vec{x}) = -f(\vec{x})$. Further, when f is odd, the sum in (6) can be restricted to only be over S 's such that $|S|$ is odd.

We now make the following observation: For fixed odd f , the expression $\mathbb{S}_f(\rho) := \sum_{|S| \text{ odd}} \hat{f}(S)^2 \rho^{|S|}$ is a polynomial in ρ (power series, actually) with nonnegative coefficients and only odd powers. This means that it is convex for $\rho \geq 0$ and concave for $\rho \leq 0$. Now suppose we keep f fixed but vary the ρ -distribution P , subject only to it having mean $1 - 2c$. Using formula (6), one sees that we can make $\text{val}_{\mathcal{G}_P^{(d)}}(f)$ as low as the value of the convex lower envelope of $\frac{1}{2} - \frac{1}{2} \mathbb{S}_f(\rho)$ at $1 - 2c$. Further, by the convexity/concavity described, one achieves this by concentrating all of P 's probability mass on at most two points: some negative number ρ_0 , and possibly also 1.

Definition 3.5 *We call a discrete probability distribution P on $[-1, 1]$ a $(1, \rho_0)$ -distribution if P puts positive probability on some $-1 \leq \rho_0 \leq 0$, nonnegative probability on 1, and zero probability on all other values in $[-1, 1]$.⁵*

⁵At this point we have defined all of the terms necessary for the definition (3) of $S(c)$.

These considerations suggest that the Gaussian mixture graphs with lowest Max-Cut are those based on $(1, \rho_0)$ -distributions. This doesn't constitute a proof, though, because we fixed the cut and the graph in the wrong order: we are supposed to fix the distribution P first and then choose the optimal cut. Ultimately, though, we prove that $(1, \rho_0)$ -distributions *are* the worst case for Gaussian mixture graphs by using the von Neumann Minimax Theorem: we can reverse the order of fixing the distribution and the cut if we allow the 'cut Player' to choose a distribution on cuts. Fortunately, the convex combination of $\mathbb{S}_f(\rho)$ polynomials has the same convexity/concavity properties as a single one, so the previous argument goes through. Unfortunately, one also has to overcome some rather severe discretization/compactness complications to use the von Neumann Theorem in this infinitary setting.

At this point we essentially have that the Gaussian mixture graphs with smallest Max-Cut are those based on $(1, \rho_0)$ -distributions. Finally, we are able to deduce that in such graphs there are optimal, one-dimensional, increasing cuts through the use of Borell's rearrangement inequality for Gaussian space [Bor85]. Borell's theorem implies that for $\rho \in [0, 1]$, the quantity $\mathbb{S}_f(\rho)$ can only increase if one 'rearranges' f 's values into an increasing, one-dimensional function. If $G = \mathcal{G}_P^{(d)}$ is a Gaussian mixture graph with P a $(1, \rho_0)$ -distribution, then formula (6) tells us that $\text{val}_G(f)$ is (up to an additive $\frac{1}{2}$) a negative linear combination of $\mathbb{S}_f(1)$ and $\mathbb{S}_f(\rho_0)$. It turns out that $\mathbb{S}_f(1)$ is just $\mathbf{E}[f^2]$, which doesn't change under rearrangement, and when f is odd $\mathbb{S}_f(\rho_0) = -\mathbb{S}_f(-\rho_0)$; hence Borell implies that this quantity decreases under rearrangement. This proves that indeed there is a one-dimensional and increasing optimal cut.

Thus we establish that (5) holds and that the right-hand side in that inequality is precisely $S(c)$.

3.4 The remaining proofs

As for the remaining proofs in the paper: the construction of optimal Dictator-vs.-Gaussian tests from Gaussian mixture graphs mimics the proof of the Majority Is Stablest theorem using the 'Invariance Principle' from [MOO05]; the $\text{poly}(1/\epsilon)$ -time algorithm for computing $S(c)$ within ϵ , promised in Theorem 2.1, involves combining the Karush-Kuhn-Tucker conditions with Borell's theorem; and, the remaining work involves careful discretization arguments.

References

- [ABH⁺05] Sanjeev Arora, Eli Berger, Elad Hazan, Guy Kindler, and Muli Safra. On non-approximability for quadratic programs. In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 206–215, 2005.
- [AMMN06] Noga Alon, Konstantin Makarychev, Yury Makarychev, and Assaf Naor. Quadratic forms on graphs. *Inventiones Mathematicae*, 163(3):499–522, 2006.
- [AN06] Noga Alon and Assaf Naor. Approximating the Cut-Norm via Grothendieck's Inequality. *SIAM Journal on Computing*, 35(4):787–803, 2006.
- [AS00] Noga Alon and Benjamin Sudakov. Bipartite subgraphs and the smallest eigenvalue. *Combinatorics, Probability & Computing*, 9(1), 2000.
- [ASZ02] Noga Alon, Benny Sudakov, and Uri Zwick. Constructing worst case instances for semidefinite programming based approximation algorithms. *SIAM Journal on Discrete Mathematics*, 15(1):58–72, 2002.
- [Aus07a] Per Austrin. Balanced Max-2Sat might not be hardest. In *Proc. 39th ACM Symp. on the Theory of Computing*, pages 189–197, 2007.
- [Aus07b] Per Austrin. Towards sharp inapproximability for any 2-CSP. In *Proc. 48th IEEE Symp. on Foundations of Comp. Sci.*, 2007.

- [AZ05] Adi Avidor and Uri Zwick. Rounding two and three dimensional solutions of the SDP relaxation of MAX CUT. In *Proc. 8th APPROX-RANDOM*, pages 14–25, 2005.
- [BBC04] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56(1-3):89–113, 2004.
- [BC06] Mark Braverman and Stephen Cook. Computing over the reals: Foundations for scientific computing. *Notices of the AMS*, 53(3):318–329, 2006.
- [Bec92] William Beckner. Sobolev inequalities, the Poisson semigroup, and analysis on the sphere S^n . *Proc. Natl. Acad. Sci.*, 89:4816–4819, 1992.
- [BGJR88] Francisco Barahona, Martin Grötschel, Michael Jünger, and Gerhard Reinelt. An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research*, 36(3):493–513, 1988.
- [BGS98] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free bits, PCPs, and nonapproximability — towards tight results. *SIAM Journal on Computing*, 27(3):804–915, 1998.
- [Bor85] Christer Borell. Geometric bounds on the Ornstein-Uhlenbeck velocity process. *Z. Wahrsch. Verw. Gebiete*, 70(1):1–13, 1985.
- [CST01] Pierluigi Crescenzi, Riccardo Silvestri, and Luca Trevisan. On weighted vs unweighted versions of combinatorial optimization problems. *Information and Computation*, 167(1):10–26, 2001.
- [CW04] Moses Charikar and Anthony Wirth. Maximizing quadratic programs: Extending Grothendieck’s Inequality. In *Proc. 45th IEEE Symp. on Foundations of Comp. Sci.*, pages 54–60, 2004.
- [DF87] Persi Diaconis and David Freedman. A dozen Finetti-style results in search of a theorem. *Annals de l’I. H. P., sec. B*, 23(S2):397–423, 1987.
- [DP93a] Charles Delorme and Svatopluk Poljak. Combinatorial properties and the complexity of max-cut approximation. *Eur. J. Combinatorics*, 14(4):313–333, 1993.
- [DP93b] Charles Delorme and Svatopluk Poljak. Laplacian eigenvalues and the maximum cut problem. *Mathematical Programming*, 62:557–574, 1993.
- [EIO02] Lars Engebretsen, Piotr Indyk, and Ryan O’Donnell. Derandomized dimensionality reduction with applications. In *Proc. 13th ACM-SIAM Symp. on Discrete Algorithms*, pages 705–712, 2002.
- [FKL02] Uriel Feige, Marek Karpinski, and Michael Langberg. Improved approximation of Max-Cut on graphs of bounded degree. *J. Algorithms*, 43(2):201–219, 2002.
- [FL92] Uriel Feige and Lászlo Lovász. Two-prover one-round proof systems, their power and their problems. In *Proc. 24th ACM Symp. on the Theory of Computing*, pages 733–744, 1992.
- [FL06] Uriel Feige and Michael Langberg. The RPR² rounding technique for semidefinite programs. *J. Algorithms*, 60(1):1–23, 2006. First appeared ICALP 2001.
- [FS02] Uriel Feige and Gideon Schechtman. On the optimality of the random hyperplane rounding technique for Max-Cut. *Random Structures & Algorithms*, 20(3):403–440, 2002.
- [Gro53] Alexandre Grothendieck. Résumé de la théorie métrique des produits tensoriels topologiques. *Bol. Soc. Mat. Sao Paulo*, 8:1–79, 1953.
- [GW95] Michel Goemans and David Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.

- [Haa87] Uffe Haagerup. A new upper bound for the complex Grothendieck constant. *Israel J. of Math.*, 60:199–224, 1987.
- [Hås01] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.
- [HLZ04] Eran Halperin, Dror Livnat, and Uri Zwick. MAX CUT in cubic graphs. *J. Algorithms*, 53(2):169–185, 2004.
- [HV04] Johan Håstad and Srinivasan Venkatesh. On the advantage over a random assignment. *Random Structures & Algorithms*, 25(2):117–149, 2004.
- [Kar72] Richard Karp. *Reducibility among combinatorial problems*, pages 85–103. Plenum Press, 1972.
- [Kar99] Howard Karloff. How good is the Goemans-Williamson MAX CUT algorithm? *SIAM Journal on Computing*, 29(1):336–350, 1999.
- [KKMO07] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O’Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM Journal on Computing*, 37(1):319–357, 2007.
- [KO06] Subhash Khot and Ryan O’Donnell. SDP gaps and UGC-hardness for MaxCutGain. In *Proc. 47th IEEE Symp. on Foundations of Comp. Sci.*, pages 217–226, 2006.
- [KV05] Subhash Khot and Nisheeth Vishnoi. The Unique Games Conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1 . In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 53–62, 2005.
- [MOO05] Elchanan Mossel, Ryan O’Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 21–30, 2005. To appear, *Annals of Mathematics*.
- [MR99] Sanjeev Mahajan and Hariharan Ramesh. Derandomizing approximation algorithms based on semidefinite programming. *SIAM Journal on Computing*, 28(5):1641–1663, 1999.
- [PR95] Svatopluk Poljak and Franz Rendl. Nonpolyhedral relaxations of graph-bisection problems. *SIAM Journal on Optimization*, 5(3):467–487, 1995.
- [PT95] Svatopluk Poljak and Zsolt Tuza. The max-cut problem — a survey. In *Special Year on Combinatorial Optimization, DIMACS series in Discrete Mathematics and Theoretical Computer Science*. AMS, 1995.
- [RRW07] Franz Rendl, Giovanni Rinaldi, and Angelika Wiegele. A branch and bound algorithm for Max-Cut based on combining semidefinite and polyhedral relaxations. In *Proc. 12th IPCO*, pages 295–309, 2007.
- [SG76] Sartaj Sahni and Teofilo Gonzales. P-Complete approximation problems. *Journal of the ACM*, 23:555–565, 1976.
- [TSSW00] Luca Trevisan, Gregory Sorkin, Madhu Sudan, and David Williamson. Gadgets, approximation, and linear programming. *SIAM Journal on Computing*, 29(6):2074–2097, 2000.
- [Zwi99] Uri Zwick. Outward rotations: A tool for rounding solutions of semidefinite programming relaxations, with applications to MAX CUT and other problems. In *Proc. 31st ACM Symp. on the Theory of Computing*, pages 679–687, 1999.

4 $\text{Gap}_{\text{SDP}}(c) \leq S(c)$: Hermite analysis and Borell's rearrangement

In this section we prove $\text{Gap}_{\text{SDP}}(c) \leq S(c)$; i.e., we show that for each $c \in [\frac{1}{2}, 1]$ and $\eta > 0$, there exists a graph G exhibiting a large SDP gap: $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq S(c) + \eta$. We remind the reader here of the definition of $S(c)$:

$$S(c) = \inf_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r).$$

4.1 SDP gaps via Gaussian mixture graphs

As described in Sections 3.2 and 3.3, the graphs we use to exhibit SDP gaps will be high-dimensional Gaussian mixture graphs based on $(1, \rho_0)$ -distributions. Since these are infinite graphs, we will need to extend a number of our basic definitions, including ‘ $\text{Sdp}(G)$ ’ and ‘ $\text{Opt}(G)$ ’. The reader may object that these will not proper SDP gap examples because the graphs are infinite and also have self-loops (one might even object that the graphs are weighted). However in Appendix B we show that these issues can be circumvented:

Proposition 4.1 *Suppose $G = \mathcal{G}_P^{(d)}$ is a Gaussian mixture graph with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$. Then for any $\epsilon > 0$, there is a finite, self-loopless, unweighted graph G' (with $n = (1/\epsilon)^{O(d)}$ vertices) with $\text{Sdp}(G') \geq c - \epsilon$ and $\text{Opt}(G') \leq s + \epsilon$.*

The proof of this proposition essentially only uses straightforward, already-known ideas [FS02, ABH⁺05, KO06]. The reader should also note that arbitrarily small losses in c are also immaterial, since we can show (essentially a priori) that $\text{Gap}_{\text{SDP}}(c)$ is continuous:

Proposition 4.2 *The function Gap_{SDP} is continuous on $[\frac{1}{2}, 1]$, and strictly increasing from $\frac{1}{2}$ to 1.*

The proof of this proposition is in Appendix A.

Extending the basic Max-Cut definitions to infinite graphs is quite straightforward; see [KO06]. Here we will just treat the special case of Gaussian mixture graphs, which require a little extra care due to the fact that they can have ‘self-loops’. To begin, we define cut and value as before: A (fractional) cut for $\mathcal{G}_P^{(d)}$ is any measurable function $f: \mathbb{R}^d \rightarrow [-1, 1]$, and

$$\text{val}_{\mathcal{G}_P^{(d)}}(f) = \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\vec{x}, \vec{y}) \text{ } \rho\text{-corr'd} \\ d\text{-dim. Gaussians}}} \left[\frac{1}{2} - \frac{1}{2} f(\vec{x}) f(\vec{y}) \right].$$

Since we allow ‘self-loops’ (i.e., P ’s with probability mass on 1), one should note that we *can’t* necessarily find ‘proper’ cuts with value at least that of fractional cuts. We define $\text{Opt}(\mathcal{G}_P^{(d)})$ to be the supremum of the value over all *fractional* cuts.

Second, we define $\text{Sdp}(\mathcal{G}_P^{(d)})$ essentially as in the SDP (2):

$$\text{Sdp}(\mathcal{G}_P^{(d)}) = \sup_{g: \mathbb{R}^d \rightarrow B_d} \mathbf{E}_{(\mathbf{u}, \mathbf{v})} \left[\frac{1}{2} - \frac{1}{2} g(\mathbf{u}) \cdot g(\mathbf{v}) \right].$$

Some comments on this definition: Again, because of self-loops, it is not necessarily true that the optimal embedding g maps into the surface of the ball S^{d-1} . As it happens, though, we are only concerned with proving lower bounds on $\text{Sdp}(\mathcal{G}_P^{(d)})$, and the embeddings we will use happen to map into S^{d-1} anyway. Second, the most natural definition of $\text{Sdp}(G)$ for an ‘infinite graph’ G would allow embeddings into B_m and have an additional sup over $m \in \mathbb{N}$. But again, we will end up only considering embeddings $\mathbb{R}^d \rightarrow S^{d-1}$ for $\mathcal{G}_P^{(d)}$, so we choose to make the above simpler definition.

Having made these definitions, the goal of this section is to prove the following two theorems:

Theorem 4.3 *Let $G = \mathcal{G}_P^{(d)}$ be a d -dimensional Gaussian mixture graph, and let $c = \text{Spread}(P) = \mathbf{E}_{\rho \sim P} [\frac{1}{2} - \frac{1}{2} \rho]$. Then $\text{Sdp}(G) \geq c - O(\sqrt{\log d/d})$, via the embedding $g: \mathbb{R}^d \rightarrow S^{d-1}$ mapping x to $x/\|x\|$.⁶*

⁶ $g(0)$ can be set arbitrarily.

Theorem 4.4 *Let $G = \mathcal{G}_P^{(d)}$ be a d -dimensional Gaussian mixture graph for which P is a $(1, \rho_0)$ -distribution. Then the optimal fractional cut for G is achieved by an increasing, odd, ‘one-dimensional’ cut; i.e., a function $s : \mathbb{R}^d \rightarrow [-1, 1]$ of the form $s(x) = r(x_1)$, where $r : \mathbb{R} \rightarrow [-1, 1]$ is increasing and odd.*

Theorem 4.3 is just a calculation; the heart of the matter is Theorem 4.4.

Before proving these theorems, let us see how together they imply $\text{Gap}_{\text{SDP}}(c) \leq S(c)$. Let P be a $(1, \rho_0)$ -distribution achieving the inf in the definition of $S(c)$ to within ϵ . Now consider $G = \mathcal{G}_P^{(d)}$. By Theorem 4.3, $\text{Sdp}(G) \geq c - O(\sqrt{\log d/d})$. On the other hand, Theorem 4.4 implies that

$$\text{Opt}(G) \leq \sup_{\substack{s: \mathbb{R}^d \rightarrow [-1, 1] \\ \text{one-dimensional, increasing, odd}}} \text{val}_G(s).$$

But when s is one-dimensional, $s(x) = r(x_1)$, it’s immediate from the definitions that $\text{val}_G(s) = \text{val}_{\mathcal{G}_P^{(1)}}(r)$. Thus we have $\text{Opt}(G) \leq S(c) + \epsilon$.

