Towards Sharp Inapproximability For Any 2-CSP

Per Austrin*
KTH – Royal Institute of Technology, Stockholm
austrin@kth.se

January 14, 2008

Abstract

We continue the recent line of work on the connection between semidefinite programming-based approximation algorithms and the Unique Games Conjecture. Given any boolean 2-CSP (or more generally, any nonnegative objective function on two boolean variables), we show how to reduce the search for a good inapproximability result to a certain numeric minimization problem. The key objects in our analysis are the vector triples arising when doing clause-by-clause analysis of algorithms based on semidefinite programming. Given a weighted set of such triples of a certain restricted type, which are "hard" to round in a certain sense, we obtain a Unique Games-based inapproximability matching this "hardness" of rounding the set of vector triples. Conversely, any instance together with an SDP solution can be viewed as a set of vector triples, and we show that we can always find an assignment to the instance which is at least as good as the "hardness" of rounding the corresponding set of vector triples. We conjecture that the restricted type required for the hardness result is in fact no restriction, which would imply that these upper and lower bounds match exactly. This conjecture is supported by all existing results for specific 2-CSPs.

As an application, we show that MAX 2-AND is hard to approximate within 0.87435. This improves upon the best previous hardness of $\alpha_{GW} + \epsilon \approx 0.87856$, and comes very close to matching the approximation ratio of the best algorithm known, 0.87401. It also establishes that balanced instances of MAX 2-AND, i.e., instances in which each variable occurs positively and negatively equally often, are not the hardest to approximate, as these can be approximated within a factor α_{GW}.

1 Introduction

Predicates on two boolean variables are fundamental in the study of constraint satisfaction problems. Given a set of constraints, each being a formula on two boolean variables, it is an easy task to find an assignment satisfying all constraints, if such an assignment exists. However, determining the maximum possible number of simultaneously satisfied constraints is well-known to be NP-hard. This problem is known as the MAX 2-CSP problem. It also has some very interesting special cases, the two most well-studied of which are the MAX CUT problem and the MAX 2-SAT problem. In the MAX CUT problem, each constraint is of the form $x_i \oplus x_j$, i.e., it is true if exactly one of the inputs are true. In the MAX 2-SAT problem, each constraint is of the form $l_i \vee l_j$, i.e., a disjunction on two literals, each literal being either a variable or a negated variable.

Given that the problem is NP-hard, much research has been focused on approximating the maximum number of satisfied constraints to within some factor α. An algorithm achieves approximation ratio α if the solution found by the algorithm is guaranteed to have value at least α times the optimum. We also allow for randomized algorithms, in which we require that the expected value (over the randomness of

*Research funded by Swedish Research Council Project Number 50394001.

the algorithm) of the solution found by the algorithm is α times the optimum. The arguably most trivial approximation algorithm is to simply pick a random assignment to the variables. For the general MAX 2-CSP problem, this algorithm achieves an approximation ratio of $1/4$. For the special cases of MAX CUT and MAX 2-SAT, it achieves ratios of $1/2$ and $3/4$, respectively. For several decades, no substantial improvements were made over this result, until a seminal paper by Goemans and Williamson [16], where they constructed a 0.7960-approximation algorithm for MAX 2-CSP, and 0.87856-approximation algorithms for MAX CUT and MAX 2-SAT. To do so, they relaxed the combinatorial problem at hand to a semidefinite programming problem, to which an optimal solution can be found with high precision, and then used a very clever technique to "round" the solution of the semidefinite programming back to a discrete solution for the original problem. This approach has since been succesfully applied to several other hard combinatorial optimization problems, yielding significant improvements over existing approximation algorithms. Examples include coloring graphs using as few colors as possible [20, 6, 17, 2], MAX BISECTION [15] and quadratic programming over the boolean hypercube [9].

Some of the results by Goemans and Williamson were subsequently improved by Feige and Goemans [13], who strengthened the semidefinite relaxation using certain triangle [16]. They obtained 0.931-approximation for MAX 2-SAT, and 0.859-approximation for MAX 2-CSP. These results were further improved by Matuura and Matsui [27, 28], who obtained 0.935-approximation for MAX 2-SAT and 0.863-approximation for MAX 2-CSP. Shortly thereafter, Lewin et al. [26] obtained further improvements, getting a 0.94016-approximation algorithm for MAX 2-SAT and a 0.87401-approximation algorithm for MAX 2-CSP, and these stand as the current best algorithms. It should be pointed out that these last two ratios arise as the minima of two complex numeric optimization problems, and, as far as we are aware, it has not yet been proved formally that these are the actual ratios, though there seems to be very little doubt that this is indeed the case.

Meanwhile, the study of *inapproximability* has seen a lot of progress, emanating from the discovery of the celebrated PCP theorem [4, 3]. In particular, Håstad [18] showed that the generalizations of MAX 2-SAT and MAX CUT from 2 to 3 variables, MAX 3-SAT and MAX 3-LIN-MOD2,[1] are NP-hard to approximate within factors $7/8+\epsilon$ and $1/2+\epsilon$, respectively. This surprisingly demonstrates that the random assignment algorithm is the best possible for these problems, assuming $P \neq NP$. On the other hand, MAX 3-CSP can be approximated to within a factor $1/2$ [34] which is tight by the result for MAX 3-LIN-MOD2.

For optimization problems with constraints acting on two variables, however, strong inapproximability results have been more elusive. The best NP-hardness results for MAX 2-CSP, MAX 2-SAT, and MAX CUT are $9/10+\epsilon \approx 0.900$, $21/22+\epsilon \approx 0.955$, and $16/17+\epsilon \approx 0.941$, respectively [33, 18]. The most promising approach to obtaining strong results for these problems is the so-called Unique Games Conjecture (UGC), introduced by Khot [21]. The UGC has established itself as one of the most important open problems in theoretical computer science, because of the many strong inapproximability results that follow from it. Examples of such results include $2-\epsilon$ hardness for VERTEX COVER [24], superconstant hardness for SPARSEST CUT [10, 25] and MULTICUT [10], hardness of approximating MAX INDEPENDENT SET within $d/\mathrm{poly}(\log d)$ in degree-d graphs [31], and approximation resistance[2] for random predicates [19].

For MAX 2-CSP problems, Khot et al. [22] showed that the UGC implies $\alpha_{GW}+\epsilon$ hardness for MAX CUT, where $\alpha_{GW} \approx 0.87856$ is the performance ratio of the original Goemans-Williamson algorithm, and in [5], we showed that the UGC implies $\alpha_{LLZ}+\epsilon$ hardness for MAX 2-SAT, where $\alpha_{LLZ} \approx 0.94016$ is the performance ratio of the algorithm of Lewin et al. (modulo the slight possibility that the performance ratio of their algorithm is smaller than indicated by existing analyses). It is interesting that the hardness ratios yielded by the Unique Games Conjecture exactly match these somewhat "odd" constants obtained

[1] Linear equations mod 2, where every equation has 3 variables.

[2] A predicate is approximation resistant if it is hard to do approximate the corresponding MAX CSP problem better than a random assignment.

from the complex numeric optimization problems arising from the SDP-based algorithms.

There are several other cases where the best inapproximability result, based on the UGC, matches the best approximation algorithm, based on a semidefinite programming approach. Examples include the MAX k-CSP problem [8, 31] and MAX CUT-GAIN [9, 23] (which is essentially a version of the MAX CUT problem where unsatisfied constraints give negative contribution rather than zero). This line of results is not a coincidence: in most cases, the choice of optimal parameters for the so called long code test (which is at the heart of the hardness result) are derived by analyzing worst-case scenarios for the semidefinite relaxation of the problem.

1.1 Our Contribution

In this paper, we continue to explore this tight connection between semidefinite programming relaxations and the UGC. We consider a generalization of predicates on two variables to what we call *fuzzy predicates*. A fuzzy predicate P on two variables is a function $P : \{\text{true}, \text{false}\}^2 \to [0, 1]$, rather than to $\{0, 1\}$ as would be the case with a regular predicate. We investigate the approximability of the MAX CSP(P) problem. Following the paradigm introduced by Goemans and Williamson, we relax this problem to a semidefinite programming problem. We then consider the following approach for rounding the relaxed solution to a boolean solution: given the SDP solution, we pick the "best" rounding from a certain class of randomized rounding methods (based on skewed random hyperplanes), where "best" is in the sense of giving a boolean assignment with maximum possible expected value. Informally, let $\alpha(P)$ denote the approximation ratio yielded by such an approach. We then have the following theorem.

Theorem 1.1. *For any fuzzy predicate P and $\epsilon > 0$, the* MAX CSP(P) *problem can be approximated within $\alpha(P) - \epsilon$ in polynomial time.*

The reason that we lose an additive ϵ is that we are not, in general, able to find the *best* rounding function, but we can come arbitarily close.

Then, we turn our attention to hardness of approximation. Here, we are able to take instances which are hard to round, in the sense that the best rounding (as described above) is not very good, and translate them into a Unique Games-based hardness result. There is, however, a caveat: in order for the analysis to work, the instance needs to satisfy a certain "positivity" condition. Again, informally, let $\beta(P)$ denote the approximation ratio when restricted to instances satisfying this condition. We then have

Theorem 1.2. *If the Unique Games Conjecture is true, then for any fuzzy predicate P and $\epsilon > 0$, the* MAX CSP(P) *problem is NP-hard to approximate within $\beta(P) + \epsilon$.*

Both $\alpha(P)$ and $\beta(P)$ are the solutions to a certain numeric minimization problem. The function being minimized is the same function in both cases, the only difference is that in $\alpha(P)$, the minimization is over a larger domain, and thus, we could potentially have $\alpha(P) < \beta(P)$. However, there are strong indications that the minimum for $\alpha(P)$ is in fact obtained within the domain of $\beta(P)$, in which case they would be equal and Theorems 1.1 and 1.2 would be tight.

Conjecture 1.3. *For any fuzzy predicate P, we have $\alpha(P) = \beta(P)$.*

Because of the difficulty of actually computing the approximation ratios $\alpha(P)$ and $\beta(P)$, it may seem to be somewhat difficult to compare these results to previous results. However, previous algorithms and hardness results for MAX CUT, MAX 2-SAT, and MAX 2-CSP can all be obtained as special cases of Theorems 1.1 and 1.2. In particular, for $P(x_1, x_2) = x_1 \oplus x_2$, the XOR predicate, it can be shown that $\alpha(P) = \beta(P) = \alpha_{GW}$.

We are also able to use Theorem 1.2 to obtain new results, in the form of an improved hardness of approximation for the MAX 2-AND problem, in which every constraint is an AND of two literals. This also implies improved hardness for the MAX 2-CSP problem – as is well-known, the MAX k-CSP problem and the MAX k-AND problem are equally hard to approximate for every k (folklore, or see e.g. [32]).

Theorem 1.4. *For the predicate $P(x_1, x_2) = x_1 \wedge x_2$, we have $\beta(P) \leq 0.87435$.*

This comes very close to matching the 0.87401-approximation algorithm of Lewin et al. It also demonstrates that balanced instances, i.e., instances in which each variable occurs positively and negatively equally often, are not the hardest to approximate, as these can be approximated within $\alpha_{GW} \approx 0.87856$ [22].

Finally, as a by-product of our results, we obtain some insight regarding the possibilites of obtaining improved results by strengthening the semidefinite program with more constraints. Traditionally, the only constraints which have been useful in the design of MAX 2-CSP algorithms are triangle inequalities of a certain form (namely, those involving the vector v_0, coding the value false). It turns out that, for very natural reasons, these are exactly the inequalities that need to be satisfied in order for the hardness result to carry through. In other words, assuming Conjecture 1.3 is true, it is UG-hard to do better than what can be achieved by adding only these triangle inequalities, and thus, it is unlikely that improvements can be made by adding additional inequalities (while still using polynomial time).

1.2 Techniques and Related Work

The main new ingredients of this paper are the generalizations of the various quantities used in previous results. In e.g. the case of MAX 2-SAT [5], one only had to consider one single angle, giving rise to two configurations of a very special form, something which made the calculations a lot easier. In this paper, on the other hand, we can have an arbitrary number of angles (and this is of course the reason why it is very difficult to actually compute the approximation ratios obtained), and the "positivity" condition needed here is significantly less restrictive than the special form used for MAX 2-SAT.

The proof of Theorem 1.2 follows the same path as previous proofs for specific predicates [22, 5], using the Majority Is Stablest theorem [29]. The main difference here is that we need a generalization of the "correlation under noise" quantities involved, to functions on different probability distributions. The proof of Theorem 1.1 primarily builds upon the work of [26] for MAX 2-SAT and MAX DI-CUT, the main difference being that a rounding function is chosen based on the semidefinite solution rather than beforehand, using a discretization technique to make the search a good rounding function feasible.

1.3 Organization

This paper is organized as follows. In Section 2, we set up some notation, define constraint satisfaction problems and the Unique Games Conjecture. In Section 3, we discuss the SDP relaxation of the MAX CSP(P) problem and define the constants $\alpha(P)$ and $\beta(P)$. In Section 4, we prove Theorem 1.1. In Section 5, we prove Theorem 1.2. In Section 6, we prove Theorem 1.4. Finally, in Section 7, we give some concluding remarks on our results.

2 Preliminaries

We associate the boolean values true and false with -1 and 1, respectively. Thus, a disjunction $x \vee y$ is true if $x = -1$ or $y = -1$, and a conjunction $x \wedge y$ is true if $x = y = -1$. We denote by $S^n = \{ v \in \mathbb{R}^{n+1} : ||v|| = 1 \}$ the n-dimensional unit sphere.

2.1 Constraint Satisfaction Problems

A *predicate* P on two boolean variables is a function $P : \{-1, 1\}^2 \to \{0, 1\}$. We generalize this to the notion of *fuzzy predicates*.

