THEORY OF COMPUTING, Volume 10 (13), 2014, pp. 341–358
www.theoryofcomputing.org

SPECIAL ISSUE: APPROX-RANDOM 2012

Approximation Algorithm for Non-Boolean Max-k-CSP

Konstantin Makarychev   Yury Makarychev*

Received October 31, 2012; Revised May 12, 2014; Published October 10, 2014

Abstract: In this paper we present a randomized polynomial-time approximation algorithm for MAX-k-CSP$_d$. In MAX-k-CSP$_d$we are given a set of predicates of arity k over an alphabet of size d. Our goal is to find an assignment that maximizes the number of satisfied constraints.

Our algorithm has approximation factor $\Omega(kd/d^k)$ (when $k \geq \Omega(\log d)$). The best previously known algorithm has approximation factor $\Omega(k \log d/d^k)$. Our bound is asymptotically optimal when $d = \Omega(d)$.

We also give an approximation algorithm for the Boolean MAX-k-CSP$_2$ problem with a slightly improved approximation guarantee.

ACM Classification: F.2.2, G.1.6

AMS Classification: 68W25

Key words and phrases: constraint satisfaction, approximation algorithm, semidefinite programming

1 Introduction

We design an approximation algorithm for MAX-k-CSP$_d$, the maximum constraint satisfaction problem with k-ary predicates and domain size d. In this problem, we are given a set $\{x_u\}_{u\in X}$ of variables and a set $\mathcal{P}$ of predicates. Each variable x_u takes values in $[d] = \{1,\dots,d\}$. Each predicate $P \in \mathcal{P}$ depends on at most k variables. Our goal is to assign values to variables so as to maximize the number of satisfied constraints.

*Supported by NSF CAREER award CCF-1150062 and NSF award IIS-1302662.

 DOI: 10.4086/toc.2014.v010a013

There has been a lot of interest in finding the approximability of MAX-k-CSP$_d$ in the complexity community motivated by the connection of MAX-k-CSP$_d$ to k-bit PCPs. Let us briefly review the known results. Samorodnitsky and Trevisan [12] showed that the Boolean MAX-k-CSP$_2$ problem cannot be approximated within a factor of $\Omega(2^{2\sqrt{k}}/2^k)$ if $\mathsf{P} \neq \mathsf{NP}$. Later Engebretsen and Holmerin [7] improved this bound to $\Omega(2^{\sqrt{2k}}/2^k)$. For non-Boolean MAX-$k$-CSP$_d$, Engebretsen [6] proved a hardness result of $d^{O(\sqrt{k})}/d^k$. Much stronger inapproximability results were obtained assuming the Unique Games Conjecture (UGC). Samorodnitsky and Trevisan [13] proved the hardness of $O(k/2^k)$ for the Boolean MAX-k-CSP$_2$. Austrin and Mossel [1] and, independently, Guruswami and Raghavendra [8] proved the hardness of $O(kd^2/d^k)$ for non-Boolean MAX-k-CSP$_d$. Moreover, Austrin and Mossel [1] proved the hardness of $O(kd/d^k)$ for every d and infinitely many k; specifically, their result holds for d and k such that $k = (d^t - 1)/(d-1)$ for some $t \in \mathbb{N}$. Based on this result of Austrin and Mossel and our matching algorithmic result (see below), we made a conjecture that the hardness of $O(kd/d^k)$ holds for every d and all sufficiently large k. Håstad proved this conjecture—he strengthened the result of Austrin and Mossel and showed the hardness of $O(kd/d^k)$ for every d and $k \geq d$. With his permission, we present his result in Section 8. Later Chan [2] proved the hardness of $O(kd/d^k)$ for every d and $k \geq d$ assuming only that $\mathsf{P} \neq \mathsf{NP}$.

