On Quadratic Threshold CSPs

Per Austrin[1], Siavosh Benabbas[2], and Avner Magen[2]

[1] Courant Institute of Mathematical Sciences
New York University
austrin@cims.nyu.edu

[2] Department of Computer Science
University of Toronto
{siavosh,avner}@cs.toronto.edu

Abstract. A predicate $P : \{-1, 1\}^k \to \{0, 1\}$ can be associated with a constraint satisfaction problem Max CSP(P). P is called "approximation resistant" if Max CSP(P) cannot be approximated better than the approximation obtained by choosing a random assignment, and "approximable" otherwise. This classification of predicates has proved to be an important and challenging open problem. Motivated by a recent result of Austrin and Mossel (Computational Complexity, 2009), we consider a natural subclass of predicates defined by signs of quadratic polynomials, including the special case of predicates defined by signs of linear forms, and supply algorithms to approximate them as follows.

In the quadratic case we prove that every *symmetric* predicate is approximable. We introduce a new rounding algorithm for the standard semidefinite programming relaxation of Max CSP(P) for any predicate $P : \{-1, 1\}^k \to \{0, 1\}$ and analyze its approximation ratio. Our rounding scheme operates by first manipulating the optimal SDP solution so that all the vectors are nearly perpendicular and then applying a form of hyperplane rounding to obtain an integral solution. The advantage of this method is that we are able to analyze the behaviour of a set of k rounded variables together as opposed to just a pair of rounded variables in most previous methods.

In the linear case we prove that a predicate called "Monarchy" is approximable. This predicate is not amenable to our algorithm for the quadratic case, nor to other LP/SDP-based approaches we are aware of.

1 Introduction

This paper studies the approximability of *constraint satisfaction problems* (CSPs). Given a predicate $P : \{-1, 1\}^k \to \{0, 1\}$, the Max CSP($P$) problem is defined as follows. An instance is given by a list of k-tuples (clauses) of literals over some set of variables $x_1, \ldots, x_n$, where a literal is either a variable or its negation. A clause is satisfied by an assignment to the variables if P is satisfied when applied to the k literals of the clause. The goal is then to find an assignment to the variables that maximizes the number of satisfied clauses. Our specific interest is predicates of the form $P(x) = \frac{1+\operatorname{sign}(Q(x))}{2}$ where $Q : \mathbb{R}^k \to \mathbb{R}$ is a quadratic

A. López-Ortiz (Ed.): LATIN 2010, LNCS 6034, pp. 332–343, 2010.

polynomial with no constant term, i.e., $Q(x) = \sum_{i=1}^{k} a_i x_i + \sum_{i \neq j} b_{ij} x_i x_j$ for some set of coefficients $a_1, \ldots, a_n$ and $b_{11}, \ldots, b_{nn}$. While this special case is arguably very rich and interesting in its own right, we give some further motivations below. But first, we give some background to the study of Max CSP(P) problems in general.

A canonical example of a Max CSP(P) problem is when $P(x_1, x_2, x_3) = x_1 \vee x_2 \vee x_3$ is a disjunction of three variables, in which case Max CSP(P) is the classic Max 3-Sat problem. Another well-known example is the Max 2-Lin(2) problem in which $P(x_1, x_2) = x_1 \oplus x_2$. As Max CSP($P$) is NP-hard for almost all choices of P (the only case for which it is not NP-hard is when P depends on at most 1 variable), much effort has been put into understanding the best possible approximation ratio achievable in polynomial time. A (randomized) algorithm is said to have approximation ratio $\alpha \leq 1$ if, given an instance with optimal value Opt, it produces an assignment with (expected) value at least $\alpha \cdot \text{Opt}$.

The arguably simplest approximation algorithm is to pick a uniformly random assignment. As this algorithm satisfies each constraint with probability $\frac{|P^{-1}(1)|}{2^k}$ it follows that it gives an approximation ratio of $\frac{|P^{-1}(1)|}{2^k}$. In their classic paper [5], Goemans and Williamson used semidefinite programming to obtain improved approximation algorithms for predicates on two variables. For instance, for Max 2-Lin(2) they gave an algorithm with approximation ratio $\alpha_{GW} \approx 0.878$. Following [5], many new approximation algorithms were found for various specific predicates, improving upon the random assignment algorithm. However, for some cases, perhaps most prominently the Max 3-Sat problem, no such progress was made. Then, in another classic paper [7], Håstad proved that Max 3-Sat is in fact NP-hard to approximate within $7/8+\epsilon$, showing that a random assignment in fact gives the best possible worst-case approximation that can be obtained in polynomial time.

