ON THE APPROXIMATION RESISTANCE OF A RANDOM PREDICATE

Johan Håstad

Abstract. A predicate is called approximation resistant if it is NP-hard to approximate the corresponding constraint satisfaction problem significantly better than what is achieved by the naive algorithm that picks an assignment uniformly at random. In this paper we study predicates of Boolean inputs where the width of the predicate is allowed to grow. Samorodnitsky and Trevisan proved that, assuming the Unique Games Conjecture, there is a family of very sparse predicates that are approximation resistant. We prove that, under the same conjecture, any predicate implied by their predicate remains approximation resistant and that, with high probability, this condition applies to a randomly chosen predicate.

Keywords. Constraint Satisfaction Problem, Approximation Algorithms, Approximation Resistance

Subject classification. 68W25, 68Q25

1. Introduction

We consider constraint satisfaction problems (CSPs) over the Boolean domain. In our model a problem is defined by a k-ary predicate P and an instance is given by a list of k-tuples of literals. The task is to find an assignment to the variables such that all the k-bit strings resulting from the list of k-tuples of literals under the assignment satisfy the predicate P. In this paper we focus on Max-CSPs which are optimization problems where we try to satisfy as many constraints as possible.

The most famous such problem is probably Max-3-Sat where $k = 3$ and P is the disjunction of the three bits. Another problem that (almost) falls into this category is Max-Cut, in which $k = 2$ and P is non-equality. In traditional Max-Cut we do not allow negations among the literals and if we do allow negation the problem becomes Max-E2-Lin-2, linear equations modulo 2 with exactly two variables in each equation.

It is a classical result that most Boolean CSPs are NP-complete. In an early result Schaefer (1978) gave a complete characterization giving only 5 cases for which the problem is in P while establishing NP-completeness in the other cases.

Of course if a CSP is NP-complete, the corresponding Max-CSP is NP-hard. The converse is false and several of Schaefer's easy satisfiability problems are in fact NP-hard as optimization problems. We turn to study approximation algorithms. An algorithm is here considered to be a C-approximation of a maximization problem if it, on each input, finds an assignment with an objective value that is at least C times that of the optimal solution. We allow randomized approximation algorithms and in such a case it is sufficient that the expected value, over the random choices of the algorithm, of the objective value satisfies the desired bound.

To define what is non-trivial is a matter of taste but hopefully there is some consensus that the following algorithm is trivial: Without looking at the instance, pick a random value for each variable. We say that an approximation ratio is non-trivial if it gives a value of C that is better than the value obtained by this trivial algorithm. We call a predicate *approximation resistant* if it is NP-hard to achieve a non-trivial approximation ratio.

It is perhaps surprising but many CSPs are approximation resistant and one basic example, as established by Håstad (2001), is Max-3-Sat. In their famous algorithm Goemans & Williamson (1995) show that Max-Cut is not approximation resistant and this result was extended by Håstad (2005) to show that no predicate that depends on two inputs from an arbitrary finite domain can be approximation resistant.

Zwick (1998) established approximability results for predicates that depend on three Boolean inputs and from this work it follows that the only predicates on three inputs that are approximation resistant are those that are implied by parity or its negation. A predicate P is implied by a predicate Q iff whenever $Q(x)$ is true so is $P(x)$; as an example, the negation of parity implies disjunction as if we know that an odd number of variables are true then they cannot all be false.

Some scattered results on (families of) predicates depending on four or more Boolean inputs do exist and some can be found in the papers by Guruswami *et al.* (1998) and Samorodnitsky & Trevisan (2000). On a more systematic note, Hast (2005) made an extensive classification of predicates of four inputs. Predicates that can be made equal by permuting the inputs and/or negating one or more inputs behave the same with respect to approximation resistance and with this notion of equivalence there are (exactly) 400 different non-constant predicates on 4 Boolean inputs. Hast proved that 79 of these are approximation resistant, established 275 to be non-trivially approximable leaving the status of 46 predicates open. Zwick (2006) has obtained numerical evidence suggesting that most of the latter predicates are in fact non-trivially approximable.

The main result of this paper is to give evidence that a random k-ary predicate for a large value of k is approximation resistant. The result is only evidence in the sense that it relies on the Unique Games Conjecture (UGC) of Khot (2002), but on the other hand we establish that a vast majority of the predicates are approximation resistant under this assumption.

We base our proof on the recent result by Samorodnitsky & Trevisan (2006) that establishes that if d is the smallest integer such that $2^d - 1 \geq k$ then there is a predicate of width k that accepts only 2^d of the 2^k possible k-bit strings and which, based on the UGC, is approximation resistant. We first extend their proof to establish that any predicate implied by their predicate is approximation resistant.

This extension is neither difficult nor surprising given the existing machinery. To prove that a predicate P is approximation resistant one needs to design a Probabilistically Checkable Proof (PCP) where the verifier reads k bits and accepts iff these bits satisfy the predicate P. To establish approximation resistance this test needs to have (almost) perfect completeness and a soundness that (almost) matches the probability that a random assignment satisfies P. To extend this to a predicate Q that is implied by P one obvious attempt is to use the same PCP up to the acceptance criteria which we change to be that the constructed string satisfies Q rather than P. As P implies Q the completeness of the test remains (almost) perfect and we need only address the soundness condition. It turns out, in our situation, that the original proof of the soundness from Samorodnitsky & Trevisan (2006) is not difficult to modify.

To establish our main result we proceed to prove that a random predicate is implied by some predicate which is equivalent to the predicate of Samorodnitsky and Trevisan. This is established by a second moment method. A standard random predicate on k bits is constructed by, for each of the 2^k inputs, flipping an unbiased coin to determine whether that input is accepted. It turns out that our results apply to other spaces of random predicates. In fact, if we construct a random predicate by accepting each input with probability k^{-c} for some $c > 0$ we still, with high probability for sufficiently large k, get an approximation resistant predicate. Here c is a number in the range $[1/2, 1]$ that depends on how close k is to the smallest number of the form $2^d - 1$ larger than k.

We make the proof more self-contained by reproving one main technical lemma of Samorodnitsky & Trevisan (2006) relating to Gowers uniformity norms and influences of variables. Our proof is similar in spirit to the original proof but significantly shorter and we hence believe it is of independent interest.