On the positive side, approximation algorithms for the problem have been developed in a series of papers by Trevisan [14], Hast [9], Charikar, Makarychev and Makarychev [4], and Guruswami and Raghavendra [8]. The best currently known algorithm for k-CSP$_d$ by Charikar et al. [4] has approximation factor of $\Omega(k \log d/d^k)$. Note that a trivial algorithm for MAX-k-CSP$_d$ that just picks a random assignment satisfies each constraint with probability at least $1/d^k$, and therefore its approximation ratio is $1/d^k$.

The problem is essentially settled in the Boolean case. We know that the optimal approximation factor is $\Theta(k/2^k)$ assuming $\mathsf{P} \neq \mathsf{NP}$. However, the best known lower and upper bounds for the non-Boolean case do not match. In this paper we present an approximation algorithm for non-Boolean MAX-k-CSP$_d$ with approximation factor $\Omega(kd/d^k)$ (for $k \geq \Omega(\log d)$). This algorithm is asymptotically optimal (when $k \geq d$)—it is within a constant factor of the upper bounds of Austrin and Mossel [1], Håstad (see Section 8), and Chan [2]. Our result improves the best previously known approximation factor of $\Omega(k \log d/d^k)$.

Related work. Raghavendra studied a more general MAX-CSP($\mathcal{P}$) problem [11].

He showed that the optimal approximation factor equals the integrality gap of the standard SDP relaxation for the problem (assuming UGC). His result applies in particular to MAX-k-CSP$_d$. However, the SDP integrality gap of MAX-k-CSP$_d$ is not known.

Overview. We use semidefinite programming (SDP) to solve the problem. In our SDP relaxation, we have an "indicator vector" u_i for every variable x_u and value i; we also have an "indicator vector" z_C for every constraint C. In the intended solution, u_i is equal to a fixed unit vector $\mathbf{e}$ if $x_u = i$, and $u_i = 0$ if $x_u \neq i$; similarly, $z_C = \mathbf{e}$ if C is satisfied, and $z_C = 0$, otherwise.

It is interesting that the best previously known algorithm for the problem [4] did not use this SDP relaxation; rather it reduced the problem to a Boolean MAX-k-CSPproblem, which it solved in turn using semidefinite programming. The only previously known algorithm [8] that directly rounded an SDP solution for MAX-k-CSP$_d$ had approximation factor $\Omega\left(k/d^7/d^k\right)$.

One of the challenges of rounding the SDP solution is that the vectors u_i might have different lengths. Consequently, we cannot just use a rounding scheme that projects vectors on a random direction and then chooses vectors that have largest projections, since this scheme will choose longer vectors with disproportionately large probabilities. (In fact, if we apply this rounding scheme, we will get an $\Omega(k/d^k)$ approximation, which is worse than the $\Omega(k\log d/d^k)$ approximation of [4].) To deal with this problem, we first develop a rounding scheme that rounds *uniform* SDP solutions, solutions in which all vectors are "short." Then we construct a randomized reduction that converts any instance to an instance with a uniform SDP solution.

Our algorithm for the uniform case is very simple. First, we choose a random Gaussian vector g. Then, for every u, we find u_i that has the largest projection on g (in absolute value), and let $x_u = i$. However, the analysis of this algorithm is quite different from analyses of similar algorithms for other problems (e. g., [10, 3, 5]): when we estimate the probability that a constraint C is satisfied, we have to analyze the correlation of all vectors u_i with vector z_C (where $\{u_i\}$ are SDP vectors for variables x_u that appear in C, z_C is the SDP vector for C), whereas the standard approach would be to look only at pairwise correlations of vectors $\{u_i\}$; this approach does not work in our case, however, since vectors corresponding to an assignment that satisfies C may have very small pairwise correlations, but vectors corresponding to assignments that do not satisfy C may have much larger pairwise correlations.

Remark 1.1. We study the problem only in the regime when $k \geq \Omega(\log d)$. In Theorem 5.1, we prove that when $k = O(\log d)$ our algorithm has approximation factor $\mathrm{e}^{\Omega(k)}/d^k$. However, in this regime, there is a very simple greedy approximation algorithm that has a better approximation factor of $\Omega(d/d^k)$. For completeness, we present this algorithm in Section 9.