Predicates which exhibit this behavior are called *approximation resistant.* One of the main open questions along this line of research is to characterize which predicates admit a non-trivial approximation algorithm, and which predicates are approximation resistant. For predicates on three variables, the work of Håstad together with work of Zwick [13] shows that a predicate is resistant iff it is implied by an XOR of the three variables, or the negation thereof, where a predicate P is said to imply a predicate P' if $P(x) = 1 \Rightarrow P'(x) = 1$. For four variables, Hast [6] made an extensive classification leaving open the status of 46 different predicates.

There have been several papers [11,3,12], mainly motivated by the soundness-query trade off for PCPs, giving increasingly general conditions under which predicates are approximation resistant. In a recent paper [2], the first author and Mossel proved that, if there exists an unbiased pairwise independent distribution on $\{-1, 1\}^k$ whose support is contained in $P^{-1}(1)$, then P is approximation resistant under the Unique Games Conjecture [10]. This condition is very general and turned out to give many new cases of resistant predicates [1]. A related result by Georgiou et al [4] that is independent of complexity assumptions, shows that under the same condition on P, the so-called Sherali-Admas hierarchy—which is in some sense the strongest version of the Linear Programming approach—does not beat a random assignment. Indeed, when it comes to algorithms, there are very few systematic results that give algorithms for large classes of predicates. One such result can be found in [6]. Given the result of [2], such systematic results can only work for predicates that do not support pairwise independence. A very natural subclass of these predicates are those of the form $\frac{1+\text{sign}(Q)}{2}$ for Q a quadratic polynomial as described above. To be more precise, the following fact from [1] is our main motivation for studying this type of predicates.

Fact 1. *A predicate P does not support pairwise independence if and only if there exists a quadratic polynomial $Q : \{-1,1\}^k \to \mathbb{R}$ with no constant term that is positive on all of $P^{-1}(1)$ (in other words, P implies a predicate of such form).*

Given that the main tool for approximation algorithms—semidefinite programming—works by optimizing quadratic forms, it seemed natural and intuitive to hope that predicates of this form are always approximable. This however turns out to be false—Håstad [8] constructs a predicate that is the sign of a quadratic polynomial and still approximation resistant. Loosely speaking, the main crux is that semidefinite programming is good for optimizing the degree-2 part of the Fourier expansion of a predicate, which unfortunately can behave very differently from P itself or the quadratic polynomial used to define P (we elaborate on this below.) However, it turns out that when we restrict our attention to the special case of symmetric predicates, this can not happen, and we can obtain an approximation algorithm, which is our first result.

Theorem 1. *Let $P : \{-1,1\}^k \to \{0,1\}$ be a predicate that is of the form $P(x) = \frac{1+\text{sign}(Q(x))}{2}$ where Q is a symmetric quadratic polynomial with no constant term. Then P is not approximation resistant.*

A very natural special case of the signs of quadratic polynomials is the case when $P(x) = \frac{1+\text{sign}(\sum a_i x_i)}{2}$ is simply the sign of a linear form, i.e., a linear threshold function. While we cannot prove that linear threshold predicates are approximable in general, we do believe this is the case, and make the following conjecture.

Conjecture 1. Let $P : \{-1,1\}^k \to \{0,1\}$ be a predicate that is a sign of a linear form with no constant term. Then P is not approximation resistant.

We view the resolution of this conjecture as a very natural and interesting open problem. As in the quadratic case, the difficulty stems from the fact that the low-degree part of P can be unrelated to the linear form used to define P. Specifically, it can be the case that the low-degree part of the arithmetization of P vanishes or becomes negative for some inputs where the linear/quadratic polynomial is positive (i.e. accepting inputs), and unfortunately this seems to make the standard SDP approach fail. The perhaps most extreme case of this phenomenon is exhibited by the predicate Monarchy : $\{-1,1\}^k \to \{0,1\}$ suggested by Håstad [8], in which the first variable (the "monarch") decides the outcome, unless all the other variables unite against it. In other words,

$$\text{Monarchy}(x) = \frac{1 + \text{sign}((k-2)x_1 + \sum_{i=2}^{k} x_i)}{2}.$$

Now, for the input $x_1 = -1$, $x_2 = \ldots = x_k = 1$, the linear part of the Fourier expansion of Monarchy takes value $-1 + o_k(1)$, whereas the linear form used to define monarchy is positive on this input, hence the value of the predicate is 1. Again, we stress that this means that known algorithms and techniques do not apply. However, in this case we are still able to achieve an approximation algorithm, which is our second result.