Having determined this Gaussian mixture graph G with $\text{Sdp}(G) \geq c - O(\sqrt{\log d/d})$ and $\text{Opt}(G) \leq S(c) + \epsilon$, we are essentially done. Using Proposition 4.1 we can convert G to a finite, self-loopless graph G' with $\text{Sdp}(G') \geq c - O(\sqrt{\log d/d})$ and $\text{Opt}(G) \leq S(c) + 2\epsilon$; since $\epsilon > 0$ is arbitrary this proves that $\text{Gap}_{\text{SDP}}(c - O(\sqrt{\log d/d})) \leq S(c)$. Now by the continuity of Gap_{SDP} (Proposition 4.2), we conclude that $\text{Gap}_{\text{SDP}}(c) \leq S(c)$.

4.2 Proof of Theorem 4.3

Theorem 4.3 *Let $G = \mathcal{G}_P^{(d)}$ be a d -dimensional Gaussian mixture graph, and let $c = \text{Spread}(P) = \mathbf{E}_{\rho \sim P}[\frac{1}{2} - \frac{1}{2}\rho]$. Then $\text{Sdp}(G) \geq c - O(\sqrt{\log d/d})$, via the embedding $g : \mathbb{R}^d \rightarrow S^{d-1}$ mapping x to $x/\|x\|$.*

Proof: As stated, let $g(x) = x/\|x\|$, which maps \mathbb{R}^d onto S^{d-1} . (The value of $g(0)$ may be set arbitrarily since the probability that one of $\mathcal{G}_P^{(d)}$ ’s ‘edges’ involves 0 is 0.) We need to show:

$$\mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\vec{x}, \vec{y}) \\ d\text{-dim, Gaussians}}} \left[\frac{1}{2} - \frac{1}{2} \frac{\vec{x}}{\|\vec{x}\|} \cdot \frac{\vec{y}}{\|\vec{y}\|} \right] \geq \mathbf{E}_{\rho \sim P} \left[\frac{1}{2} - \frac{1}{2}\rho \right] - O(\sqrt{\log d/d}).$$

Clearly it suffices to prove the following:

$$\text{for all } \rho \in [-1, 1], \quad \mathbf{E}_{\substack{(\vec{x}, \vec{y}) \\ d\text{-dim, Gaussians}}} \left[\frac{\vec{x}}{\|\vec{x}\|} \cdot \frac{\vec{y}}{\|\vec{y}\|} \right] \leq \rho + O(\sqrt{\log d/d}). \quad (7)$$

This can be considered a standard probability result. Inside the expectation, in the numerator, we have

$$\vec{x} \cdot \vec{y} = \sum_{i=1}^n \mathbf{x}_i \mathbf{y}_i,$$

and the summands $\mathbf{x}_i \mathbf{y}_i$ are i.i.d. real-valued random variables. The expectation of $\mathbf{x}_i \mathbf{y}_i$ is ρ , and the variance and third absolute moment are bounded by absolute constants. Thus the Berry-Esseen theorem implies that $\mathbf{x} \cdot \mathbf{y}$ will be in the range $\rho d \pm O(\sqrt{d \log d})$ except with probability at most $O(1/\sqrt{d})$. In the denominator, it is well-known (and a similar argument shows) that $\|\vec{x}\|$ and $\|\vec{y}\|$ will each be in the range $\sqrt{d} \pm O(\sqrt{\log d})$ except with probability at most $O(1/\sqrt{d})$. Hence except with probability at most $O(1/\sqrt{d})$ we have that

$$\frac{\vec{x}}{\|\vec{x}\|} \cdot \frac{\vec{y}}{\|\vec{y}\|} \leq \frac{\rho d + O(\sqrt{d \log d})}{(\sqrt{d} - O(\sqrt{\log d}))(\sqrt{d} - O(\sqrt{\log d}))} \leq \rho + O(\sqrt{\log d/d}).$$

Since $\frac{\vec{x}}{\|\vec{x}\|} \cdot \frac{\vec{y}}{\|\vec{y}\|}$ is bounded above by 1 always, we gain at most $O(1/\sqrt{d})$ in the exceptional cases, and conclude that (7) indeed holds. \square

4.3 Proof of Theorem 4.4

Before proceeding with the proof of Theorem 4.4 we record here the basic facts from ‘Hermite analysis’ we will use throughout this work.

The space of functions $L^2(\mathbb{R}^d)$ under the Gaussian distribution has a countable orthonormal basis given by products of normalized Hermite polynomials. These products are indexed by vectors $S \in \mathbb{N}^d$; we use the notation $|S|$ for $\sum_{i=1}^d S_i$, which is also the degree of the product polynomial H_S . We can express any such function f via its ‘Hermite expansion’,

$$f(x) = \sum_{S \in \mathbb{N}^d} \widehat{f}(S) H_S(x),$$

with convergence in L^2 -norm. We make frequent use of the following definition:

Definition 4.5 *Given $f \in L^2(\mathbb{R}^d)$ and $\rho \in [-1, 1]$, the noise stability of f at ρ is*

$$\mathbb{S}_\rho(f) = \mathbf{E}_{\substack{(\vec{x}, \vec{y}) \\ d\text{-dim. Gaussians}}}^{\rho\text{-corr'd}} [f(\vec{x})f(\vec{y})].$$

(Note that we reversed the notational position of ρ and g in Section 3.3 for clarity of exposition.) The following basic facts about Hermite expansions are well known; see, e.g., [KO06] and the references therein.

Proposition 4.6

1. $\mathbb{S}_\rho(f) = \sum_{S \in \mathbb{N}^d} \rho^{|S|} \widehat{f}(S)^2$.
2. $\mathbb{S}_1(f) = \sum_{S \in \mathbb{N}^d} \widehat{f}(S)^2 = \mathbf{E}[f^2]$.
3. If f is an odd function (i.e., $f(-x) = -f(x)$), then $\widehat{f}(S) = 0$ unless $|S|$ is odd.
4. If f is an odd function then $\mathbb{S}_{-\rho}(f) = -\mathbb{S}_\rho(f)$.

We also immediately deduce the following fact:

Proposition 4.7 *Assume f is an odd function. Then as a function of ρ , $\mathbb{S}_\rho(f)$ is a power series with nonnegative coefficients, odd powers of ρ only, and radius of convergence at least 1. In particular it is an odd function of ρ , strictly increasing on $[-1, 1]$, 0 at 0, concave on $[-1, 0]$, and convex on $[0, 1]$.*

We now proceed with the proof:

Theorem 4.4 *Let $G = \mathcal{G}_P^{(d)}$ be a d -dimensional Gaussian mixture graph for which P is a $(1, \rho_0)$ -distribution. Then the optimal fractional cut for G is achieved by an increasing, odd, ‘one-dimensional’ cut; i.e., a function $s : \mathbb{R}^d \rightarrow [-1, 1]$ of the form $s(x) = r(x_1)$, where $r : \mathbb{R} \rightarrow [-1, 1]$ is increasing and odd.*

Proof: Suppose P has weight p on the point $-1 \leq \rho_0 \leq 0$ and weight $1 - p$ on the point 1. Let (f_i) be a sequence of measurable fractional cuts, $f_i : \mathbb{R}^d \rightarrow [-1, 1]$, for which $\text{val}_G(f_i) \nearrow \text{Opt}(G)$. We have

$$\text{val}_G(f_i) = \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\vec{x}, \vec{y}) \\ d\text{-dim. Gaussians}}}^{\rho\text{-corr'd}} \left[\frac{1}{2} - \frac{1}{2} f_i(\vec{x}) f_i(\vec{y}) \right] = \frac{1}{2} - \frac{1}{2} \mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(f_i)],$$

and hence

$$1 - 2\text{val}_G(f_i) = (1 - p)\mathbb{S}_1(f_i) + p\mathbb{S}_{\rho_0}(f_i). \quad (8)$$

Consider now replacing f_i by f_i^{odd} , the function $\mathbb{R}^d \rightarrow [-1, 1]$ given by $f_i^{\text{odd}}(x) = (f_i(x) - f_i(-x))/2$. It is well known that $\widehat{f_i^{\text{odd}}}(S)$ equals $\widehat{f_i}(S)$ for odd $|S|$ and is 0 for even $|S|$. Thus when we make this replacement, $\mathbb{S}_1(f_i) = \sum_S \widehat{f_i}(S)^2$ only decreases, and similarly $\mathbb{S}_{\rho_0}(f_i) = \sum_S \widehat{f_i}(S)^2 \rho_0^{|S|}$ only decreases (using the fact that

$\rho_0 \leq 0$). Thus (8) only decreases, and hence $\text{val}_G(f_i)$ can only increase. Thus we may assume each f_i is odd.

Given this assumption and using Proposition 4.6.4,

$$1 - 2\text{val}_G(f_i) = (8) = (1 - p)\mathbf{E}[f_i^2] - p\mathbb{S}_{-\rho_0}(f_i). \quad (9)$$

We now appeal to the Gaussian rearrangement inequality of Borell [Bor85], which implies that for any function $f_i \in L^2(\mathbb{R}^d)$ and any nonnegative ρ ,

$$\mathbb{S}_\rho(f_i) \leq \mathbb{S}_\rho(f_i^*);$$

here f_i^* is the Gaussian rearrangement of f_i , an increasing, one-dimensional function.⁷ Suppose then we replace each f_i by f_i^* . Since it holds that $\mathbf{E}[(f_i^*)^2] = \mathbf{E}[f_i^2]$, the first term in (9) does not change. But $-\rho_0$ is nonnegative, so we can use Borell's result to conclude that the second term $\mathbb{S}_{-\rho_0}(f_i)$ only increases. Hence (9) only decreases under Gaussian rearrangement and thus $\text{val}_G(f_i)$ only increases. Thus we may replace all of the f_i 's by their Gaussian rearrangements. Note that an odd function, when rearranged, is still odd.

We now have a sequence of one-dimensional, odd, increasing functions $r_i : \mathbb{R} \rightarrow [-1, 1]$, with $\text{val}_G(r_i) \nearrow \text{Opt}(G)$ (we abuse notation here slightly instead of writing $\text{val}_G(s_i)$ where $s_i : \mathbb{R}^d \rightarrow [-1, 1]$ is defined by $s_i(x) = r(x_1)$). It is well known that using a Helly-type proof we can pass to a subsequence that converges a.e. to an increasing, one-dimensional function r , which must also be odd. Dominated convergence then implies that $\text{val}_G(r) = \text{Opt}(G)$. \square

5 $\text{Gap}_{\text{SDP}}(c) \geq S(c)$: Discretized RPR² and Minimax

In this section we show that $\text{Gap}_{\text{SDP}}(c) \geq S(c)$. As described in Section 3.2, the idea will be to randomly find cuts in a given embedded graph by trying the RPR² algorithm with 'all' increasing, odd rounding functions. Of course, we actually only try 'all' such functions up to some discretization. Specifically:

Definition 5.1 Given $\epsilon > 0$, let \mathcal{I}_ϵ denote the partition of $\mathbb{R} \setminus \{0\}$ into intervals,

$$\mathcal{I}_\epsilon = \{\pm(-\infty, -B], \pm(-B, -B + \epsilon^2], \pm(-B + \epsilon^2, -B + 2\epsilon^2], \dots, \pm(-2\epsilon^2, \epsilon^2], \pm(-\epsilon^2, \epsilon^2)\},$$

where $B = B(\epsilon)$ is the smallest integer multiple of ϵ^2 exceeding $\sqrt{2 \ln(1/\epsilon)}$. We say that a function $r : \mathbb{R} \rightarrow [-1, 1]$ is ϵ -discretized if the following hold:

- r is identically -1 on $(-\infty, -B]$, 0 at 0 , and identically 1 on $[B, \infty)$.
- r 's values on the finite intervals in \mathcal{I}_ϵ are from the set $\epsilon\mathbb{Z} \cap (-1, 1)$.

Note that the number of different ϵ -discretized r 's is $2^{\tilde{O}(1/\epsilon^2)}$.

The main theorem we prove in this section is the following:

Theorem 5.2 There is a universal constant⁸ $K < \infty$ such that for all $c \in [\frac{1}{2}, 1]$,

$$\inf_{\substack{\text{discrete dists } P \text{ on } [-1, 1] \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{G_P}(r) \quad (10)$$

is within $\pm K\epsilon$ of

$$S(c) = \inf_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{G_P}(r).$$

⁷Borell only proves this for f_i Lipschitz and nonnegative, but both conditions are inessential; the first can be removed by standard approximation arguments and the second simply by adding a sufficiently large constant. Alternatively, one can use the alternate proof of Borell's theorem due to Beckner [Bec92].

⁸In future results in this section, different K 's may have different values; however they never depend on c or ϵ .

Aside from discretization issues, the main idea here is using Hermite analysis and the von Neumann Minimax Theorem to show that ‘worst’ ρ -distribution is a $(1, \rho_0)$ -distribution. Incidentally, the discretization issues are not just necessary because we want a finitary algorithm; in fact, discretization is also necessary for the employ of the Minimax Theorem (which also requires a finitary setting, or at least some kind of continuity and compactness).

Let us explain how we can use Theorem 5.2 algorithmically:

Theorem 5.3 *Let G be any (discrete) embedded graph with spread c . If we run Algorithm 3.4 on G , trying RPR^2 on G with all possible increasing, odd ϵ -discretized rounding functions r , then at least one will achieve, in expectation, a cut of value at least $S(c) - O(\epsilon)$. In particular, there exists a cut in G with value at least $S(c)$.*

Proof: Given any r , the observation (4) from Section 3.2 implies that $\text{Alg}_{RPR^2}(G) = \text{val}_{G_P}(r)$. Thus the suggested algorithm achieves at least (10), which by Theorem 5.2 is at least $S(c) - K\epsilon$. As for the last statement in the theorem, we’ve in particular shown that there exists some cut $f_\epsilon : V \rightarrow [-1, 1]$ with value at least $S(c) - K\epsilon$. Taking $\epsilon \rightarrow 0$ we can get a sequence of cuts f_i with $\limsup \text{val}_G(f_i) \geq S(c)$. But since each cut is just a point in the compact, finite-dimensional cube $[-1, 1]^{|V|}$ and since $\text{val}_G(\cdot)$ is continuous, we can extract a limiting cut f with value at least $S(c)$. \square

Corollary 5.4 *For each $c \in [\frac{1}{2}, 1]$ it holds that $\text{Gap}_{\text{SDP}}(c) \geq S(c)$. Indeed, there is an algorithm which, given any graph G with $\text{Sdp}(G) \geq c$ and any $\epsilon > 0$, runs in time $\text{poly}(|V|) \cdot 2^{\tilde{O}(1/\epsilon^2)}$ and with high probability outputs a proper cut in G with value at least $S(c) - \epsilon$.*

Proof: Given G , we can solve the semidefinite program and find an isomorphic embedded graph G' with spread at least c . It is quite easy to decrease the spread of an embedded graph arbitrarily; for example, map each $x \in S^{n-1}$ to $(tx, \sqrt{1-t^2}) \in S^n$ for a $t \in [0, 1]$ of one’s choosing. Thus we may assume that G' has spread exactly c . Now the algorithm from Theorem 5.3 (which has the dominating running time stated) is used to obtain a cut with value at least $S(c) - O(\epsilon)$. As $\epsilon > 0$ can be arbitrarily small, this establishes $\text{Gap}_{\text{SDP}}(c) \geq S(c)$.

Some minor algorithmic details are discussed more carefully in Appendix C. One we need to mention explicitly is that our algorithm cannot solve the SDP exactly. Instead, we can use it to find an isomorphic graph with spread exactly $c - \epsilon^2$. Then the algorithm will find a cut with value at least $S(c - \epsilon^2) - O(\epsilon)$. Since we now know $S = \text{Gap}_{\text{SDP}}$, we can inspect the proof of Proposition 4.2 and conclude that $S(c - \epsilon^2) \geq S(c) - O(\epsilon^2)$ if c is bounded away from 1, and we can use the fact that $\text{Gap}_{\text{SDP}}(1 - \delta) = 1 - \arccos(-1 + 2\delta)/\pi = 1 - \Theta(\sqrt{\delta})$ (from Goemans-Williamson) to conclude that $S(c - \epsilon^2) \geq S(c) - O(\epsilon)$ if c is close to 1. \square

We discuss the issue of the running time’s dependence of ϵ in Section 7.2.

Combining Corollary 5.4 with the results of Section 4 completes the proof that

$$\text{Gap}_{\text{SDP}}(c) = S(c).$$

The remainder of this section is devoted to proving Theorem 5.2. The proof will proceed by transforming (10) into $S(c)$ in several steps. Each step will modify the range of either the inf or sup, while changing the overall value by at most $K\epsilon$.

5.1 Discretizing distributions

The first step involves showing we can discretize the *distributions* P appearing in (10). This will facilitate our application of the Minimax Theorem.