Other Results. We also apply our SDP rounding technique to the Boolean MAX-k-CSPProblem. We give an algorithm that has approximation guarantee $\approx 0.62\,k/2^k$ for sufficiently large k. That slightly improves the best previously known guarantee of $\approx 0.44\,k/2^k$ [4]. We present this result in Section 6.

In Section 2, we formally define the problem and present our SDP relaxation. In Section 3, we give an algorithm for rounding uniform SDP solutions. In Section 4, we present a reduction that reduces an arbitrary instance to an instance with a uniform solution. In Section 5, we put all pieces of our algorithm together and prove Theorem 5.1, the main result of this paper. In Section 6, we apply our techniques to Boolean MAX-k-CSP. In Section 7, we prove an inequality for Gaussian random variables, which we use in the analysis of the algorithm (we use this inequality in Section 3; however, we choose to describe its proof in Section 7 since the proof is elementary but technical). In Section 8, we present Håstad's hardness result for MAX-k-CSP$_d$. Finally, in Section 9, we present a simple greedy approximation algorithm for MAX-k-CSP$_d$ that performs better than our SDP algorithm when $k = O(\log d)$.

2 Preliminaries

We apply the approximation preserving reduction of Trevisan [14] to transform a general instance of MAX-k-CSP$_d$ to an instance where each predicate is a conjunction of terms of the form $x_u = i$. The reduction replaces a predicate P, which depends on variables $x_{v_1}, \ldots, x_{v_k}$, with a set of clauses

$$\{(x_{v_1} = i_1) \wedge \cdots \wedge (x_{v_k} = i_k) : P(i_1, \ldots, i_k) \text{ is true}\}\,.$$

Then it is sufficient to solve the obtained instance. We refer the reader to [14] for details. We assume below that each predicate is a clause of the form $(x_{v_1} = i_1) \wedge \cdots \wedge (x_{v_k} = i_k)$.

Definition 2.1 (Constraint satisfaction problem)**.** An instance $\mathcal{I}$ of MAX-CSP$_d$ consists of

- a set X of "indices,"
- a set $\{x_u\}_{u \in X}$ of variables (there is one variable x_u for every index $u \in X$),
- a set $\mathcal{C}$ of clauses.

Each variable x_u takes values in the domain $[d] = \{1, \ldots, d\}$. Each clause $C \in \mathcal{C}$ is a set of pairs (u,i) where $u \in X$ and $i \in [d]$. An assignment $x = x^*$ satisfies a clause C if, for every $(u,i) \in C$, we have $x_u^* = i$. We assume that no clause C in $\mathcal{C}$ contains pairs (u,i) and (u,j) with $i \neq j$ (no assignment satisfies such clause). The length of a clause C is $|C|$. The support of C is $\mathrm{supp}(C) = \{u : (u,i) \in C\}$.

The value of an assignment x^* is the number of constraints in $\mathcal{C}$ satisfied by x^*. Our goal is to find an assignment of maximum value. We denote the value of an optimal assignment by $OPT = OPT(\mathcal{I})$.

In the MAX-k-CSP$_d$ problem, we additionally require that all clauses in $\mathcal{C}$ have length at most k.

We consider the following semidefinite programming (SDP) relaxation for MAX-CSP$_d$. For every index $u \in X$ and $i \in [d]$, we have a vector variable u_i; for every clause C, we have a vector variable z_C.

$$\text{maximize: } \sum_{C \in \mathcal{C}} \|z_C\|^2$$

subject to:

$$\begin{aligned}
&\sum_{i=1}^{d} \|u_i\|^2 \leq 1 && \text{for every } u \in X,\\
&\langle u_i, u_j \rangle = 0 && \text{for every } u \in X, i, j \in [d]\ (i \neq j),\\
&\langle u_i, z_C \rangle = \|z_C\|^2 && \text{for every } C \in \mathcal{C},\ (u,i) \in C,\\
&\langle u_j, z_C \rangle = 0 && \text{for every } C \in \mathcal{C},\ (u,i) \in C \text{ and } j \neq i.
\end{aligned}$$