Of course the contribution of this paper heavily depends on how one views the Unique Games Conjecture, UGC. At the least one can conclude that it will be difficult to give a non-trivial approximation algorithm for a random predicate. Our results also point to the ever increasing need to settle the UGC.

An outline of the paper is as follows. We start by establishing some notation and giving some definitions in Section 2. We prove the lemmas relating to Gowers uniformity in Section 3 and proceed, in Section 4, to establish that any predicate implied by the predicate used by Samorodnitsky and Trevisan is approximation resistant. We then present our applications of this theorem by first establishing that a random predicate is approximation resistant in Section 5 and that all very dense predicates are approximation resistant in Section 6. We stress that all our inapproximability results rely on the UGC. We end with some concluding remarks in Section 7.

This is the full version of the conference version given in Håstad (2007).

2. Preliminaries

We consider functions mapping $\{-1,1\}^n$ into the real numbers and usually into the closed interval $[-1,1]$. In this paper we use $\{-1,1\}$ as the value set of Boolean variables but still call the values “bits”. For $x, x' \in \{-1,1\}^n$ we let $x \cdot x'$ denote the coordinate-wise product. In $\{0,1\}^n$-notation this is the simply the exclusive-or of vectors.

For any $\alpha \subseteq [n]$ we have the character χ_α which is defined by

$$\chi_\alpha(x) = \prod_{i \in \alpha} x_i$$

and the Fourier expansion is given by

$$f(x) = \sum_{\alpha \subseteq [n]} \hat{f}_\alpha \chi_\alpha(x),$$

where

$$\hat{f}_\alpha = 2^{-n} \sum_x f(x) \chi_\alpha(x).$$

These numbers satisfy the Parseval identity

$$\sum_\alpha \hat{f}_\alpha^2 = 2^{-n} \sum_x f(x)^2,$$

and as we are mostly considering functions that map into the real interval $[-1, 1]$ these sums are usually bounded by 1.

We are interested in coding elements $[L]$, by the *long code.* This code was introduced by Bellare *et al.* (1998) and an element v is coded by a function $A : \{-1, 1\}^L \to \{-1, 1\}$ defined by $A(x) = x_v$. Our main PCP consists of a number of tables that are supposed to be long codes. We cannot trust a malicious prover to give tables that are long codes but we limit the power of such a prover by requiring a table to be *folded* which is the same as saying that $A(-x) = -A(x)$. This is enforced by requiring the table only to contain elements for inputs with $x_0 = 1$. The value when $x_0 = -1$ is defined to be $-A(-x)$. It is important to us that any folded table is unbiased, i.e. has expectation 0.

Permutations of $[L]$ play an important role in this papers and we extend permutations to act on $\{-1, 1\}^L$ by defining $\pi(x)_j = x_{\pi(j)}$. For a set $S \subseteq \{-1, 1\}^L$ we let $\pi(S)$ be the set

$$\{\pi(x)|x \in S\}.$$

For two sets α and β we let $\alpha\Delta\beta$ be the symmetric difference of the two sets.

The influence of a variable x_i on a function f, denoted by $\text{inf}_i f$, is the expected variance of f when all variables, except x_i, are fixed to random variables. It is well known that

$$\text{inf}_i = \sum_{i\in\alpha} \hat{f}_\alpha^2.$$

The following lemma from Samorodnitsky & Trevisan (2000) is useful.

LEMMA 2.1. *Let $(f_j)_{j=1}^k$, $\{-1, 1\}^n \to [-1, 1]$ be k functions, and*

$$f(x) = \prod_{j=1}^k f_j(x).$$

Then, for every $i \in [n]$, $\text{inf}_i(f) \leq k \sum_{j=1}^k \text{inf}_i(f_j)$.

The pairwise cross-influence of a set of functions $(f_j)_{j=1}^k$ is defined to be the maximal simultaneous influence in any two of the functions or more formally

$$\text{cinf}_i(f_j)_{j=1}^k = \max_{j_1\neq j_2} \min(\text{inf}_i(f_{j_1}), \text{inf}_i(f_{j_2})).$$

Let P be a predicate on k Boolean inputs. An instance of the problem Max-CSP(P) is given by a list of k-tuples of literals. The task is to find the assignment to the variables that maximizes the number of k-tuples that satisfy P.

An algorithm is a C-approximation if it, for any instance I of this problem, produces an assignment which satisfies at least $C \cdot Opt(I)$ constraints where $Opt(I)$ is the number of constraints satisfied by an optimal solution.

Let $d(P)$ be the fraction of k-bit strings accepted by P. The trivial algorithm that just picks an assignment uniformly at random satisfies, on the average, a $d(P)$-fraction of the constraints and as an optimal solution cannot satisfy more than all the constraints this yields a (randomized) $d(P)$-approximation algorithm. We have the following definition.

Definition 2.2. *A predicate P is* approximation resistant *if, for any $\epsilon > 0$, it is NP-hard to approximate Max-CSP(P) within $d(P) + \epsilon$.*

Some predicates have an even stronger property and to state it let us first formally define implication among predicates.

Definition 2.3. *A predicate P is implied by a predicate Q if, for any x, whenever $Q(x)$ is true so is $P(x)$.*

A different way of formalizing this fact is, of course, that $Q^{-1}(1) \subseteq P^{-1}(1)$. We are now ready for a definition of an even stronger notion of hardness.

Definition 2.4. *A predicate P is* hereditary approximation resistant *if any predicate Q implied by P is approximation resistant.*

3. Gowers Uniformity and Influence

Gowers (1998, 2001) introduced the notion of dimension-d uniformity norm $U^d(f)$ which was used in an essential way by Samorodnitsky & Trevisan (2006). Their result says that if a function does not have an influential variable and is unbiased then the dimension-d uniformity norm is small. More importantly for their application, Samorodnitsky & Trevisan (2006) also proved that if a set of functions has small cross-influences and at least one function is unbiased then a product similar to the product giving the U^d-norm is small. We slightly extend their result by allowing a small bias of the involved functions. Allowing this extension makes it possible to give a short, direct proof by induction.

We want to emphasize that the results obtained by Samorodnitsky and Trevisan are sufficient for us but we include the results of this section since we believe that our proofs are simpler and that the extension might be interesting on its own and possibly useful in some other context.