Theorem 2. *The predicate* Monarchy *is* not *approximation resistant.*

This shows that there is some hope in overcoming the apparent barriers to proving Conjecture 1.

Techniques: Our starting point in both our algorithms is the standard SDP relaxation of Max CSP(P). The main difficulty in rounding the solutions of these SDPs is that current rounding algorithms offer no analysis of the joint distribution of the outcome of the rounding for k variables, when $k > 2$. (Interestingly, when some modest value $k = 3$ is used, often some numerical methods are employed to complete the analysis [9,13].) Unfortunately, such analysis seems essential to understanding the performance of the algorithm for Max CSP(P) as each constraint depends on k variables. Indeed, even a local argument would have to argue about the outcome of the rounding algorithm for k variables together.

For Theorem 1, we give a new, simpler proof of a theorem by Hast [6], giving a general condition on the low-degree part of the Fourier Expansion which guarantees a predicate is approximable (Theorem 4). We then show that this condition holds for predicates which are defined by symmetric quadratic polynomials. The basic idea behind our new algorithm is as follows. First, observe that the SDP solution in which all vectors are perpendicular is easy to analyze when the usual hyperplane rounding is employed, as in this case the obtained integral values are distributed uniformly. This motivates the following approach: start with the perpendicular configuration and then perturb the vectors in the direction of the optimal SDP solution. This perturbation acts as a differentiation operator, and as such allows for a "linear snapshot" of what is typically a complicated system. For each clause we analyze the probability that hyperplane rounding outputs a satisfying assignment, as a function of the inner products of vectors involved. Now, the object of interest is the gradient of this function at "zero". The hope is that since the optimal SDP solution (almost) satisfies this clause, it has a positive inner product with the gradient, and so can act as a global recipe that works for all clauses. It is important to stress that since we are only concerned with getting an approximation algorithm that works slightly better than random we can get away with this linear simplification. We show that this condition on the gradient translates into a condition on the low-degree part of the Fourier expansion of the predicate.

As it turns out, the predicate Monarchy which we tackle in Theorem 2 does not exhibit the aforementioned desirable property. In other words, the gradient above does not generally have a positive inner product with an optimal SDP solution. Instead, we show that when all vectors are sufficiently far from $\pm\mathbf{v}_0$ it is possible to get a similar guarantee on the gradient using high (but not-too-high) moments of the vectors. We can then handle vectors which are very close to $\pm\mathbf{v}_0$ separately by rounding them deterministically to ±1.

Organization: The rest of the paper is organized as follows. First, we introduce some definitions and preliminaries including the standard SDP relaxation of Max CSP(P) in Section 2. Then, in Section 3 we give our new algorithm for this SDP relaxation and characterize the predicates for which it gives an approximation ratio better than a random assignment. We then take a closer look at signs of symmetric quadratic forms in Section 4 and show that these satisfy the condition of the previous section, proving Theorem 1. In Section 5 we give the approximation algorithm for the Monarchy predicate and the ideas behind its somewhat tedious analysis. Finally, we give a discussion and some directions for future work in Section 6.

2 Preliminaries

In what follows $\mathbb{E}$ stands for expectation. For any positive integer n we use the notation $[n]$ for the set $\{1,\ldots,n\}$. For a finite set S (often a subset of $[n]$) we use the notation $\{-1,1\}^S$ for the set of all $-1,1$ vectors indexed by elements of S. For example, $|\{-1,1\}^S| = 2^{|S|}$.

We use φ and Φ for the probability density function and the cumulative distribution function of a standard normal random variable, respectively. We use the notation $\mathbb{S}^{n-1}$ for the $n-1$ dimensional sphere,i.e. , the set of unit vectors in $\mathbb{R}^n$.

Throughout the paper, we take the convention that $\text{sign}(0) = -1$.