Definition 5.5 *Let $c \in [\frac{1}{2}, 1]$ be given and fixed. We say that a discrete distribution P on $[-1, 1]$ is η -discretized if its support is contained in $\eta\mathbb{Z} \cup \{-1, 1\}$.*

Lemma 5.6 *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2}, 1]$,*

$$(10) = \inf_{\substack{\text{discrete dists } P \text{ on } [-1, 1] \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r)$$

is within $\pm K\epsilon$ of

$$\inf_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r). \quad (11)$$

Proof: In fact, (11) is clearly at least (10), since the inf is over a smaller set. To show the difference is at most $O(\epsilon)$ it suffices to show that every discrete distribution P on $[-1, 1]$ with mean $1 - 2c$ can be converted into an ϵ^7 -discretized distribution P' with mean $1 - 2c$ such that

$$\begin{aligned} & |\text{val}_{\mathcal{G}_P}(r) - \text{val}_{\mathcal{G}_{P'}}(r)| \leq O(\epsilon) \\ \Leftrightarrow & \left| \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \rho\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] - \mathbf{E}_{\rho \sim P'} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \rho\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] \right| \leq O(\epsilon) \end{aligned} \quad (12)$$

holds for every ϵ -discretized, increasing, odd r .

The conversion of P to P' proceeds as follows. For each atom ρ_i of P , choose $\rho'_i \leq \rho''_i$ to be the two values in $\epsilon^7\mathbb{Z} \cup \{-1, 1\}$ which straddle ρ_i as closely as possible. Write also $\rho_i = \lambda_i \rho'_i + (1 - \lambda_i) \rho''_i$, $\lambda_i \in [0, 1]$. We form P' by replacing each atom ρ_i with probability mass p_i in P with the pair of atoms ρ'_i, ρ''_i with masses $p_i \lambda_i, p_i(1 - \lambda_i)$, respectively. We have that P' is indeed an ϵ^7 -discretized distribution with the same mean as P , namely $1 - 2c$.

Note that $|\rho'_i - \rho_i|, |\rho''_i - \rho_i| \leq \epsilon^7$ always. It's easy now to see that (12) will follow if we can show

$$\left| \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \rho'_i\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] - \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \rho_i\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] \right| \leq O(\epsilon) \quad (13)$$

holds for all ϵ -discretized increasing odd r , using only $|\rho'_i - \rho_i| \leq \epsilon^7$. Now the left side of (13) is equal to $|\mathbb{S}_{\rho'_i}(r) - \mathbb{S}_{\rho_i}(r)|$, and r here is odd. Thus by the increasing/concavity/convexity properties of $\mathbb{S}_\rho(r)$ given in Proposition 4.7, we immediately see that the largest possible of $|\mathbb{S}_{\rho'_i}(r) - \mathbb{S}_{\rho_i}(r)|$ value would occur when $\rho'_i = 1$ and $\rho_i = 1 - \epsilon^7$ (or equivalently, $\rho'_i = -1, \rho_i = -1 + \epsilon^7$). Thus the proof of (13) and hence the theorem follows Claim 5.7 below. \square

Claim 5.7 *For every fixed ϵ -discretized, increasing, odd r ,*

$$\left| \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) 1\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] - \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) (1 - \epsilon^7)\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})] \right| \leq O(\epsilon).$$

Proof: Write $\eta = \epsilon^7$. Since 1-correlated Gaussians are identical, we are comparing

$$\mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) (1 - \eta)\text{-corr'd} \\ \text{Gaussians}}} [r(\mathbf{x})r(\mathbf{y})]$$

with $\mathbf{E}[r(\mathbf{x})^2]$. Using the fact that r is ϵ -discretized, it suffices to show that when (\mathbf{x}, \mathbf{y}) is a pair of $(1 - \eta)$ -correlated Gaussians, the probability that \mathbf{x} and \mathbf{y} land in different intervals from \mathcal{I}_ϵ (recall Definition 5.1) is at most $O(\epsilon)$. We will first give up on the half-infinite intervals in \mathcal{I}_ϵ ; using the fact that \mathbf{x} and \mathbf{y} are both individually distributed as Gaussians, the probability that either of them ends up at least $B \geq \sqrt{2 \ln(1/\epsilon)}$ in absolute value is at most $O(\epsilon)$ anyway. Also, the probability that either lands on 0 is 0. It remains to consider the intervals of the form $I = [t, t + \epsilon^2)$, where $0 \leq t < B$ (the case of negative intervals will be the

same). The probability density function for \mathbf{x} is nearly constant over the interval I ; in particular, the ratio between its values at t and $t + \epsilon^2$ is $\exp(\epsilon^2 t + \epsilon^4/2)$, which is close to 1 (since $t < B = O(\sqrt{\log(1/\epsilon)})$). Even just using that it is at most 2, we conclude that conditioned on \mathbf{x} falling into I , the probability that \mathbf{x} falls into $[t + 2\epsilon^3, t + \epsilon^2 - \epsilon^3]$ is at least $1 - O(3\epsilon^3/\epsilon^2) = 1 - O(\epsilon)$.

By losing $O(\epsilon)$ probability, we will assume this happens. In this case, \mathbf{y} is distributed as $(1 - \eta)\mathbf{x} + \sqrt{1 - (1 - \eta)^2}N(0, 1)$, where $N(0, 1)$ is a standard normal. Note that $(1 - \eta)\mathbf{x} = \mathbf{x} - \eta\mathbf{x} \geq \mathbf{x} - \eta B \geq \mathbf{x} - \epsilon^3$, since $\eta\mathbf{x} \leq \epsilon^7 B \ll \epsilon^3$. Hence we have $(1 - \eta)\mathbf{x} \in [t + \epsilon^3, t + \epsilon^2 - \epsilon^3]$. Given this, the conditional probability that \mathbf{y} won't also fall into I is at most the probability that $\sqrt{1 - (1 - \eta)^2}N(0, 1)$ will exceed ϵ^3 in absolute value. But the standard deviation of this normal is $O(\sqrt{\eta}) = O(\epsilon^{3.5})$, so the probability it will exceed ϵ^3 in absolute value is exponentially small in ϵ , certainly smaller than $O(\epsilon)$. Thus we've shown that except with probability at most $O(\epsilon)$, \mathbf{x} and \mathbf{y} will fall into the same interval from \mathcal{I}_ϵ , and this completes the proof of the claim. \square

5.2 Minimax

The next step in the proof of Theorem 5.2 is to reinterpret the space of ϵ^7 -discretized distributions P with mean $1 - 2c$:

Fact 5.8 *Any ϵ^7 -discretized distribution P with mean $1 - 2c$ can be expressed as a convex combination of 2-point ϵ^7 -discretized distributions each with mean $1 - 2c$ (and vice versa, clearly).*

Here, by a '2-point distribution' we mean one whose support is on at most two points (i.e., either one or two points).

Proof: This fact can be considered standard. One proof sketch is the following: Given any ϵ^7 -discretized P with mean $1 - 2c$, pick any two points which straddle $1 - 2c$ and on which P has positive probability mass (the two points may coincide in case P has mass on $1 - 2c$). Such a pair must exist because P has mean $1 - 2c$. Take the mean- $(1 - 2c)$ probability distribution over this pair and 'remove it from P ' (i.e., subtract and rescale) to the greatest extent possible. This will preserve the mean of P being $1 - 2c$, and it will also cause P to have support on (at least) one fewer point. Repeat this process until P is empty; the pairs extracted give the required combination of 2-point distributions. \square

The next step is to reverse the inf/min and max in (11) using the von Neumann Minimax theorem.

Lemma 5.9

$$(11) = \min_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) \quad (14)$$

$$= \max_{\substack{\text{probability distributions } R \text{ over} \\ \epsilon\text{-discretized, increasing} \\ \text{odd } r: \mathbb{R} \rightarrow [-1, 1]}} \min_{\substack{2\text{-point } \epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \mathbf{E}_{\mathbf{r} \sim R} [\text{val}_{\mathcal{G}_P}(\mathbf{r})]. \quad (15)$$

Proof: Note that (11), which has an inf, is not precisely the same as (14), which has a min. We will show that (11) equals (15) using the Minimax theorem. Since a corollary of the Minimax theorem is that the inf's and sup's involved are achieved, this will imply that (11) is equal to (14) and that we can write min and max everywhere.

Consider a zero-sum game between a 'Distribution Player' and a 'Function Player'. Acting simultaneously, the Distribution Player chooses a 2-point ϵ^7 -discretized probability distribution P with mean $1 - 2c$, and the Function Player chooses an increasing, odd, ϵ -discretized $r: \mathbb{R} \rightarrow [-1, 1]$. The payoff is $\text{val}_{\mathcal{G}_P}(r)$ to the Function Player from the Distribution Player.

Note that both players choose from a finite set of strategies; for the Distribution Player, this uses the fact that for any pair of discretized points, there is at most one distribution with mean $1 - 2c$ supported on

this pair. Therefore we may apply the von Neumann Minimax theorem. We conclude that the game has some value, which is achieved in both of the following scenarios: (a) the Function Player goes first and gets to choose a mixed strategy, and then the Distribution Player goes second and gets to choose a pure strategy; and, (b) the Distribution Player goes first and gets to choose a mixed strategy, and the Function Player goes second and gets to choose a pure strategy. The value in (a) is clearly (15). As for the value in (b), we claim it equals (14). This follows from Fact 5.8, along with the fact that if we identify a P with a convex combination of 2-point distributions Q , then for any r ,

$$\begin{aligned} \mathbf{E}_{Q \sim P} [\text{val}_{\mathcal{G}_Q}(r)] &= \mathbf{E}_{Q \sim P} \mathbf{E}_{\rho \sim Q} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \text{Gaussians}}} \mathbf{E}_{\rho\text{-corr'd}} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{x}) r(\mathbf{y}) \right] \\ &= \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \text{Gaussians}}} \mathbf{E}_{\rho\text{-corr'd}} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{x}) r(\mathbf{y}) \right] = \text{val}_{\mathcal{G}_P}(r). \end{aligned}$$

Hence (14) equals (15) and the proof is complete. \square

5.3 More Minimax; Convexity and Concavity

In the next step, we use the special properties of $\mathbb{S}_\rho(r)$ for odd r given in Proposition 4.7, along with further Minimax-based reasoning, to deduce that the ‘Distribution Player’ essentially may as well use a $(1, \rho_0)$ -distribution. This idea was discussed in Section 3.3.

Definition 5.10 *We say an ϵ^7 -discretized distribution P is almost- $(1, \rho_0)$ if it is the mixture of two $(1, \rho_0)$ -distributions for which the two ρ_0 values are neighboring (or equal) discretized values.*

Lemma 5.11

$$\begin{aligned} (14) &= \min_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) \\ &= \min_{\substack{\epsilon^7\text{-discretized almost-}(1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r). \quad (16) \end{aligned}$$

Proof: Let P^* denote an ϵ^7 -discretized distribution with mean $1 - 2c$ achieving the min in (14); i.e., an optimal mixed strategy for the Distribution Player. Let R^* denote a distribution over ϵ -discretized, increasing, odd r achieving the max in (15); i.e., an optimal mixed strategy for the Function Player. The Minimax Theorem further implies that P^* is an optimal strategy for the Distribution Player given that the Function Player uses R^* . I.e., P^* is a minimizing choice for P in the following:

$$\min_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \mathbf{E}_{\mathbf{r} \sim R^*} [\text{val}_{\mathcal{G}_P}(\mathbf{r})].$$

Now

$$\mathbf{E}_{\mathbf{r} \sim R^*} [\text{val}_{\mathcal{G}_P}(\mathbf{r})] = \mathbf{E}_{\mathbf{r} \sim R^*} \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \text{Gaussians}}} \mathbf{E}_{\rho\text{-corr'd}} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{x}) r(\mathbf{y}) \right] = \frac{1}{2} - \frac{1}{2} \mathbf{E}_{\rho \sim P} \mathbf{E}_{\mathbf{r} \sim R^*} [\mathbb{S}_\rho(\mathbf{r})],$$

and so it follows that P^* is a maximizing choice for P in the following:

$$\max_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \mathbf{E}_{\rho \sim P} \mathbf{E}_{\mathbf{r} \sim R^*} [\mathbb{S}_\rho(\mathbf{r})].$$

Suppose we fix a particular odd r . We now have the special properties of $\mathbb{S}_\rho(r)$ as a function of ρ given in Proposition 4.7. We also claim that the convexity and concavity of this function are essentially *strict*; i.e., $\mathbb{S}_\rho(r)$ is not linear on any open interval. For otherwise, by analyticity, $\frac{d^2}{d\rho^2} \mathbb{S}_\rho(r)$ would have to be 0 everywhere on $[-1, 1]$, implying that r is equal (in the L^2 sense) to a linear function. But an ϵ -discretized

function cannot be linear, since it is constantly -1 on $(-\infty, -B]$ and constantly 1 on $[B, \infty)$.

Next, note that all of the properties mentioned in Proposition 4.7 are maintained under finite convex combinations, in particular because first and second derivatives are linear. Hence if we define

$$q(\rho) = \mathbf{E}_{\mathbf{r} \sim R^*} [\mathbb{S}_\rho(\mathbf{r})],$$

we conclude that $q(\rho)$ is also an odd function of ρ , strictly increasing on $[-1, 1]$, 0 at 0 , concave on $[-1, 0]$, convex on $[0, 1]$, and not linear on any open interval. An illustration of what q may look like is given in Figure 1.

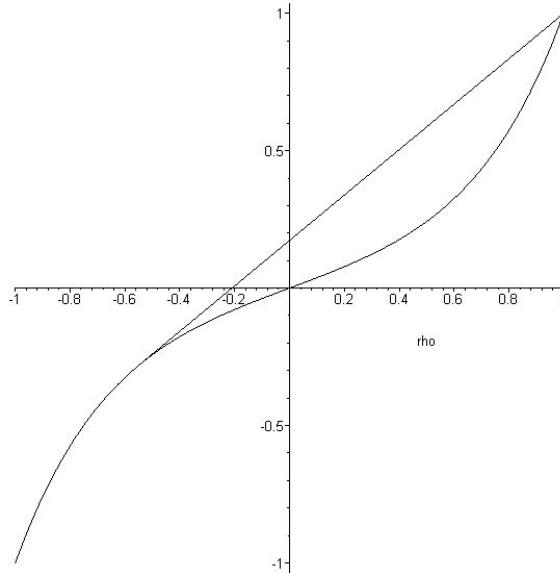


Figure 1: Illustrative $q(\rho)$, with least concave upper bound $\bar{q}(\rho)$.

Recall now that P^* is a maximizing choice for P in

$$\max_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1 - 2c}} \mathbf{E}_{\rho \sim P} [q(\rho)].$$

To complete the proof, we will show that this forces P^* to be almost- $(1, \rho_0)$. Suppose we first disregard the constraint of being ϵ^7 -discretized. Then it is easy to see that the maximum value in the above is equal to $\bar{q}(1 - 2c)$, where \bar{q} denotes the least concave upper bound of the function q . We have that \bar{q} equals q on some interval $[-1, \rho_0]$, where $\rho_0 < 0$, and is a straight line joining $q(\rho_0)$ and $q(1)$ on $[\rho_0, 1]$. Further, in this case there would be a unique maximizing P^* : either the 1-point distribution concentrated on $1 - 2c$, if $1 - 2c \leq \rho_0$, or the $(1, \rho_0)$ -distribution with mean $1 - 2c$, if $1 - 2c \geq \rho_0$.

Now we reintroduce the constraint that P^* must be ϵ^7 -discretized. Let \tilde{q} denote the piecewise linear function which interpolates q 's values on the discretized points $\epsilon^7\mathbb{Z}$. We now have that the maximum value of $\mathbf{E}_{\rho \sim P} [q(\rho)]$ is equal to $\tilde{q}(1 - 2c)$, where again \tilde{q} is the least concave upper bound of \tilde{q} . The function \tilde{q} is still odd, strictly increasing, concave on $[-1, 0]$, and convex on $[0, 1]$; hence again the function \tilde{q} equals q on some interval $[-1, \rho_0]$, where $\rho_0 < 0$, and is a straight line joining $q(\rho_0)$ and $q(1)$ on $[\rho_0, 1]$. The only difference now is that the point ρ_0 is not necessarily unique; there may be two consecutive possibilities, if the 'secant' at one of the possible ρ_0 's is parallel to one of the line segments touching $q(\rho_0)$. (Note that there cannot be more than two possible ρ_0 's, since otherwise the graph of q would have three distinct collinear points on $[-1, 0]$ and would thus be linear on some open interval.) We conclude that any maximizing P^*

must have all of its support among 1 and the (at most) two discretized values that straddle ρ_0 ; i.e., P^* must be almost- $(1, \rho_0)$. \square

Finally, we can convert almost- $(1, \rho_0)$ -distributions to $(1, \rho_0)$ -distributions:

Lemma 5.12 *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2}, 1]$,*

$$(16) = \min_{\substack{\epsilon^7\text{-discretized almost-}(1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r)$$

is within $\pm K\epsilon$ of

$$\min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r). \quad (17)$$

Proof: We sketch the proof, which uses the same ideas used in the proof of Lemma 5.6. We need to show that any almost- $(1, \rho_0)$ -distribution P with mean $1 - 2c$ can be converted into a $(1, \rho_0)$ -distribution P' with mean $1 - 2c$ in a such a way that $\text{val}(r)$ changes by at most $O(\epsilon)$ for every ϵ -discretized, increasing, odd r . If P is already a $(1, \rho_0)$ -distribution then we are done. Otherwise, it has support on two neighboring discretized values, say $\rho'_0 < \rho''_0$. Since the mean of P is $1 - 2c$ we must have $\rho'_0 < 1 - 2c$. We now form P' by pushing the weight λ that P gave to ρ''_0 onto ρ'_0 . This changes the mean by $\lambda(\rho''_0 - \rho'_0) \leq \epsilon^7$, but we can compensate for this by shifting a small amount of weight (at most $2\epsilon^7$) onto the support point 1. One bounds the change in $\text{val}(r)$ caused by these shifts by $O(\epsilon) + O(\epsilon^7)$ via $|\rho'_0 - \rho''_0| \leq \epsilon^7$ and Claim 5.7. \square

5.4 Undiscretizing

We have now reached (17), which is very close to $S(c)$; the only difference is that we have discretized distributions and functions. We now ‘undiscretize’:

Lemma 5.13 *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2}, 1]$,*

$$(17) = \min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r).$$

is within $\pm K\epsilon$ of

$$\inf_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) = S(c). \quad (18)$$

Proof: It is straightforward to see that the ideas from Lemma 5.6 can be used to replace the min in (17) with the inf from (18), changing the value of (17) by at most $O(\epsilon)$. Thus we concentrate on discretizing the functions. To that end, fix any $(1, \rho_0)$ -distribution P (in fact, our argument will hold for *any* distribution on $[-1, 1]$). We will show that for any increasing, odd $r: \mathbb{R} \rightarrow [-1, 1]$, there is an ϵ -discretized, increasing, odd $r': \mathbb{R} \rightarrow [-1, 1]$ with $|\text{val}_{\mathcal{G}_P}(r) - \text{val}_{\mathcal{G}_P}(r')| \leq O(\epsilon)$. This will complete the proof.