Definition 2.1. A *fuzzy predicate* P on two boolean variables is a function $P : \{-1, 1\}^2 \to [0, 1]$.

Note that, with general objective functions from $\{-1, 1\}^2$ to $\mathbb{R}$ in mind, the upper bound is without loss of generality, since we can always scale down any nonnegative objective function so that it takes values in $[0, 1]$ and thus becomes a fuzzy predicate.

Definition 2.2. An instance Ψ of the MAX CSP(P) problem, for a fuzzy predicate P, consists of a set of clauses and a weight function wt. Each clause ψ is a pair of literals (l_1, l_2) (a literal is either a variable or a negation of a variable), and the weight function associates with each clause ψ a nonnegative weight $\mathrm{wt}(\psi)$. We abuse notation slightly by identifying Ψ with both the instance and the set of clauses. Given an assignment $x = (x_1, \ldots, x_n)$ to the variables occurring in Ψ, and a clause $\psi = (s_1 x_i, s_2 x_j)$ (where $s_1, s_2 \in \{-1, 1\}$), we denote the restriction of x to ψ by $x|_\psi = (s_1 x_i, s_2 x_j)$. The value of an assignment x to the variables occuring in Ψ is then given by

$$\mathrm{Val}_\Psi(x) = \sum_{\psi \in \Psi} \mathrm{wt}(\psi) P(x|_\psi), \tag{1}$$

and the value of Ψ is the maximum possible value of an assignment

$$\mathrm{Val}(\Psi) = \max_x \mathrm{Val}_\Psi(x). \tag{2}$$

For convenience, we will assume (without loss of generality) that the weights are normalized so that $\mathrm{wt}(\cdot)$ is just a probability distribution on the clauses, i.e., that $\sum_{\psi \in \Psi} \mathrm{wt}(\psi) = 1$ (so $0 \leq \mathrm{Val}(\Psi) \leq 1$).

Definition 2.3. The MAX CSP$^+(P)$ problem is the special case of MAX CSP(P) where there are no negated literals (i.e. each clause is a pair of variables).

An example of the MAX CSP(P) problem which is of special interest for us is the MAX 2-AND problem, which is obtained by letting P be the predicate which is 1 if both of the inputs are true, and 0 otherwise. A well-known example of the MAX CSP$^+(P)$ problem is the MAX CUT problem, which is obtained by letting P be the predicate which is 1 if the inputs are different, and 0 if they are equal.

Any fuzzy predicate P can be arithmetized as $P(x_1, x_2) = \hat{P}_0 + \hat{P}_1 x_1 + \hat{P}_2 x_2 + \hat{P}_3 x_1 x_2$, for some constants $\hat{P}_0$, $\hat{P}_1$, $\hat{P}_2$ and $\hat{P}_3$. Thus, the MAX CSP(P) problem can be viewed as a certain special case of the integer quadratic programming problem. Throughout the remainder of this paper, we fix some arbitrary fuzzy predicate P and its corresponding coefficients $\hat{P}_0 \ldots \hat{P}_3$.

2.2 The Unique Games Conjecture

The Unique Games Conjecture was introduced by Khot [21] as a possible means to obtain new strong inapproximability results. As is common, we will formulate it in terms of a Label Cover problem.

Definition 2.4. An instance

$$X = (V, E, \mathrm{wt}, [L], \{\sigma_e^v, \sigma_e^w\}_{e=\{v,w\} \in E})$$

of UNIQUE LABEL COVER is defined as follows: given is a weighted graph $G = (V, E)$ (which may have multiple edges) with weight function $\mathrm{wt} : E \to [0, 1]$, a set $[L]$ of allowed labels, and for each edge $e = \{v, w\} \in E$ two permutations $\sigma_e^v, \sigma_e^w \in \mathfrak{S}_L$ such that $\sigma_e^w = (\sigma_e^v)^{-1}$, i.e., they are each other's inverse. We say that a function $\ell : V \to [L]$, called a labelling of the vertices, satisfies an edge $e = \{v, w\}$ if $\sigma_e^v(\ell(v)) = \ell(w)$, or equivalently, if $\sigma_e^w(\ell(w)) = \ell(v)$. The value of ℓ is the total weight of edges satisfied by it, i.e.,

$$\mathrm{Val}_X(\ell) = \sum_{\substack{e \\ \ell \text{ satisfies } e}} \mathrm{wt}(e) \tag{3}$$

The value of X is the maximum fraction of satisfied edges for any labelling, i.e.,

$$\mathrm{Val}(X) = \max_{\ell} \mathrm{Val}_X(\ell). \tag{4}$$

Without loss of generality, we will always assume that $\sum_e \mathrm{wt}(e) = 1$, i.e., that wt is in fact a probability distribution over the edges of X. We denote by $E(v)$ the subset of edges adjacent to v, i.e., $E(v) = \{\, e \mid v \in e \,\}$. The probability distribution wt induces a natural probability distribution on the vertices of X where the probability of choosing v is $\frac{1}{2}\sum_{e \in E(v)} \mathrm{wt}(e)$, and wt also induces a natural distribution on the edges of $E(v)$ where the probability of choosing $e \in E(v)$ is $\frac{\mathrm{wt}(e)}{\sum_{e \in E(v)} \mathrm{wt}(e)}$.

Whenever we speak of choosing a random element of V, E or $E(v)$, it will be according to these probability distributions, but to simplify the presentation, we will simply refer to it as a random element. For the same reason we will refer to a fraction c of the elements of V, E or $E(V)$ when in fact we mean a set of vertices/edges with probability mass c.

A UNIQUE LABEL COVER problem where G is bipartite can be viewed as a two-prover (one-round) game in which the acceptance predicate of the verifier is such that given the answer for one of the provers, there is always a unique answer from the other prover such that the verifier accepts. The probability that the verifier accepts assuming that the provers use an optimal strategy is then $\mathrm{Val}(X)$. Hence the terminology "Unique Games". We will be interested in the gap version of the UNIQUE LABEL COVER problem, which we define as follows.

Definition 2.5. GAP-UNIQUE LABEL COVER$_{\eta,\gamma,L}$ is the problem, given a UNIQUE LABEL COVER instance X with label set $[L]$, of determining whether $\mathrm{Val}(X) \geq 1 - \eta$ or $\mathrm{Val}(X) \leq \gamma$.

Khot's Unique Games Conjecture (UGC) asserts that the gap version is hard to solve for arbitrarily small η and γ, provided we take a sufficiently large label set.

Conjecture 2.6 (Unique Games Conjecture [21])**.** *For every $\eta > 0$, $\gamma > 0$, there is a constant $L > 0$ such that* GAP-UNIQUE LABEL COVER$_{\eta,\gamma,L}$ *is NP-hard.*

Note that even if the UGC turns out to be false, it might still be the case that GAP-UNIQUE LABEL COVER$_{\eta,\gamma,L}$ is hard in the sense of not being solvable in polynomial time, and such a (weaker) hardness would also apply to all other problems for which hardness has been shown under the UGC.

2.3 Influence and Correlation Under Noise

Fourier analysis of Boolean functions is a crucial tool in most strong inapproximability results. As in previous results [22, 5], the key ingredient in the proof of our hardness result is (a generalization of) the so-called Majority Is Stablest Theorem [29]. In this section, we describe this result and the exact formulation we use. Since we need to work with biased distributions rather than the standard uniform ones, we will review some important concepts. With the exception of Proposition 2.14, the propositions in this section are well-known, and proofs can be found in e.g. [5], full version. We denote by μ_q^n the probability distribution on $\{-1,1\}^n$ where each bit is set to -1 with probability q, independently, and we let B_q^n be the probability space $\left(\{-1,1\}^n, \mu_q^n\right)$.

We define a scalar product on the space of functions from B_q^n to $\mathbb{R}$ by

$$\langle f, g \rangle = \mathop{\mathbb{E}}_{x \in B_q^n}[f(x)g(x)], \tag{5}$$

and for each $S \subseteq [n]$ the function $U_q^S : B_q^n \to \mathbb{R}$ by $U_q^S(x) = \prod_{i \in S} U_q(x_i)$ where

$$U_q(x_i) = \begin{cases} -\sqrt{\frac{1-q}{q}} & \text{if } x_i = -1 \\ \sqrt{\frac{q}{1-q}} & \text{if } x_i = 1 \end{cases}.$$

Proposition 2.7. *The set of functions $\{U_q^S\}_{S\subseteq[n]}$ forms an orthonormal basis w.r.t. the scalar product $\langle\cdot,\cdot\rangle$.*

Thus, any function $f : B_q^n \to \mathbb{R}$ can be written as

$$f(x) = \sum_{S\subseteq[n]} \hat{f}_S U_q^S(x),$$

where the coefficients $\hat{f}_S = \langle f, U_q^S \rangle = \mathbb{E}_x[f(x)U_q^S(x)]$ are the Fourier coefficients of the function f. It is a straight-forward exercise to verify the basic identities $\langle f, g\rangle = \sum_{S\subseteq[n]} \hat{f}_S \hat{g}_S$, $\mathbb{E}_x[f(x)] = \hat{f}_\emptyset$ and $\mathrm{Var}_x[f(x)] = \sum_{S\neq\emptyset} \hat{f}_S^2$. We will also use $\|f\| := \sqrt{\langle f, f\rangle}$ to denote the L_2 norm of a function $f : B_q^n \to \mathbb{R}$, and remind the reader of the Cauchy-Schwartz inequality

$$|\langle f, g\rangle| \leq \|f\| \cdot \|g\|. \tag{6}$$

Definition 2.8. The *long code* of an integer $i \in [n]$ is the function $f : \{-1,1\}^n \to \{-1,1\}$ defined by $f(x) = x_i$.

Definition 2.9. A function $f : \{-1,1\}^n \to \mathbb{R}$ is said to be *folded over true* if $f(x) = -f(-x)$ for every x.

Definition 2.10. The *influence* of the variable i on the function $f : B_q^n \to \mathbb{R}$ is

$$\mathrm{Inf}_i(f) = \mathop{\mathbb{E}}_x\left[\mathop{\mathrm{Var}}_{x_i}[f(x)\,|\,x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n]\right] \tag{7}$$

The influence of the variable i is a measure of how much the variable i is able to change the value of f once we have fixed the other $n-1$ variables randomly (according to the distribution μ_q^{n-1}).

Proposition 2.11.

$$\mathrm{Inf}_i(f) = \sum_{\substack{S\subseteq[n]\\ i\in S}} \hat{f}_S^2. \tag{8}$$

Motivated by the Fourier-representation formulation of influence, we define the slightly stronger concept of low-degree influence, crucial to PCP applications.

Definition 2.12. For $k \in \mathbb{N}$, the *low-degree influence* of the variable i on the function $f : B_q^n \to \mathbb{R}$ is

$$\mathrm{Inf}_i^{\leq k}(f) = \sum_{\substack{S\subseteq[n]\\ i\in S\\ |S|\leq k}} \hat{f}_S^2. \tag{9}$$

A nice property of the low-degree influence is the fact that for functions into $[-1,1]$, $\sum_i \mathrm{Inf}_i^{\leq k}(f) \leq k$, implying that the number of variables having low-degree influence more than, say, τ, must be small (think of k and τ as constants not depending on the number of variables n). Very informally, one can think of the low-degree influence as a measure of how close the function f is to depending on only a few variables, i.e., for the case of boolean-valued functions, how close f is to being the long code of i (or its negation). Note that a long code is the extreme case of a function with large low-degree influence, in the sense that it has one variable with $\mathrm{Inf}_i^{\leq 1}(f) = 1$, and all other variables having influence 0.

Next, we introduce the *correlation under $\tilde{\rho}$-noise* between two functions $f : B_{q_1}^n \to \mathbb{R}$ and $g : B_{q_2}^n \to \mathbb{R}$. For functions into $\{-1,1\}$, the correlation under noise measures how likely f and g are to take the same value on two random inputs with a certain correlation. For $f = g$, this is simply the well-studied *noise stability* of f.

x_i	y_i	Probability
1	1	$\frac{1+\xi_1+\xi_2+\rho}{4}$
1	-1	$\frac{1+\xi_1-\xi_2-\rho}{4}$
-1	1	$\frac{1-\xi_1+\xi_2-\rho}{4}$
-1	-1	$\frac{1-\xi_1-\xi_2+\rho}{4}$

Table 1: Distribution of x and y

Definition 2.13. The *correlation under $\tilde{\rho}$-noise* between $f : B_{q_1}^n \to \mathbb{R}$ and $g : B_{q_2}^n \to \mathbb{R}$ is given by

$$\mathbb{S}_{\tilde{\rho}}(f,g) = \mathop{\mathbb{E}}_{x,y}[f(x)g(y)], \tag{10}$$

where the i:th bits of x and y are drawn from $B_{q_1}^n$ and $B_{q_2}^n$ with correlation coefficient $\tilde{\rho}$ (independently of the other bits).

Note that we can write

$$\tilde{\rho} = \mathop{\mathbb{E}}_{x_i,y_i}\left[\frac{(x_i - \mathbb{E}[x_i])(y_i - \mathbb{E}[y_i])}{\sqrt{\operatorname{Var}[x_i]\operatorname{Var}[y_i]}}\right] = \frac{\rho - \xi_1\xi_2}{\sqrt{1-\xi_1^2}\sqrt{1-\xi_2^2}} \tag{11}$$

where $\xi_1 = \mathbb{E}[x_i] = 1 - 2q_1$, $\xi_2 = \mathbb{E}[y_i] = 1 - 2q_2$, and $\rho = \mathbb{E}[x_i y_i]$. Thus, the distribution of the i:th bits of x and y can be written out explicitly as in Table 1.

We define $\mathbb{S}_{\tilde{\rho}}(f) = \mathbb{S}_{\tilde{\rho}}(f,f)$ to be the noise stability of the function f.