2.1 Fourier Representation

Consider the set of real functions with domain $\{-1,1\}^k$ as a vector space. It is well known that the following set of functions called the characters form a complete basis for this space, $\chi_S(x) \stackrel{\text{def}}{=} \prod_{i\in S} x_i$. In fact if we define inner products of functions as $f \cdot g \stackrel{\text{def}}{=} \mathbb{E}_x\left[f(x)g(x)\right]$ this basis will be orthonormal and every function will have a unique *Fourier expansion* when written in this basis,

$$f = \sum_{S\subseteq[k]} \widehat{f}(S)\chi_S, \qquad\qquad \widehat{f}(S) \stackrel{\text{def}}{=} f \cdot \chi_S.$$

$\widehat{f}(S)$'s are often called the *Fourier coefficients* of f. We write $f^{=d}$ for the part of the function that is of degree d, i.e.,

$$f^{=d}(x) = \sum_{|S|=d} \widehat{f}(\{S\})\chi_S(x).$$

2.2 Semidefinite Relaxation

For any fixed P, Max CSP(P) has a natural SDP relaxation that can be seen in Figure 1. The essence of this relaxation is that each $I_{S,*}$ is a distribution, often called a *local distribution*, over all possible assignments to the variables in set S as enforced by (1). Whenever, S_1 and S_2 intersect (2) guarantees that their marginal distributions on the intersection agree. Also, (3) and (4) ensure that $\mathbf{v}_0 \cdot \mathbf{v}_i$ and $\mathbf{v}_i \cdot \mathbf{v}_j$ are equal to the bias of variable x_i and the correlation of the variables x_i and x_j in the local distributions respectively. The clauses of the instance are $C_1, \ldots, C_m$, with C_i being an application of P (possibly with some variables negated) on the set of variables T_i. The objective function is the fraction of the clauses that are satisfied.

Observe that the reason this SDP is not an exact formulation but a relaxation is that these distributions are defined only on sets of size up to k. It is worth mentioning that this program is weaker than the kth round of the Lasserre hierarchy for this problem while stronger than the kth round of the Sheralli-Adams hierarchy. From here on the only things we use in the rounding algorithms are the vectors $\mathbf{v}_0, \ldots, \mathbf{v}_n$ and the existence of the local distributions.

$$
\begin{array}{rcll}
\text{Maximize} & & \displaystyle\frac{1}{m}\sum_{i=1}^{m}\sum_{\omega\in\{-1,1\}^{T_i}} C_i(\omega)\, I_{T_i,\omega} & \\
\text{Where,} & \forall S \subset [n], |S| \le k, \omega \in \{-1,1\}^S & I_{S,\omega} \in [0,1] & \\
& \forall i \in [n] & \mathbf{v}_i \in \mathbb{R}^{n+1}, \|\mathbf{v}_i\|_2^2 = 1 & \\
& & \mathbf{v}_0 \in \mathbb{R}^{n+1}, \|\mathbf{v}_0\|_2^2 = 1 & \\
\text{Subject to} & \forall S \subset [n], |S| \le k & \displaystyle\sum_{\omega\in\{-1,1\}^S} I_{S,\omega} = 1 & (1) \\
& \forall S \subset S' \subset [n], |S'| \le k, \omega \in \{-1,1\}^S & \displaystyle\sum_{\omega'\in\{-1,1\}^{S'}} I_{S',\omega'} = I_{S,\omega} & (2) \\
& & \omega' \text{ is an extension of } \omega & \\
& \forall i \in [n] & I_{\{i\},(1)} - I_{\{i\},(-1)} = \mathbf{v}_0 \cdot \mathbf{v}_i & (3) \\
& \forall i, j \in [n] & I_{\{i,j\},(1,1)} - I_{\{i,j\},(-1,1)} & \\
& & -I_{\{i,j\},(1,-1)} + I_{\{i,j\},(-1,-1)} = \mathbf{v}_i \cdot \mathbf{v}_j & (4)
\end{array}
$$

Fig. 1. Standard SDP relaxation of Max CSP(P)

3 (ϵ, η)-Hyperplane Rounding

In this section we define a rounding scheme for the semidefinite program of Max CSP(P) and proceed to analyze its performance. The rounding scheme is based on the usual hyperplane rounding but is more flexible in that it uses two parameters ϵ and η where ϵ is a sufficiently small constant and η is an arbitrary real number. We will then formalize a (sufficient) condition involving P and η under which our approximation algorithm has approximation factor better than that of a random assignment. In the next section we show that this condition is satisfied (for some η) by signs of symmetric quadratic polynomials.