Denote the optimal SDP value by $SDP = SDP(\mathcal{I})$. Consider the optimal solution x^* to an instance $\mathcal{I}$ and the corresponding SDP solution defined as follows:

$$u_i = \begin{cases} \mathbf{e}, & \text{if } x_u^* = i; \\ 0, & \text{otherwise;} \end{cases} \qquad z_C = \begin{cases} \mathbf{e}, & \text{if } C \text{ is satisfied;} \\ 0, & \text{otherwise;} \end{cases}$$

where $\mathbf{e}$ is a fixed unit vector. It is easy to see that this is a feasible SDP solution and its value equals $OPT(\mathcal{I})$. Therefore, $SDP(\mathcal{I}) \geq OPT(\mathcal{I})$.

Definition 2.2. We say that an SDP solution is uniform if $\|u_i\|^2 \leq 1/d$ for every $u \in X$ and $i \in [d]$.

Definition 2.3. Let ξ be a standard Gaussian variable with mean 0 and variance 1. We denote

$$\begin{aligned}
&\Phi(t) = \Pr(|\xi| \leq t) = \frac{1}{\sqrt{2\pi}} \int_{-t}^{t} \mathrm{e}^{-x^2/2} dx, \quad \text{and}\\
&\bar{\Phi}(t) = 1 - \Phi(t) = \Pr(|\xi| > t)\,.
\end{aligned}$$

We will use the following lemma, which we prove in Section 7.

Lemma 2.4. *For every $t > 0$ and $\beta \in (0,1]$, we have*

$$\bar{\Phi}(\beta t) \leq \bar{\Phi}(t)^{\beta^2}.$$

We will also use the following result of Šidák [15].

Theorem 2.5 (Šidák [15]). *Let $\xi_1, \dots, \xi_r$ be jointly Gaussian random variables with mean zero and an arbitrary covariance matrix. Then for any positive $t_1, \dots, t_r$,*

$$\Pr\left(|\xi_1| \leq t_1, |\xi_2| \leq t_2, \dots, |\xi_r| \leq t_r\right) \geq \prod_{i=1}^{r} \Pr\left(|\xi_i| \leq t_i\right).$$

3 Rounding uniform SDP solutions

In this section we present a rounding scheme for uniform SDP solutions.

Lemma 3.1. *There is a randomized polynomial-time algorithm that given an instance $\mathcal{I}$ of the* MAX-CSP$_d$ *problem (with $d \geq 57$) and a uniform SDP solution, outputs an assignment x such that for every clause $C \in \mathcal{C}$:*

$$\Pr\left(C \text{ is satisfied by } x\right) \geq \frac{\min(\|z_C\|^2 |C| d/8, \mathrm{e}^{|C|})}{2d^{|C|}}.$$

Proof. We use the rounding algorithm described in Figure 1 below.

Rounding Scheme for Uniform SDP solutions
Input: an instance of the MAX-CSP$_d$ problem and a uniform SDP solution.
Output: an assignment x.

- Choose a random Gaussian vector g so that every component of g is distributed as a Gaussian variable with mean 0 and variance 1, and all components are independent.
- For every $u \in X$, let $x'_u = \arg\max_i |\langle u_i, g\rangle|$.
- For every $u \in X$, choose x''_u uniformly at random from $[d]$ (independently for different u).
- With probability $1/2$ return assignment x'; with probability $1/2$ return assignment x''.

Figure 1: Rounding Scheme for Uniform SDP solutions.

For every clause C, let us estimate the probabilities that assignments x' and x'' satisfy C. It is clear that x'' satisfies C with probability $d^{-|C|}$. We prove now that x' satisfies C with probability at least $d^{-3|C|/4}$ if $\|z_C\|^2 \geq 8/(|C|d)$.