Proposition 2.14. *For x and y chosen as in Table 1, we have*

$$\mathbb{E}[U_{q_1}^S(x)U_{q_2}^T(y)] = \begin{cases} \tilde{\rho}^{|S|} & \text{if } S = T \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

The following proof was suggested by Marcus Isaksson.

Proof. The case when $S \neq T$ is immediately clear, since $\mathbb{E}[U_{q_1}(x_i)] = \mathbb{E}[U_{q_2}(y_i)] = 0$. For the $S = T$ case, it suffices to prove that $\mathbb{E}[U_{q_1}(x_i)U_{q_2}(y_i)] = \tilde{\rho}$. But this follows immedediately from the fact that U_{q_1} can be written as

$$U_{q_1}(x_i) = \frac{x_i - \mathbb{E}[x_i]}{\sqrt{\operatorname{Var}[x_i]}}, \tag{13}$$

and similarly for U_{q_2}, implying that $\mathbb{E}[U_{q_1}(x_i)U_{q_2}(y_i)]$ equals the correlation coefficient between x_i and y_i, which, by definition, equals $\tilde{\rho}$. ☐

Thus, we can write the correlation under noise between f and g as

$$\mathbb{S}_{\tilde{\rho}}(f,g) = \mathop{\mathbb{E}}_{x,y}\left[\sum_{S,T} \hat{f}_S U_{q_1}^S(x)\hat{g}_T U_{q_2}^T(y)\right] = \sum_{S\subseteq[n]} \tilde{\rho}^{|S|}\hat{f}_S\hat{g}_S \tag{14}$$

2.4 Functions in Gaussian Space

We denote by $\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ the standard normal density function, by $\Phi(x) = \int_{-\infty}^{x} \phi(t)\mathrm{d}t$ the standard normal distribution function, and by Φ^{-1} the inverse of Φ.

As with functions on the hypercube, we define a scalar product on functions $f, g : \mathbb{R}^n \to \mathbb{R}$ by $\langle f, g \rangle = \mathbb{E}_x[f(x)g(x)]$ (we abuse notation slightly by using the same notation as for scalar products on functions from the hypercube) where the expectation is over a standard n-dimensional Gaussian, i.e. each component being a standard $N(0,1)$ random variable. The *Ornstein-Uhlenbeck* operator U_ρ on functions $f : \mathbb{R}^n \to \mathbb{R}$ is defined as:

$$U_\rho f(x) = \mathop{\mathbb{E}}_{y}\left[f(\rho x + \sqrt{1-\rho^2}y)\right], \tag{15}$$

where the expected value is over a standard n-dimensional Gaussian y. Note that $\rho x + \sqrt{1-\rho^2}y$ is a standard n-dimensional Gaussian where each coordinate has covariance ρ with the corresponding coordinate in x. For $\mu \in [-1,1]$ we denote by $\chi_\mu : \mathbb{R} \to [0,1]$ the indicator function of an interval $(-\infty, t)$, where t is chosen so that $\mathbb{E}[\chi_\mu] = \frac{1-\mu}{2}$, i.e. $t = \Phi^{-1}\left(\frac{1-\mu}{2}\right)$.

Definition 2.15. For $\rho, \mu_1, \mu_2 \in [-1,1]$, define

$$\Gamma_\rho(\mu_1, \mu_2) = \langle \chi_{\mu_1}, U_\rho \chi_{\mu_2} \rangle = \Pr[X_1 \le t_1 \wedge X_2 \le t_2], \tag{16}$$

where $t_i = \Phi^{-1}\left(\frac{1-\mu_i}{2}\right)$ and where $X_1, X_2 \in N(0,1)$ with covariance ρ.

In other words, Γ_ρ is just the bivariate normal distribution function with a transformation on the input. Analogously to noise stability, we define $\Gamma_\rho(\mu) = \Gamma_\rho(\mu, \mu)$. The following properties of Γ_ρ will be useful.

Proposition 2.16 ([5], Lemma 2.1)**.** *For all* $\rho \in [-1,1]$, $\mu_1, \mu_2 \in [-1,1]$, *we have*

$$\Gamma_\rho(-\mu_1, -\mu_2) = \Gamma_\rho(\mu_1, \mu_2) + \mu_1/2 + \mu_2/2 \tag{17}$$

The following proposition is easily derived from [5] (full version), Proposition D.1.

Proposition 2.17. *For any* $\mu_1, \mu_1', \mu_2, \mu_2' \in [-1,1]$ *and* $\rho \in (-1,1)$, *we have*

$$\left|\Gamma_\rho(\mu_1, \mu_2) - \Gamma_\rho(\mu_1', \mu_2')\right| \le \frac{|\mu_1 - \mu_1'| + |\mu_2 - \mu_2'|}{2} \tag{18}$$

2.5 Thresholds are Extremely Correlated Under Noise

For proving hardness of MAX CUT, Khot et al. [22] made a conjecture called Majority Is Stablest, essentially stating that any boolean function with noise stability significantly higher than the majority function must have a variable with high low-degree influence (and thus in a vague sense be similar to a Long Code). Majority Is Stablest was subsequently proved by Mossel et al. [29], using a very powerful invariance principle which, essentially, allows for considering the corresponding problem over Gaussian space instead. For our result, we will use a strengthening of Majority is Stablest to two functions on the biased hypercube.

Theorem 2.18. *For any* $\epsilon > 0$, $q_1 \in (0,1), q_2 \in (0,1)$ *and* $\rho \in (-1,1)$ *there are* $\tau > 0$, $k \in \mathbb{N}$ *such that for any two functions* $f : B_{q_1}^n \to [0,1]$ *and* $g : B_{q_2}^n \to [0,1]$ *satisfying* $\mathbb{E}[f] = \frac{1-\mu_f}{2}$, $\mathbb{E}[g] = \frac{1-\mu_g}{2}$ *and*

$$\min\left(\mathrm{Inf}_i^{\le k}(f), \mathrm{Inf}_i^{\le k}(g)\right) \le \tau$$

for all $i \in [n]$, *the following holds:*

$$\mathbb{S}_\rho(f,g) \le \left\langle \chi_{\mu_f}, U_{|\rho|}\chi_{\mu_g} \right\rangle + \epsilon \tag{19}$$

$$\mathbb{S}_\rho(f,g) \ge \left\langle \chi_{\mu_f}, U_{|\rho|}(1 - \chi_{-\mu_g}) \right\rangle - \epsilon \tag{20}$$

In the terminology of [12], the setting of Theorem 2.18 corresponds to the case of a reversible noise operator, rather than a symmetric one as was studied there. It is known that the results also hold in the reversible case [11] (and in fact even in the non-reversible case [29]), but for completeness, we give a proof (following the same lines as the proof of [12]) in Appendix A. Using elementary manipulations, we obtain the following Corollary.

Corollary 2.19. *Let $\epsilon > 0$, $q_1, q_2 \in (0,1)$ and $\rho \in (-1,1)$. Then there are $\tau > 0$, $k \in \mathbb{N}$ such that for all functions $f : B_{q_1}^n \to [-1,1]$, $g : B_{q_2}^n \to [-1,1]$ satisfying $\mathbb{E}[f] = \mu_f$, $\mathbb{E}[g] = \mu_g$, and $\min(\mathrm{Inf}_i^{\leq k}(f), \mathrm{Inf}_i^{\leq k}(g)) \leq \tau$ for all i, we have*

$$4\Gamma_{-|\rho|}(\mu_f, \mu_g) - \epsilon \leq \mathbb{S}_\rho(f,g) - \mu_f - \mu_g + 1 \leq 4\Gamma_{|\rho|}(\mu_f, \mu_g) + \epsilon \tag{21}$$

Proof. Set $\tilde{f} = \frac{1-f}{2}$, $\tilde{\mu}_f = \mathbb{E}[\tilde{f}] = \frac{1-\mu_f}{2}$, and define $\tilde{g}$ and $\tilde{\mu}_g$ analogously. Thus, $\mathbb{S}_\rho(f,g) = 4\,\mathbb{S}_\rho(\tilde{f},\tilde{g}) - 2\tilde{\mu}_f - 2\tilde{\mu}_g + 1 = 4\,\mathbb{S}_\rho(\tilde{f},\tilde{g}) + \mu_f + \mu_g - 1$. By Theorem 2.18,

$$\mathbb{S}_\rho(\tilde{f},\tilde{g}) \geq \left\langle \chi_{\mu_f}, U_{|\rho|}(1 - \chi_{-\mu_g}) \right\rangle - \epsilon/4 \tag{22}$$

for any f, g where every variable has sufficiently small low-degree influence in at least one of the functions. Now, note that

$$\begin{aligned} (U_{|\rho|}(1-\chi_{-\mu_g}))(x) &= \Pr_y\left[|\rho|x + \sqrt{1-\rho^2}y \geq \Phi^{-1}(1-\tilde{\mu}_g)\right] \\ &= \Pr_y\left[-|\rho|x + \sqrt{1-\rho^2}y \leq \Phi^{-1}(\tilde{\mu}_g)\right] = U_{-|\rho|}\chi_{\mu_g}(x). \end{aligned}$$

Combining this with Equation (22) and the definition of Γ_ρ, we get

$$\mathbb{S}_\rho(f,g) \geq 4\Gamma_{-|\rho|}(\mu) + \mu_f + \mu_g - 1 - \epsilon. \tag{23}$$

The upper bound follows similarly, using Equation (19). ☐

3 Semidefinite Relaxation

One approach to solving integer quadratic programming problems which has turned out to be remarkably successful over the years is to relax the original problem to a semidefinite programming problem. This approach was first used in the seminal paper by Goemans and Williamson [16] where they gave the first approximation algorithms for MAX CUT, MAX 2-SAT, and MAX DI-CUT with a non-trivial approximation ratio (ignoring lower order terms).

For solving integer quadratic programming over the hypercube, where each variable is restricted to ±1, the standard approach is to first homogenize the program by introducing a variable x_0 which is supposed to represent the value false and then replace each term x_i by $x_0 x_i$. We then relax each variable $x_i \in \{-1,1\} = S^0$ with a vector $v_i \in S^n$ (i.e. a unit vector in $\mathbb{R}^{n+1}$), so that each term $x_i x_j$ becomes the scalar product $v_i \cdot v_j$.

In addition, we add the following inequality constraints to the program for all triples of vectors v_i, v_j, v_k.

$$v_i \cdot v_j + v_j \cdot v_k + v_i \cdot v_k \geq -1 \qquad -v_i \cdot v_j + v_j \cdot v_k - v_i \cdot v_k \geq -1 \tag{24}$$

$$v_i \cdot v_j - v_j \cdot v_k - v_i \cdot v_k \geq -1 \qquad -v_i \cdot v_j - v_j \cdot v_k + v_i \cdot v_k \geq -1 \tag{25}$$

These are equivalent to triangle inequalities of the form $||v_i - v_j||^2 + ||v_j - v_k||^2 \geq ||v_i - v_k||^2$, which clearly hold for the case that all vectors lie in a zero-dimensional subspace of S^n (so this is still a relaxation of the original integer program), but is not necessarily true otherwise. There are of course many other valid inequalities which could also be added, considering k-tuples of variables rather than just triples. In particular, adding *all* valid constraints makes the optimum for the semidefinite program equal the discrete optimum [14] (but there are an exponential number of constraints to consider). Such higher-order constraints have not received much attention, and from what is known today, it seems that the only ones which actually help are the triangle inequalities. In particular, the only inequalities which have been used when analyzing the performance of approximation algorithms, are those of the triangle inequalities which involve the vector v_0. The results of this paper shed some light on why this is the case – these are exactly the inequalities we need in order for the hardness of approximation to work out. Thus, assuming Conjecture 1.3 and the Unique Games Conjecture, it is unlikely that adding other valid inequalities (while still being able to solve the SDP in polynomial time) will help achieve a better approximation ratio, as that would imply $P = NP$.

In general, we cannot find the exact optimum of a semidefinite program. It is however possible to find the optimum to within an additive error of ϵ in time polynomial in $\log 1/\epsilon$ [1]. We ignore this small point for notational convenience and assume that we can solve the semidefinite program exactly.

Given a vector solution $\{v_i\}_{i=0}^n$, the relaxed value of a clause $\psi \in \Psi$ depends only on the three (possibly negated) scalar products $v_0 \cdot v_i$, $v_0 \cdot v_j$, and $v_i \cdot v_j$, where x_i and x_j are the two variables occuring in ψ. Most of the time, we do not care about the actual vectors, but only be interested in these triples of scalar products.

Definition 3.1. A *scalar product configuration* θ, or just a *configuration* for short, is a triple of real numbers (ξ_1, ξ_2, ρ) satisfying

$$\begin{aligned} \xi_1 + \xi_2 + \rho &\geq -1 & \qquad -\xi_1 + \xi_2 - \rho &\geq -1 \\ \xi_1 - \xi_2 - \rho &\geq -1 & \qquad -\xi_1 - \xi_2 + \rho &\geq -1 \end{aligned} \tag{26}$$

A *family of configurations* Θ is a finite set $X = \{\theta_1, \ldots, \theta_k\}$ of configurations, endowed with a probability distribution P. We routinely abuse notation by identifying Θ both with the set X and the probability space (X, P).

A configuration can be viewed as representing three vectors v_0, v_1, v_2, where $v_0 \cdot v_i = \xi_i$, and $v_1 \cdot v_2 = \rho$. Note that the inequalities in Equation (26) then correspond exactly to those of the triangle inequalities (24) which involve v_0. The important feature of these inequalities is that they precisely guarantee that Table 1 gives a valid probability distribution, which will be crucial in order for the hardness result to work out. It can also be shown that these inequalities ensures the existence of vectors v_0, v_1, v_2 with the corresponding scalar products.