Given an instance of Max CSP(P), our algorithm first solves the standard SDP relaxation of the problem (Figure 1.) Then, it employs the rounding scheme outlined in Figure 2 to get an integral solution.

```
   INPUT: $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{S}^n$.
   OUTPUT: $x_1, \ldots, x_n \in \{-1, 1\}$.
1. Define unit vectors $\mathbf{w}_0, \mathbf{w}_1, \ldots, \mathbf{w}_n \in \mathbb{S}^n$ such that for all $0 \leq i \neq j$,
       $\mathbf{w}_i \cdot \mathbf{w}_j = \epsilon(\mathbf{v}_i \cdot \mathbf{v}_j)$,
2. Let $g \in \mathbb{R}^{n+1}$ be a random $(n+1)$-dimensional Gaussian.
3. Assign each $x_i$ as,
       $x_i = \begin{cases} 1 & \text{if } \mathbf{w}_i \cdot g > -\eta(\mathbf{w}_0 \cdot \mathbf{w}_i), \\ -1 & \text{otherwise.} \end{cases}$
```

Fig. 2. (ϵ, η)-Hyperplane Rounding

Note that when $\epsilon = 0$ the rounding scheme above simplifies to assigning all x_i's uniformly and independently at random which satisfies $\frac{|P^{-1}(1)|}{2^k}$ fraction of all clauses in expectation. For non-zero ϵ, η will determine how much weight is given to the position of $\mathbf{v}_0$ compared to the correlation of the variables.

Notice that in the pursuit of a rounding algorithm that has approximation ratio better than $\frac{|P^{-1}(1)|}{2^k}$ it is possible to assume that the optimal integral solution is arbitrary close to 1 (in terms of k) as otherwise random assignment already delivers an approximation factor better than $\frac{|P^{-1}(1)|}{2^k}$. In particular, the optimal vector solution can be assumed to be that good. This observation is in fact essential to our analysis. But, for the sake of simplicity first consider the case where the value of the vector solution is precisely 1. Fix a clause, say $P(x_1, x_2, \ldots, x_k)$. (In general, without loss of generality we can assume that the current clause is on *k variables* as opposed to *k literals*. This is simply because one can assume that $\neg x_i$ is a separate variable from x_i with SDP vector $-\mathbf{v}_i$.) Since the SDP value is 1, every clause (and this clause in particular) is completely satisfied by the SDP, hence the local distribution $I_{[k],*}$ is supported on the set of satisfying assignments of P. The hope now is that when ϵ increases from zero to some small positive value this distribution helps to boost the probability of satisfying the clause (a little) beyond $\frac{|P^{-1}(1)|}{2^k}$. This becomes a question of differentiation. Specifically, consider the probability of satisfying the clause at hand as a function of ϵ. We want to show that for some $\epsilon > 0$ the value of this function is bigger than its value at zero which is closely related to the derivative of the function at zero. We can show,

Theorem 3. *For any fixed η, the probability that $P(x_1, \ldots, x_k)$ is satisfied by the assignment behaves as follows at $\epsilon = 0$:*

$$\Pr\left[(x_1, \ldots, x_k) \in P^{-1}(1)\right] = \frac{|P^{-1}(1)|}{2^k}$$

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}\Pr\left[(x_1, \ldots, x_k) \in P^{-1}(1)\right] = \frac{2\eta}{\sqrt{2\pi}}\sum_{i=1}^{k}\widehat{P}(\{i\})\mathbf{v}_0 \cdot \mathbf{v}_i + \frac{2}{\pi}\sum_{i<j}\widehat{P}(\{i,j\})\mathbf{v}_i \cdot \mathbf{v}_j. \tag{5}$$

Now, the inner products $\mathbf{v}_i \cdot \mathbf{v}_j$ are equal to the moments of the local distributions $I_{\{i,j\},*}$, which in turn agree with those of the local distribution $I_{[k],*}$. It follows that,

$$\frac{2\eta}{\sqrt{2\pi}}\sum_{i=1}^{k}\widehat{P}(\{i\})\mathbf{v}_0 \cdot \mathbf{v}_i + \frac{2}{\pi}\sum_{i<j}\widehat{P}(\{i,j\})\mathbf{v}_i \cdot \mathbf{v}_j = \mathop{\mathbb{E}}_{\omega \sim I_{[k],*}}\left[\frac{2\eta}{\sqrt{2\pi}}P^{=1}(\omega) + \frac{2}{\pi}P^{=2}(\omega)\right]. \tag{6}$$

Thus, in order for the derivative in (5) to be positive for all possible values of the $\mathbf{v}_i$'s, it is necessary and sufficient that $\frac{2\eta}{\sqrt{2\pi}}P^{=1}(\omega) + \frac{2}{\pi}P^{=2}(\omega)$ is positive for every $\omega \in P^{-1}(1)$. We can then show the following Theorem.