THEOREM 3.1. *Let* $f\colon \{-1,1\}^n \to [-1,1]$ *be a function with* $\max_i \inf_i(f) \le \epsilon$ *and* $|E[f]| \le \delta$, *then*

$$\left| E_{x^1,\dots x^d}\left[\prod_{S\subseteq[d]} f\left(\prod_{i\in S} x^i\right)\right]\right| \le \delta + (2^{d-1}-1)\sqrt{\epsilon}.$$

PROOF. We prove the theorem by induction over d. Clearly it is true for $d = 1$ as the quantity to estimate equals $|f(1^n)E[f]|$.

For the induction step let $g^{x^d}(x) = f(x)f(x\cdot x^d)$. Then, by Lemma 2.1, $\max_i \inf_i g^{x^d} \le 4\epsilon$. Furthermore

$$E_x[g^{x^d}] = 2^{-n}\sum_x f(x)f(x\cdot x^d) = f * f(x^d) \tag{3.2}$$

and let us for notational simplicity denote this function by $h(x^d)$. As convolution turns into multiplication on the Fourier transform side we have that $\hat{h}_\alpha = \hat{f}_\alpha^2$. For any $\alpha \ne \emptyset$ we have $\hat{f}_\alpha^2 \le \max_i \inf_i(f) \le \epsilon$ and hence

$$\|h\|_2^2 = \sum_\alpha \hat{h}_\alpha^2 = \sum_\alpha \hat{f}_\alpha^4 \le \hat{f}_\emptyset^4 + \epsilon\sum_{\alpha\ne\emptyset}\hat{f}_\alpha^2 \le \delta^4 + \epsilon.$$

This implies, using the Cauchy-Schwartz inequality, that

$$E_{x^d}[|E_x[g^{x^d}(x)]|] = E_{x^d}[|h(x_d)|] \le \sqrt{\delta^4+\epsilon} \le \delta^2 + \sqrt{\epsilon} \le \delta + \sqrt{\epsilon}. \tag{3.3}$$

Now

$$\left| E_{x^1,\dots x^d}\left[\prod_{S\subseteq[d]} f\left(\prod_{i\in S} x^i\right)\right]\right| \le E_{x^d}\left| E_{x^1,\dots x^{d-1}}\left[\prod_{S\subseteq[d-1]} g^{x^d}\left(\prod_{i\in S} x^i\right)\right]\right|,$$

which, by induction, is bounded by

$$E_{x^d}\left[|E_x[g^{x^d}]| + (2^{d-2}-1)\sqrt{4\epsilon}\right] \le \delta + (2^{d-1}-1)\sqrt{\epsilon}.$$

□

Theorem 3.1 is really a warmup for Theorem 3.4 below which is the theorem needed for our application. For any reader interested in Theorem 3.1 in itself let us point out that it is possible to get a sharper estimate by not doing the wasteful replacement of δ^2 by δ in (3.3).

We now turn to the more general situation.

THEOREM 3.4. *Let $(f_S)_{S\subseteq[d]}$ be a set of functions $\{-1,1\}^n \to [-1,1]$, with $\max_i \mathrm{cinf}_i(f_S) \le \epsilon$ and $\min_{S\neq\emptyset} |E[f_S]| \le \delta$, then*

$$\left| E_{x^1,\ldots x^d}\left[\prod_{S\subseteq[d]} f_S(\prod_{i\in S} x^i)\right]\right| \le \delta + (2^d-2)\sqrt{\epsilon}.$$

PROOF. We use induction over d. The base case $d=1$ is again straightforward and let us do the induction step.

By a change of variables we can assume that $|E[f_{[d]}]| \le \delta$. Now define a new set of functions by

$$g_S^{x^d}(x) = f_S(x) f_{S\cup\{d\}}(x\cdot x^d),$$

for any $S \subseteq [d-1]$. The cross-influence of this set of functions is, by Lemma 2.1, bounded by 4ϵ. Let $h(x^d)$ be the average of $g_{[d-1]}^{x^d}$ and as, similarly to (3.2) we have that h is the convolution of $f_{[d-1]}$ and $f_{[d]}$ we conclude that $\hat{h}_\alpha = \hat{f}_{[d-1],\alpha}\hat{f}_{[d],\alpha}$. From this we get

$$\begin{aligned} \|h\|_2^2 &= \sum_\alpha \hat{h}_\alpha^2 = \hat{f}^2_{[d-1],\emptyset}\hat{f}^2_{[d],\emptyset} + \sum_{\alpha\neq\emptyset} \hat{f}^2_{[d-1],\alpha}\hat{f}^2_{[d],\alpha} \le \\ &\le \delta^2 + \sum_{\alpha\neq\emptyset} \min(\hat{f}^2_{[d-1],\alpha}, \hat{f}^2_{[d],\alpha})(\hat{f}^2_{[d-1],\alpha} + \hat{f}^2_{[d],\alpha}) \le \\ &\le \delta^2 + \sum_{\alpha\neq\emptyset} \epsilon(\hat{f}^2_{[d-1],\alpha} + \hat{f}^2_{[d],\alpha}) \le \delta^2 + 2\epsilon. \end{aligned}$$

Using induction and the Cauchy-Schwartz inequality as in the previous proof we get

$$\left| E_{x^1,\ldots x^d}\left[\prod_{S\subseteq[d]} f_S\left(\prod_{i\in S} x^i\right)\right]\right| \le E_{x^d}\left| E_{x^1,\ldots x^{d-1}}\left[\prod_{S\subseteq[d-1]} g_S^{x^d}\left(\prod_{i\in S} x^i\right)\right]\right| \le$$

$$E_{x^d}\left[|E[g^{x^d}_{[d-1]}]| + (2^{d-1}-2)\sqrt{4\epsilon}\right] \le \delta + 2\sqrt{\epsilon} + (2^d-4)\sqrt{\epsilon} \le \delta + (2^d-2)\sqrt{\epsilon}.$$

□

4. The ST-predicate

Fix k and let d be such that $2^{d-1} \le k \le 2^d - 1$. For any integer i with $0 \le i \le 2^d - 1$ let $\hat{i} \subseteq [d]$ be the set whose characteristic vector is equal to the binary expansion of i. We define $P_{ST}(x)$, a predicate on k-bit strings, to be true if for all triplets i_1, i_2, and i_3 such that $\hat{i}_1 \Delta \hat{i}_2 = \hat{i}_3$ we have $x_{i_1} x_{i_2} = x_{i_3}$. Note that P_{ST} depends on k but as k (and d) remains fixed we suppress this dependence.