Definition 3.2. The relaxed value of a configuration $\theta = (\xi_1, \xi_2, \rho)$ is given by

$$P_{\mathrm{relax}}(\theta) = P_{\mathrm{relax}}(\xi_1, \xi_2, \rho) = \hat{P}_0 + \hat{P}_1\xi_1 + \hat{P}_2\xi_2 + \hat{P}_3\rho$$

Analogously to the notation $x|_\psi$ for discrete solutions, we denote by $v|_\psi = (s_1 v_0 \cdot v_i, s_2 v_0 \cdot v_j, s_1 s_2 v_i \cdot v_j)$ the configuration arising from the clause $\psi = (s_1 x_i, s_2 x_j)$ for the vector solution $v = \{v_i\}_{i=0}^n$. The relaxed value of the clause ψ is then simply given by $P_{\mathrm{relax}}(v|_\psi)$.

Often we view the solution to the SDP as just the family of configurations $\Theta = \{\, v|_\psi \mid \psi \in \Psi \,\}$ with the probability distribution where $\Pr_{\theta \in \Theta}[\theta = v|_\psi] = \mathrm{wt}(\psi)$. The relaxed value of an assignment of vectors $\{v_i\}_{i=0}^n$ is then given by

$$\text{SDP-Val}_\Psi(\{v_i\}) = \sum_{\psi \in \Psi} \mathrm{wt}(\psi) P_{\mathrm{relax}}(v|_\psi) = \mathop{\mathbb{E}}_{\theta \in \Theta}[P_{\mathrm{relax}}(\theta)]. \tag{27}$$

Given a vector solution $\{v_i\}$, one natural attempt at an approximation algorithm is to set x_i true with probability $\frac{1-\xi_i}{2}$ (where $\xi_i = v_i \cdot v_0$), independently—the intuition being that the linear term ξ_i gives an indication of "how true" x_i should be. This assignment has the same expected value on the linear terms as the vector solution, and the expected value of a quadratic term $x_i x_j$ is $\xi_i \xi_j$. However, typically there is some correlation between the vectors v_i and v_j, so that the scalar product $v_i \cdot v_j$ contributes more than $\xi_i \xi_j$ to the objective function. To quantify this, write the vector v_i as

$$v_i = \xi_i v_0 + \sqrt{1 - \xi_i^2}\tilde{v}_i, \tag{28}$$

where $\xi_i = v_i \cdot v_0$, and $\tilde{v}_i$ is the part of v_i orthogonal to v_0, normalized to a unit vector (if $\xi_i = \pm 1$, we define $\tilde{v}_i$ to be a unit vector orthogonal to all other vectors v_j). Then, we can rewrite the quadratic term $v_i \cdot v_j$ as

$$v_i \cdot v_j = \xi_i \xi_j + \sqrt{1 - \xi_i^2}\sqrt{1 - \xi_j^2}\tilde{v}_i \cdot \tilde{v}_j. \tag{29}$$

As it turns out, the relevant parameter when analyzing the quadratic terms is the scalar product $\tilde{v}_i \cdot \tilde{v}_j$, i.e. how much better we do than if the variables would have been independent (scaled by an appropriate factor). Motivated by this, we make the following definition.

Definition 3.3. The *inner angle* $\tilde{\rho}(\theta)$ of a configuration $\theta = (\xi_1, \xi_2, \rho)$ is

$$\tilde{\rho}(\theta) = \frac{\rho - \xi_1 \xi_2}{\sqrt{1 - \xi_1^2}\sqrt{1 - \xi_2^2}}. \tag{30}$$

In the case that $\xi_1 = \pm 1$ or $\xi_2 = \pm 1$, we define $\tilde{\rho}(\theta) = 0$.

Note that, in the notation above, the advantage is exactly the scalar product $\tilde{v}_i \cdot \tilde{v}_j$. We are now ready to define the "positivity condition", alluded to in Section 1.1.

Definition 3.4. A configuration $\theta = (\xi_1, \xi_2, \rho)$ is *positive* if $\hat{P}_3 \cdot \tilde{\rho}(\theta) \geq 0$.

Intuitively, positive configurations should be more difficult to handle, since they are the configurations where we need to do something better than just setting the variables independently in order to get a good approximation ratio.

What Goemans and Williamson [16] originally did to round the vectors back to boolean variables, was to pick a random hyperplane through the origin, and decide the value of the variables based on whether their vectors are on the same side of the hyperplane as v_0 or not. Feige and Goemans [13] suggested several generalizations of this approach, using preprocessing (e.g. first rotating the vectors) and/or more elaborate choices of hyperplanes. In particular, consider a rounding scheme where we pick a random vector $r \in \mathbb{R}^{n+1}$ and then set the variable x_i to true if

$$r \cdot \tilde{v}_i \leq T(v_0 \cdot v_i) \tag{31}$$

for some threshold function $T : [-1, 1] \to \mathbb{R}$. This scheme (and more general ones) was first analyzed by Lewin et al. [26].

To describe the performance ratio yielded by this scheme, we begin by setting up some notation.

Definition 3.5. A *rounding function* is a continuous function $R : [-1, 1] \to [-1, 1]$ which is odd, i.e. satisfies $R(\xi) = -R(-\xi)$. We denote by $\mathcal{R}$ the set of all such functions.

The reason that we require a rounding function to be odd is that a negated literal $-x_i$ should be treated the opposite way as x_i. A rounding R is in one-to-one corresponce with a threshold function T as described above by the simple relation $R(x) = 1 - 2\Phi(T(x))$, where Φ is the normal distribution function (it will turn out to be more convenient to describe the rounding in terms of R rather than in terms of T).

Definition 3.6. The *rounded value* of a configuration θ with respect to a rounding function $R \in \mathcal{R}$ is

$$P_{\text{round}}(\theta, R) = P_{\text{relax}}\left(R(\xi_1), R(\xi_2), 4\Gamma_{\tilde{\rho}(\theta)}(R(\xi_1), R(\xi_2)) + R(\xi_1) + R(\xi_2) - 1\right), \tag{32}$$

This seemingly arbitrary definition is motivated by the following lemma (which essentially traces back to Lewin et al. [26], though they never made it explicit).

Lemma 3.7. *There is a polynomial-time algorithm which, given a* MAX CSP(P) *instance* Ψ*, a semidefinite solution* $\{v_i\}_{i=0}^n$ *to* Ψ*, and a (polynomial-time computable) rounding function* $R \in \mathcal{R}$*, finds an assignment to* Ψ *with expected value*

$$\mathop{\mathbb{E}}_{\theta \in \Theta}\left[P_{\text{round}}(\theta, R)\right], \tag{33}$$

Proof. The algorithm works as described above: First, we pick a random vector $r \in \mathbb{R}^{n+1}$ (i.e. each coordinate of r is a standard $N(0,1)$ random variable). Then, we set the variable x_i to true if

$$\tilde{v}_i \cdot r \leq T(v_i \cdot v_0), \tag{34}$$

where we define the threshold function T as

$$T(x) = \Phi^{-1}\left(\frac{1 - R(x)}{2}\right). \tag{35}$$

To analyze the performance of this algorithm, we need to analyze the expected values $\mathbb{E}[x_i]$ and $\mathbb{E}[x_i x_j]$.

We begin with the linear terms. These are easy, because $\tilde{v}_i \cdot r$ is just a standard $N(0,1)$ random variable, implying that x_i is set to true with probability $\frac{1-R(\xi_i)}{2}$. Thus, we have that the expected value $\mathbb{E}[x_i] = R(\xi_i)$.

For the quadratic terms, we analyze the probability that two variables x_i and x_j are rounded to the same value. It is readily verified that the covariance between the two scalar products $\tilde{v}_i \cdot r$ and $\tilde{v}_j \cdot r$ is $\tilde{\rho}$, and thus, the probability that both $\tilde{v}_i \leq T(v_i \cdot v_0)$ and $\tilde{v}_j \leq T(v_j \cdot v_0)$ is simply $\Gamma_{\tilde{\rho}}(R(\xi_i), R(\xi_j))$. By symmetry, the probability that both x_i and x_j are set to false is then $\Gamma_{\tilde{\rho}}(-R(\xi_i), -R(\xi_j))$. Using Proposition 2.16, the expected value of $x_i x_j$ is then given by

$$\begin{aligned}\mathbb{E}[x_i x_j] &= 2\left(\Gamma_{\tilde{\rho}}\left(R(\xi_i), R(\xi_j)\right) + \Gamma_{\tilde{\rho}}\left(-R(\xi_i), -R(\xi_j)\right)\right) - 1 \\ &= 4\Gamma_{\tilde{\rho}}\left(R(\xi_i), R(\xi_j)\right) + R(\xi_i) + R(\xi_j) - 1,\end{aligned} \tag{36}$$

Thus, the expected value of the solution found (over the random choice of r) is given by

$$\begin{aligned}&\mathop{\mathbb{E}}_{(\xi_1,\xi_2,\rho)\in\Theta}\left[\hat{P}_0 + \hat{P}_1 R(\xi_1) + \hat{P}_2 R(\xi_2) + \hat{P}_3(4\Gamma_{\tilde{\rho}}(R(\xi_1), R(\xi_2)) + R(\xi_1) + R(\xi_2) - 1)\right] \\ &\qquad = \mathop{\mathbb{E}}_{\theta\in\Theta}\left[P_{\text{round}}(\theta, R)\right],\end{aligned} \tag{37}$$

and we are done. □

We remark that the rounding procedure used in the proof of Lemma 3.7 is from the class of roundings Lewin et al. [26] called $\mathcal{THRESH}^-$. The rounding function R specifies an arbitrary rounding procedure from $\mathcal{THRESH}^-$.[3]

A statement similar to Lemma 3.7 holds for MAX CSP$^+(P)$, the difference being that, since there are no longer any negated literals, we can change the definition of a rounding function slightly and not require it to be odd (which could potentially give us a better algorithm). Motivated by Lemma 3.7, we make the following sequence of definitions

[3] In the notation of [26], we have $S(x) = T(x)\sqrt{1-x^2}$, or equivalently, $R(x) = 1 - 2\Phi(S(x)/\sqrt{1-x^2})$.

Definition 3.8. The approximation ratio of a rounding R for a family of configurations Θ is given by

$$\alpha_P(\Theta, R) = \frac{\mathbb{E}_{\theta\in\Theta}\left[P_{\text{round}}(\theta, R)\right]}{\mathbb{E}_{\theta\in\Theta}\left[P_{\text{relax}}(\theta)\right]} \tag{38}$$

Definition 3.9. The approximation ratio of a family of configurations Θ is given by

$$\alpha_P(\Theta) = \max_{R\in\mathcal{R}} \alpha_P(\Theta, R). \tag{39}$$

Definition 3.10. The approximation ratios of P for families of k configurations and families of k positive configurations, respectively, are given by (recall the definition of a positive configuration from Definition 3.4)

$$\alpha_P(k) = \min_{|\Theta|=k} \alpha_P(\Theta), \qquad \beta_P(k) = \min_{\substack{|\Theta|=k \\ \text{every } \theta\in\Theta \text{ is positive}}} \alpha_P(\Theta) \tag{40}$$

We would like to point out that we do not require that the family of configurations Θ can be derived from an SDP solution to some MAX CSP(P) instance Ψ – we only require that each configuration in Θ satisfies the inequalities in Equation (26). In other words, we have a lot more freedom when searching for a Θ which makes $\alpha_P(k)$ or $\beta_P(k)$ small, than we would have when searching for MAX CSP(P) instances and corresponding vector solutions.

Finally, we define

$$\alpha(P) = \lim_{k\to\infty} \alpha_P(k), \qquad \beta(P) = \lim_{k\to\infty} \beta_P(k). \tag{41}$$

These are the approximation ratios arising in Theorems 1.1 and 1.2. Ideally, of course, we would like to prove hardness of approximating MAX CSP(P) within $\alpha(P)$ rather than $\beta(P)$, getting rid of the requirement that every $\theta \in \Theta$ must be positive. The reason that we need it shows up when we do the proof of soundness for the PCP constructed in Section 5, and we have not been able to get around this. However, as we state in Conjecture 1.3, we do not *believe* that this restriction affects the approximation ratio achieved: by the intuition above, positive configurations seem to be the ones that are hard to round, so restricting our attention to such configurations should not be a problem. And indeed, the configurations we use to show hardness for MAX 2-AND are all positive, as are all configurations which have appeared in previous proofs of hardness for 2-CSPs (e.g. for MAX CUT and MAX 2-SAT).

4 The Approximation Algorithm

The approximation algorithm for MAX CSP(P) (Theorem 1.1) is based on the following theorem.

Theorem 4.1. *For any $\epsilon > 0$, the value of a* MAX CSP(P) *instance on k clauses can be approximated within $\alpha_P(k) - \epsilon$ in time polynomial in k.*

Note that this theorem immediately implies Theorem 1.1 since $\alpha_P(k) \geq \alpha(P)$. We remark that the exact value of $\alpha_P(k)$ is virtually impossible to compute for large k, making it somewhat hard to compare Theorem 4.1 with existing results. However, for MAX CUT, MAX 2-SAT and MAX 2-AND, it is not hard to prove that $\alpha(P)$ is at least the performance ratio of existing algorithms.

Proof. Let Ψ be a MAX CSP(P) instance and $\{v_i\}_{i=0}^n$ be an optimal solution to the semidefinite relaxation of Ψ. Note that, if we could find an optimal rounding function R for Ψ, the theorem would follow immediately from Lemma 3.7 (and we wouldn't need the ϵ). However, since we can not in general hope to find an optimal R, we'll discretize the set of possible angles and find the best rounding for the modified problem (for which there will be only a constant number of possible solutions).