Theorem 4. *Suppose that there exists an $\eta \in \mathbb{R}$ such that*

$$\frac{2\eta}{\sqrt{2\pi}}P^{=1}(\omega) + \frac{2}{\pi}P^{=2}(\omega) > 0 \tag{7}$$

for every $\omega \in P^{-1}(1)$. Then P is approximable.

As mentioned in the Techniques section, this theorem is not new. It was previously found by Hast [6]. However, his algorithm and analysis is completely different from ours (using different algorithms to optimize the linear and quadratic parts of the predicate, and case analysis depending on the behaviour of the integral solution). Our algorithm is considerably more direct, and its analysis is simpler.

The general strategy for the proof, which is deferred to the full version is as follows. We will concentrate on a clause that is almost satisfied by the SDP solution. By the condition and Theorem 3 the first derivative of the probability that this clause is satisfied by the rounded solution is at least some positive global constant (say δ) at $\epsilon = 0$. We will then show that provided that ϵ is small enough the second derivative of this probability is bounded in absolute value by, say, Γ at any point in $[0, \epsilon]$. Now we can apply Taylor's theorem to show that if ϵ is small enough the probability of success is at least $\frac{|P^{-1}(1)|}{2^k} + \delta\epsilon - \Gamma\epsilon^2/2$ which for $\epsilon = \delta/\Gamma$ is at least $\frac{|P^{-1}(1)|}{2^k} + \delta^2/2\Gamma$.

4 Signs of Symmetric Quadratic Polynomials

In this section we study signs of symmetric quadratic polynomials, and give a proof of Theorem 1. Consider a predicate $P : \{-1, 1\}^k \to \{0, 1\}$ that is the sign of a symmetric quadratic polynomial with no constant term, i.e.,

$$P(x) = \frac{1 + \operatorname{sign}\left(\alpha \sum x_i + \beta \sum x_i x_j\right)}{2}$$

for some constants α and β. We would like to apply the (ϵ, η)-rounding scheme to Max CSP(P), which in turn requires us to understand the low-degree Fourier coefficients of P. Note that because of symmetry, the value of a Fourier coefficient $\widehat{P}(S)$ depends only on $|S|$.

We will prove that "morally" the degree-2 Fourier coefficient of P and β have the same sign and that if one of them is 0 then so is the other. This statement is not quite true (consider for instance the predicate $P(x_1, x_2) = \frac{1+\operatorname{sign}(x_1+x_2)}{2} = \frac{1+x_1+x_2+x_1x_2}{4}$), however it is always true that by slightly adjusting β (without changing P), we can assure that this is the case.

Theorem 5. *For any P of the above form, there exists β' with the property that $\beta' \cdot \widehat{P}(\{1,2\}) \geq 0$ and $\beta' = 0$ iff $\widehat{P}(\{1,2\}) = 0$, satisfying*

$$P(x) = \frac{1 + \operatorname{sign}\left(\alpha \sum x_i + \beta' \sum x_i x_j\right)}{2}.$$

Due to space considerations the proof of this theorem is deferred to the full version. We are now ready to prove Theorem 1.

Theorem 1 (restated). *Let $P : \{-1,1\}^k \to \{0,1\}$ be a predicate that is of the form $P(x) = \frac{1+\operatorname{sign}(Q(x))}{2}$ where Q is a symmetric quadratic polynomial with no constant term. Then P is not approximation resistant.*

Proof. Without loss of generality, we can take $Q(x) = \alpha \sum x_i + \beta \sum x_i x_j$ where β satisfies the property of β' in Theorem 5.

If $\widehat{P}(\{1,2\}) = \beta = 0$, we set $\eta = \alpha / \widehat{P}(\{1\})$ (note that in this case we can assume that α, hence $\widehat{P}(\{1\})$ is non-zero as otherwise P is the trivial predicate that is always false). We then have, for every $x \in P^{-1}(1)$,

$$\frac{2\eta}{\sqrt{2\pi}} P^{=1}(x) + \frac{2}{\pi} P^{=2}(x) = \frac{2\alpha}{\sqrt{2\pi}} \sum x_i,$$

which is positive by the definition of P. If $\widehat{P}(\{1,2\}) \neq 0$, we set $\eta = \sqrt{\frac{2}{\pi}} \frac{\alpha}{\widehat{P}(\{1\})} \cdot \frac{\widehat{P}(\{1,2\})}{\beta}$. In this case for every $x \in P^{-1}(1)$,

$$\frac{2\eta}{\sqrt{2\pi}} P^{=1}(x) + \frac{2}{\pi} P^{=2}(x) = \frac{2\widehat{P}(\{1,2\})}{\pi\beta} \left(\alpha \sum x_i + \beta \sum x_i x_j\right) > 0,$$

since β agrees with $\widehat{P}(\{1,2\})$ in sign and $Q(x) > 0$. In either cases, using Theorem 4 and the respective choices of η we conclude that P is approximable.