Claim 3.2. *Suppose $C \in \mathcal{C}$ is a clause such that $\|z_C\|^2 \geq 8/(|C|d)$ and $d \geq 57$. Then the probability that the assignment x' satisfies C is at least $d^{-3|C|/4}$.*

Proof. Denote $s = |C|$. We assume without loss of generality that for every $u \in \operatorname{supp}(C)$, $(u,1) \in C$. Note that for $(u,i) \in C$, we have $\|z_C\|^2 = \langle z_C, u_i\rangle \leq \|z_C\| \cdot \|u_i\| \leq \|z_C\|/\sqrt{d}$ (here we use that the SDP solution is uniform and therefore $\|u_i\|^2 \leq 1/d$). Thus $\|z_C\|^2 \leq 1/d$. In particular, $s = |C| \geq 8$ since $\|z_C\|^2 \geq 8/(|C|d)$.

For every $u \in \operatorname{supp}(C)$, let $u_1^\perp = u_1 - z_C$. Let $\gamma_{u,1} = \langle g, u_1^\perp\rangle$ and $\gamma_{u,i} = \langle g, u_i\rangle$ for $i \geq 2$. Let $\gamma_C = \langle g, z_C\rangle$. All variables $\gamma_{u,i}, \gamma_C$ are jointly Gaussian random variables. Using that for every two vectors v and w, $\mathbb{E}[\langle g,v\rangle \cdot \langle g,w\rangle] = \langle v,w\rangle$, we get

$$\mathbb{E}[\gamma_C \cdot \gamma_{u,1}] = \langle z_C, u_1 - z_C\rangle = \langle z_C, u_1\rangle - \|z_C\|^2 = 0;$$
$$\mathbb{E}[\gamma_C \cdot \gamma_{u,i}] = \langle z_C, u_i\rangle = 0 \quad \text{for } i \geq 2.$$

Therefore, all variables $\gamma_{u,i}$ are independent from γ_C. (However, for $u', u'' \in \operatorname{supp}(C)$ variables $\gamma_{u',i}$ and $\gamma_{u'',j}$ are not necessarily independent.) Let $M = \bar{\Phi}^{-1}(1/d^{s/2})/\sqrt{sd/8}$. We write the probability that x' satisfies C,

$$\begin{aligned}
\Pr\big(x' \text{ satisfies } C\big) &= \Pr\big(\arg\max_i |\langle g, u_i\rangle| = 1 \text{ for every } u \in \operatorname{supp}(C)\big)\\
&= \Pr\left(|\langle g, u_1\rangle| > |\langle g, u_i\rangle| \text{ for every } u \in \operatorname{supp}(C), i \in \{2,\dots,d\}\right)\\
&= \Pr\left(|\gamma_{u,1} + \gamma_C| > |\gamma_{u,i}| \text{ for every } u \in \operatorname{supp}(C), i \in \{2,\dots,d\}\right)\\
&\geq \Pr(|\gamma_{u,1}| \leq M/2, \text{ and } |\gamma_{u,i}| \leq M/2\\
&\qquad \text{for every } u \in \operatorname{supp}(C), i \in \{2,\dots,d\} \;\big|\; |\gamma_C| > M) \cdot \Pr\left(|\gamma_C| > M\right).
\end{aligned}$$

Since all variables $\gamma_{u,i}$ are independent from γ_C,

$$\Pr\big(x' \text{ satisfies } C\big) \geq \Pr\left(|\gamma_{u,i}| \leq M/2 \text{ for every } u \in \operatorname{supp}(C), i \in \{1,\dots,d\}\right) \cdot \Pr\left(|\gamma_C| > M\right).$$

By Šidák's Theorem (Theorem 2.5), we have

$$\Pr\big(x' \text{ satisfies } C\big) \geq \Big(\prod_{u \in \operatorname{supp}(C)} \prod_{i=1}^{d} \Pr\left(|\gamma_{u,i}| \leq M/2\right)\Big) \cdot \Pr\left(|\gamma_C| > M\right). \tag{3.1}$$

We compute the variance of vectors $\gamma_{u,i}$. We use that $\operatorname{Var}[\langle g,v\rangle] = \|v\|^2$ for every vector v and that the SDP solution is uniform.