So let r be given. Define the increasing, odd, ϵ -discretized function $r': \mathbb{R} \rightarrow [-1, 1]$ as follows: On each finite interval I in \mathcal{I}_ϵ , we will take r' to be identically equal to the value of r on the midpoint of I ,⁹ rounded to the nearest integer multiple of ϵ (or ± 1 , if one of these is closer). As necessary, we will also take r' to be identically -1 on $(-\infty, -B]$ and identically 1 on $[B, \infty)$. We now argue that $\text{val}_{\mathcal{G}_P}(r')$ is within $\pm O(\epsilon)$ of $\text{val}_{\mathcal{G}_P}(r)$.

The idea is that $|r - r'| \leq \epsilon$ except on a set of small Gaussian measure. We will give up on the two half-infinite intervals and include them in the exceptional set. As for the finite intervals in \mathcal{I}_ϵ , since r is increasing and bounded in $[-1, 1]$, for at most $1/\epsilon$ of these intervals can r increase by more than ϵ . On the intervals where it increases by less than ϵ , we indeed have $|r - r'| \leq \epsilon$. Hence $|r - r'|$ fails on at most $1/\epsilon$

⁹Since we are working in $L^2(\mathbb{R})$, technically here we mean the value of any increasing representative of r 's equivalence class.

intervals of width ϵ^2 , plus perhaps the two half-infinite intervals $\pm(-\infty, B]$. Note that the total Gaussian measure of these intervals is at most $O(\epsilon)$. It is thus easy to see that

$$\text{val}_{\mathcal{G}_P}(r) = \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \text{Gaussians}}} \rho\text{-corr'd} \left[\frac{1}{2} - \frac{1}{2} r(\mathbf{x}) r(\mathbf{y}) \right]$$

is within $\pm O(\epsilon)$ of $\text{val}_{\mathcal{G}_P}(r')$: The probability that either \mathbf{x} or \mathbf{y} falls into the ‘bad’ intervals is at most $2 \cdot O(\epsilon)$, since \mathbf{x} and \mathbf{y} are each individually distributed as standard Gaussians. In this case, the difference in values is at most 1. Otherwise, we have that $|r(\mathbf{x}) - r'(\mathbf{x})|, |r(\mathbf{y}) - r'(\mathbf{y})| \leq \epsilon$, and then the difference in values is at most $O(\epsilon)$. \square

Combining all of the Lemmas 5.6, 5.9, 5.11, 5.12, 5.13, we have proved Theorem 5.2.

We end with the following observation:

Corollary 5.14 *Each sup in the definition of $S(c)$, as well as the inf, is achieved. Hence*

$$S(c) = \min_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \max_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r).$$

Proof: (Sketch.) The fact that the sup is achieved for each P is proved in Theorem 4.4. The fact that the inf is achieved can be deduced by taking a converging subsequence of ρ_0 ’s, and using the discretization Lemmas 5.6 and 5.13 to show that the max’s for close values of ρ_0 are close. \square

6 Estimating $S(c)$ efficiently

This section is devoted to the proof of Theorem 2.1:

Theorem 2.1 *There is an algorithm that, on input $c \in [\frac{1}{2}, 1]$ and $\epsilon > 0$, runs in time $\text{poly}(1/\epsilon)$ and computes $S(c)$ to within $\pm\epsilon$.*

As Lemma 5.13 shows, $S(c)$ is within $\pm O(\epsilon)$ of

$$(17) = \min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r).$$

Since we can enumerate all $\text{poly}(1/\epsilon)$ many ϵ^7 -discretized $(1, \rho_0)$ -distributions, it is clearly sufficient to show we can efficiently estimate

$$\max_{\substack{\epsilon\text{-discretized } r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) \tag{19}$$

for any $(1, \rho_0)$ -distribution P . In fact, for technical reasons, we will show how to estimate a slightly different quantity. Specifically, instead of using the rounding function discretization described in Definition 5.1, we will use a different one:

Definition 6.1 *Let $\epsilon > 0$ be such that $1/\epsilon^2$ is an odd integer. We define \mathcal{J}_ϵ to be the partition of \mathbb{R} into $1/\epsilon^2$ intervals of equal Gaussian measure ϵ^2 .¹⁰ We say that a function $r: \mathbb{R} \rightarrow [-1, 1]$ is ϵ^2 -equidiscretized if r is constant on each of the intervals in \mathcal{J}_ϵ .*

¹⁰Which partition points are included in which intervals is immaterial.

We will show how to estimate

$$\sup_{\substack{\epsilon^2\text{-equidiscretized } r:\mathbb{R}\rightarrow[-1,1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) \quad (20)$$

to within $\pm O(\epsilon)$ in time $\text{poly}(1/\epsilon)$, whenever P is a $(1, \rho_0)$ -distribution. Although this quantity is not directly comparable to (19), nevertheless with only minor modifications to the proof of Lemma 5.13 one can show that $S(c)$ is also within $\pm O(\epsilon)$ of

$$\min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \sup_{\substack{\epsilon^2\text{-equidiscretized } r:\mathbb{R}\rightarrow[-1,1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r).$$

(To see this, first note that the function discretization step hardly changes. Second, the proof of Lemma 5.6 goes through with ϵ^2 -equidiscretized functions as well because the intervals in \mathcal{J}_ϵ are only wider than the intervals in \mathcal{I}_ϵ .) Thus efficient estimation of (20) for $(1, \rho_0)$ -distributions is sufficient to establish Theorem 2.1.

The reason for our redefinition of discretization is the following: it allows us to drop the conditions ‘increasing, odd’ from the optimization problem (20). Specifically:

Proposition 6.2 *Let P be a $(1, \rho_0)$ -distribution and consider the following optimization problem:*

$$\sup_{\epsilon^2\text{-equidiscretized } r:\mathbb{R}\rightarrow[-1,1]} \text{val}_{\mathcal{G}_P}(r). \quad (21)$$

There exists an optimal solution r^ achieving the sup which is both increasing and odd.*

Proof: The proof is essentially identical to that of Theorem 4.4; the key point is that performing Gaussian rearrangement on an ϵ^2 -equidiscretized function yields another ϵ^2 -equidiscretized function. \square

We now consider (21). Suppose P has weight $1 - p$ on the point 1 and weight p on the point ρ_0 ; of course, $p = 2c/(1 - \rho_0)$. Let us index the intervals in \mathcal{J}_ϵ from left to right as I_{-m}, \dots, I_m , where $m = (1/\epsilon^2 - 1)/2$. We identify an ϵ^2 -equidiscretized function r with the length- $(2m + 1)$ vector giving its value on each interval; we will write r_j for the entry corresponding to I_j , $-m \leq j \leq m$. Finally, we write W_ρ for the $(2m + 1) \times (2m + 1)$ matrix whose (j, k) entry equals the probability that a ρ -correlated pair of Gaussians (\mathbf{x}, \mathbf{y}) will satisfy $\mathbf{x} \in I_j, \mathbf{y} \in I_k$. Now

$$\text{val}_{\mathcal{G}_P}(r) = \frac{1}{2} - \frac{1}{2} \left((1 - p) \sum_{-m \leq j, k \leq m} W_1(j, k) r_j r_k + p \sum_{-m \leq j, k \leq m} W_{\rho_0}(j, k) r_j r_k \right),$$

and hence the optimization problem (21) is equivalent to the problem

$$\begin{aligned} & \text{minimize} && r^\top ((1 - p)W_1 + pW_{\rho_0})r, \\ & \text{subject to} && -1 \leq r_j \leq 1 \quad \text{for all } -m \leq j \leq m. \end{aligned}$$

We now consider the Karush-Kuhn-Tucker conditions for this quadratic program and conclude that any optimal solution r must satisfy

$$\sum_{-m \leq k \leq m} ((1 - p)W_1(j, k) + pW_{\rho_0}(j, k))r_k = 0, \quad \text{for all } j \text{ such that } -1 < r_j < 1. \quad (22)$$

These necessary conditions for the optimality of a rounding function were already determined by Feige and Langberg [FL06].

The key observation that lets us make efficient use of the conditions is that we know from Proposition 6.2 that there is an optimal *increasing* odd r^* . In particular, there is some $0 \leq m_0 \leq m$ such that

$$\begin{aligned} r_j^* &= -1, && \text{for all } j < -m_0, \\ r_j^* &= 1, && \text{for all } j > m_0, \\ -1 < r_j^* &< 1, && \text{for all } -m_0 \leq j \leq m_0. \end{aligned} \quad (23)$$

Thus algorithmically, we can try all possible values for m_0 , incurring only an $O(1/\epsilon^2)$ factor slowdown. For each choice, we assume an r^* satisfying the conditions (23), and we solve (22) for the remaining unknown values; i.e., we solve the square system

$$\sum_{-m_0 \leq k \leq m_0} ((1-p)W_1(j, k) + pW_{\rho_0}(j, k))r_k = b_j \quad \text{for all } -m_0 \leq j \leq m_0, \quad (24)$$

where $b_j = \sum_{k < -m_0} ((1-p)W_1(j, k) + pW_{\rho_0}(j, k)) - \sum_{k > m_0} ((1-p)W_1(j, k) + pW_{\rho_0}(j, k))$. We are guaranteed that there exists an optimal, feasible solution r^* satisfying (24) for at least one value of m_0 .

6.1 Evading singularity

The above discussion suggests a $\text{poly}(1/\epsilon)$ time algorithm for computing (20) exactly. There are two problems we need to circumvent, however. The first problem is that, algorithmically, we cannot compute the values $W_\rho(j, k)$ — or even the endpoints of the intervals in \mathcal{J}_ϵ — exactly. The more challenging problem is that the square system (24) may be singular, in which case it may produce infinitely solutions that would need to be tried. As we will see, once we take care of the latter problem, the former will follow.

Let us write the square system (24) more compactly as

$$((1-p)M_{1, m_0} + pM_{\rho_0, m_0})s = b, \quad (25)$$

where M_{ρ, m_0} represents the square submatrix of W_ρ corresponding to indices $-m_0 \dots m_0$, and s represents the truncation of the vector r to these indices. We may assume here that $m_0 \geq 1$, since there is nothing to solve for if $m_0 = 0$ (note that r_0^* must be 0 by oddness). Write $M_{\rho_0, m_0, p} = (1-p)M_{1, m_0} + pM_{\rho_0, m_0}$.

We are concerned about the possibility that $\det(M_{\rho_0, m_0, p}) = 0$. More generally, we are concerned if the condition number $\kappa(M_{\rho_0, m_0, p})$ is very large, since in this case our inability to calculate the M_{ρ, m_0} matrices precisely would lead to very inaccurate solutions to (25). Since the matrix $M_{\rho_0, m_0, p}$ is symmetric, its condition number is

$$\kappa(M_{\rho_0, m_0, p}) = |\lambda_{\max}(M_{\rho_0, m_0, p})|/|\lambda_{\min}(M_{\rho_0, m_0, p})|,$$

where λ_{\max} and λ_{\min} denote largest and smallest eigenvalues in absolute value. Since each M_{ρ, m_0} is a submatrix of the stochastic matrix W_ρ , its maximum eigenvalue is at most 1; hence we need only worry about the smallest eigenvalue of $M_{\rho_0, m_0, p}$. Since M_{1, m_0} is a multiple of the identity matrix, it can be simultaneously diagonalized with M_{ρ_0, m_0} , and hence the eigenvalues of $M_{\rho_0, m_0, p}$ are precisely

$$\{(1-p) + p\lambda_{\rho_0, m_0}(j)\}_{-m_0 \leq j \leq m_0},$$

where the $\lambda_{\rho_0, m_0}(j)$'s are the eigenvalues of M_{ρ_0, m_0} . It is easy to see that for any particular $\lambda_{\rho_0, m_0}(j)$, the set of p 's for which $(1-p) + p\lambda_{\rho_0, m_0}(j)$ is in the range $(-\delta, \delta)$ is an interval of width at most 2δ . Hence we deduce the following:

Proposition 6.3 *For each ρ_0 , the set*

$$B_{\rho_0} := \bigcup_{1 \leq m_0 \leq m} \{p : \kappa(M_{\rho_0, m_0, p}) > 1/\delta\}$$

is a collection of at most $m \cdot (2m+1) = O(1/\epsilon^4)$ intervals of width at most 2δ each.

Our trick now will be to give up on these ‘bad’ p 's; or rather, the ‘bad’ c -values with which they are associated. Recalling the relationship $p = 2c/(1-\rho_0) \Leftrightarrow c = (1-\rho_0)p/2$, we have that

$$C := \bigcup_{\epsilon^7\text{-discretized}\rho_0} \{(1-\rho_0)p/2 : p \in B_{\rho_0}\}$$

is a collection of at most $O(1/\epsilon^{11})$ intervals of width at most 2δ each. And, whenever $c \notin C$, we are assured that the square system (25) has a matrix with condition number at most $1/\delta$.

We now set $\delta = \epsilon^{15}$ and use the following algorithm for estimating $S(c)$. Given c , we try to estimate $S(c')$ for all values $c' = c + t\epsilon^{14}$, for t an integer with $|t| \leq 1/\epsilon^{12}$. If we manage to succeed for some c' , then the resulting estimate for $S(c')$ will also be a $\pm O(\epsilon)$ estimate for $S(c)$, since $|c' - c| \leq \epsilon^2$ (and see the proof of Corollary 5.4 regarding the continuity of S). There are at most $O(1/\epsilon^{11})$ ‘bad’ intervals comprising C , and each has width at most 2δ . Since $2\delta \ll \epsilon^{14}$, each such interval contains at most one possible c' ; but, there are $2/\epsilon^{12} + 1 \gg O(1/\epsilon^{11})$ possible c' , and hence at least one choice must fall outside C . Hence we will succeed for at least one c' .

7 On $S(c)$ and running times

7.1 On $S(c)$

As we have shown, $S(c)$ can be computed to within $\pm\epsilon$ in time $\text{poly}(1/\epsilon)$; we believe this result justifies our claim that $S(c)$ is ‘explicit’. A reasonable way to understand the notion of ‘explicitness’ would be with respect to the ‘bit model’ of Braverman and Cook [BC06]; in that setting, our $\text{poly}(1/\epsilon)$ time algorithm would correspond to a fairly liberal notion of ‘explicit’, with a $\text{polylog}(1/\epsilon)$ time algorithm corresponding to a fairly demanding notion of ‘explicit’. The latter notion is the level of explicitness one has for, e.g., $\frac{1}{\pi} \arccos(1-2c)$. On the other hand, some less explicit-looking bounds have been given for related problems; for example, Haagerup’s bound [Haa87] for the complex Grothendieck constant is $8/\pi(k_0 + 1)$, where k_0 is the unique solution of the equation

$$\frac{\pi(k+1)}{8k} = \int_0^{\pi/2} \frac{\cos^2 t}{\sqrt{1-k^2 \sin^2 t}} dt$$

in the interval $[0, 1]$. This value can surely be computed to within $\pm\epsilon$ in time $\text{poly}(1/\epsilon)$; it may well also be computable in time $\text{polylog}(1/\epsilon)$ but this is, at least, not immediately obvious.

We in fact used the algorithm behind Theorem 2.1 to approximate $S(c)$ for the values .505, .510, .515, ..., .840 (with the values $S(.5) = .5$ and $S(c) = \arccos(1-2c)/\pi$ for $c \geq .844$ being already known). The values we found are given in the table in Appendix E. We were not completely formal about the approximation process and thus the results in Appendix E should not be considered rigorous. In particular, the approximations of the matrices W_ρ were done numerically in Matlab; also, the problem of singularity discussed in Section 6.1 did not seem to arise and so we disregarded it. We can also report that the best rounding functions r arising in the algorithm were very close to being s -linear, in all cases; they became only slightly rounded near $\pm s$ (convex near $-s$, concave near s).