We will use the simple facts that we always have $\mathrm{Val}(\Psi) \geq \hat{P}_0 \geq \max(|\hat{P}_1|, |\hat{P}_2|, |\hat{P}_3|)$ (to see that the second inequality holds, note that otherwise there would be x_1, x_2 such that $P(x_1, x_2) < 0$).

Construct a new SDP solution $\{u_i\}_{i=0}^n$ by letting $u_0 = v_0$, and, for each $1 \leq i \leq n$, letting u_i be the vector v_i rotated towards or away from v_0 so that $u_0 \cdot u_i$ is an integer multiple of ϵ' (where ϵ' will be chosen small enough). In other words, we have $|u_0 \cdot u_i - v_0 \cdot v_i| \leq \epsilon'/2$. For the quadratic terms, Feige and Goemans [13] proved that for $i, j \geq 1$, we have

$$u_i \cdot u_j = \zeta_i \zeta_j + \tilde{\rho}_{ij} \cdot \sqrt{1 - \zeta_i^2} \sqrt{1 - \zeta_j^2}, \tag{42}$$

where we define $\zeta_i := u_0 \cdot u_i$ and $\tilde{\rho}_{ij} := \frac{v_i \cdot v_j - \xi_i \xi_j}{\sqrt{1-\xi_i^2}\sqrt{1-\xi_j^2}}$. In other words, the rotation does not affect the value of $\tilde{\rho}_{ij}$. Thus, we have

$$v_i \cdot v_j - u_i \cdot u_j = \xi_i \xi_j - \zeta_i \zeta_j + \tilde{\rho}_{ij} \left(\sqrt{1 - \xi_i^2} \sqrt{1 - \xi_j^2} - \sqrt{1 - \zeta_i^2} \sqrt{1 - \zeta_j^2} \right). \tag{43}$$

Since $|\xi_i - \zeta_i| \leq \epsilon'/2$, we have that $|\xi_i \xi_j - \zeta_i \zeta_j| \leq 2\epsilon'$, and since

$$\left| \sqrt{1 - \xi_i^2} - \sqrt{1 - \zeta_i^2} \right| \leq \sqrt{1 - (1 - \epsilon'/2)^2} \leq \sqrt{\epsilon'}, \tag{44}$$

we have

$$\left| \tilde{\rho}_{ij} \left(\sqrt{1 - \xi_i^2} \sqrt{1 - \xi_j^2} - \sqrt{1 - \zeta_i^2} \sqrt{1 - \zeta_j^2} \right) \right| \leq 4 |\tilde{\rho}_{ij}| \sqrt{\epsilon'} \leq 4 \sqrt{\epsilon'}. \tag{45}$$

Thus, we get that

$$|v_i \cdot v_j - u_i \cdot u_j| \leq 2\epsilon' + 4\sqrt{\epsilon'}. \tag{46}$$

However, the vectors $\{u_i\}_{i=0}^n$ could possibly violate some of the triangle inequalities. To remedy this, we adjust it slightly, by again defining a new SDP solution $\{v_i'\}_{i=0}^n$ as follows (ϵ'' will be chosen momentarily)

$$v_i' = \sqrt{1 - \epsilon''} u_i + \sqrt{\epsilon''} w_i, \tag{47}$$

for $i \in \{0, \ldots, n\}$. Here, each w_i is a unit vector which is orthogonal to every other w_j, and to all the v_i' vectors (such a set of w_i vectors is trivial to construct by embedding all vectors in $\mathbb{R}^{2(n+1)}$). These new vectors satisfy $v_i' \cdot v_j' = (1 - \epsilon'') u_i \cdot u_j$ for all $i \neq j$. And since the original SDP solution $\{v_i\}_{i=0}^n$ satisfies the triangle inequalities, we have that

$$\begin{aligned} u_i \cdot u_j + u_j \cdot u_k + u_k \cdot u_i &\geq -1 - 6\epsilon' - 12\sqrt{\epsilon'} & (48)\\ v_i' \cdot v_j' + v_j' \cdot v_k' + v_k' \cdot v_i' &\geq -(1 + 6\epsilon' + 12\sqrt{\epsilon'})(1 - \epsilon''). & (49) \end{aligned}$$

Letting $\epsilon'' = 6\epsilon' + 12\sqrt{\epsilon'}$, the right hand side is at least -1, and this triangle inequality is satisfied. The other three sign combinations are handled identically. In other words, $\{v_i'\}_{i=0}^n$ is a feasible SDP solution. Its value can be lower-bounded by

$$\begin{aligned} \text{SDP-Val}(\{v_i\}) - \text{SDP-Val}(\{v_i'\}) &\leq |\hat{P}_1|(\epsilon'/2 + \epsilon'') + |\hat{P}_2|(\epsilon'/2 + \epsilon'') + |\hat{P}_3|(2\epsilon' + 4\sqrt{\epsilon'} + \epsilon'') \\ &\leq |\hat{P}_0|(21\epsilon' + 40\sqrt{\epsilon''}). & (50) \end{aligned}$$

Choosing ϵ' small enough (e.g. $\epsilon' = (\epsilon/122)^2$), this is bounded by $\frac{\epsilon}{2}\,\mathrm{Val}(\Psi)$.

Now, consider an optimal rounding function R for $\{v'_i\}$, and construct a new rounding function R' by letting $R'(\xi)$ be the nearest integer multiple of $\epsilon/8$ (so that $|R(\xi) - R'(\xi)| \le \epsilon/16$ for all ξ). We then have for any configuration $\theta' = (\xi'_1, \xi'_2, \rho')$

$$P_{\text{round}}(\theta', R) - P_{\text{round}}(\theta', R') \le |\hat{P}_1|\epsilon/16 + |\hat{P}_2|\epsilon/16 + |\hat{P}_3|(4\epsilon/16 + \epsilon/16 + \epsilon/16) \le \frac{\epsilon}{2}\,\mathrm{Val}(\Psi). \quad (51)$$

To see this, we refer to Proposition 2.17, which implies that

$$\left|\Gamma_{\tilde{\rho}}(R(\xi'_1), R(\xi'_2)) - \Gamma_{\tilde{\rho}}(R'(\xi'_1), R'(\xi'_2))\right| \le \epsilon/16 \quad (52)$$

Note that the number of possible R' is constant, roughly $(16/\epsilon)^{1/\epsilon'}$. Thus, we can find a rounding which is at least as good as R' in polynomial time by simply trying all possible choices of R', evaluating each one, and picking the best function found. Using Lemma 3.7, this means that we can find a solution to Ψ with expected value at least

$$\begin{aligned}
\mathop{\mathbb{E}}_{\theta' \in \Theta'}\left[P_{\text{round}}(\theta', R')\right] &\ge \mathop{\mathbb{E}}_{\theta' \in \Theta'}\left[P_{\text{round}}(\theta', R)\right] - \frac{\epsilon}{2}\,\mathrm{Val}(\Psi) \\
&= \alpha_P(\Theta')\,\text{SDP-Val}(\{v'_i\}) - \frac{\epsilon}{2}\,\mathrm{Val}(\Psi) \\
&\ge \alpha_P(\Theta')\,\text{SDP-Val}(\{v_i\}) - \epsilon\,\mathrm{Val}(\Psi) \\
&\ge (\alpha_P(k) - \epsilon)\,\mathrm{Val}(\Psi), \quad (53)
\end{aligned}$$

where Θ' denotes the set of configurations arising from the SDP solution $\{v'_i\}_{i=0}^n$. $\square$

We remark that the running time of the algorithm has a quite bad dependency on ϵ; it scales as $(1/\epsilon)^{\Omega(1/\epsilon^2)}$.

5 The PCP Reduction

Theorem 1.2 immediately follows from the following Theorem 5.1 below. Taking k large enough so that $\beta_P(k) \le \beta(P)+\epsilon$ and invoking Theorem 5.1 gives hardness of approximating MAX CSP(P) within $\beta(P)+2\epsilon$.

Theorem 5.1. *Assuming the Unique Games Conjecture, it is NP-hard to approximate* MAX CSP(P) *within* $\beta_P(k) + \epsilon$ *for any* $\epsilon > 0$ *and* $k \in \mathbb{N}$.

We prove Theorem 5.1 by constructing a PCP verifier which checks a supposed long coding of a good assignment to a UNIQUE LABEL COVER instance, and decides whether to accept or reject based on the evaluation of the predicate P on certain bits of the long codes. The verifier is parametrized by a family of k positive configurations $\Theta = \{\theta_1, \ldots, \theta_k\}$ and a probability distribution on Θ. Again, we point out that the requirement that the configurations of Θ are positive is by necessity rather than by choice, and if we could get rid of it, the hardness of approximation yielded would exactly match the approximation ratio from Theorem 1.1. The set Θ corresponds to a set of vector configurations for the semidefinite relaxation of MAX CSP(P). When proving soundness, i.e., in the case that there is no good assignment to the UNIQUE LABEL COVER instance, we prove that the best strategy for the prover corresponds to choosing a good rounding function R for the family of configurations Θ. Choosing a set of configurations which are hard to round, we obtain the desired result.

Since we can negate variables freely, we will assume that the purported long codes are folded over true (by selecting, for each pair $(x, -x)$ of inputs one representative, say x, and then look up the value at $-x$ by reading the value at x and negating the answer). Intuitively, this ensures that the prover's rounding function is odd, i.e. that $R(\xi) = -R(-\xi)$. For a permutation $\sigma \in \mathfrak{S}_L$ and a bitstring $x \in \{-1, 1\}^L$, we denote by

Algorithm 1: The verifier $\mathcal{V}_\Theta$

$\mathcal{V}_\Theta(X, \Sigma = \{f_v\}_{v\in V})$

(1) Pick a random configuration $\theta = (\xi_1, \xi_2, \rho) \in \Theta$ according to the distribution on Θ.

(2) Pick a random $v \in V$.

(3) Pick $e_1 = \{v, w_1\}$ and $e_2 = \{v, w_2\}$ randomly from $E(v)$.

(4) Pick $x_1, x_2 \in \{-1, 1\}^L$ such that each bit of x_i is picked independently with expected value ξ_i and that the j:th bits of x_1 and x_2 are ρ-correlated for $j = 1, \ldots, L$.

(5) For $i = 1, 2$, let $b_i = f_{w_i}(\sigma^v_{e_i} x_i)$ (folded over true).

(6) Accept with probability $P(b_1, b_2)$.

$\sigma x \in \{-1, 1\}^L$ the string x permuted according to σ, i.e., $\sigma x = x_{\sigma(1)} x_{\sigma(2)} \ldots x_{\sigma(L)}$. The verifier is given in Algorithm 1. Note that, because θ is a configuration, Equation (26) guarantees that we can choose x_1 and x_2 with the desired distribution in step (4).

We now analyze the completeness and soundness of the verifier. Arithmetizing the acceptance predicate, we find that the acceptance probability of $\mathcal{V}_\Theta$ can be written as

$$\mathop{\mathbb{E}}_{\theta\in\Theta}\left[\mathop{\mathbb{E}}_{v,e_1,e_2,x_1,x_2}\left[\hat{P}_0 + \hat{P}_1 f_{w_1}(\sigma^v_{e_1} x_1) + \hat{P}_2 f_{w_2}(\sigma^v_{e_2} x_2) + \hat{P}_3 f_{w_1}(\sigma^v_{e_1} x_1) f_{w_2}(\sigma^v_{e_1} x_2) \;\middle|\; \theta\right]\right] \tag{54}$$

5.1 Completeness

Lemma 5.2 (Completeness)**.** *If* $\mathrm{Val}(X) \geq 1 - \eta$*, then there is a proof* Σ *such that*

$$\Pr[\mathcal{V}_\Theta(X, \Sigma) \textit{ accepts}] \geq (1 - 2\eta) \mathop{\mathbb{E}}_{\theta\in\Theta}[P_{\mathrm{relax}}(\theta)] \tag{55}$$

Proof. Fix a labelling ℓ of the vertices of X such that the fraction of satisfied edges is at least $1 - \eta$, and let $f_v : \{-1, 1\}^L \to \{-1, 1\}$ be the Long Code of the label of the vertex v. Note that for a satisfied edge $\{v, w\}$ and an arbitrary biststring $x \in \{-1, 1\}^L$, $f_w(\sigma^v_e x)$ equals the value of the $\ell(v)$:th bit of x.

Fix a choice of $\theta = (\xi_1, \xi_2, \rho)$. By the union bound, the probability that any of the two edges e_1, e_2 chosen by $\mathcal{V}_\Theta$ are not satisfied is at most 2η. For a choice of edges that *are* satisfied, the expected value of $f_{w_i}(\sigma^v_{e_i} x_i)$ is the expected value of the $\ell(v)$:th bit of x_i, i.e. ξ_i, and the expected value of $f_{w_1}(\sigma^v_{e_1} x_1) f_{w_2}(\sigma^v_{e_2} x_2)$ is the expected value of the $\ell(v)$:th bit of $x_1 x_2$, i.e. ρ.