$$\operatorname{Var}[\gamma_{u,1}] = \|u_1^\perp\|^2 = \|u_1 - z_C\|^2 = \|u_1\|^2 - 2\langle u_1, z_C\rangle + \|z_C\|^2 = \|u_1\|^2 - \|z_C\|^2 \leq \|u_1\|^2 \leq 1/d;$$
$$\operatorname{Var}[\gamma_{u,i}] = \|u_i\|^2 \leq 1/d \quad \text{for } i \geq 2.$$

Hence since $\Phi(t)$ is an increasing function and $\bar{\Phi}(\beta t) \leq \bar{\Phi}(t)^{\beta^2}$ (by Lemma 2.4), we have

$$\begin{aligned}
\Pr\left(|\gamma_{u,i}| \leq M/2\right) &= \Phi(M/(2\sqrt{\operatorname{Var}[\gamma_{u,i}]})) \geq \Phi(\sqrt{d}M/2) = 1 - \bar{\Phi}(\sqrt{d}\,M/2)\\
&\geq 1 - \bar{\Phi}(\sqrt{sd/8}\,M)^{2/s} = 1 - (d^{-s/2})^{2/s} = 1 - d^{-1}
\end{aligned}$$

(recall that we defined M so that $\bar{\Phi}(\sqrt{sd/8}M) = d^{-s/2}$). Similarly, $\mathrm{Var}[\gamma_C] = \|z_C\|^2 \geq 8/(sd)$ (by the condition of the lemma). We get (using the fact that $\bar{\Phi}(t)$ is a decreasing function),

$$\Pr(|\gamma_C| > M) = \bar{\Phi}(M/\sqrt{\mathrm{Var}[\gamma_C]}) \geq \bar{\Phi}(M\sqrt{sd/8}) = d^{-s/2}.$$

Plugging in bounds for $\Pr(|\gamma_{u,i}| \leq M/2)$ and $\Pr(|\gamma_C| > M)$ into (3.1), we obtain

$$\Pr\left(x' \text{ satisfies } C\right) \geq (1-d^{-1})^{ds} d^{-s/2} \geq d^{-3s/4}.$$

Here, we used that $(1-d^{-1})^d \geq d^{-1/4}$ for $d \geq 57$ (the inequality $(1-d^{-1})^d \geq d^{-1/4}$ holds for $d \geq 57$ since it holds for $d = 57$ and the left hand side, $(1-d^{-1})^d$, is an increasing function, the right hand side, $d^{-1/4}$, is a decreasing function). □

We conclude that if $\|z_C\|^2 \leq 8/(|C|d)$ then the algorithm chooses assignment x'' with probability $1/2$ and this assignment satisfies C with probability at least $1/d^{|C|} \geq \|z_C\|^2 |C| d/(8d^{|C|})$. So C is satisfied with probability at least, $1/d^{|C|} \geq \|z_C\|^2 |C| d/(16d^{|C|})$; if $\|z_C\|^2 \geq 8/(|C|d)$ then the algorithm chooses assignment x' with probability $1/2$ and this assignment satisfies C with probability at least $d^{-3|C|/4} \geq \mathrm{e}^{|C|}/d^{|C|}$ (since $e \leq 57^{1/4} \leq d^{1/4}$). In either case,

$$\Pr(C \text{ is satisfied}) \geq \frac{\min(\|z_C\|^2|C|d/8, \mathrm{e}^{|C|})}{2d^{|C|}}. \qquad \square$$

Remark 3.3. We note that we did not try to optimize all constants in the statement of Lemma 3.1. By choosing all parameters in our proof appropriately, it is possible to show that for every constant $\varepsilon > 0$, there is a randomized rounding scheme, $\delta > 0$ and d_0 such that for every instance of MAX-CSP$_d$ with $d \geq d_0$ the probability that each clause C is satisfied is at least $\min((1-\varepsilon)\|z_C\|^2 \cdot |C|\,d, \delta \cdot \mathrm{e}^{\delta|C|})/d^{|C|}$.