5 Monarchy

In this section we prove that for $k > 4$ the Monarchy predicate is not approximation resistant. Notice that Monarchy is defined only for $k > 2$, and that the case $k = 3$ coincides with the predicate majority that is known not to be approximation resistant. Further, the case $k = 4$ is handled by [6].[1]

Just like the algorithm for symmetric predicates we first solve the natural semidefinite program of Monarchy, and then use a rounding algorithm to construct an integral solution out of the vectors. The rounding algorithm, which is given in Figure 3, has two parameters $\epsilon > 0$ and an odd positive integer ℓ, both depending on k. These will be fixed in the proof.

```
   INPUT: "biases" $b_1 = \mathbf{v}_0 \cdot \mathbf{v}_1, \ldots, b_n = \mathbf{v}_0 \cdot \mathbf{v}_n$.
   OUTPUT: $x_1, \ldots, x_n \in \{-1, 1\}$.
1. Choose a parameter $\tau \in [1/2(k-2)^2, 1/(k-2)^2]$ uniformly at random.
2. For all $i$,
   (a) If $b_i > 1 - \tau$ or $b_i < -1 + \tau$, set $x_i$ to 1 or $-1$ respectively.
   (b) Otherwise, set $x_i$ (independent of all other $x_j$'s), randomly to $-1$ or
       1 such that $\mathbb{E}[x_i] = \epsilon b_i^\ell$. In particular, set $x_i = 1$ with probability
       $(1 + \epsilon b_i^\ell)/2$ and $x_i = -1$ with probability $(1 - \epsilon b_i^\ell)/2$.
```

Fig. 3. Rounding SDP solutions for Monarchy

Remark 1. As the reader may have observed, the "geometric" power of SDP is not used in the above rounding scheme, and indeed a linear programming relaxation of the problem would suffice for the algorithm we propose. However, in the interest of consistency and being able to describe the the techniques in a language comparable to Theorem 1 we elected to use the SDP framework.

Here is the outline of the analysis. We first ignore the greedy ingredient (2a above). Notice that for $\epsilon = 0$ the rounding gives a uniform assignment to the variables, hence the expected value of the obtained solution is 1/2. As long as $\epsilon > 0$ is small enough, the probability of success for a clause is essentially only affected by the degree-one Fourier coefficients of Monarchy. Now, fix a clause and assume that the SDP solution completely satisfies it. Specifically, consider the clause Monarchy$(x_1, \ldots, x_k)$, and define $b_1, \ldots, b_k$ as the corresponding biases. Notice that all Fourier coefficients of Monarchy are positive. This implies that the rounding scheme above will succeed with probability that is essentially 1/2 plus some *positive linear combination* of the ϵb_i^ℓ. Our objective is then to fix ℓ that would make the value of this combination positive (and independent from n). It turns out that the maximal b_i in magnitude (call it b_j) is always positive in this case. Oversimplifying, imagine that $|b_j| \geq |b_i| + \xi$ for all i different than j where ξ is some positive constant. Clearly in this setting it is easy to take ℓ (a function of k) that makes the effect of all b_i other than b_j vanish, ensuring a positive addition to the probability as desired so that overall the expected fraction of satisfied clauses is more than 1/2.

[1] In the notation of [6], Monarchy on 4 variables is the predicate `0000000101111111`, which is listed as approximable in Table 6.6. We remark that this is not surprising since Monarchy in this case is simply a majority in which x_1 serves as a tie-breaker variable.

More realistically, the above slack ξ does not generally exist. However, we can show that a similar condition holds provided that the $|b_i|$ are bounded away from 1. This condition suffices to prove that the rounding algorithm works for clauses that do not have any variables with bias very close to ±1. The case where there are b_i that are very close to 1 in magnitude is where the greedy ingredient of the algorithm (2a) is used, and it can be shown that when τ is roughly $1/k^2$, this ingredient works.In particular, we can show that for each clause, if rule (2a) is used to round one of the variables, it is used to round essentially every variable in the clause. Also, if this happens, the clause is going to be satisfied with high probability by the rounded solution.