The accepted strings form a linear space of dimension d and the following procedure for picking a random string accepted by P_{ST} is a good way to visualize the predicate. For each i that is a power of two, set x_i to a random bit. For other values of i set

$$x_i = \prod_{j \in \hat{i}} x_{2^j}.$$

We consider Max-CSP(P_{ST}) and have the following theorem obtained by Samorodnitsky and Trevisan.

Theorem 4.1. *Samorodnitsky & Trevisan (2006) Assuming the UGC, for any* $\epsilon > 0$, *it is NP-hard to approximate Max-CSP(*P_{ST}*) within* $2^{d-k} + \epsilon$.

Equivalently, the theorem says that P_{ST}, assuming UGC, is approximation resistant. We are interested in the following extension.

Theorem 4.2. *Assuming UGC,* P_{ST} *is hereditary approximation resistant.*

It is satisfying to note that for $k = 3$ the predicate P_{ST} is simply parity and hence this instance of the theorem was proved by Håstad (2001) without using the UGC.

Proof. Let Q be any predicate of arity k implied by P_{ST}. Our proof is very similar to the proof of Samorodnitsky & Trevisan (2006) but we use a slightly different terminology. We assume that the reader is familiar with Probabilistically Checkable Proofs (PCPs) and their relation to inapproximability result for Max-CSPs. Details of the connection can be found in many places, two possible places being the papers by Håstad (2001) and Bellare *et al.* (1998), but let us give a very quick summary.

To prove that Max-CSP(Q) is approximation resistant we need, for any $\gamma > 0$, to design a PCP for an NP-hard problem where the acceptance condition is given by the predicate Q with the following properties. For positive instances there is a proof with acceptance probability at least $1 - \gamma$ while for negative instances the maximal acceptance probability is at most $d(Q) + \gamma$. It is also needed that the verifier uses $O(\log n)$ random bits when checking proofs of statements of size n. The latter property implies that the proof is of polynomial size.

As in Samorodnitsky & Trevisan (2006) we use a form of the UGC which, using the terminology of that paper, is called the *k-ary unique games.* We have variables $(v_i)_{i=1}^n$ taking values in a finite domain of size L, which we assume to be $[L]$. We are given a list of m constraints where the ℓ'th constraint, C_ℓ, is given by a k-tuple, $(v_{i_j^\ell})_{j=1}^k$ of variables and k permutations $(\pi_j^\ell)_{j=1}^k$. An assignment V *strongly satisfies* C_ℓ iff the k elements $\pi_j^\ell(V(v_{i_j}^\ell))$ are all equal and V *weakly satisfies* C_ℓ iff these values are not all distinct. The following result, originally derived in Khot & Regev (2003), is stated in the following form in Samorodnitsky & Trevisan (2006).

THEOREM 4.3. *If the UGC is true, then for every integer k and real number $\epsilon > 0$ there is an $L = L(k, \epsilon)$ such that, given a k-ary unique game problem with alphabet size L and m constraints, it is NP-hard to distinguish the case in which there is an assignment that strongly satisfies at least $(1-\epsilon)m$ constraints from the case where every assignment weakly satisfies at most ϵm constraints.*

We proceed to construct a PCP based on this k-ary unique game problem. The test is as described in Samorodnitsky & Trevisan (2006) but slightly reformulated.

The proof consists of n tables, A_i, each describing a function $\{-1,1\}^L \to \{-1,1\}$, which are accessed with the folding mechanism described in Section 2. To guide the intuition, let us note that, as we describe below in the proof of Lemma 4.4, an honest prover constructing a correct proof for a positive instance defines A_i to be the long code of the value for v_i.

We use noise vectors $\mu \in \{-1,1\}^L$ which has the property that μ_v is picked randomly and independently and for each $v \in [L]$ and equals 1 with probability $1-\delta$. The real number δ is a crucial parameter and need to be chosen as a function of γ.

We are now ready to define our central PCP.

Q-test(δ).

1. Pick a random k-ary constraint, C_ℓ given by variables $(v_{i_j^\ell})_{j=1}^k$, and permutations $(\pi_j^\ell)_{j=1}^k$.

2. Pick d independent random unbiased $x^i \in \{-1,1\}^L$ and k independent noise functions $\mu^j \in \{-1,1\}^L$.

3. Let $y^j = \prod_{i \in \hat{j}} x^i$ and $b_j = A_{i_j^\ell}(\pi_j^\ell(y^j) \cdot \mu^j)$.

4. Accept iff $Q(b) = Q(b_1, b_2, \dots b_k)$ is true.

We first address completeness.

Lemma 4.4. *For any $\gamma > 0$ there exists $\delta, \epsilon > 0$ such that if there is an assignment that strongly satisfies $(1-\epsilon)m$ constraints in the k-ary unique game problem then the verifier in Q-test(δ) can be made to accept with probability $1-\gamma$.*

Proof. Let V be an assignment that satisfies $(1-\epsilon)m$ constraints. Consider the proof where A_i is the long code of $V(v_i)$. Suppose the chosen constraint, C_ℓ, is satisfied by V and let v be the common value of $\pi_j^\ell(V(v_{i_j^\ell}))$. Then assuming that $\mu^j_{V(v_{i_j^\ell})} = 1$ for all j we have

$$b_j = y^j_{\pi_j^\ell(V(v_{i_j^\ell}))} \cdot \mu^j_{V(v_{i_j^\ell})} = y_v^j = \prod_{i \in \hat{j}} x_v^i.$$

It follows that b satisfies P_{ST} and hence also Q. As each $\mu_v = 1$ with probability $1-\delta$ the probability of acceptance is at least $(1-\delta)^k(1-\epsilon)$ which, for sufficiently small ϵ and δ, is at least $1-\gamma$. □

Let us turn to the more challenging task of analyzing the soundness.