4 Rounding arbitrary SDP solutions

In this section we show how to round an arbitrary SDP solution.

Lemma 4.1. *There is a randomized polynomial-time algorithm that given an instance $\mathcal{I}$ of the* MAX-CSP$_d$ *problem (with $d \geq 113$) and an SDP solution, outputs an assignment x such that for every clause $C \in \mathcal{C}$:*

$$\Pr(C \textit{ is satisfied by } x) \geq \frac{\min(\|z_C\|^2|C|d/64, 2\mathrm{e}^{|C|/8})}{4d^{|C|}}.$$

Proof. For every index u, we sort all vectors u_i according to their length. Let S_u be the indices of $\lceil d/2 \rceil$ shortest vectors among u_i, and $L_u = [d] \setminus S_u$ be the indices of $\lfloor d/2 \rfloor$ longest vectors among u_i (we break ties arbitrarily). For every clause C let $r(C) = |\{(u,i) \in C : i \in S_u\}|$.

Claim 4.2. *For every $i \in S_u$, we have $\|u_i\|^2 \leq 1/|S_u|$.*

Proof. Let $i \in S_u$. Note that $\|u_i\|^2 + \sum_{j \in L_u} \|u_j\|^2 \leq 1$ (this follows from SDP constraints). There are at least $\lceil d/2 \rceil$ terms in the sum, and $\|u_i\|^2$ is the smallest among them (since $i \in S_u$). Thus $\|u_i\|^2 \leq 1/\lceil d/2 \rceil = 1/|S_u|$. □

We use a combination of two rounding schemes: one of them works well on clauses C with $r(C) \geq |C|/4$, the other on clauses C with $r(C) \leq |C|/4$.

Lemma 4.3. *There is a polynomial-time randomized rounding algorithm that given a* MAX-CSP$_d$ *instance* $\mathcal{I}$ *with* $d \geq 113$ *outputs an assignment* x *such that every clause* C *with* $r(C) \geq |C|/4$ *is satisfied with probability at least*

$$\frac{\min(\|z_C\|^2\,|C|\,d/64, \mathrm{e}^{|C|/4})}{2d^{|C|}}.$$

Proof. We will construct a sub-instance $\mathcal{I}'$ with a uniform SDP solution and then solve $\mathcal{I}'$ using Lemma 3.1. To this end, we first construct a partial assignment x. For every $u \in X$, with probability $|L_u|/d = \lfloor d/2 \rfloor/d$, we assign a value to x_u uniformly at random from L_u; with probability $1 - |L_u|/d = |S_u|/d$, we do not assign any value to x_u. Let $A = \{u : x_u \text{ is assigned}\}$. Let us say that a clause C *survives* the partial assignment step if for every $(u,i) \in C$ either $u \in A$ and $i = x_u$, or $u \notin A$ and $i \in S_u$.

The probability that a clause C survives is

$$\prod_{(u,i)\in C, i\in L_u} \Pr\left(x_u \text{ is assigned value } i\right) \prod_{(u,i)\in C, i\in S_u} \Pr\left(x_u \text{ is unassigned}\right)$$
$$= \left(\frac{\lfloor d/2\rfloor}{d}\cdot\frac{1}{\lfloor d/2\rfloor}\right)^{|C|-r(C)}\cdot\left(\frac{\lceil d/2\rceil}{d}\right)^{r(C)} = \frac{\lceil d/2\rceil^{r(C)}}{d^{|C|}}.$$

For every surviving clause C, let $C' = \{(u,i) : u \notin A\}$. Note that for every $(u,i) \in C'$ we have $i \in S_u$. We get a sub-instance $\mathcal{I}'$ of our problem on the set of unassigned variables $\{x_u : u \notin A\}$ with the set of clauses $\{C' : C \in \mathcal{C} \text{ survives}\}$. The length of each clause C' equals $r(C)$. In sub-instance $\mathcal{I}'$, we require that each variable x_u takes values in S_u. Thus $\mathcal{I}'$ is an instance of MAX CSP$_{d'}$ problem with $d' = |S_u| = \lceil d/2 \rceil$.