The last complication stems from the fact that the clauses are generally not completely satisfied by the SDP solution. However, using an averaging argument, it is enough to only deal with clauses that are *almost* satisfied by the SDP solution. For any such clause the SDP induces a probability distribution on the variables that is mostly supported on satisfying assignments, compared to *only* on satisfying assignments in the above ideal setting. As such, the corresponding b_i's can be thought of as a perturbed version of the biases in that ideal setting. Unfortunately, the greedy ingredient of the algorithm is very sensitive to such small perturbations. In particular, if the biases are very close to the set threshold, τ, a small perturbation can break the method. To avoid this, we choose the actual threshold randomly, and we manage to argue that only a small fraction of the clauses end up in such unfortunate configurations.

This completes the high level description of the proof of the following theorem. Due to space considerations we leave the complete proof for the full version.

Theorem 2 (restatement of Theorem 2). *The predicate* Monarchy *is* not *approximation resistant.*

6 Discussion

We have given algorithms for two cases of Max CSP(P) problems not previously known to be approximable. The first case, signs of symmetric quadratic forms, follows from the condition that the low-degree part of the Fourier expansion behaves "roughly" like the predicate in the sense of Theorem 4. The second case, Monarchy, is interesting since it does not satisfy the condition of Theorem 4. As far as we are aware, this is the first example of a predicate which does not satisfy this property but is still approximable. Monarchy is of course only a very special case of Conjecture 1, and we leave the general form open.

A further interesting special case of the conjecture is a generalization of Monarchy called "republic", defined as $\operatorname{sign}(\frac{k}{2}x_1 + \sum_{i=2}^{k} x_i)$. In this case the x_1 variable needs to get a 1/4 fraction of the other variables on its side. We do not even know how to handle this seemingly innocuous example.

It is interesting that the condition on P for our (ϵ, η)-rounding to succeed turned out to be precisely the same as the condition previously found by Hast [6], with a completely different algorithm. It would be interesting to know whether this is a coincidence or whether there is a larger picture there that we can not yet see.

As we mentioned in the introduction, there are very few results which give approximation algorithms for large classes of predicates, and it would be very interesting if new such algorithms could be devised.

References

1. Austrin, P., Håstad, J.: Randomly Supported Independence and Resistance. In: ACM Symposium on Theory of Computing (STOC), pp. 483–492 (2009)
2. Austrin, P., Mossel, E.: Approximation Resistant Predicates from Pairwise Independence. Computational Complexity 18(2), 249–271 (2009)
3. Engebretsen, L., Holmerin, J.: More Efficient Queries in PCPs for NP and Improved Approximation Hardness of Maximum CSP. In: Diekert, V., Durand, B. (eds.) STACS 2005. LNCS, vol. 3404, pp. 194–205. Springer, Heidelberg (2005)
4. Georgiou, K., Magen, A., Tulsiani, M.: Optimal sherali-adams gaps from pairwise independence. In: Dinur, I., Jansen, K., Naor, J., Rolim, J. (eds.) APPROX 2009 and RANDOM 2009. LNCS, vol. 5687, pp. 125–139. Springer, Heidelberg (2009)
5. Goemans, M.X., Williamson, D.P.: Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. Journal of the ACM 42, 1115–1145 (1995)
6. Hast, G.: Beating a Random Assignment – Approximating Constraint Satisfaction Problems. Ph.D. thesis, KTH – Royal Institute of Technology (2005)
7. Håstad, J.: Some Optimal Inapproximability Results. Journal of the ACM 48(4), 798–859 (2001)
8. Håstad, J.: Personal communication (2009)
9. Karloff, H., Zwick, U.: A 7/8-Approximation Algorithm for MAX 3SAT? In: IEEE Symposium on Foundations of Computer Science (FOCS), p. 406 (1997)
10. Khot, S.: On the Power of Unique 2-prover 1-round Games. In: ACM Symposium on Theory of Computing (STOC), pp. 767–775 (2002)
11. Samorodnitsky, A., Trevisan, L.: A PCP characterization of NP with optimal amortized query complexity. In: ACM Symposium on Theory of Computing (STOC), pp. 191–199 (2000)
12. Samorodnitsky, A., Trevisan, L.: Gowers Uniformity, Influence of Variables, and PCPs. In: ACM Symposium on Theory of Computing (STOC), pp. 11–20 (2006)
13. Zwick, U.: Approximation Algorithms for Constraint Satisfaction Problems Involving at Most Three Variables Per Constraint. In: ACM-SIAM Symposium on Discrete Algorithms, SODA (1998)