7.2 On the running time of the rounding algorithm

As shown in Corollary 5.4, our Max-Cut rounding algorithm is efficient (polynomial) in terms of its dependence on n , the number of vertices; indeed, the running time is dominated by the time for semidefinite programming. To get a cut that is provably within ϵ of $S(\text{Opt}(G))$, however, our algorithm’s dependence on ϵ is exponential, $2^{\tilde{O}(1/\epsilon^2)}$. As we will discuss in Appendix C, all known RPR² algorithms have at least some ϵ dependence as well. This dependence is at least $\text{poly}(1/\epsilon)$, from converting expectation results to high probability results; in some papers, it is exponential (as in the derandomized Goemans-Williamson algorithm from [EIO02]).

In practice, we feel this issue is not very important. As mentioned in the previous section, we observed that using RPR² with s -linear rounding functions (as Feige and Langberg suggested) seems nearly optimal. In particular, it seems to achieve cuts that are within about 10^{-4} of $S(c)$, across all values of c . Further, one can precompute a table of which value of ‘ s ’ to use for ‘each’ possible value of c (suitably discretized) — and the algorithm knows what c is after solving the SDP. Thus in practice one can achieve within 10^{-4} of $S(c)$ with *no* real running time overhead. If error smaller than 10^{-4} is desired, it seems one can perform a local search for a better rounding function, starting from the appropriate s -linear function and modifying it slightly near $\pm s$.

Finally, given our $\text{poly}(1/\epsilon)$ time algorithm for approximating $S(c)$ to within $\pm\epsilon$, we believe that our rounding algorithm should also be able to have this improved dependence. Since this is not the main focus of our paper, we will only briefly describe the technicalities that would need to be overcome. Given an embedded graph G with ρ -distribution P , the idea would *not* be to try to solve the Karush-Kuhn-Tucker conditions for \mathcal{G}_P — since in general we have no promise that the optimal rounding function for \mathcal{G}_P is increasing, we wouldn't be able to effectively try all possibilities for where it is ± 1 . Instead, one might simply try to use all of the rounding functions constructed in the determination of $S(c)$. This seems as though it should work: the proof of Theorem 5.2 using the Minimax Theorem seems to imply that a convex combination of the optimal rounding functions for $(1, \rho_0)$ -distributions will achieve at least $S(c)$ for \mathcal{G}_P .

Unfortunately, several technical problems crop up. First, the Minimax proof only implies that ‘nearly’ $(1, \rho_0)$ -distributions are the worst case, and it is unclear if we can effectively enumerate these, since the weight to distribute to the three points is not completely determined by c . Second, even if we circumvent this problem, the Minimax theorem only implies that some convex combination of *all* the optimal rounding functions for $(1, \rho_0)$ -distribution will be good for \mathcal{G}_P ; however, our algorithm for computing $S(c)$ only finds the *increasing* ones. This problem too might be circumventable if one could prove *strict* increase in Borell's rearrangement inequality assuming the function is not already monotone. Such an ‘equality condition’ result is probably true, but is currently unknown. Finally, even if both of these issues were fixed, we still have the problem that the Karush-Kuhn-Tucker conditions might be a singular system and thus have multiple (and possibly very many) solutions, all of which theoretically might need to be combined by the ‘Function Player’.

8 Dictator-vs.-Gaussian tests

In this section we discuss Long Code tests and give the definitions necessary for our ‘Dictator-vs.-Gaussian’ tests. The subsequent two sections are devoted to the proof that $\text{Gap}_{\text{Test}}(c) = S(c)$.

We begin with an essential observation: 2-query Long Code tests are nothing more than embedded graphs (see Definition 3.1), with the vertex set being further restricted to lie within the discrete cube. To make the connection clearer, we treat the discrete cube as lying on the unit sphere:

Definition 8.1 We write $\mathbb{B}^n = \{-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}\}^n$ for the discrete cube, since it is convenient to have $\mathbb{B}^n \subseteq S^{n-1}$.

Definition 1.9 defines a 2-query, \neq -based Long Code test to be a probability distribution on pairs $(\mathbf{x}, \mathbf{y}) \in \mathbb{B}^n \times \mathbb{B}^n$. Since we think of the Long Code test as testing $f(\mathbf{x}) \neq f(\mathbf{y})$ and since \neq is symmetric, there is no loss in generality if we insist that the probability distribution be symmetric in \mathbf{x} and \mathbf{y} . But such a symmetric distribution on $\mathbb{B}^n \times \mathbb{B}^n$ is identical to a weighted undirected graph G on \mathbb{B}^n , with self-loops allowed. Note that this is an embedded graph, with the additional property that the vertex set is (a subset of) \mathbb{B}^n . Further, if $f : \mathbb{B}^n \rightarrow \{-1, 1\}$ is the function being tested, then $\frac{1}{2} - \frac{1}{2}f(\mathbf{x})f(\mathbf{y})$ is 1 if $f(\mathbf{x}) \neq f(\mathbf{y})$ and 0 if $f(\mathbf{x}) = f(\mathbf{y})$. Hence the probability that f passes the test is just $\text{val}_G(f)$. Extending this definition to functions $f : \mathbb{B}^n \rightarrow [-1, 1]$, we have the following:

Definition 8.2 A Dictator-vs.-Gaussian test for n -bit functions $f : \mathbb{B}^n \rightarrow [-1, 1]$ is an embedded graph T whose vertex set is \mathbb{B}^n . The value of the test on f is $\text{val}_T(f)$, and this is sometimes referred to as the probability that T passes/accepts f .

Our notion of the ‘completeness’ of a Dictator-vs.-Gaussian test is essentially as in Definition 1.10: the least probability with which one of the Dictators passes:

Definition 8.3 The i th Dictator function $\chi_i : \mathbb{B}^n \rightarrow \{-1, 1\}$ is defined by $\chi_i(x) = \sqrt{n} \cdot x_i$.

Definition 8.4 The completeness of an n -bit Dictator-vs.-Gaussian test T is

$$\text{Completeness}(T) = \min_{i \in [n]} \{\text{val}_T(\chi_i)\}$$

The average of the probabilities with which Dictators pass a test T is precisely its spread:

Proposition 8.5 *Given an n -bit Dictator-vs.-Gaussian test $T = (\mathbb{B}^n, E)$, we have*

$$\text{Spread}(T) = \text{avg}_{i \in [n]} \{\text{val}_T(\chi_i)\}.$$

Hence $\text{Spread}(T) \geq \text{Completeness}(T)$.

Proof:

$$\text{Spread}(T) = \mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim E} \left[\frac{1}{2} - \frac{1}{2} \mathbf{x} \cdot \mathbf{y} \right] = \mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim E} \left[\frac{1}{2} - \frac{1}{2} \sum_{i=1}^n \mathbf{x}_i \mathbf{y}_i \right] = \frac{1}{n} \sum_{i=1}^n \mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim E} \left[\frac{1}{2} - \frac{1}{2} \mathbf{x}_i \mathbf{y}_i \right] = \text{avg}_{i \in [n]} \{\text{val}_T(\chi_i)\}.$$

□

As discussed in Section 1.5 we use a weakened soundness notion for Dictator-vs.-Gaussian tests; specifically, these tests only need to reject functions that are sufficiently ‘Gaussianic’. This soundness condition allows us to get large completeness/soundness gaps despite using only 2 queries, and is also precisely what is needed for Theorems 1.13, 1.14, and 1.15. The notion of being ‘Gaussianic’ is, for all intents and purposes, the same as the notion of having small ‘low-degree influences’ introduced in [KKMO07] and used in previous papers on Unique Games-hardness. We use the very slightly different notion of Gaussianic functions because we feel it is more natural. To make this definition we need to recall the basics of Fourier analysis of boolean functions.

Analogous to the Hermite analysis described in Section 4.3, the space of functions $L^2(\mathbb{B}^n)$ under the uniform distribution has a complete orthonormal basis given by the monomials $(\chi_S)_{S \subseteq [n]}$:

$$\chi_S(x) = \prod_{i \in S} (\sqrt{n} \cdot x_i).$$

One can uniquely express any function $f : \mathbb{B}^n \rightarrow \mathbb{R}$ via its Fourier expansion,

$$f = \sum_{S \subseteq [n]} \hat{f}(S) \chi_S.$$

We now introduce Gaussianic functions:

Definition 8.6 *For $0 \leq \epsilon, \delta \leq 1$, we say a function $f : \mathbb{B}^n \rightarrow [-1, 1]$ is (ϵ, δ) -Gaussianic if for each $i \in [n]$,*

$$\text{Inf}_i^{(1-\delta)}(f) \leq \epsilon,$$

where we define the $(1 - \delta)$ -attenuated influence of i on f to be

$$\text{Inf}_i^{(1-\delta)}(f) = \sum_{\substack{S \subseteq [n] \\ i \in S}} (1 - \delta)^{|S|-1} \hat{f}(S)^2.$$

Note that this definition becomes stricter when ϵ or δ decreases; we think of functions as being ‘more Gaussianic’ when δ and (especially) ϵ are small. As an example, Dictator functions χ_i are the antithesis of being Gaussianic; in particular, if $\epsilon < 1$ then χ_i is not (ϵ, δ) -Gaussianic even for $\delta = 1$.¹¹ On the other hand, the Majority function is extremely Gaussianic; specifically, $(O(\frac{1}{\sqrt{n}}), 0)$ -Gaussianic. The name ‘Gaussianic’ was chosen based on the ‘Invariance Principle’ from [MOO05], which essentially states that if $f : \mathbb{B}^n \rightarrow [-1, 1]$ is very Gaussianic, then the distribution of

$$\sum_{S \subseteq [n]} \hat{f}(S) \prod_{i \in S} \mathbf{X}_i$$

is nearly unchanged whether one takes the \mathbf{X}_i ’s to be independent ± 1 bits or independent $N(0, 1)$ Gaussians.

Having defined Gaussianic functions, we give the soundness notion for our tests:

¹¹We take $0^0 = 1$ in the definition.

Definition 8.7 The (ϵ, δ) -soundness of a Dictator-vs.-Gaussianic test T for functions $f : \{-1, 1\}^n \rightarrow [-1, 1]$ is

$$\text{Soundness}_{\epsilon, \delta}(T) = \max\{\text{val}_T(f) : f \text{ is } (\epsilon, \delta)\text{-Gaussianic}\}.$$

Given this definition, the most natural Property Testing question to ask is how far apart completeness and soundness can be for Dictator-vs.-Gaussianic tests:

Definition 8.8 We call the pair (c, s) a Dictator-vs.-Gaussianic test (ϵ, δ) -gap if for all sufficiently large n , there is a Dictator-vs.-Gaussianic test $T^{(n)}$ for functions $f : \mathbb{B}^n \rightarrow [-1, 1]$ with $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}_{\epsilon, \delta}(T^{(n)}) \leq s$. We call the pair (c, s) simply a Dictator-vs.-Gaussianic test gap if $\forall \eta > 0, \exists \epsilon, \delta > 0$ such that $(c, s + \eta)$ is a Dictator-vs.-Gaussianic test (ϵ, δ) -gap.

Definition 8.9 The Dictator-vs.-Gaussianic gap curve is the function $\text{Gap}_{\text{Test}} : [\frac{1}{2}, 1] \rightarrow [\frac{1}{2}, 1]$ defined by

$$\text{Gap}_{\text{Test}}(c) = \min\{s : (c, s) \text{ is a Dictator-vs.-Gaussianic test gap}\}.$$

(It is immediate from the definitions that this min is achieved; i.e., we needn't write inf.)

The reader is reminded that in Section 1.5 we described three applications of establishing a (c, s) Dictator-vs.-Gaussianic test gap, Theorems 1.13, 1.14, and 1.15. In the next section we will show that $\text{Gap}_{\text{Test}}(c) \leq S(c)$; substituting this into these theorems yields our results from Section 2; the subsequent section will be devoted to the inequality $\text{Gap}_{\text{Test}}(c) \geq S(c)$, whose proof completes the result $\text{Gap}_{\text{Test}}(c) = S(c)$. Although the inequality $\text{Gap}_{\text{Test}}(c) \geq \text{Gap}_{\text{SDP}}(c)$ was already implicitly proved in [KV05], we will give an alternate direct proof which clarifies the connection between SDP rounding algorithms and Dictator-vs.-Gaussianic testing. Finally, in the last section we will connect Dictator-vs.-Gaussianic tests with the SDP-hardness constructions in [Kar99, AS00, ASZ02] and prove Theorem 1.15, extending a result of Feige and Schechtman [FS02].

9 $\text{Gap}_{\text{Test}}(c) \leq S(c)$: Invariance Principle

To upper-bound $\text{Gap}_{\text{Test}}(c)$, we need to determine Dictator-vs.-Gaussianic tests with completeness at least c for which all Gaussianic functions pass with small probability. Studying just how small this soundness can be is very similar to searching for the largest possible SDP gap, discussed in Section 3. For example, given a particular test T on \mathbb{B}^n with $\text{Completeness}(T) \geq c$ and $\text{Soundness}_{\epsilon, \delta}(T) \leq s$, one can symmetrize it with respect to all $2^n n!$ symmetries of \mathbb{B}^n , forming T' . Then one still has $\text{Completeness}(T') \geq c$ and $\text{Soundness}_{\epsilon, \delta}(T') \leq s$, and furthermore T' has the property that the probability of choosing a pair (x, y) depends only on its Hamming distance; i.e., only on $\langle x, y \rangle$. Just as we switched from $\mathcal{S}_P^{(d)}$ (which insisted on $\langle x, y \rangle$ being precisely ρ) to the analytically-easier $\mathcal{G}_P^{(d)}$, it is natural to switch to the version of symmetrized tests with independence across coordinates:

Definition 9.1 We define the noise sensitivity mixture test $\mathcal{T}_P^{(n)}$ on \mathbb{B}^n by analogy with Gaussian mixture graphs. In particular we define (\mathbf{x}, \mathbf{y}) to be ρ -correlated n -bit strings if \mathbf{x} is drawn uniformly from \mathbb{B}^n and \mathbf{y} is formed by taking $\mathbf{y}_i = \mathbf{x}_i$ with probability $\frac{1}{2} + \frac{1}{2}\rho$ and $\mathbf{y}_i = -\mathbf{x}_i$ with probability $\frac{1}{2} - \frac{1}{2}\rho$, independently across i .

We remark that a ρ -correlated pair (\mathbf{x}, \mathbf{y}) has $\langle \mathbf{x}, \mathbf{y} \rangle$ tightly concentrated around ρ , and that further:

Fact 9.2 $\text{Completeness}(\mathcal{T}_P^{(n)}) = \text{Spread}(P) = \mathbf{E}_{\rho \sim P}[\frac{1}{2} - \frac{1}{2}\rho]$.

Also, given $f : \mathbb{B}^n \rightarrow \mathbb{R}$ we use the notation

$$\mathbb{S}_\rho(f) = \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \rho\text{-corr'd} \\ n\text{-bit strings}}} [f(\mathbf{x})f(\mathbf{y})].$$

The reader is warned that we use the notation $\mathbb{S}_\rho(f)$ for both $f : \mathbb{B}^n \rightarrow \mathbb{R}$ and $f \in L^2(\mathbb{R}^n)$ with the Gaussian distribution. For more on noise sensitivity tests, see [KKMO07].

Having decided that the best Dictator-vs.-Gaussianic gaps will occur essentially with noise sensitivity mixture tests, the ideas from Section 3.3 again apply. The Hermite and Fourier formulas for noise stability are the same and we again conclude that the optimal mixture should come from a $(1, \rho_0)$ -distribution. This provides an explanation for why such tests were useful in [KO06].

Finally, to upper-bound the value of Gaussianic functions on noise sensitivity $(1, \rho_0)$ -mixture tests, we use the Invariance Principle of [MOO05] to reduce to the analysis of the Max-Cut in Gaussian mixture graphs. Then Theorem 4.4 can be used to get an upper bound of $S(c)$. More precisely, we prove the following theorem:

Theorem 9.3 *Let P be any $(1, \rho_0)$ -distribution and let T denote the Dictator-vs.-Gaussianic test $\mathcal{T}_P^{(n)}$. Then for any $\tau > 0$,*

$$\text{Soundness}_{\tau, \Omega(1/\log(1/\tau))}(T) \leq \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) + O(\log(1/\tau)^{-1/8}).$$

Before proving Theorem 9.3, let us see how it implies the desired result:

Corollary 9.4 $\text{Gap}_{\text{Test}}(c) \leq S(c)$.