Thus, the probability that $\mathcal{V}_\Theta$ accepts is at least

$$\mathop{\mathbb{E}}_{\theta\in\Theta}\left[(1 - 2\eta)(\hat{P}_0 + \hat{P}_1 \xi_1 + \hat{P}_2 \xi_2 + \hat{P}_3 \rho)\right] = (1 - 2\eta) \mathop{\mathbb{E}}_{\theta\in\Theta}[P_{\mathrm{relax}}(\theta)], \tag{56}$$

and the proof is complete. □

5.2 Soundness

Lemma 5.3 (Soundness)**.** *For every* $\epsilon > 0$ *there is a* $\gamma > 0$ *such that if* $\mathrm{Val}(X) \leq \gamma$*, then for any proof* Σ*, we have*

$$\Pr[\mathcal{V}_\Theta(X, \Sigma) \textit{ accepts}] \leq \max_{R\in\mathcal{R}} \mathop{\mathbb{E}}_{\theta\in\Theta}[P_{\mathrm{round}}(\theta, R)] + \epsilon. \tag{57}$$

Proof. For $\xi \in [-1, 1]$ and $v \in V$, define $g^\xi_v : B^n_{(1-\xi)/2} \to \{-1, 1\}$ by

$$g^\xi_v(x) = \mathop{\mathbb{E}}_{e=\{v,w\}\in E(v)}\left[f_w(\sigma^v_e x)\right], \tag{58}$$

and define the function $R_v(\xi) := \mathbb{E}\left[g_v^\xi(x)\right]$. Note that since we fold the purported Long Codes over true, we have that both g_v^ξ and R_v are odd functions, and in particular that $R_v \in \mathcal{R}$. We remark that for a fixed v and different values of ξ, the functions g_v^ξ are the same function, but since the probability distributions of their inputs have an almost disjoint support (in the probabilistic sense), we might as well treat them as independent of each other.

We can now write $\mathcal{V}_\Theta$:s acceptance probability as

$$\begin{aligned}\Pr[\mathcal{V}_\Theta \text{ accepts}] &= \mathop{\mathbb{E}}_{\theta}\left[\mathop{\mathbb{E}}_{v,x_1,x_2}\left[\hat{P}_0 + \hat{P}_1 g_v^{\xi_1}(x_1) + \hat{P}_2 g_v^{\xi_2}(x_2) + \hat{P}_3 g_v^{\xi_1}(x_1) g_v^{\xi_2}(x_2) \mid \theta\right]\right] \\ &= \mathop{\mathbb{E}}_{\theta,v}\left[\hat{P}_0 + \hat{P}_1 R_v(\xi_1) + \hat{P}_2 R_v(\xi_2) + \hat{P}_3\, \mathbb{S}_{\tilde{\rho}(\theta)}(g_v^{\xi_1}, g_v^{\xi_2})\right], \end{aligned} \tag{59}$$

Assume that the probability that $\mathcal{V}_\Theta$ accepts is at least

$$\begin{aligned}&\Pr[\mathcal{V}_\Theta \text{ accepts}] \geq \mathop{\mathbb{E}}_{\theta,v}\left[P_{\text{round}}(\theta, R_v)\right] + \epsilon \\ &= \mathop{\mathbb{E}}_{\theta,v}\left[\hat{P}_0 + \hat{P}_1 R_v(\xi_1) + \hat{P}_2 R_v(\xi_2) + \hat{P}_3(4\Gamma_{\tilde{\rho}}(R_v(\xi_1), R_v(\xi_2)) + R_v(\xi_1) + R_v(\xi_2) - 1)\right] + \epsilon. \end{aligned} \tag{60}$$

Combining this with Equation (59), this implies that there exists a $\theta = (\xi_1, \xi_2, \rho) \in \Theta$ such that

$$\mathop{\mathbb{E}}_{v}\left[\hat{P}_3 \cdot \left(\mathbb{S}_{\tilde{\rho}(\theta)}(g_v^{\xi_1}, g_v^{\xi_2}) - 4\Gamma_{\tilde{\rho}(\theta)}(R_v(\xi_1), R_v(\xi_2)) - R_v(\xi_1) - R_v(\xi_2) + 1\right)\right] \geq \epsilon. \tag{61}$$

Using the fact that the absolute value of the expression in the expectancy is bounded by $2|\hat{P}_3|$, this implies that for a fraction $\epsilon' := \frac{\epsilon}{3|\hat{P}_3|}$ of all $v \in V$, we have

$$\hat{P}_3 \cdot \mathbb{S}_{\tilde{\rho}(\theta)}(g_v^{\xi_1}, g_v^{\xi_2}) \geq \hat{P}_3\left(4\Gamma_{\tilde{\rho}(\theta)}(R_v(\xi_1), R_v(\xi_2)) + R_v(\xi_1) + R_v(\xi_2) - 1\right) + \epsilon'. \tag{62}$$

Let V_{good} be the set of all such v. Using that θ is a positive configuration (i.e. $\hat{P}_3\tilde{\rho}(\theta) \geq 0$), we then get that for $v \in V_{\text{good}}$,

$$\mathbb{S}_{\tilde{\rho}(\theta)}(g_v^{\xi_1}, g_v^{\xi_2}) \geq 4\Gamma_{|\tilde{\rho}(\theta)|}(R_v(\xi_1), R_v(\xi_2)) + R_v(\xi_1) + R_v(\xi_2) - 1 + \epsilon'/|\hat{P}_3| \tag{63}$$

if $\hat{P}_3 > 0$, or

$$\mathbb{S}_{\tilde{\rho}(\theta)}(g_v^{\xi_1}, g_v^{\xi_2}) \leq 4\Gamma_{-|\tilde{\rho}(\theta)|}(R_v(\xi_1), R_v(\xi_2)) + R_v(\xi_1) + R_v(\xi_2) - 1 - \epsilon'/|\hat{P}_3| \tag{64}$$

if $\hat{P}_3 < 0$. In either case, Majority is stablest (Corollary 2.19) implies that there are constants τ and k (depending only on ϵ, θ, and P) such that for any $v \in V_{\text{good}}$ we have $\mathrm{Inf}_i^{\leq k}(g_v^{\xi_1}) \geq \tau$ (and also that $\mathrm{Inf}_i^{\leq k}(g_v^{\xi_2}) \geq \tau$, though we will not use that). Fixing θ and dropping the bias parameter ξ_1 for the remainder of the proof, we have that for any $v \in V_{\text{good}}$,

$$\tau \leq \mathrm{Inf}_i^{\leq k}(g_v) \leq \mathop{\mathbb{E}}_{e=\{v,w\}}\left[\mathrm{Inf}_{\sigma_e^v(i)}^{\leq k}(f_w)\right], \tag{65}$$

and since $\mathrm{Inf}_{\sigma_e^v(i)}^{\leq k}(f_w) \leq 1$ for all e, this implies that for at least a fraction $\tau/2$ of all edges $e = \{v, w\} \in E(v)$, we have $\mathrm{Inf}_{\sigma_e^v(i)}^{\leq k}(f_w) \geq \tau/2$. For $v \in V$, let

$$C(v) = \{\, i \in L \mid \mathrm{Inf}_i^{\leq k}(f_v) \geq \tau/2 \vee \mathrm{Inf}_i^{\leq k}(g_v) \geq \tau \,\}. \tag{66}$$

Intuitively, the criterion $\mathrm{Inf}_i^{\leq k}(f_v) \geq \tau/2$ means that the purported Long Codes of the label of v suggests i as a suitable label for v, and the criterion $\mathrm{Inf}_i^{\leq k}(g_v) \geq \tau$ means that many of the purported Long Codes for the neighbors of v suggests that v should have the label i. By the fact that $\sum_i \mathrm{Inf}_i^{\leq k}(f_v) \leq k$, we must have $|C(v)| \leq 2k/\tau + k/\tau = 3k/\tau$.

We now define a labelling by picking independently for each $v \in V$ a (uniformly) random label $i \in C(v)$ (or an arbitrary label in case $C(v)$ is empty). For a label $v \in V_{\mathrm{good}}$ with $\mathrm{Inf}_i^{\leq k}(g_v) \geq \tau$, the probability that v is assigned label i is $1/|C(v)| \geq \tau/3k$. Furthermore, by the above reasoning and the definition of C, at least a fraction $\tau/2$ of the edges $e = \{v, w\}$ from v will satisfy $\sigma_e^v(i) \in C(w)$. For such an edge, the probability that w is assigned the label $\sigma_e^v(i)$ is $1/|C(w)| \geq \tau/3k$. Thus, the expected fraction of satisfied edges adjacent to any $v \in V_{\mathrm{good}}$ is at least $\tau/2 \cdot (\tau/3k)^2$, and so the expected fraction of satisfied edges in total[4] is at least $\epsilon' \cdot \frac{\tau^3}{18k^2}$ and thus there is an assignment satisfying at least this total weight of edges. Note that this is a positive constant that depends only on ϵ and θ, and P. Making sure that $\gamma < \frac{\epsilon'\tau^3}{18k^2}$, we get a contradiction on the assumption of the acceptance probability (Equation (60)), implying that the soundness is at most

$$\begin{aligned} \Pr[\mathcal{V}_\Theta \text{ accepts } \Sigma] &\leq \mathop{\mathbb{E}}_{\theta, v}[P_{\mathrm{round}}(\theta, R_v)] + \epsilon & (67) \\ &\leq \max_{R \in \mathcal{R}} \mathop{\mathbb{E}}_{\theta \in \Theta}[P_{\mathrm{round}}(\theta, R)] + \epsilon, & (68) \end{aligned}$$

and we are done. □

5.3 Wrapping It Up

Combining the two lemmas and picking η small enough, we get that it is Unique Games-hard to approximate MAX CSP(P) within

$$\max_{R \in \mathcal{R}} \frac{\mathbb{E}_{\theta \in \Theta}[P_{\mathrm{round}}(\theta, R)]}{\mathbb{E}_{\theta \in \Theta}[P_{\mathrm{relax}}(\theta)]} + \mathcal{O}(\epsilon) = \alpha_P(\Theta) + \mathcal{O}(\epsilon)\,. \quad (69)$$

Picking a Θ with $|\Theta| = k$ that minimizes $\alpha_P(\Theta)$, we obtain Theorem 5.1.

6 Application to MAX 2-AND

Using the machinery developed in Sections 3 and 5, we are able to obtain an upper bound of $\beta(P) \leq 0.87435$ for the case when $P(x_1, x_2) = x_1 \wedge x_2$, i.e., the MAX 2-AND problem, establishing Theorem 1.4. We do this by exhibiting a set Θ of $k = 4$ (positive) configurations on 2 distinct non-zero ξ-values (and a probability distribution on the elements of Θ), such that $\alpha_P(\Theta) < 0.87435$.

Before doing this, let us start with an even simpler set of configurations, sufficient to give an inapproximability of 0.87451, only marginally worse than 0.87435. This set of configurations $\Theta = \{\theta_1, \theta_2\}$ contains only one non-zero ξ-value, and is given by

$$\begin{aligned} \theta_1 &= (0, -\xi, 1 - \xi) \quad \text{with probability } 0.64612 \\ \theta_2 &= (0, \xi, 1 - \xi) \quad \text{with probability } 0.35377, \end{aligned}$$

where $\xi = 0.33633$.

[4] We remind the reader of the convention of Section 2.2 that the choices of random vertices and edges are according to the probability distributions induced by the weights of the edges, and so choosing a random $v \in V$ and then a random $e \in E(v)$ is equivalent to just choosing a random $e \in E$.

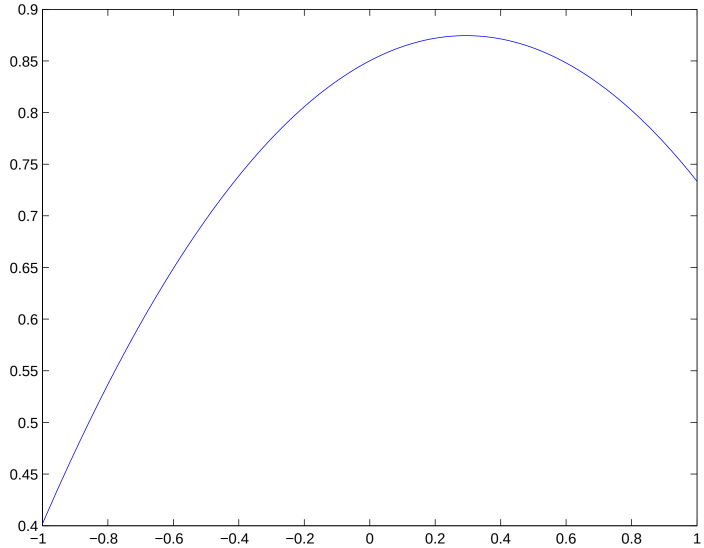


Figure 1: Approximation ratio as a function of R

To compute the hardness factor given by this set of configurations, we must compute

$$\alpha_P(\Theta) = \max_{R \in \mathcal{R}} \frac{\mathbb{E}_{\theta\in\Theta}[P_{\text{round}}(\theta, R)]}{\mathbb{E}_{\theta\in\Theta}[P_{\text{relax}}(\theta)]}. \tag{70}$$

Since $P(x_1, x_2) = \frac{1-x_1-x_2+x_1x_2}{4}$ we have that for an arbitrary configuration $\theta = (\xi_1, \xi_2, \rho)$,

$$\begin{aligned}
P_{\text{relax}}(\theta) &= \frac{1-\xi_1-\xi_2+\rho}{4} \\
P_{\text{round}}(\theta, R) &= \frac{1 - R(\xi_1) - R(\xi_2) + 4\Gamma_{\tilde{\rho}(\theta)}(R(\xi_1), R(\xi_2)) + R(\xi_1) + R(\xi_2) - 1}{4} \\
&= \Gamma_{\tilde{\rho}(\theta)}(R(\xi_1), R(\xi_2)).
\end{aligned}$$

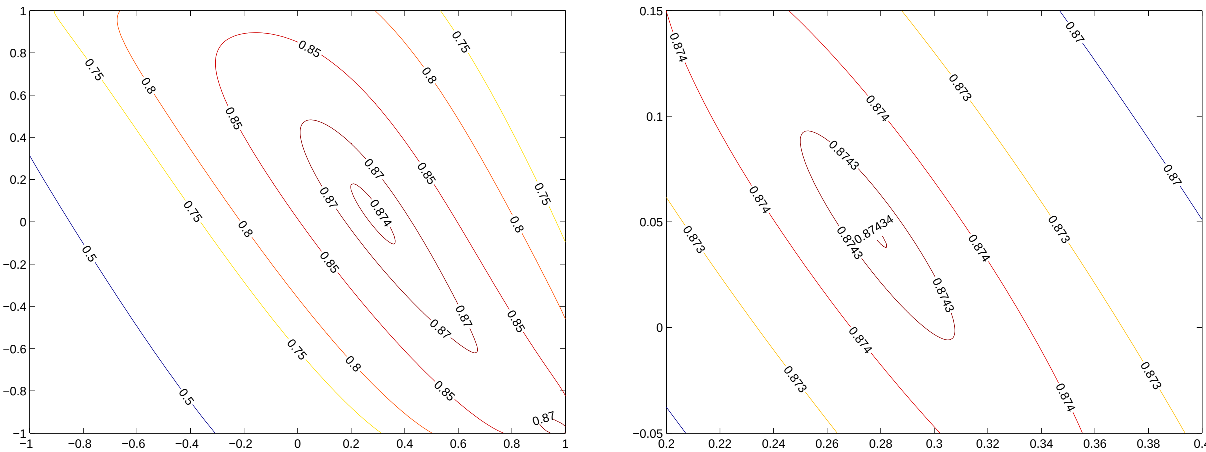
In our case, using the two configurations given above, R is completely specified by its value on the angle ξ (since $R(0) = 0$ and $R(-\xi) = -R(\xi)$). Figure 1 gives a plot of the right-hand side of Equation (70), as a function of the value of $R(\xi)$. The maximum turns out to occur at $R(\xi) \approx 0.29412$, and gives a ratio of approximately 0.87450517. Thus, we see that $\alpha_P(\Theta) \le 0.87451$. We remark that it is not very difficult to make this computation rigorous—it can be proven analytically that the curve of Figure 1 is indeed convex, and so the only maximum can be computed to within high precision (using easy bounds on the derivative) using a simple ternary search.