Lemma 4.5. *For any $\gamma > 0$, $\delta > 0$ there exist $\epsilon = \epsilon(k, \delta, \gamma) > 0$ such that if the verifier in Q-test(δ) accepts with probability at least $d(Q) + \gamma$ there exists an assignment that weakly satisfies at least a ϵm constraints in the k-ary unique game problem.*

Proof. We assume that the verifier accepts with probability $d(Q) + \gamma$ and turn to define a (randomized) assignment that weakly satisfies a fraction of the constraints that only depends on k, δ and γ.

Consider the multilinear representation of Q

$$Q(b) = \sum_\beta \hat{Q}_\beta \prod_{j \in \beta} b_j.$$

The constant term $\hat{Q}_\emptyset$ is exactly $d(Q)$ and hence if the verifier accepts with probability $d(Q) + \gamma$ there must be some nonempty β such that

$$|E[\prod_{j \in \beta} b_j]| \geq 2^{-k}\gamma, \tag{4.6}$$

where the expectation is taken over a random C_ℓ and the random choices of x^i and μ^j.

Let us first study the expectation over the noise vectors. Towards this end let us define

$$B_j(y) = E_\mu[A_j(y \cdot \mu)],$$

which is useful as $E_{\mu^j}(b_j) = B_{i_j}(\pi_j(y^j))$. It is a standard fact (for a proof see Håstad (2001)) that

$$\hat{B}_{j,\beta} = (1 - 2\delta)^{|\beta|}\hat{A}_{j,\beta}$$

and hence

$$\sum_{|\beta| \geq t} \hat{B}^2_{j,\beta} \leq (1 - 2\delta)^{2t} \tag{4.7}$$

for any t. Now set $\Gamma = 2^{-2(d+k+2)}\gamma^2$ and let $t = \Theta(\delta^{-1} \log \Gamma^{-1})$ be such that

$$(1 - 2\delta)^{2t} \leq \Gamma/2,$$

and define

$$T_j = \{i \mid \text{inf}_i B_j \geq \Gamma\}.$$

As

$$\text{inf}_i B_j = \sum_{i \in \beta} \hat{B}^2_{j,\beta}, \tag{4.8}$$

by (4.7) and the definition of t, if $i \in T_j$ then we must have

$$\sum_{\beta | i \in \beta \wedge |\beta| < t} \hat{B}^2_{j,\beta} \geq \Gamma/2. \tag{4.9}$$

We conclude that

$$|T_j| \leq 2/\Gamma \sum_{\beta, i | i \in \beta \wedge |\beta| < t} \hat{B}^2_{j,\beta} \leq 2t/\Gamma.$$

Now, consider the probabilistic assignment that for each v_j chooses a random element of T_j. If T_j is empty we choose an arbitrary value for v_j. Let us analyze the expected number of constraints satisfied by this assignment.

By (4.6) we know that for at least $2^{-k}\gamma m/2$ of the constraints C_ℓ we have

$$\left| E_{x^i, \mu^j} \left[\prod_{j \in \beta} b_j \right] \right| \geq 2^{-k}\gamma/2. \tag{4.10}$$

Fix any such constraint and for $j \notin \beta$ or $k < j \leq 2^d - 1$ set $h_{\hat{j}}$ to be identically one while if $j \in \beta$ define $h_{\hat{j}}$ by

$$h_{\hat{j}}(y) = B_{i^\ell_j}(\pi^\ell_j(y)).$$

As $y^j = \prod_{i \in \hat{j}} x^i$ these definitions imply that

$$E_\mu \left[\prod_{j \in \beta} b_i \right] = \prod_{S \subseteq [d]} h_S \left(\prod_{i \in S} x^i \right) \tag{4.11}$$

and hence we are in a position to apply Theorem 3.4. Note first that, by folding, each h that is non-constant is in fact unbiased and hence, as β is non-empty, the minimum bias of the set of functions is 0.

We now claim that the maximal cross-influence of the function set h_S is at least Γ. Indeed suppose that this is not the case. Then, by Theorem 3.4, the expectation of (4.11), over the choice of vectors x^i, is at most

$$(2^d - 2)\sqrt{\Gamma} < 2^d 2^{-(d+k+2)}\gamma \leq 2^{-k}\gamma/2$$

contradicting (4.10).

As the cross-influence is at least Γ we have $j_1, j_2 \in \beta$ and an i such that $\mathrm{inf}_i h_{\hat{j}_k} \geq \Gamma$ for $k = 1, 2$. By definition,

$$\mathrm{inf}_i h_{\hat{j}_k} = \mathrm{inf}_{(\pi^\ell_{j_k})^{-1}(i)}(B_{i^\ell_{j_k}}).$$

We conclude that i belongs to $\pi^\ell_{j_k}(T_{i^\ell_{j_k}})$ for $k = 1, 2$ and thus our probabilistic assignment weakly satisfies C_ℓ with probability at least

$$\frac{1}{|T_{i^\ell_{j_1}}|} \cdot \frac{1}{|T_{i^\ell_{j_2}}|} \geq \frac{\Gamma^2}{4t^2}.$$

As this conclusion holds for at least $2^{-k}\gamma m/2$ constraints our probabilistic assignment weakly satisfies, on the average, at least

$$\frac{2^{-k}\gamma\Gamma^2 m}{8t^2}$$

constraints. Clearly there exists a standard, deterministic assignment that satisfies the same number of constraints. This finishes the proof of Lemma 4.5. □

As stated before Lemma 4.4 and Lemma 4.5 together with the fact that the acceptance criteria of Q-test(δ) is given by Q is sufficient to prove Theorem 4.2. Note that the randomness used by the verifier is bounded by $O(\log n)$ and most of the randomness is used to choose a random constraint as all other random choices only require $O(1)$ random bits. □

5. Random Predicates

Clearly two predicates P and Q that can be obtained from each other by permuting the inputs and/or negating some inputs are equivalent with respect to approximation resistance. Thus Theorem 4.2 does not only apply to P_{ST} but also to any predicate which is equivalent to it.

Consider the following space of random predicates.

Definition 5.1. *Let $Q_{p,k}$ be the probability space of predicates in k variables where each input is accepted with probability p.*

A uniformly random predicate corresponds to a predicate from $Q_{1/2,k}$ but we consider also smaller values of p. Whenever needed in calculations we assume $p \leq 1/2$.