Proof: Let P be the $(1, \rho_0)$ -distribution with mean $1 - 2c$ achieving the minimum in the definition of $S(c)$ (or rather, in Corollary 5.14). Writing $T = \mathcal{T}_P^{(n)}$, we have $\text{Completeness}(T) = c$ by Fact 9.2. Now by definition,

$$\sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r)$$

is precisely $S(c)$. Hence Theorem 9.3 implies that the (ϵ, δ) -soundness of T can be made at most $S(c)$ plus an arbitrarily small amount, by taking ϵ and δ sufficiently small. This establishes $\text{Gap}_{\text{Test}}(c) \leq S(c)$. \square

9.1 Proof of Theorem 9.3

The proof is an extension of the proof of the Majority Is Stablest theorem from [MOO05]. Let P , T , and τ be as in the statement of the theorem, and let $f: \mathbb{B}^n \rightarrow [-1, 1]$ be a $(\tau, \Omega(1/\log(1/\tau)))$ -Gaussianic function. We need to show that

$$\text{val}_T(f) = \mathbf{E}_{\rho \sim P} \mathbf{E}_{\substack{(\mathbf{x}, \mathbf{y}) \\ \rho\text{-corr'd} \\ n\text{-bit strings}}} \left[\frac{1}{2} - \frac{1}{2} f(\mathbf{x}) f(\mathbf{y}) \right] = \frac{1}{2} - \frac{1}{2} \mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(f)],$$

is, up to an additive $O(\log(1/\tau)^{-1/8})$, at most

$$\sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_P}(r) = \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \left(\frac{1}{2} - \frac{1}{2} \mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(r)] \right).$$

Equivalently, we must show

$$\mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(f)] \geq \inf_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(r)] - O(\log(1/\tau)^{-1/8}). \quad (26)$$

Let us write p for the weight of P on ρ_0 . Then the left side of (26) is

$$(1 - p)\mathbf{E}[f^2] + p\mathbb{S}_{\rho_0}(f).$$

As in the proof of Theorem 4.4, this quantity can only decrease if we replace f by f^{odd} , in which case it becomes

$$(1 - p)\mathbf{E}[f^2] - p\mathbb{S}_{-\rho_0}(f), \quad (27)$$

analogous to (9). (Note that a similar formula will arise on the right side of (26), since the r 's are odd.) Since f^{odd} has the same Fourier expansion as f except with the even-degree terms dropped, we have that $\text{Inf}_i^{(1-\delta)}(f^{\text{odd}}) \leq \text{Inf}_i^{(1-\delta)}(f)$, and hence $f = f^{\text{odd}}$ is still (ϵ, δ) -Gaussianic.

We now set $\gamma = O\left(\frac{\log \log(1/\tau)}{\log(1/\tau)}\right)$ and distinguish the two cases $\rho_0 \leq -1 + 3\gamma$ and $\rho_0 > -1 + 3\gamma$:

Case 1: $\rho_0 \leq -1 + 3\gamma$. In this case we use $\mathbb{S}_{-\rho_0}(f) \leq \mathbb{S}_1(f) = \mathbf{E}[f^2]$ to deduce that (27) is at least $1 - 2p$. On the other hand, by taking $r = \text{sgn}$ (which is increasing and odd), we conclude that the term on the right side of (26) satisfies

$$\inf_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \mathbf{E}_{\rho \sim P} [\mathbb{S}_\rho(r)] \leq (1-p)\mathbf{E}[\text{sgn}^2] - p\mathbb{S}_{-\rho_0}(\text{sgn}) = (1-p) - p(1 - \Theta(\sqrt{\gamma})) = 1 - 2p + \Theta(\sqrt{\gamma}),$$

where we used the estimate $\mathbb{S}_{1-\delta}(\text{sgn}) = 1 - \Theta(\sqrt{\delta})$. Since $\Theta(\sqrt{\gamma}) \ll O(\log(1/\tau)^{-1/8})$, the proof of (26) in this case is complete.

Case 2: $\rho_0 > -1 + 3\gamma$. In this case we follow the arguments from [MOO05]'s proof of the Majority Is Stablest theorem. Write $\rho = -\rho_0 < 1 - 3\gamma$, and express $\rho = \rho' \cdot (1 - \gamma)^2$. We let $g \in L^2(\mathbb{R}^n)$ be the multilinear polynomial

$$g(x_1, \dots, x_n) = \sum_{S \subseteq [n]} (1 - \gamma)^{|S|} \widehat{f}(S) \prod_{i \in S} x_i,$$

and we let $\tilde{g}: \mathbb{R}^n \rightarrow [-1, 1]$ be the function defined by

$$\tilde{g}(\vec{x}) = \begin{cases} g(\vec{x}) & \text{if } |g(\vec{x})| \leq 1, \\ \text{sgn}(g(\vec{x})) & \text{else.} \end{cases}$$

We note that f being odd implies that both g and \tilde{g} are odd. Since

$$\mathbf{E}[f^2] = \sum_{S \subseteq [n]} \widehat{f}(S)^2 = \sum_{S \in \mathbb{N}^n} \widehat{g}(S)^2 = \mathbf{E}[g^2] \geq \mathbf{E}[\tilde{g}^2],$$

we have

$$(27) \geq (1-p)\mathbf{E}[\tilde{g}^2] - p\mathbb{S}_\rho(f).$$

Further, using the fact that f is $(\tau, \Omega(1/\log(1/\tau)))$ -Gaussianic, the Invariance Principle-based arguments in [MOO05] imply that

$$|\mathbb{S}_\rho(f) - \mathbb{S}_{\rho'}(\tilde{g})| \leq \tau^{\Omega(\gamma)}.$$

Hence we have

$$(27) \geq (1-p)\mathbf{E}[\tilde{g}^2] - p\mathbb{S}_{\rho'}(\tilde{g}) - \tau^{\Omega(\gamma)} = (1-p)\mathbf{E}[\tilde{g}^2] + p\mathbb{S}_{-\rho'}(\tilde{g}) - \tau^{\Omega(\gamma)} = \left(1 - 2\text{val}_{\mathcal{G}_{\rho'}^{(n)}}(\tilde{g})\right) - \tau^{\Omega(\gamma)},$$

where the first equality uses the fact that \tilde{g} is odd and where P' the probability distribution that puts weight $1 - p$ on 1 and weight p on $-\rho'$. But P' is a $(1, \rho_0)$ -distribution', and hence Theorem 4.4 implies that

$$\text{val}_{\mathcal{G}_{P'}^{(n)}}(\tilde{g}) \leq \sup_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \text{val}_{\mathcal{G}_{P'}^{(n)}}(r).$$

Thus we have

$$(27) \geq \inf_{\substack{r: \mathbb{R} \rightarrow [-1, 1] \\ \text{increasing, odd}}} \mathbf{E}_{\rho \sim P'} [\mathbb{S}_\rho(r)] - \tau^{\Omega(\gamma)}.$$

By taking the constant in the definition of γ large enough we get $\tau^{\Omega(\gamma)} \ll O(\log(1/\tau)^{-1/8})$. Thus to complete the proof of (26), we only need to relate the inf with P to the inf with P' , using the fact that $|(-\rho') - \rho_0| \leq O(\gamma)$. This can be done by using the discretization Lemmas 5.6 and 5.13; the resulting error term is at most $O(\gamma^{1/7}) \leq O(\log(1/\tau)^{-1/8})$, as required.

10 $\text{Gap}_{\text{Test}}(c) \geq S(c)$: RPR² algorithms imply testing lower bounds

In this section we discuss ‘lower bounds’ for the Dictator-vs.-Gaussian testing problem; i.e., proofs that any test T with $\text{Completeness}(T) = c$ cannot have $\text{Soundness}_{\epsilon, \delta}(T)$ which is too small. As mentioned earlier, Khot and Vishnoi’s Theorem 1.14 can be used to get such lower bounds: it gives a long translation of a (c, s) Dictator-vs.-Gaussian test gap into a $(c - \eta, s + \eta)$ SDP gap (with triangle inequality, even), for arbitrarily small η . This means that an SDP-rounding guarantee can be used to rule out the existence of strong Dictator-vs.-Gaussian tests. A similar idea arises from the earlier Theorem 1.13, which shows that a (c, s) Dictator-vs.-Gaussian test gap can be translated into a $c - \eta$ vs. $s + \eta$ UGC-hardness result for Max-Cut. Since one feels it is unlikely that the Unique Games Conjecture would be disproved via an elaborate reduction to Max-Cut followed by a too-strong SDP-rounding algorithm, Theorem 1.13 also suggests that SDP-rounding algorithms should be able to prove Dictator-vs.-Gaussian testing lower bounds.

In this section we show explicitly and directly that RPR² algorithms give rise to Dictator-vs.-Gaussian testing lower bounds. More specifically, the following theorem implies (and indeed is slightly stronger than) the result $\text{Gap}_{\text{Test}}(c) \geq S(c)$:

Theorem 10.1 *Let $\epsilon > 0$ be given. Then for all $n \geq O(1/\epsilon^7)$, if T is any Dictator-vs.-Gaussian test for functions $f : \mathbb{B}^n \rightarrow [-1, 1]$ satisfying $\text{Completeness}(T) \geq c$, then $\text{Soundness}_{\epsilon, 0}(T) \geq S(c) - \epsilon$.*

Proof: Let T be a such a test. As described in Section 8, T can be thought of as an embedded graph on the vertex set $\mathbb{B}^n \subseteq S^{n-1}$. Write P for the ρ -distribution of T , and recall from Proposition 8.5 that $\text{Spread}(P) \geq \text{Completeness}(T) \geq c$.

Imagine we now run our RPR² Algorithm 3.4 on T , with the discretization parameter set to $\epsilon' := \epsilon/K$. By Theorem 5.3, it will at some point hit upon an ϵ' -discretized, increasing, odd rounding function $r^* : \mathbb{R} \rightarrow [-1, 1]$ which satisfies

$$\text{Alg}_{\text{RPR}^2}(T) = \text{val}_{G_P}(r^*) \geq S(\text{Spread}(P)) - O(\epsilon') \geq S(c) - \epsilon/2, \quad (28)$$

assuming K is a sufficiently large constant. (Here we also used that S is increasing.) Recall that when we run the RPR² algorithm with r^* , it chooses a random n -dimensional Gaussian \vec{Z} and outputs the fractional cut $f_{\vec{Z}} : \mathbb{B}^n \rightarrow [-1, 1]$ defined by

$$f_{\vec{Z}}(x) = r^*(x \cdot \vec{Z}).$$

Thus (28) is equivalent to

$$\mathbf{E}_{\vec{Z}}[\text{val}_T(f_{\vec{Z}})] \geq S(c) - \epsilon/2.$$

Our goal is now to show the intuitively plausible claim that that $f_{\vec{Z}}$ is very likely to be a Gaussian boolean function:

Claim 10.2 *With probability at least $1 - O(1/n)$ over the choice of \vec{Z} , the function $f_{\vec{Z}}$ is $(O(\sqrt{\ln n/n})/\epsilon^2, 0)$ -Gaussian.*

With our choice of $n \geq O(1/\epsilon^7)$, this claim implies that with probability at least $1 - \epsilon/2$ the function $f_{\vec{Z}}$ is $(\epsilon, 0)$ -Gaussian. This in turn completes the proof of the theorem, since it implies

$$\mathbf{E}_{\vec{Z}}[\text{val}_T(f_{\vec{Z}}) \mid f_{\vec{Z}} \text{ is } (\epsilon, 0)\text{-Gaussian}] \geq S(c) - \epsilon/2 - \epsilon/2.$$

Thus there must exist an $(\epsilon, 0)$ -Gaussian $f : \mathbb{B}^n \rightarrow [-1, 1]$ with $\text{val}_T(f) \geq S(c) - \epsilon$, and we conclude that $\text{Soundness}_{\epsilon, 0}(T) \geq S(c) - \epsilon$ as needed.

Proof: (of Claim 10.2.) Given \vec{Z} , let us write $f = f_{\vec{Z}}$ for notational simplicity. Let us also write $\gamma = O(\sqrt{\ln n/n})/\epsilon^2$. We need to show that with probability at least $1 - O(1/n)$,

$$\gamma \geq \text{Inf}_i^{(0)}(f) = \text{Inf}_i(f) = \mathbf{E}_{\mathbf{x} \in \mathbb{B}^n} \left[\left(\frac{f(\mathbf{x}^{(i=1)}) - f(\mathbf{x}^{(i=-1)})}{2} \right)^2 \right] \quad \text{for all } 1 \leq i \leq n. \quad (29)$$

Here we have used the notation $\mathbf{x}^{(i=b)}$ for the string \mathbf{x} with the i th coordinate set to b/\sqrt{n} , along with the well-known alternate definition of boolean influences (see [KKMO07]). In fact, we will show that (29) holds whenever both of the following hold:

$$|Z_i| \leq 2\sqrt{\ln n} \quad \text{for all } 1 \leq i \leq n; \quad (30)$$

$$\frac{1}{2}n \leq \|\vec{Z}\|_2^2 \leq \frac{3}{2}n. \quad (31)$$

Since $|Z_i| \leq 2\sqrt{\ln n}$ for each i except with probability at most $O(1/n^2)$, we get that (30) holds except with probability $O(1/n)$. It's also well known (and the proof is sketched in the proof of Theorem 4.3) that (31) holds except with exponentially small probability in n . Thus both (30) and (31) hold except with probability at most $O(1/n)$, as necessary.

Let us henceforth fix $\vec{Z} = \vec{Z}$ satisfying (30) and (31). We wish to prove now that (29) holds. We will show that it holds for $i = n$, and the fact that it holds for $1 \leq i < n$ will follow by an identical argument. So we must prove that

$$\begin{aligned} \gamma &\geq \mathbf{E}_{\mathbf{x} \in \mathbb{B}^n} \left[\left(\frac{f(\mathbf{x}^{(n=1)}) - f(\mathbf{x}^{(n=-1)})}{2} \right)^2 \right] \\ &= \frac{1}{4} \mathbf{E}_{\mathbf{x} \in \mathbb{B}^{n-1}} \left[\left(r^* \left(\sum_{i=1}^{n-1} Z_i \mathbf{x}_i + \frac{Z_n}{\sqrt{n}} \right) - r^* \left(\sum_{i=1}^{n-1} Z_i \mathbf{x}_i - \frac{Z_n}{\sqrt{n}} \right) \right)^2 \right]. \end{aligned}$$

Using the fact that r is ϵ' -discretized, we can even show the following stronger result:

$$\mathbf{Pr}_{\mathbf{x} \in \mathbb{B}^{n-1}} \left[\sum_{i=1}^{n-1} Z_i \mathbf{x}_i \pm \frac{Z_n}{\sqrt{n}} \text{ fall into different intervals from } \mathcal{I}_{\epsilon'} \right] \leq \gamma. \quad (32)$$

Let σ^2 denote $\sum_{i=1}^{n-1} Z_i^2/n$, which by (30) and (31) satisfies $\frac{1}{3} \leq \sigma^2 \leq \frac{3}{2}$. Now the random variable $\sum_{i=1}^{n-1} Z_i \mathbf{x}_i$ has distribution close to that of a mean-zero Gaussian with variance σ^2 ; more specifically, using the Berry-Esseen Theorem we have that for every interval I ,

$$\left| \mathbf{Pr} \left[\sum_{i=1}^{n-1} Z_i \mathbf{x}_i \in I \right] - \mathbf{Pr}[N(0, \sigma^2) \in I] \right| \leq O \left(\frac{\max_i |Z_i|}{\sigma \sqrt{n}} \right) = O(\sqrt{\log n/n}). \quad (33)$$

The analysis is now very similar to the analysis in Claim 5.7. Given any interval $J \in \mathcal{I}_{\epsilon'}$, let J' denote the subinterval gotten by moving the boundary points inwards by $3\sqrt{\ln n/n}$. The analysis from Claim 5.7 implies that a standard Gaussian will fall into one of the J' intervals except with probability $O(\sqrt{\ln n/n}/\epsilon'^2)$, and only the constant in the $O(\cdot)$ changes if we consider instead a Gaussian with variance $\sigma^2 \in [\frac{1}{3}, \frac{3}{2}]$. Hence the same is true of the random variable $\sum_{i=1}^{n-1} Z_i \mathbf{x}_i$, using (33). But whenever this random variable falls into some J' , we get that $\sum_{i=1}^{n-1} Z_i \mathbf{x}_i \pm \frac{Z_n}{\sqrt{n}}$ are both in the associated J , since $|Z_i| \leq 2\sqrt{\ln n}$. Since we took $\gamma = O(\sqrt{\ln n/n})/\epsilon'^2$, we have that (32) indeed holds, as needed. \square

(Theorem 10.1) \square

11 Hardness results for RPR² algorithms

In this section we revisit the constructions of Karloff [Kar99], Alon and Sudakov [AS00], and Alon, Sudakov, and Zwick [ASZ02]. The purpose of these constructions is to demonstrate that the *analysis* of the Goemans-Williamson approximation guarantee is tight (and likewise for the Zwick [Zwi99] approximation guarantee, in the case of [ASZ02]). For now we discuss [Kar99, AS00], returning to [ASZ02] at the end of the section.

The works [Kar99, AS00] consider the graph T on \mathbb{B}^n in which a pair of vertices (x, y) is connected if and only if the vertices' inner product is exactly $1 - 2c$; here c is any rational parameter in $(\frac{1}{2}, 1)$.¹² The authors show (for infinitely many n) that the identity map is an optimal SDP embedding, and hence $\text{Opt}(T) = \text{Sdp}(T) = c$. On the other hand, since every edge in the embedded graph connects vectors with inner product $1 - 2c$, the expected value of the cut output by the GW algorithm (RPR² with the rounding function sgn) is only $\arccos(1 - 2c)/\pi$. Thus (in expectation, at least) the GW approximation curve satisfies $\text{Apx}_{GW}(c) \leq \arccos(1 - 2c)/\pi$.

As the reader can clearly see, this construction can be viewed as a Dictator-vs.-Gaussian test with completeness c . Indeed, the noise sensitivity test of [KKMO07] is almost identical to it; the only difference is that the noise sensitivity test picks edges with *expected* inner product $1 - 2c$ rather than precise inner product $1 - 2c$. The 'soundness' result used in [Kar99, AS00] is that the average value among 'random halfspace functions' $\text{sgn}(x \cdot \vec{Z})$ is at most $\arccos(1 - 2c)/\pi$. As we saw in Section 10, these random halfspace functions are almost surely Gaussian.