Let us now turn to the larger set of configurations, based on four configurations, mentioned earlier. This set of configurations $\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4\}$ is as follows:

$$\begin{aligned}
\theta_1 &= (0, -\xi_A, 1-\xi_A) && \text{with probability } 0.52850 \\
\theta_2 &= (0, \xi_A, 1-\xi_A) && \text{with probability } 0.05928 \\
\theta_3 &= (\xi_A, -\xi_B, 1-\xi_A-\xi_B) && \text{with probability } 0.29085 \\
\theta_4 &= (-\xi_A, \xi_B, 1-\xi_A-\xi_B) && \text{with probability } 0.12137,
\end{aligned}$$

where $\xi_A = 0.31988$ and $\xi_B = 0.04876$.

As before, to compute the approximation ratio given by Θ, we need to find the best R for Θ, and again, such an R is completely specified by its values on the non-zero ξ-values. In other words, we now need to



(a) The entire range of R

(b) Restricted to the critical area

Figure 2: Approximation ratio as a function of R

specify the values of R on the two angles ξ_A and ξ_B. Figure 2(a) gives a contour plot of approximation ratio, as a function of the values of $R(\xi_A)$ and $R(\xi_B)$. There are now two local maxima, one around the point $(R(\xi_A), R(\xi_B)) \approx (0.27846, 0.044376)$, and one around the point $(1, -1)$. Figure 2(b) gives a contour plot of the area around the first point. This maximum turns out to be approximately 0.87434075. At the point $(1, -1)$ (which is indeed the other maximum), the approximation ratio is approximately 0.87434007. Thus, we have $\alpha_P(\Theta) \leq 0.87435$.

In general, given Θ (and a probability distribution on its elements), the very problem of computing $\alpha_P(\Theta)$ is a difficult numeric optimization problem. However, for the Θ we use, the number of distinct ξ-values used is small, so that computing $\alpha_P(\Theta)$ in this case is a numeric optimization problem in 2 variables, which we are able to handle.

It seems likely that additional improvements can be made by using more and more ξ-values, though these improvements will be quite small. Indeed, using larger Θ we are able to improve upon Theorem 1.4, but the improvements we have been able to make are minute (of order 10^{-5}), and it becomes a lot more difficult to verify them. Note that θ_1 and θ_2 used in the larger set of configurations are very similar to the first set of configurations—they are of the same form, and the ξ-value used is only slightly different. It appears that it is useful to follow this pattern when adding even more configurations: the values of ξ_A and ξ_B are adjusted sligtly, and we add two configurations of the form $(\pm\xi_B, \mp\xi_C, 1 - \xi_B - \xi_C)$. Essentially this type of sequence of configurations has appeared before, see e.g. the analysis of lower bounds for certain MAX DI-CUT algorithms in [35].

7 Concluding Remarks

We remark that it is a fairly straightforward task to adapt these results to the MAX CSP$^+(P)$ problem, obtaining statements analogous to Theorems 1.1 and 1.2. The only difference is that we drop the requirement that a rounding function has to be odd (since we cannot fold the long codes over true anymore, we would not be able to enforce such a constraint). However, in doing so, we also lose the possibility to force a rounding function R to satisfy $R(0) = 0$. The configurations that we use for proving hardness of MAX 2-AND rely heavily on this property, and it is for this reason that those results do not apply to the MAX DI-CUT problem directly. In other words, we are not able to obtain a statement similar to Theorem 1.4 for the MAX DI-CUT problem. Whether this is because the MAX DI-CUT problem is easier to approximate than MAX 2-AND, or whether we just have to spend some more time searching for a “bad” set of configurations, we do not know, but we conjecture that the latter is true and that they are equally hard. However, today we do not even know whether balanced instances of the MAX DI-CUT problem are the hardest or not.

If P is monotone, the MAX $\mathrm{CSP}^{+}(P)$ problem is trivially solvable, so there are cases where MAX $\mathrm{CSP}^{+}(P)$ is easier than MAX $\mathrm{CSP}(P)$. Lacking results on MAX DI-CUT, it would be interesting to determine whether there are other examples than these trivial ones. A good candidate would probably be an “almost monotone” P (recall that P is real-valued.).

Recently, O’Donnell and Wu have done a complete analysis of the “approximability curve” of the MAX CUT problem, exhibiting an algorithm, integrality gap, and UGC-based hardness result which all match [30]. It will be interesting to see whether their results can be extended to other MAX 2-CSP problems.

7.1 Acknowledgements

I would like to thank Johan Håstad for luring me into the world of 2-CSPs in the first place, as well as for his patience and many insightful comments along the way. I would also like to thank Marcus Isaksson for suggesting the short proof of Proposition 2.14.

References

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1995.

[2] S. Arora, E. Chlamtac, and M. Charikar. New Approximation Guarantee for Chromatic Number. In *STOC 2006*, pages 205–214, 2006.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof Verification and the Hardness of Approximation Problems. *Journal of the ACM*, 45(3):501–555, 1998.

[4] S. Arora and S. Safra. Probabilistic Checking of Proofs: A New Characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998.

[5] P. Austrin. Balanced Max 2-Sat Might Not be the Hardest. In *ACM Symposium on Theory of Computing (STOC)*, pages 189–197, 2007.

[6] A. Blum and D. Karger. An $\tilde{\mathcal{O}}(n^{3/14})$-coloring algorithm for 3-colorable graphs. *Information Processing Letters*, 61(1):49–53, 1997.

[7] C. Borell. Geometric bounds on the Ornstein-Uhlenbeck velocity process. *Z. Wahrsch. Verw. Gebiete*, 70(1):1–13, 1985.

[8] M. Charikar, K. Makarychev, and Y. Makarychev. Approximation Algorithm for the Max k-CSP Problem, 2006.

[9] M. Charikar and A. Wirth. Maximizing Quadratic Programs: Extending Grothendieck’s Inequality. In *FOCS 2004*, pages 54–60, 2004.

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

[11] I. Dinur, E. Friedgut, and O. Regev. Independent sets in graph powers are almost contained in juntas. *Geometric and Functional Analysis*, 2006. to appear.

[12] I. Dinur, E. Mossel, and O. Regev. Conditional hardness for approximate coloring. In *STOC 2006*, pages 344–353, 2006.

[13] U. Feige and M. Goemans. Aproximating the Value of Two Prover Proof Systems, With Applications to MAX 2SAT and MAX DICUT. In *ISTCS 1995*, pages 182–189, 1995.

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

[15] A. M. Frieze and M. Jerrum. Improved Approximation Algorithms for MAX k-CUT and MAX BISECTION. *Algorithmica*, 18(1):67–81, 1997.

[16] M. X. Goemans and D. P. Williamson. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. *Journal of the ACM*, 42:1115–1145, 1995.

[17] E. Halperin, R. Nathaniel, and U. Zwick. Coloring k-colorable graphs using smaller palettes. In *SODA 2001*, pages 319–326, 2001.

[18] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.

[19] J. Håstad. On the approximation resistance of a random predicate. To appear in RANDOM-APPROX, 2007.

[20] D. R. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. *Journal of the ACM*, 45(2):246–265, 1998.

[21] S. Khot. On the power of unique 2-prover 1-round games. In *STOC 2002*, pages 767–775, 2002.

[22] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal Inapproximability Results for Max-Cut and Other 2-Variable CSPs? In *FOCS 2004*, pages 146–154, 2004.

[23] S. Khot and R. O'Donnell. SDP gaps and UGC-hardness for MAXCUTGAIN. In *FOCS 2006*, pages 217–226, 2006.

[24] S. Khot and O. Regev. Vertex Cover Might be Hard to Approximate to within $2-\epsilon$. In *IEEE Conference on Computational Complexity*, pages 379–, 2003.

[25] S. Khot and N. K. Vishnoi. The Unique Games Conjecture, Integrality Gap for Cut Problems and Embeddability of Negative Type Metrics into l_1. In *FOCS 2005*, pages 53–62, 2005.

[26] M. Lewin, D. Livnat, and U. Zwick. Improved rounding techniques for the MAX 2-SAT and MAX DI-CUT problems. In *IPCO 2002*, volume 2337 of *Lecture Notes in Computer Science*, pages 67–82, 2002.

[27] S. Matuura and T. Matsui. 0.863-Approximation Algorithm for MAX DICUT. In *RANDOM-APPROX*, pages 138–146, 2001.

[28] S. Matuura and T. Matsui. 0.935-Approximation Randomized Algorithm for MAX 2SAT and Its Derandomization. Technical Report METR 2001-03, Department of Mathematical Engineering and Information Physics, the University of Tokyo, Japan, 2001.

[29] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. To appear in Annals of Mathematics, 2007.

[30] R. O'Donnell and Y. Wu. An optimal SDP algorithm for Max-Cut, and equally optimal Long Code tests. Manuscript, 2007.

[31] A. Samorodnitsky and L. Trevisan. Gowers uniformity, influence of variables, and PCPs. In *STOC 2006*, pages 11–20, 2006.

[32] L. Trevisan. Parallel Approximation Algorithms by Positive Linear Programming. *Algorithmica*, 21:72–88, 1998.

[33] L. Trevisan, G. B. Sorkin, M. Sudan, and D. P. Williamson. Gadgets, Approximation, and Linear Programming. *SIAM Journal on Computing*, 29(6):2074–2097, 2000.

[34] U. Zwick. Approximation Algorithms for Constraint Satisfaction Problems Involving at Most Three Variables Per Constraint. In *SODA 1998*, 1998.

[35] U. Zwick. Analyzing the MAX 2-SAT and MAX DI-CUT approximation algorithms of Feige and Goemans. Manuscript, 2000.

A Proof of Theorem 2.18

In this section, we prove Theorem 2.18. The proof is essentially the same as the proof of Dinur et al. [12] for a similar theorem. They consider a more general class of noise operators than the ones we need and functions over the m-ary hypercube rather than just the Boolean hypercube. On the other hand, they only consider functions on the uniform distribution.

Theorem (Theorem 2.18 restated)**.** *For any $\epsilon > 0$, $q_1 \in (0,1), q_2 \in (0,1)$ and $\rho \in (-1,1)$ there is a $\tau > 0$, $k \in \mathbb{N}$ such that for any two functions $f : B_{q_1}^n \to [0,1]$ and $g : B_{q_2}^n \to [0,1]$ satisfying $\mathbb{E}[f] = \frac{1-\mu_f}{2}$, $\mathbb{E}[g] = \frac{1-\mu_g}{2}$ and*

$$\min\left(\mathrm{Inf}_i^{\leq k}(f), \mathrm{Inf}_i^{\leq k}(g)\right) \leq \tau$$

for all $i \in [n]$, the following holds:

$$\mathbb{S}_\rho(f,g) \leq \left\langle \chi_{\mu_f}, U_{|\rho|}\chi_{\mu_g}\right\rangle + \epsilon \tag{71}$$

$$\mathbb{S}_\rho(f,g) \geq \left\langle \chi_{\mu_f}, U_{|\rho|}(1-\chi_{-\mu_g})\right\rangle - \epsilon \tag{72}$$

Proof. First, note that it suffices to prove Equation (71), since if it is true, we have

$$\begin{aligned} \mathbb{S}_\rho(f,g) &= \mathbb{S}_\rho(f,\mathbf{1}) - \mathbb{S}_\rho(f,\mathbf{1}-g) \\ &\geq \left\langle \chi_{\mu_f}, U_{|\rho|}\mathbf{1}\right\rangle - \left\langle \chi_{\mu_f}, U_{|\rho|}\chi_{-\mu_g}\right\rangle - \epsilon \\ &= \left\langle \chi_{\mu_f}, U_{|\rho|}(1-\chi_{-\mu_g})\right\rangle - \epsilon, \end{aligned} \tag{73}$$

where we note that $\mathbb{S}_\rho(f,\mathbf{1}) = \left\langle \chi_{\mu_f}, U_{|\rho|}\mathbf{1}\right\rangle = \frac{1-\mu_f}{2}$.