We want to analyze the probability that a random predicate from $Q_{p,k}$ is implied by a negated and/or permuted variant of P_{ST} and let us give a ball park estimate for what values of p and k it is reasonable to believe that this is the case.

We have $k!$ permutations of the inputs and 2^k possible ways to negate the inputs. Thus the expected number of P_{ST}-equivalent predicates that imply a random predicate from $Q_{p,d}$ is

$$p^{2^d} 2^k k!.$$

For this number to be at least one we need, ignoring low order terms, that

$$p \geq k^{-k2^{-d}}.$$

This lower bound is between k^{-1} and $k^{-1/2}$ and in particular it is smaller than any constant. In fact this rough estimate is very close to what we can establish and thus we are, up to error estimates, able to get the strongest possible result using this approach. The proof works by an application of the second moment method. A problem to be overcome is that some pairs of P_{ST}-equivalent predicate have large intersection of their accepted sets. To address this problem we pick a large subset of the P_{ST}-equivalent predicates with bounded size intersections. We proceed to state our main theorem.

Theorem 5.2. *Assuming UGC and suppose $2^{d-1} \leq k \leq 2^d - 1$ then, there is a value c of the form $c = k2^{-d}(1 - o(1))$, such that, with probability $1 - o(1)$, a random predicate chosen according to $Q(p, k)$ with $p = k^{-c}$ is hereditary approximation resistant.*

PROOF. In view of Theorem 4.2 we need only prove that a random predicate from $Q_{p,k}$ with high probability is implied by some predicate which can be obtained from P_{ST} by negations and/or a permutation of the inputs.

Let us denote the set accepted by P_{ST} by L. It is a linear space of dimension d. Negating one or more inputs gives an affine space that is either L or completely disjoint from L. We get 2^{k-d} disjoint affine spaces denoted by $L+\alpha$ where α ranges over a suitable set of cardinality 2^{k-d}. We can also permute the coordinates and this gives a total of $k!2^{k-d}$ sets

$$\pi(L+\alpha)$$

A set of the form

$$\pi(L+\alpha)\cap\pi'(L+\beta)$$

is an affine space which is either empty or of dimension of the linear space

$$\pi(L)\cap\pi'(L).$$

We prove that for random choices of π and π' this dimension is likely to be small.

LEMMA 5.3. *Let d_0 and k_0 be sufficiently large constants and let r be a number such that $2^{d-r}\geq d_0$ and assume that $k\geq k_0$. Let L be the linear space accepted by P_{ST}. Then, if π and π' are two random permutations we have*

$$Pr[dim(\pi(L)\cap\pi'(L))\geq r]\leq 2^{(2-r)k}.$$

PROOF. We can clearly assume that π' is the identity. Let $(x_i)_{i=1}^r$ and $(y_i)_{i=1}^r$ and be any two sets, each of r independent points from L. We analyze the probability that $\pi(x^i)=y^i$ for $1\leq i\leq r$.

Fixing j, consider the r bits $(x_j^i)_{i=1}^r$ as an integer, t_j, in the range $0\leq t_j\leq 2^r-1$. Suppose the number m appears for k_m^x different values of j. Let the numbers k_m^y be defined similarly. Unless

$$k_m^x=k_m^y \tag{5.4}$$

for all m, the probability that $\pi(x^i)=y^i$ for all $1\leq i\leq r$ is 0. On the other hand if (5.4) holds then the probability of equality is exactly

$$\frac{\prod_{m=0}^{2^r-1}k_m!}{k!}. \tag{5.5}$$

We claim that $k_m^x \leq 2^{d-r}$ for each m. To see this, suppose we extend the x-vectors to length 2^d by adding the linearly dependent coordinates to get a point in the linear space accepted by P_{ST} on inputs of length $2^d - 1$ and also add a constant coordinate 1. We claim that in such a case any set of r linearly independent vectors x^i results in $k_m = 2^{d-r}$ for any m in the above notation.

To see this consider the r-bit strings that appear in positions 2^j of the x^i. By the definition of P_{ST} the values that appear as r-bit strings giving numbers m are all nonempty exclusive-ors of these strings. The added constant one coordinate corresponds to the empty exclusive-or. This implies that a random coordinate has an r-bit string that is a random point from a linear space and as the x^i are linearly independent it must be the full space which is observed with uniform probability.

Let us use $k_m^x \leq 2^{d-r}$ to estimate (5.5). First note that for any $n_1 \leq n_2$ we have $(n_1!)^{n_2} \leq (n_2)!^{n_1}$. This is true as each side is a product of $n_1 n_2$ integers and the it easy to find a matching where each factor in $(n_1!)^{n_2}$ is matched to factor of at least the same size in $(n_2)!^{n_1}$. We conclude that

$$k_m^x! \leq (2^{d-r}!)^{k_m^x 2^{r-d}}$$

and as $\sum_{m=0}^{2^r-1} k_m = k$ we have

$$\prod_{m=0}^{2^r-1} k_m^x! \leq (2^{d-r}!)^{k2^{r-d}}.$$

By Stirling's formula ($n! = \sqrt{2\pi n}(n/e)^n(1 + O(1/n))$) we can conclude, for sufficiently large value of $d - r$, that

$$2^{d-r}! \leq 3 \cdot 2^{(d-r)/2} 2^{(d-r)2^{d-r}} e^{-2^{d-r}}.$$

Using $k! \geq (k/e)^k$ we get the upper bound

$$(5.6) \qquad (3 \cdot 2^{(d-r)/2})^{k2^{r-d}} \left(\frac{2^{d-r}}{k}\right)^k$$

for (5.5) and thus the probability that $\pi(x^i) = \pi(y^i)$ for $1 \leq i \leq r$. The first factor of (5.6) is, as a function of $d - r$, of the form $(1 + o(1))^k$ and thus for sufficiently large values of $d-r$ it is bounded by $(3/2)^k$. As $k \geq 2^{d-1}$ the second factor of (5.6) is bounded by $2^{(1-r)k}$. Finally there are

$$\binom{2^d}{r}^2 \leq 2^{2dr}$$

ways to chose the points $(x^i)_{i=1}^r$ and $(y^i)_{i=1}^r$. As $r \leq d$ this number is bounded by $2^{O((\log k)^2)}$ and hence it is of the form of the form $(1+o(1))^k$. Thus a total estimate for the event of the lemma is

$$(3/2+o(1))^k 2^{(1-r)k} \leq 2^{(2-r)k}$$

and the proof is complete. □

From now on let us fix values of k and r such that Lemma 5.3 is true.