The result from [Kar99, AS00] has some additional strengths and weaknesses. One strength is that the SDP embedding used has all of its unit vectors on the discrete cube \mathbb{B}^n ; hence these points satisfy the triangle inequalities, and indeed satisfy all 'valid' inequalities (see [Kar99]). Thus $\text{Apx}_{GW}(c)$ is still at most $\arccos(1 - 2c)/\pi$ even if the SDP with triangle inequalities is used. A weakness of the original result was that it only stated that the *expected* value of the cut GW produces is at most $\arccos(1 - 2c)/\pi$; it said nothing, e.g., about what happens if the GW algorithm is run several times and the best resulting cut is selected. For the noise sensitivity version of the test, a result in [KKMO07] shows that GW achieves at most $\arccos(1 - 2c)/\pi + o(1)$ with high probability. However, Feige and Schechtman [FS02] showed an even better result:

Theorem 11.1 ([FS02]) *For any rational $c \in (\frac{1}{2}, 1)$ and any $\eta > 0$, there are optimally embedded graphs G , with arbitrarily large numbers of vertices, satisfying:*

- $\text{Opt}(G) = \text{Sdp}(G) = c$;
- *the vectors in G satisfy the triangle inequalities;*
- *every halfspace cut has value at most $\arccos(1 - 2c)/\pi + \eta$.*

The conclusion from this result is that running the RPR² algorithm A with the rounding function sgn cannot achieve $\text{Apx}_A(c) > \arccos(1 - 2c)/\pi$, even if: (i) A uses the SDP with triangle inequalities; and, (ii) A is not required to choose \vec{Z} at random but is allowed to use the best possible \vec{Z} of length \sqrt{n} . (When $r = \text{sgn}$, the length of \vec{Z} is irrelevant and may as well be fixed.)

Feige and Schechtman prove Theorem 11.1 (non-constructively) as follows: They begin with the embedded graph T on \mathbb{B}^n constructed in [Kar99, AS00]. They then essentially take G to consist of m disjoint copies of T , each embedded in a random n -dimensional subspace of \mathbb{R}^d . If $d \gg n^2 \log m$, then the triangle inequalities hold in G with high probability; on the other hand, if d is not too large then it can be shown that every halfspace cut of G has value at most $\arccos(1 - 2c)/\pi + \eta$.

We now prove a generalization of Theorem 11.1. We would like to emphasize that our proof follows Feige and Schechtman's extremely closely. The following theorem implies our promised Theorem 1.15 from Section 1.5:

Theorem 11.2 *Suppose (c, s) is a Dictator-vs.-Gaussian gap, and $\eta > 0$. Fix any RPR² rounding function r which is piecewise constant.¹³ Then there are embedded graphs G in S^{d-1} , with arbitrarily large numbers of vertices, satisfying:*

- $\text{Opt}(G) \geq c$;

¹²The earlier work of [Kar99] was slightly more complicated as it only included vertices with Hamming weight exactly $n/2$.

¹³As all functions implemented on a discrete computer must be.

- the vectors in G satisfy the triangle inequalities;
- every fractional cut $f_{\vec{z}}$ of the form $f_{\vec{z}}(u) = r(u \cdot \vec{z})$ satisfies $\text{val}_G(f_{\vec{z}}) \leq s + \eta$, as long as $\|\vec{z}\|_2 = \Theta(\sqrt{d})$.

Proof: Select $\epsilon, \delta > 0$ and a family $(T^{(n)})$ of Dictator-vs.-Gaussian tests, with $T^{(n)}$ operating on \mathbb{B}^n , such that $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}_{\epsilon, \delta}(T_n) \leq s + \eta/3$, for all sufficiently large n . We would also like to assume that each $T^{(n)}$ is regular, meaning that each $x \in \mathbb{B}^n$ participates in the test with the same probability. We can ensure this by symmetrizing each $T^{(n)}$ with respect to the $2^n n!$ symmetries of \mathbb{B}^n , as discussed in Section 9. (Alternatively, the Dictator-vs.-Gaussian tests we will actually use, constructed in Section 9, are already regular.)

As in [FS02], we take G to be m equally weighted disjoint copies of $T^{(n)}$, embedded on the unit d -dimensional sphere S^{d-1} with independent random orientations. Since $\text{Completeness}(T^{(n)}) \geq c$, certainly $\text{Opt}(G) = \text{Opt}(T^{(n)}) \geq c$. Also, as shown in [FS02], if $d \gg n^2 \log m$ then the vectors in G satisfy the triangle inequalities with high probability; this uses the fact that the vectors in $T^{(n)}$ satisfy the triangle inequalities. It remains to analyze $\text{val}_G(f_{\vec{z}})$ for all possible fractional cuts $f_{\vec{z}}(u) := r(u \cdot \vec{z})$ where $\|\vec{z}\|_2 = \Theta(\sqrt{d})$. For concreteness, assume that this means $(1/c)\sqrt{d} \leq \|\vec{z}\|_2 \leq c\sqrt{d}$ for some $c > 0$.

Let us consider the piecewise constant function r . Choose a small enough $\gamma > 0$ so that the set

$$\mathcal{B} := \bigcup \{[t - \gamma, t + \gamma] : t \text{ is a point of discontinuity for } r\}$$

has total measure at most $\epsilon \eta / O(\sqrt{c})$. Following [FS02], we now take a γ -net \mathcal{N} for the set $\{(1/c)\sqrt{d} \leq \|\vec{z}\|_2 \leq c\sqrt{d}\}$; this can have cardinality $O(c\sqrt{d}/\gamma)^d$. We show that, with high probability over the orientations of G , both of the following hold for all $\vec{v} \in \mathcal{N}$:

1. $\text{val}_G(f_{\vec{v}}) \leq s + 2\eta/3$;
2. the fraction of vertices \vec{u} of G for which $\vec{u} \cdot \vec{v} \in \mathcal{B}$ is at most $\eta/6$.

Having shown this, it follows that $\text{val}_G(f_{\vec{z}}) \leq s + \eta$ for all $(1/c)\sqrt{d} \leq \|\vec{z}\|_2 \leq c\sqrt{d}$. To see this for a given \vec{z} , take \vec{v} to be the closest net point. Then for every $\vec{u} \in G$ we have $|\vec{u} \cdot \vec{z} - \vec{u} \cdot \vec{v}| \leq \|\vec{u}\| \cdot \|\vec{z} - \vec{v}\| \leq \gamma$. It follows that $f_{\vec{z}}(\vec{u}) = f_{\vec{v}}(\vec{u})$ except possibly when $\vec{u} \cdot \vec{v} \in \mathcal{B}$. But this occurs only for at most an $\eta/6$ fraction of vertices in G , and hence at most an $\eta/6$ fraction of edge weight, by regularity. It follows that $|\text{val}_G(f_{\vec{z}}) - \text{val}_G(f_{\vec{v}})| \leq 2\eta/6$, and hence $\text{val}_G(f_{\vec{z}}) \leq s + \eta$, as required.

It remains to prove that items (1) and (2) above indeed hold with high probability. Fix any $\vec{v} \in \mathcal{N}$ and let T_1, \dots, T_m denote the randomly oriented copies of $T^{(n)}$ making up G . In analyzing some T_i vis-a-vis \vec{v} , we imagine instead that the orientation of T_i is fixed and \vec{v} is chosen randomly from the surface of the sphere of radius $\|\vec{v}\|_2$. In this framework, let \vec{Y} denote the projection of the random \vec{v} onto the n -dimensional subspace containing T_i . Now the projection of a random vector from the surface of a sphere onto a lower-dimensional subspace yields a distribution which is close to Gaussian. In particular, since we are already assuming $d \gg n^2 \log m \geq O(n^2)$, the results in [DF87] imply that the variation distance between \vec{Y} and the n -dimensional Gaussian distribution with coordinate variances equal to $\|\vec{v}\|_2 / \sqrt{d} \in [1/c, c]$ is at most $O(n/d) = O(1/n)$. If \vec{Y} were truly drawn from that Gaussian distribution, then we would have the following (cf. the proof of Claim 10.2):

- the expected fraction of vertices \vec{u} of T_i for which $\vec{u} \cdot \vec{Y} \in \mathcal{B}$ is at most $O(\sqrt{c}|\mathcal{B}|)$;
- $|\mathbf{Y}_i| \leq O(\sqrt{c \ln n})$ for all $1 \leq i \leq n$;
- $\frac{1}{2c}n \leq \|\vec{Y}\|_2^2 \leq \frac{3c}{2}n$.

Similar to the proof of Claim 10.2, the last two of these imply that $f_{\vec{v}}$ is a $(\epsilon, 0)$ -Gaussian cut for T_i , as long as $O(\sqrt{c \ln n}/n) \leq \gamma$ and $O(\sqrt{c}|\mathcal{B}|) \leq \epsilon$. The latter holds by design; the former holds so long as we take $n \geq \text{poly}(c/\gamma)$. But when $f_{\vec{v}}$ is a $(\epsilon, 0)$ -Gaussian cut for T_i , we have $\text{val}_{T_i}(f_{\vec{v}}) \leq s + \eta/3$. Note also that

$O(\sqrt{c}|\mathcal{B}|) \leq \eta/24$ by design.

Overall, we conclude that for each i independently we have $\text{val}_{T_i}(f_{\vec{v}}) \leq s + \eta/3$, except with probability at most $O(1/n)$ over the choice of orientations. If we ensure that $n \geq O(1/\eta)$, we conclude that the expected value of $\text{val}_{T_i}(f_{\vec{v}})$ is at most $s + \eta/2$. Similarly, we can conclude that the expected fraction of vertices \vec{u} of T_i for which $\vec{u} \cdot \vec{v} \in \mathcal{B}$ is at most $\eta/12$. Since $\text{val}_G(f_{\vec{v}}) = \text{avg}_{i \in [m]} \text{val}_{T_i}(f_{\vec{v}})$, a Chernoff bound implies that item (1) above holds except with probability at most $\exp(-O(\eta^2 m))$. Similarly, item (2) above holds except with probability at most $\exp(-O(\eta^2 m))$. If we take $m \gg d \log d$ then this probability will be much smaller than $O(c\sqrt{d}/\gamma)^{-d}$ (treating c , γ , and η as constants), and so we get that both items (1) and (2) hold with high probability for all net points simultaneously, by a union bound.

As in [FS02], the overall constraints we have on m and d are that $n^2 \log m \ll d \ll m/\log m$, and this can clearly be realized. \square

We end this section by discussing the issue of self-loops and the construction of Alon, Sudakov, and Zwick [ASZ02]. If we use Theorem 1.15 with the noise sensitivity $(1, \rho_0)$ -mixture tests constructed in Section 9, we get a hard instance for RPR², but one that might be considered slightly unsatisfactory: this is because the embedded graph G constructed has self-loops. However one can't simply dismiss embedded graphs with self-loops, because optimally embedded graphs *can* have self-loops. In fact, Alon, Sudakov, and Zwick's construction is the following: for each $(1, \rho_0)$ -mixture distribution, they construct a self-loopless graph for which the optimal SDP embedding is essentially the noise sensitivity $(1, \rho_0)$ -mixture test. More precisely, it is the version in which vertices are connected if their inner product is *exactly* ρ_0 or 1. The technique of [ASZ02] involves taking the $(1, \rho_0)$ -mixture test and replacing the self-loops by cliques, similar to the self-loop removal technique discussed in Appendix B. Indeed, using Alon, Sudakov, and Zwick's construction, we can even ensure that the hard embedded graphs for RPR² that we get from Theorem 1.15 are optimally embedded, as in Theorem 11.1.

Appendices

A $\text{Gap}_{\text{SDP}}(c)$ is continuous

In this appendix we prove Proposition 4.2. The fact that $\text{Gap}_{\text{SDP}}(c)$ is increasing on $[\frac{1}{2}, 1]$ is immediate from the definition (since if $c' > c$, the inf for c' is over a subset of the inf for c). We mainly focus on the proof that $\text{Gap}_{\text{SDP}}(c)$ is continuous on $(\frac{1}{2}, 1)$; this requires only a simple trick — the use of the *isolated edge*. The proof of continuity at 1 requires appealing to Goemans-Williamson, and the continuity at $\frac{1}{2}$ is trivial. Finally, the proof that $\text{Gap}_{\text{SDP}}(c)$ is *strictly* increasing requires an *isolated clique* trick, plus an appeal to a result of Zwick [Zwi99].

Definition A.1 *Given a graph G and a parameter $0 \leq \epsilon \leq 1$, we define the graph $G \sqcup \text{edge}_\epsilon$ to be the graph in which G 's edge-weights are scaled by a factor of $1 - \epsilon$, and then two new vertices are added, with an edge between them of weight ϵ .*

The following is easy to verify:

Proposition A.2 $\text{Sdp}(G \sqcup \text{edge}_\epsilon) = (1 - \epsilon)\text{Sdp}(G) + \epsilon$ and $\text{Opt}(G \sqcup \text{edge}_\epsilon) = (1 - \epsilon)\text{Opt}(G) + \epsilon$.

We now prove:

Proposition A.3 $\text{Gap}_{\text{SDP}}(c)$ is continuous on $(\frac{1}{2}, 1)$.

Proof: We first prove right-continuity on $(\frac{1}{2}, 1)$. Suppose $c \in (\frac{1}{2}, 1)$, and let $s = \text{Gap}_{\text{SDP}}(c)$. Given any sufficiently small $\epsilon > 0$, assume $c < c' < c + (1 - c)\epsilon/2 < 1$. By the definition of $\text{Gap}_{\text{SDP}}(c) = s$ we can find some graph G with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s + \epsilon/2$. Let $\tilde{G} = G \sqcup \text{edge}_{\epsilon/2}$. Then we have $\text{Sdp}(\tilde{G}) \geq (1 - \epsilon/2)c + \epsilon/2 = c + (1 - c)\epsilon/2 > c'$, and further, $\text{Opt}(\tilde{G}) \leq (1 - \epsilon/2)(s + \epsilon/2) + \epsilon/2 \leq s + \epsilon$. This proves $\text{Gap}_{\text{SDP}}(c') \leq s + \epsilon$. Since Gap_{SDP} is increasing, we have proven right-continuity at c .

The proof of left-continuity on $(\frac{1}{2}, 1)$ is similar. Suppose $c \in (\frac{1}{2}, 1)$, and let $s = \text{Gap}_{\text{SDP}}(c)$. Given any sufficiently small $\epsilon > 0$, assume $\frac{1}{2} < c - 2\epsilon(1 - c) < c' < c$. For any graph G with $\text{Sdp}(G) \geq c'$, let $\tilde{G} = G \sqcup \text{edge}_{2\epsilon}$. We have $\text{Sdp}(\tilde{G}) \geq (1 - 2\epsilon)c' + 2\epsilon = c' + 2\epsilon(1 - c') \geq c' + 2\epsilon(1 - c) \geq c$ and also $\text{Opt}(\tilde{G}) = (1 - 2\epsilon)\text{Opt}(G) + 2\epsilon$. By the definition of $\text{Gap}_{\text{SDP}}(c) = s$, it holds that $\text{Opt}(\tilde{G}) \geq s$. Hence $(1 - 2\epsilon)\text{Opt}(G) + 2\epsilon \geq s$ which implies $\text{Opt}(G) \geq s - (1 - \text{Opt}(G))2\epsilon \geq s - \epsilon$. This proves $\text{Gap}_{\text{SDP}}(c') \geq s - \epsilon$. Since Gap_{SDP} is increasing, we have proven left-continuity at c . \square

We next check continuity at the endpoints, $c = \frac{1}{2}, 1$. It's easy to see that if $\text{Sdp}(G) = 1$ then G must be bipartite and so $\text{Opt}(G) = 1$. Hence $\text{Gap}_{\text{SDP}}(1) = 1$. Next, by taking the sequence of complete graphs K_m (each with total edge-weight 1), which satisfy $\text{Opt}(K_m) \leq \frac{1}{2} + \frac{1}{m} \rightarrow \frac{1}{2}$ as $m \rightarrow \infty$, we see that $\text{Gap}_{\text{SDP}}(\frac{1}{2}) = \frac{1}{2}$. Thus to check continuity at the endpoints we need to show that $\lim_{c \rightarrow (1/2)^+} \text{Gap}_{\text{SDP}}(c) = \frac{1}{2}$ and $\lim_{c \rightarrow 1^-} \text{Gap}_{\text{SDP}}(c) = 1$.

The first of these follows simply because $\text{Gap}_{\text{SDP}}(c)$ is sandwiched between $\frac{1}{2}$ and c for all c . For the second of these, suppose G is any graph with $\text{Sdp}(G) \geq 1 - \epsilon$. The analysis of Goemans and Williamson [GW95] implies that one can find a cut in G with value at least $1 - O(\sqrt{\epsilon})$. Thus $\text{Gap}_{\text{SDP}}(1 - \epsilon) \geq 1 - O(\sqrt{\epsilon})$, and so $\lim_{c \rightarrow 1^-} \text{Gap}_{\text{SDP}}(c) = 1$ as claimed.

Finally, we check that $\text{Gap}_{\text{SDP}}(c)$ is strictly increasing. For this we introduce isolated cliques:

Definition A.4 Given a graph G and two parameters $m \in \mathbb{N}$ and $0 \leq \epsilon \leq 1$, we define the graph $G \sqcup K_{m,\epsilon}$ to be the graph in which G 's edge-weights are scaled by a factor of $1 - \epsilon$, and then an isolated m -clique is added, whose total edge-weight is ϵ .