The proof will be based on the following Lemma:

Lemma A.1. *Let $q_1 \in (0,1), q_2 \in (0,1)$ and $\rho \in (-1,1)$. Then for any $\epsilon > 0$, $\eta < 1$, there exists $\tau > 0$ and $k > 0$ such that for any functions $f : B_{q_1}^n \to [0,1], g : B_{q_2}^n \to [0,1]$ satisfying $\mathbb{E}[f] = \frac{1-\mu_f}{2}$, $\mathbb{E}[g] = \frac{1-\mu_g}{2}$,*

$$\max\left(\mathrm{Inf}_i^{\leq k}(f), \mathrm{Inf}_i^{\leq k}(g)\right) \leq \tau \quad \forall i \tag{74}$$

and

$$\sum_{|S|\geq d} \hat{f}_S^2 \leq \eta^{2d}, \sum_{|S|\geq d} \hat{g}_S^2 \leq \eta^{2d} \quad \forall d \tag{75}$$

it holds that

$$\mathbb{S}_\rho(f,g) \leq \left\langle \chi_{\mu_f}, U_{|\rho|}\chi_{\mu_g} \right\rangle + \epsilon \tag{76}$$

Note that the Fourier coefficients of f and g are with respects to different measures. Before proving Lemma A.1, we show how to use it to complete the proof of Theorem 2.18.

Pick $\eta < 1$ large enough so that $|\rho|^j(1-\eta^{2j}) < \epsilon/4$ for all j, and let τ', k' be the values given by Lemma A.1 with the parameters q_1, q_2, ρ, $\epsilon/4$ and η. Set k large enough so that both $|\rho|^k \leq \epsilon/4$ and $k \geq k'$. Let

$$S_f = \{\, i \mid \mathrm{Inf}_i^{\leq k}(f) \geq \tau' \,\}, \qquad S_g = \{\, i \mid \mathrm{Inf}_i^{\leq k}(g) \geq \tau' \,\} \tag{77}$$

Define $f' : B_{q_1}^S \to [0,1]$ and $g' : B_{q_2}^S \to [0,1]$ by

$$f' = \sum_{\substack{S\subseteq[n] \\ S\cap S_f=\emptyset}} \eta^{|S|}\hat{f}_S U_{q_1}^S \tag{78}$$

$$g' = \sum_{\substack{S\subseteq[n] \\ S\cap S_g=\emptyset}} \eta^{|S|}\hat{g}_S U_{q_2}^S \tag{79}$$

Now, for $i \in S_f$, we have $\mathrm{Inf}_i^{\leq k'}(f') = 0$, whereas for $i \notin S_f$, we have $\mathrm{Inf}_i^{\leq k'}(f') \leq \mathrm{Inf}_i^{\leq k}(f) \leq \tau'$, and similarly for g'. Thus, we have that $\max(\mathrm{Inf}_i^{\leq k}(f'), \mathrm{Inf}_i^{\leq k}(g')) \leq \tau'$ for every i. Furthermore,

$$\sum_{|S|\geq d} \widehat{f'}_S^2 \leq \eta^{2d} \sum_S \hat{f}_S^2 \leq \eta^{2d}, \tag{80}$$

and similarly for g', so Lemma A.1 gives that

$$\mathbb{S}_\rho(f',g') \leq \left\langle \chi_{\mu_f}, U_{|\rho|}\chi_{\mu_g} \right\rangle + \epsilon/4 \tag{81}$$

What remains is to bound the difference between $\mathbb{S}_\rho(f,g)$ and $\mathbb{S}_\rho(f',g')$. We have

$$\begin{aligned}
|\,\mathbb{S}_\rho(f,g) - \mathbb{S}_\rho(f',g')| &= \left| \sum_{\substack{S\cap S_f=\emptyset \\ S\cap S_g=\emptyset}} \rho^{|S|}\left(1-\eta^{2|S|}\right)\hat{f}_S\hat{g}_S + \sum_{S\cap(S_f\cup S_g)\neq\emptyset} \rho^{|S|}\hat{f}_S\hat{g}_S \right| \\
&\leq \sum_{\substack{S\cap S_f=\emptyset \\ S\cap S_g=\emptyset}} \frac{\epsilon}{4}|\hat{f}_S\hat{g}_S| + \sum_{\substack{S\cap(S_f\cup S_g)\neq\emptyset \\ |S|\leq k}} |\hat{f}_S\hat{g}_S| + \sum_{\substack{S\cap(S_f\cup S_g)\neq\emptyset \\ |S|\geq k}} \left|\rho^k\hat{f}_S\hat{g}_S\right| \\
&\leq \sum_{S\subseteq[n]} \frac{\epsilon}{2}|\hat{f}_S\hat{g}_S| + \sum_{\substack{S\cap(S_f\cup S_g)\neq\emptyset \\ |S|\leq k}} |\hat{f}_S\hat{g}_S|
\end{aligned} \tag{82}$$

By Cauchy-Schwartz, the first term is bounded by $\frac{\epsilon}{2} \cdot \|f\| \cdot \|g\| \leq \epsilon/2$. The second term is bounded by (again using Cauchy-Schwartz)

$$\sum_{i\in S_f\cup S_g} \sum_{\substack{i\in S \\ |S|\leq k}} |\hat{f}_S\hat{g}_S| \leq \sum_{i\in S_f\cup S_g} \sqrt{\mathrm{Inf}_i^{\leq k}(f)}\sqrt{\mathrm{Inf}_i^{\leq k}(g)} \tag{83}$$

Now, we have that both $|S_f|$ and $|S_g|$ are bounded by $2k/\tau'$. Furthermore, at least one of $\mathrm{Inf}_i^{\le k}(f)$ and $\mathrm{Inf}_i^{\le k}(g)$ is bounded by τ (the value of which we have not yet determined), and since both are bounded by 1 we have

$$\sum_{i \in S_f \cup S_g} \sqrt{\mathrm{Inf}_i^{\le k}(f)}\sqrt{\mathrm{Inf}_i^{\le k}(g)} \le 4k/\tau' \cdot \sqrt{\tau} \tag{84}$$

Setting $\tau \le \left(\frac{\epsilon\tau'}{k}\right)^2$, this is at most $\epsilon/4$. Thus, we conclude that

$$\mathbb{S}_\rho(f,g) \le \mathbb{S}_\rho(f',g') + 3\epsilon/4 \le \langle \chi_{\mu_f}, U_{|\rho|}\chi_{\mu_g} \rangle + \epsilon, \tag{85}$$

and we are done. □

A.1 Proof of Lemma A.1

What remains is the proof of Lemma A.1. Before proceeding with this, we have to introduce some new notation.

Definition A.2 (Real analogue of a function)**.** Let $f : B_q^n \to \mathbb{R}$ be a function with Fourier expansion

$$f = \sum_{S \subseteq [n]} \hat{f}_S U_q^S \tag{86}$$

We define the real analogue $\tilde{f} : \mathbb{R}^n \to \mathbb{R}$ to be

$$\tilde{f}(z_1, \ldots, z_n) = \sum_{S \subseteq [n]} \hat{f}_S \tilde{U}^S(z_1, \ldots, z_n), \tag{87}$$

where $\tilde{U}^S(z_1, \ldots, z_n) = \prod_{i \in S} z_i$.

Note that the set of functions $\{\tilde{U}^S\}_{S \subseteq [n]}$ forms an orthonormal basis (w.r.t. the scalar product defined in Section 2.4). It is a fairly straightforward exercise to verify that

$$\left\langle \tilde{f}, U_\rho \tilde{g} \right\rangle = \sum_{S \subseteq [n]} \rho^{|S|} \hat{f}_S \hat{g}_S = \mathbb{S}_\rho(f,g) \tag{88}$$

for any $\rho \in [-1,1]$.

Definition A.3. For any function f with range $\mathbb{R}$ define

$$\mathrm{chop}(f)(x) = \begin{cases} f(x) & \text{if } f(x) \in [0,1] \\ 0 & \text{if } f(x) < 0 \\ 1 & \text{if } f(x) > 1 \end{cases} \tag{89}$$

The proof of Lemma A.1 relies on two powerful theorems. The first is a version of Mossel et al.'s invariance principle.

Theorem A.4 (Mossel et al. [29], Theorem 3.20 under hypothesis **H3**)**.** *For any $q \in (0,1)$, $\tau > 0$ and $0 < \eta < 1$, let $K = \log(1/\min(q, 1-q))$, $k = \log(1/\tau)/K$. Then for any $f : U_q^n \to [0,1]$ satisfying*

$$\mathrm{Inf}_i^{\le k}(f) \le \tau \quad \forall i \quad \text{and} \quad \sum_{|S| \ge d} \hat{f}_S^2 \le \eta^{2d} \quad \forall d, \tag{90}$$

the following holds:

$$\|\tilde{f} - \mathrm{chop}(\tilde{f})\| \le \tau^{\Omega((1-\eta)/K)} \tag{91}$$

The second is the following powerful theorem of Borell. [7]

Theorem A.5 (Borell [7]). *Let $\rho \in [0,1]$ and $F, G : \mathbb{R}^n \to [0,1]$ with $\mathbb{E}[F] = \frac{1-\mu_f}{2}$, $\mathbb{E}[G] = \frac{1-\mu_g}{2}$. Then*

$$\langle F, U_\rho G \rangle \leq \left\langle \chi_{\mu_f}, U_\rho \chi_{\mu_g} \right\rangle \tag{92}$$

Note that Theorem A.5 implies that $\langle F, U_{-\rho} G \rangle \leq \left\langle \chi_{\mu_f}, U_\rho \chi_{\mu_g} \right\rangle$. To see this, take $G'(x) = G(-x)$, so that $\langle F, U_{-\rho} G \rangle = \langle F, U_\rho G' \rangle$ and $\mathbb{E}[G] = \mathbb{E}[G']$). Thus, we have $\langle F, U_\rho G \rangle \leq \left\langle \chi_{\mu_f}, U_{|\rho|} \chi_{\mu_g} \right\rangle$ for any $\rho \in [-1,1]$.

We are now ready to prove the Lemma.

Proof of Lemma A.1. Let $\mu'_f = \frac{1-\mathbb{E}[\mathrm{chop}(\tilde{f})]}{2}$, $\mu'_g = \frac{1-\mathbb{E}[\mathrm{chop}(\tilde{g})]}{2}$. Set $\epsilon' = \epsilon/3$. Pick τ small enough so that Theorem A.4 gives that both $\|\operatorname{chop}(\tilde{f}) - \tilde{f}\| \leq \epsilon'$ and $\|\operatorname{chop}(\tilde{g}) - \tilde{g}\| \leq \epsilon'$, and pick k accordingly. Now, we have

$$\begin{aligned} \mathbb{S}_\rho(f,g) &= \left\langle \tilde{f}, U_\rho \tilde{g} \right\rangle \\ &= \left\langle \operatorname{chop}(\tilde{f}), U_\rho \operatorname{chop}(\tilde{g}) \right\rangle + \\ &\qquad \left\langle \tilde{f} - \operatorname{chop}(\tilde{f}), U_\rho \operatorname{chop}(\tilde{g}) \right\rangle + \left\langle U_\rho \tilde{f}, \tilde{g} - \operatorname{chop}(\tilde{g}) \right\rangle, \end{aligned} \tag{93}$$

where we used that $\left\langle \tilde{f}, U_\rho \tilde{g} \right\rangle = \left\langle U_\rho \tilde{f}, \tilde{g} \right\rangle$. By Cauchy-Schwartz, the last two terms are bounded by

$$\|\tilde{f} - \operatorname{chop}(\tilde{f})\| \cdot \|U_\rho \operatorname{chop}(\tilde{g})\| + \|U_\rho \tilde{f}\| \cdot \|\tilde{g} - \operatorname{chop}(\tilde{g})\|, \tag{94}$$

which in turn is bounded by $2\epsilon'$, since both $\|U_\rho \operatorname{chop}(\tilde{g})\|$ and $\|U_\rho \tilde{f}\|$ are at most 1. Thus,

$$\mathbb{S}_\rho(f,g) \leq \left\langle \operatorname{chop}(\tilde{f}), U_\rho \operatorname{chop}(\tilde{g}) \right\rangle + 2\epsilon' \tag{95}$$

Applying Borell's theorem to $\mathrm{chop}(\tilde{f})$ and $\mathrm{chop}(\tilde{g})$, we have

$$\left\langle \operatorname{chop}(\tilde{f}), U_\rho \operatorname{chop}(\tilde{g}) \right\rangle \leq \left\langle \chi_{\mu'_f}, U_{|\rho|} \chi_{\mu'_g} \right\rangle \tag{96}$$

To relate this to $\left\langle \chi_{\mu_f}, U_{|\rho|} \chi_{\mu_g} \right\rangle$, note that we have

$$\begin{aligned} |\mu_f - \mu'_f| &= |\mathbb{E}[\tilde{f} - \operatorname{chop}(\tilde{f})]|/2 = \left| \left\langle \tilde{f} - \operatorname{chop}(\tilde{f}), \mathbf{1} \right\rangle \right| /2 \\ &\leq \|\tilde{f} - \operatorname{chop}(\tilde{f})\|/2 \leq \epsilon'/2, \end{aligned} \tag{97}$$

and similarly for $|\mu_g - \mu'_g|$. Applying Proposition 2.17, this gives

$$\left\langle \chi_{\mu'_f}, U_{|\rho|} \chi_{\mu'_g} \right\rangle \leq \left\langle \chi_{\mu_f}, U_{|\rho|} \chi_{\mu_g} \right\rangle + \epsilon'/2, \tag{98}$$

In conclusion, we have $\mathbb{S}_\rho(f,g) \leq \left\langle \chi_{\mu_f}, U_{|\rho|} \chi_{\mu_g} \right\rangle + 3\epsilon'$, as desired. $\square$