Let $R = 2^{k(r-2)}$ and we claim that we can choose R permutations $(\pi_i)_{i=1}^R$ such that

$$dim(\pi_i(L) \cap \pi_j(L)) \leq r$$

for any $i \neq j$. To see this first pick $2R$ permutations $(\sigma_j)_{j=1}^{2R}$ uniformly at random. The expected number of pairs (i,j), $i<j$, with

$$dim(\sigma_i(L) \cap \sigma_j(L)) \geq r+1 \tag{5.7}$$

is, by Lemma 5.3, bounded by

$$\binom{2R}{2} 2^{(1-r)k} \leq 2R^2 2^{(1-r)k} \leq R$$

and hence there is a choice of $2R$ permutations such that the number of pairs violating (5.7) is bounded by R. Erase one of the two permutations in each such pair (and possibly some additional permutations to make the remaining number of permutations exactly R), and we have the desired permutations $(\pi_i)_{i=1}^R$, which we fix once and for all.

Let $X_{i,\alpha}$ be the indicator variable for the event that a random predicate from $Q_{k,p}$ is identically one on the set

$$\pi_i(L+\alpha),$$

and define

$$X = \sum_{i,\alpha} X_{i,\alpha}.$$

If $X \neq 0$ there is some P_{ST}-equivalent predicate that implies the randomly chosen predicate and thus to prove Theorem 5.2 we need only estimate $Pr[X=0]$ and we proceed to do this using the second moment method. Clearly

$$E[X] = p^{2^d} 2^{k-d} R. \tag{5.8}$$

The variance of X equals

$$E\left[\sum_{i_1,i_2,\alpha_1,\alpha_2} (X_{i_1,\alpha_1} - p^{2^d})(X_{i_2,\alpha_2} - p^{2^d})\right]. \tag{5.9}$$

We have the following lemma.

LEMMA 5.10. *We have* $E[(X_{i_1,\alpha_1} - p^{2^d})(X_{i_2,\alpha_2} - p^{2^d})] = 0$ *if* $\pi_{i_1}(L + \alpha_1)$ *and* $\pi_{i_2}(L + \alpha_2)$ *are disjoint while if the size of the intersection is* K *it is bounded by*

$$p^{2^{d+1}-K}.$$

PROOF. In fact

$$E[(X_{i_1,\alpha_1} - p^{2^d})(X_{i_2,\alpha_2} - p^{2^d})] = E[X_{i_1,\alpha_1}X_{i_2,\alpha_2}] - p^{2^{d+1}} = p^{2^{d+1}-K} - p^{2^{d+1}}.$$

□

Let us estimate the sum (5.9) and first consider terms with $i_1 = i_2$. If $\alpha_1 = \alpha_2$ then

$$E\left[(X_{i_1,\alpha_1} - p^{2^d})(X_{i_2,\alpha_2} - p^{2^d})\right] \le p^{2^d}$$

while if $\alpha_1 \neq \alpha_2$ then the expectation is 0 and thus the total contribution to (5.9) of terms with $i_1 = i_2$ is bounded by $E[X]$.

Next consider terms with $i_1 \neq i_2$ and let us fix α_1 and analyze

$$\sum_{\alpha_2} E\left[(X_{i_1,\alpha_1} - p^{2^d})(X_{i_2,\alpha_2} - p^{2^d})\right]. \tag{5.11}$$

If

$$dim(\pi_{i_1}(L) \cap \pi_{i_2}(L)) = r'$$

then (5.11) has $2^{d-r'}$ terms with set-intersection size $2^{r'}$ while all other intersections are empty leading to the upper estimate

$$2^{d-r'}p^{2^{d+1}-2^{r'}} \le 2^{d-r}p^{2^{d+1}-2^r}$$

(using the assumption $p \le 1/2$) for the sum (5.11). Summing over all i_1, i_2 and α_1 we get

$$(5.12)\sigma^2(X) \le E[X] + R^2 2^{k-d}2^{d-r}p^{2^{d+1}-2^r} = E[X] + R^2 2^{k-r}p^{2^{d+1}-2^r}.$$

By Chebychev's inequality we have

$$\begin{aligned} Pr[X=0] &\le \frac{\sigma^2(X)}{E[X]^2} \le \frac{1}{E[X]} + \frac{R^2 2^{k-r}p^{2^{d+1}-2^r}}{R^2 2^{2(k-d)}p^{2^{d+1}}} && (5.13) \\ &\le \frac{1}{E[X]} + 2^{2d-(k+r)}p^{-2^r}. && (5.14) \end{aligned}$$

Set $p = k^{-c}$ for some $c \leq 1$. We need to prove that for some c of the form $2^{-d}k(1-o(1))$ and for a suitable choice of r the probability in (5.14) is $o(1)$. Note first that provided

$$2^r \log k < (k+r) - 2d - \omega(1)$$

the second term of (5.14) is small. This is possible to achieve with $r = d - \Theta(\log d)$. Note that this choice also ensures $d - r \in \omega(1)$ as required by Lemma 5.3.

Fixing this value of r, the first term of (5.14) is $o(1)$ provided that

$$p^{2^d} \geq 2^{(2-r)k}$$

which with, $p = k^{-c}$, is equivalent to

$$c \leq k2^{-d} \cdot \frac{r-2}{\log k}. \tag{5.15}$$

As the second factor of the bound in (5.15) is $(1-o(1))$ we have proved Theorem 5.2. □

As we have many more permutations π_i than translations α one might be given the (false) impression that the translations are not necessary. In fact this does not seem to be the case and let us argue why the translations are indeed needed.

To make sure the variance is small it has to be the case that most terms in the expression (5.9) are small or preferably 0 and this does not follow if we only use the permutations π_i.