Using the fact that $\text{Opt}(K_{m,1}) \leq \frac{1}{2} + \frac{1}{m}$, one can check:

Proposition A.5 $\text{Sdp}(G \sqcup K_{m,\epsilon}) \geq (1 - \epsilon)\text{Sdp}(G) + \epsilon/2$ and $\text{Opt}(G \sqcup K_{m,\epsilon}) \leq (1 - \epsilon)\text{Opt}(G) + (\frac{1}{2} + \frac{1}{m})\epsilon$.

We now have:

Proposition A.6 $\text{Gap}_{\text{SDP}}(c)$ is strictly increasing on $[\frac{1}{2}, 1]$.

Proof: It's enough to check this on $(\frac{1}{2}, 1)$. So suppose $\frac{1}{2} < c < c' < 1$, and write $s' = \text{Gap}_{\text{SDP}}(c')$. Zwick [Zwi99] was the first to show that $c' > \frac{1}{2}$ implies $s' > \frac{1}{2}$; Charikar and Wirth [CW04] specifically proved that $\text{Sdp}(G) \geq \frac{1}{2} + \gamma$ implies $\text{Opt}(G) \geq \frac{1}{2} + \Omega(\gamma/\log(1/\gamma))$. Thus we have $s' > \frac{1}{2}$. Write $\epsilon = (c' - c)/c'$. Select m large enough that $s' - \frac{1}{2} - \frac{1}{m}$ is still strictly positive. Finally, take $\delta > 0$ so that $\delta < (s' - \frac{1}{2} - \frac{1}{m})\epsilon$.

By definition of $\text{Gap}_{\text{SDP}}(c') = s'$, we can find a graph G' with $\text{Sdp}(G') \geq c'$ and $\text{Opt}(G') \leq s' + \delta$. Let $G = G' \sqcup K_{m,\epsilon}$. Then $\text{Sdp}(G) \geq (1 - \epsilon)c' + \epsilon/2 \geq (1 - \epsilon)c' = c$. Further, $\text{Opt}(G) \leq (1 - \epsilon)(s' + \delta) + (\frac{1}{2} + \frac{1}{m})\epsilon \leq s' + (\frac{1}{2} + \frac{1}{m} - s')\epsilon + \delta < s' - \delta + \delta = s'$. We conclude that $\text{Gap}_{\text{SDP}}(c) < s' = \text{Gap}_{\text{SDP}}(c')$. Thus $\text{Gap}_{\text{SDP}}(c)$ is indeed strictly increasing. \square

B SDP gaps based on infinite, self-looped graphs

In this appendix we prove Proposition 4.1.

Proof: Write $G_0 = G$. We will transform G_0 into G_1 , an infinite graph on vertex set B_d ; then G_1 into G_2 , a finite graph (with self-loops); then G_2 into G_3 , a self-loopless graph; then G_3 into G_4 , an unweighted graph. The desired graph will then be $G' = G_4$. The first transformation uses the idea of embedded graphs, and the remaining transformations are all previously known.

Let $g : \mathbb{R}^d \rightarrow B_d$ achieving the sup in the definition of $\text{Sdp}(G_0)$ to within ϵ . Let G_1 be the infinite graph on B_d given by pushing forward G_0 via g , i.e., $G_1(A, B) = G_0(g^{-1}(A), g^{-1}(B))$ (here we're identifying a graph with the probability measure defining its 'edge weights'). We immediately get $\mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim G_1} [\frac{1}{2} - \frac{1}{2} \mathbf{x} \cdot \mathbf{y}] \geq c - \epsilon$. We can think of this as saying:

$$\text{'Sdp}(G_1) \geq c - \epsilon, \quad (34)$$

with the identity mapping as the embedding. Further,

$$\text{Opt}(G_1) \leq s, \quad (35)$$

because for any fractional cut $h : B_d \rightarrow [-1, 1]$ for G_1 , the cut $h \circ g : \mathbb{R}^d \rightarrow [-1, 1]$ for G_0 achieves the same value, $\mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim G_1} [\frac{1}{2} - \frac{1}{2} h(\mathbf{x})h(\mathbf{y})] = \mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim G_0} [\frac{1}{2} - \frac{1}{2} (h \circ g(\mathbf{x}))(h \circ g(\mathbf{y}))]$.

We next discretize G_1 in the manner of, say, Feige and Schechtman [FS02]. Choose an ϵ -net \mathcal{N} within B_d of size at most $O(1/\epsilon)^d$. Further, partition B_d into Voronoi cells based on \mathcal{N} , with a disjoint cell C_v for each $v \in \mathcal{N}$. Now define the (finite) graph G_2 on \mathcal{N} by taking $G_2(u, v) = G_1(C_u, C_v)$ (again, we identify a graph with its edge distribution). We claim

$$\text{Sdp}(G_2) \geq c - 3\epsilon. \quad (36)$$

To see this, recall that the identity embedding for G_1 achieves $\mathbf{E}_{(\mathbf{x}, \mathbf{y}) \sim G_1} [\frac{1}{2} - \frac{1}{2} \mathbf{x} \cdot \mathbf{y}] \geq c - \epsilon$. Now if \mathbf{x} is in the cell C_u and \mathbf{y} is in the cell C_v , then $\mathbf{x} \cdot \mathbf{y} = (u + \eta_1) \cdot (v + \eta_2)$ for some vectors η_1, η_2 of length at most ϵ ; this implies $|\mathbf{x} \cdot \mathbf{y} - u \cdot v| \leq 3\epsilon$. Since we can draw from G_2 by drawing $(\mathbf{x}, \mathbf{y}) \sim G_1$ and then taking (u, v) such that $\mathbf{x} \in C_u$ and $\mathbf{y} \in C_v$, we conclude that $\mathbf{E}_{(u, v) \sim G_2} [\frac{1}{2} - \frac{1}{2} u \cdot v] \geq c - \epsilon - \frac{3}{2}\epsilon$. We conclude that (36) holds with the identity map as the embedding. The fact that

$$\text{Opt}(G_2) \leq s \quad (37)$$

follows for the same reason as (35) — any cut for G_2 can be extended to an equally good cut for G_1 .

We now eliminate self-loops from G_2 , forming G_3 , using the construction in the appendix of Khot and O'Donnell [KO06], which itself is based on a trick of Arora, Berger, Hazan, Kindler, and Safra [ABH⁺05]. It is shown therein that for any $\epsilon > 0$, we can take G_3 to have $O(1/\epsilon)^2$ times as many vertices as G_2 , and satisfy

$$\text{Sdp}(G_3) \geq \text{Sdp}(G_2) \geq c - 3\epsilon, \quad (38)$$

and

$$\text{Opt}(G_3) \leq \text{Opt}(G_2) \leq s + \epsilon. \quad (39)$$

Finally, we form $G' = G_4$ from G_3 , converting weighted edges to unweighted edges. There is a simple randomized way to do this (see, e.g., [BGS98, CST01]), taking a weighted graph on m vertices into an unweighted one on $\text{poly}(m/\epsilon)$ vertices, such that

$$\text{Sdp}(G_4) \geq \text{Sdp}(G_3) - \epsilon \geq c - 4\epsilon, \quad (40)$$

and

$$\text{Opt}(G_4) \leq \text{Opt}(G_3) + \epsilon \leq s + 2\epsilon. \quad (41)$$

Since G_3 has $O(1/\epsilon)^{d+2}$ vertices, our G_4 has $n = (1/\epsilon)^{O(d)}$ vertices, as claimed. The proof follows after replacing ϵ by $\epsilon/4$. \square

C RPR² — implementation issues

In this section we mention a few implementation issues that arise in the use of the RPR² framework and discuss how they affect our algorithmic guarantees. All of these issues have been considered before; see [GW95, MR99, FL06, FS02, EIO02].

Exact solving of the SDP. The SDP-solving guarantee one actually has is that a solution within ϵ of optimum can be found in time $\text{poly}(n) \cdot \log(1/\epsilon)$. We have already treated this issue in the proof of Corollary 5.4. Another related issue is that the vectors returned by the SDP-solver may not lie precisely on the unit sphere, something we assumed in our analysis. This can be taken care of by shrinking all vectors slightly so that they lie within the unit ball, and then adding a fictitious extra coordinate with tiny values to make the vectors have length exactly 1.

Choosing Gaussian random variables. Again, this can not be done precisely, but the approximation methods of Mahajan and Ramesh [MR99] shows that one can incur ϵ loss at the expense only of $\text{poly}(n, 1/\epsilon)$ time.

Expectation vs. high probability vs. deterministic. Our results have been concerned with showing the expected value of the fractional cut produced by the (randomized) RPR² algorithm is at least $S(c)$. One can turn this into a high-probability result, losing only an additive ϵ in cut value, by using $\text{poly}(n, 1/\epsilon)$ independent repetitions. Alternatively, one can derandomize the RPR² framework, again losing only an additive ϵ in the cut value, via the method of conditional expectations; this can be done in $\text{poly}(n, 1/\epsilon)$ time [MR99] or $O(n) \cdot 2^{\text{poly}(1/\epsilon)}$ time [EIO02]. Having done either of these, one has a fractional cut with value at least $S(c) - \epsilon$. This can be converted into a proper cut with at least the same value by the method of conditional expectations.

Multiple rounding functions. As discussed in Section 1.4, we also want to try a *collection* \mathcal{R} of rounding functions. For a high-probability results, we can simply repeat the algorithm $O(|\mathcal{R}| \log |\mathcal{R}|)$ times for each rounding function and this will achieve what the best of them does. Alternatively, we can just use the derandomized algorithms once for each $r \in \mathcal{R}$.

Proper cuts when G has self-loops. Given a graph G with self-loops, we cannot actually find proper cuts with value at least $S(\text{Sdp}(G))$. For example, if G consists of a single self-loop then $\text{Sdp}(G) = \frac{1}{2}$ (via the embedding mapping the vertex to 0), but there is no proper cut of value $\frac{1}{2}$. The way to interpret our guarantee for graphs G with self-loops is as follows: First, remove the self-loops from G , forming G' — note that this does not change the value of the optimal proper cut. Then our algorithm achieves at least $S(\text{Sdp}(G')) - \epsilon \geq S(O(G)) - \epsilon$, where $O(G)$ denotes the value of the optimal proper cut in G .

D Improved asymptotics of $S(\frac{1}{2} + \epsilon)$

As described in Section 2.3, Charikar and Wirth [CW04] established $\text{Gap}_{\text{SDP}}(\frac{1}{2} + \epsilon) \geq \frac{1}{2} + \Omega(\epsilon/\ln(1/\epsilon))$ and Khot and O'Donnell [KO06] established $\text{Gap}_{\text{SDP}}(\frac{1}{2} + \epsilon) \leq \frac{1}{2} + O(\epsilon/\ln(1/\epsilon))$. In this appendix we carefully examine these proofs and conclude the following:

Theorem D.1 $\text{Gap}_{\text{SDP}}(\frac{1}{2} + \epsilon) = S(\frac{1}{2} + \epsilon) = \frac{1}{2} + (\frac{1}{2} \pm o(1)) \cdot \epsilon/\ln(1/\epsilon)$.

Proof: We upper-bound $S(c)$ essentially by repeating the argument in [KO06], paying more attention to the constants. Take P to be the $(1, \rho_0)$ -distribution with weight $p = \frac{2}{3} + \frac{4}{3}\epsilon$ on $\rho_0 = -\frac{1}{2}$ and weight $\frac{1}{3} - \frac{4}{3}\epsilon$ on 1. Now if $r : \mathbb{R} \rightarrow [-1, 1]$ is any odd one-dimensional rounding function, we have

$$\begin{aligned} \text{val}_{\mathcal{G}_P}(r) &= \frac{1}{2} - \frac{1}{2} \left[\left(\frac{1}{3} - \frac{4}{3}\epsilon\right) \mathbb{S}_1(r) + \left(\frac{2}{3} + \frac{4}{3}\epsilon\right) \mathbb{S}_{-1/2}(r) \right] = \frac{1}{2} - \sum_{\text{odd } s} \left(\frac{1}{6} - \frac{2\epsilon}{3} + \left(\frac{1}{3} + \frac{2\epsilon}{3}\right) \left(-\frac{1}{2}\right)^s \right) \widehat{r}(s)^2 \\ &\leq \frac{1}{2} + \epsilon \widehat{r}(1)^2 - \sum_{\text{odd } s \geq 3} \left(\frac{1}{8} - \frac{3}{4}\epsilon \right) \widehat{r}(s)^2 = \frac{1}{2} + \epsilon \widehat{r}(1)^2 - \left(\frac{1}{8} - \frac{3}{4}\epsilon\right) \mathbf{E}[(r - Lr)^2], \end{aligned} \quad (42)$$

where L denotes the ‘projection to degree 1’ operator; i.e., $Lr(x) = \widehat{r}(x)x$. As in [KO06] we consider the value of $\sigma^2 := \widehat{r}(1)^2 = \mathbf{E}[(Lr)^2]$, the variance of the Gaussian $Lr(x)$. Using $|r| \leq 1$, we lower-bound

$$\mathbf{E}[(r - Lr)^2] \geq \mathbf{E}[\mathbf{1}_{\{|Lr| \geq 1\}} \cdot (\text{sgn}(r) - Lr)^2],$$

which asymptotically is $\sigma^{\Theta(1)} \cdot \exp(-1/2\sigma^2)$. If $\sigma \gg 1/\sqrt{2\ln(1/\epsilon)}$ then the final term in (42) will exceed ϵ , making the overall quantity less than $\frac{1}{2}$. Thus in upper-bounding (42) we can assume $\sigma \leq (1 + o(1))/\sqrt{2\ln(1/\epsilon)}$, and thus we get an upper bound of $\frac{1}{2} + (\frac{1}{2} + o(1))\epsilon/\ln(1/\epsilon)$, as claimed.

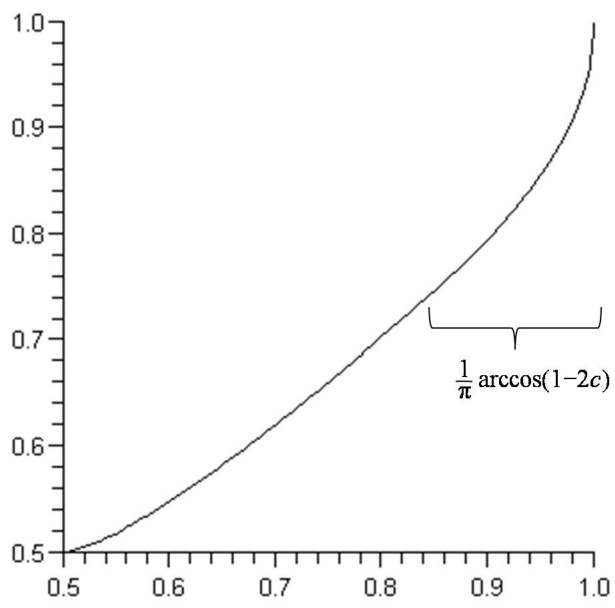
To lower-bound $\text{Gap}_{\text{SDP}}(\frac{1}{2} + \epsilon)$ we refer to [CW04, equation (11)], which shows that

$$\text{Gap}_{\text{SDP}}(\frac{1}{2} + \epsilon) \geq \frac{1}{2} + \frac{\epsilon}{T^2} - 4e^{-T^2/2}$$

for every $T \geq 1$. By taking $T = (1 - o(1)) \cdot \sqrt{2\ln(1/\epsilon)}$, we get a lower bound of $\frac{1}{2} + (\frac{1}{2} - o(1)) \cdot \epsilon/\ln(1/\epsilon)$. \square

E Approximate values of $S(c)$

c	$S(c)$	c	$S(c)$	c	$S(c)$	c	$S(c)$
0.505	0.5008	0.590	0.5414	0.675	0.6012	0.760	0.6694
0.510	0.5021	0.595	0.5446	0.680	0.6050	0.765	0.6736
0.515	0.5036	0.600	0.5478	0.685	0.6089	0.770	0.6778
0.520	0.5053	0.605	0.5510	0.690	0.6127	0.775	0.6820
0.525	0.5072	0.610	0.5544	0.695	0.6167	0.780	0.6862
0.530	0.5092	0.615	0.5577	0.700	0.6206	0.785	0.6905
0.535	0.5113	0.620	0.5611	0.705	0.6245	0.790	0.6947
0.540	0.5136	0.625	0.5646	0.710	0.6285	0.795	0.6990
0.545	0.5160	0.630	0.5681	0.715	0.6325	0.800	0.7033
0.550	0.5185	0.635	0.5716	0.720	0.6365	0.805	0.7076
0.555	0.5211	0.640	0.5752	0.725	0.6406	0.810	0.7119
0.560	0.5238	0.645	0.5788	0.730	0.6446	0.815	0.7162
0.565	0.5265	0.650	0.5825	0.735	0.6487	0.820	0.7206
0.570	0.5294	0.655	0.5861	0.740	0.6528	0.825	0.7249
0.575	0.5323	0.660	0.5898	0.745	0.6569	0.830	0.7293
0.580	0.5352	0.665	0.5936	0.750	0.6611	0.835	0.7336
0.585	0.5383	0.670	0.5974	0.755	0.6652	0.840	0.7380



$S(c)$ vs. c .