On a more formal level, consider the second term of (5.14). The factor p^{-2^r} is large and comes from the fact that we do have intersection of sizes up to 2^r between our subspaces. The saving factor (which is roughly 2^{-k}) comes from the fact that only a fraction about 2^{d-k} of the pairs of sets in fact have any intersections at all.

One can wonder about reasonable values for p for small values of k. Particularly good values for k are numbers of the form $2^d - 1$ as this gives an unusually sparse predicate P_{ST}. Numerical simulations suggests that a random predicate on 7 bits that accepts M of the 128 inputs has a probability at least 1/2 of being implied by a P_{ST}-equivalent predicate iff $M \geq 60$. Thus it seems like the asymptotic bound of density essentially k^{-1} is approached slowly.

6. Very Dense Predicates

As P_{ST} only accepts 2^d inputs we can derive approximation resistance of many predicates but let us here give only one immediate application.

Theorem 6.1. *Let $2^{d-1} \leq k \leq 2^d - 1$ and P be any predicate that accepts at least $2^k + 1 - 2^{k-d}$ inputs, then, assuming the UGC, P is approximation resistant.*

Proof. We use the same notation as used in the proof of Theorem 5.2.

We need to prove that any such predicate is implied by a P_{ST}-equivalent predicate. This time we need only apply negations and consider $L + \alpha$ for all the 2^{k-d} different representatives α. As P only rejects $2^{k-d} - 1$ different inputs and the sets $L + \alpha$ are disjoint, one such set is included in the accepted inputs of P. The corresponding suitable negated form of P_{ST} hence implies P and Theorem 6.1 follows from Theorem 4.2. □

It is an interesting question how dense a non-trivially approximable predicate can be. Let d_k be the maximum value of $d(P)$ for all predicates on k variables which are not approximation resistant. We have $d_2 = d_3 = 3/4$ and Hast (2005) proved that $d_4 = \frac{13}{16}$ and, as we can always ignore any input, d_k is an increasing function of k. It is not obvious whether d_k tends to one as k tends to infinity.

Our results show that, assuming the UGC, dense predicates which can be non-trivially approximated need to be extremely structured as they cannot be implied by any P_{ST}-equivalent predicate.

7. Concluding Remarks

The key result in the current paper is to prove that, assuming the UGC, P_{ST} is hereditary approximation resistant. This is another result indicating that the more inputs accepted by the predicate P, the more likely it is to be approximation resistant. One could be tempted to conclude that all approximation resistant predicates are in fact hereditary approximation resistant. We would like to point that this is false and Hast (2005) has an example of two predicates P and Q where P is approximation resistant, P implies Q and Q is not approximation resistant.

That a predicate is approximation resistant is almost the ultimate hardness. There is a stronger notion; approximation resistance on satisfiable instances. In such a case no efficient algorithm is able to do significantly better than picking a random assignment even in the case when the instance is satisfiable.

An example of a predicate which is approximation resistant but not approximation resistant on satisfiable instances is Max-E3-Lin-2, linear equations modulo 2 with three variables in each equation. In this case, for a satisfiable instance, it is easy to find an assignment that satisfies all constraints by Gaussian elimination.

In most cases, however, approximation resistant predicates have turned out to be approximation resistant also on satisfiable instances and it would seem reasonable to conjecture that a random predicate is indeed approximation resistant on satisfiable instances. If true it seems hard to prove this fact using the Unique Games Conjecture in that the non-perfect completeness of UGC would tend to produce instances of the CSP which are not satisfiable. There are variants of the unique games conjecture of Khot (2002) which postulate hardness of label cover problems with perfect completeness but it would be much nicer to take a different route not relying on any conjectures.

Another open problem is of course to establish approximation resistance in absolute terms and not to rely on the UGC or, more ambitiously, to prove the UGC.

Acknowledgements

I am grateful to Per Austrin for useful comments on the current manuscript.

References

M. Bellare, O. Goldreich & M. Sudan (1998). Free Bits, PCPs and Non-Approximability—Towards tight Results. *SIAM Journal on Computing* **27**, 804–915.

M. Goemans & D. Williamson (1995). Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM* **42**, 1115–1145.

T. Gowers (1998). A new proof of Szemerédi's theorem for progressions of length four. *Geometric and Functional Analysis* **8**, 529–551.

T. Gowers (2001). A new proof of Szemerédi's theorem. *Geometric and Functional Analysis* **11**, 465–588.

V. Guruswami, D. Lewin, M. Sudan & L. Trevisan (1998). A tight characterization of NP with 3 query PCPs. In *Proceedings of 39th Annual IEEE Symposium on Foundations of Computer Science*, 8–17. IEEE, Palo Alto.

G. Hast (2005). *Beating a random assignment.* KTH, Stockholm. Ph.D Thesis.

J. Håstad (2001). Some optimal inapproximability results. *Journal of ACM* **48**, 798–859.

J. Håstad (2005). Every 2-CSP Allows Nontrivial Approximation. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computation*, 740–746. Final version accepted for publication in Computational Complexity.

J. Håstad (2007). On the Approximation Resistance of a Random Predicate. In *Proceedings of RANDOM 2007 and APPROX 2007, LNCS 4627*, 149–163.

S. Khot (2002). On the power of unique 2-Prover 1-Round games. In *Proceedings of 34th ACM Symposium on Theory of Computating*, 767–775.

S. Khot & O. Regev (2003). Vertex Cover Might be Hard to Approximate to within $2-\varepsilon$. In *Proc. of 18th IEEE Annual Conference on Computational Complexity (CCC)*, 379–386.

A. Samorodnitsky & L. Trevisan (2000). A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, 191–199.

A. Samorodnitsky & L. Trevisan (2006). Gowers Uniformity, Influence of Variables and PCPs. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 11–20.

T. Schaefer (1978). The complexity of satisfiability problems. In *Conference record of the Tenth annual ACM Symposium on Theory of Computing*, 216–226.

U. Zwick (1998). Approximation algorithms for constraint satisfaction problems involving at most three variables per constraint. In *Proceedings 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, 201–210. ACM.

U. Zwick (2006). Personal Communication.

Manuscript received October 11, 2007

Johan Håstad
Royal Institute of Technology
Stockholm Sweden
johanh@kth.se
http://www.csc.kth.se/~johanh/