Now we transform the SDP solution for $\mathcal{I}$ to an SDP solution for $\mathcal{I}'$: we let $z_{C'} = z_C$ for surviving clauses C, remove vectors u_i for all $u \in A$, $i \in [d]$ and remove vectors z_C for non-surviving clauses C. By Claim 4.2, this SDP solution is a uniform solution for $\mathcal{I}'$ (i. e., $\|u_i\|^2 \leq 1/d'$ for every $u \notin A$ and $i \in S_i$; note that $\mathcal{I}'$ has alphabet size d'). We run the rounding algorithm from Lemma 3.1. The algorithm assigns values to unassigned variables x_u. For every *surviving* clause C, we get

$$\Pr\left(C \text{ is satisfied by } x\right) = \Pr\left(C' \text{ is satisfied by } x\right) \geq \frac{\min(\|z_C\|^2|C'|d'/8, \mathrm{e}^{|C'|})}{2d'^{|C'|}}$$
$$= \frac{\min(\|z_C\|^2 r(C)d'/8, \mathrm{e}^{r(C)})}{2d'^{r(C)}} \geq \frac{\min(\|z_C\|^2|C|d/64, \mathrm{e}^{|C|/4})}{2d'^{r(C)}}.$$

Therefore, for every clause C,

$$\Pr\left(C \text{ is satisfied by } x\right) \geq \Pr\left(C \text{ is satisfied by } x \mid C \text{ survives}\right)\Pr\left(C \text{ survives}\right)$$
$$\geq \frac{\min(\|z_C\|^2|C|d/64, \mathrm{e}^{|C|/4})}{2d'^{r(C)}} \times \frac{\lceil d/2\rceil^{r(C)}}{d^{|C|}}$$
$$= \frac{\min(\|z_C\|^2|C|d/64, \mathrm{e}^{|C|/4})}{2d^{|C|}}.$$
□

Finally, we describe an algorithm for clauses C with $r(C) \le |C|/4$.

Lemma 4.4. *There is a polynomial-time randomized rounding algorithm that given an* MAX-CSP$_d$ *instance $\mathcal{I}$ outputs an assignment x such that every clause C with $r(C) \le |C|/4$ is satisfied with probability at least* $\mathrm{e}^{|C|/8}/d^{|C|}$.

Proof. We do the following independently for every vertex $u \in X$. With probability $3/4$, we choose x_u uniformly at random from L_u; with probability $1/4$, we choose x_u uniformly at random from S_u. The probability that a clause C with $r(C) \le |C|/4$ is satisfied equals

$$\prod_{(u,i)\in C, i\in L_u} \frac{3}{4|L_u|} \prod_{(u,i)\in C, i\in S_u} \frac{1}{4|S_u|} = \frac{1}{d^{|C|}} \cdot \left(\frac{3d}{4|L_u|}\right)^{|C|-r(C)} \left(\frac{d}{4|S_u|}\right)^{r(C)}$$

$$\ge \frac{1}{d^{|C|}} \cdot \left(\frac{3d}{4|L_u|}\right)^{3|C|/4} \left(\frac{d}{4|S_u|}\right)^{|C|/4} \ge \frac{1}{d^{|C|}} \cdot \left(\left(\frac{3}{2}\right)^{3/4} \left(\frac{d}{2(d+1)}\right)^{1/4}\right)^{|C|}.$$

Note that

$$\left(\frac{3}{2}\right)^{3/4} \left(\frac{d}{2(d+1)}\right)^{1/4} \ge \left(\frac{3}{2}\right)^{3/4} \left(\frac{113}{2\cdot 114}\right)^{1/4} \ge \mathrm{e}^{1/8}.$$

Therefore, the probability that the clause is satisfied is at least $\mathrm{e}^{|C|/8}/d^{|C|}$. □

We run the algorithm from Lemma 4.3 with probability $1/2$ and the algorithm from Lemma 4.4 with probability $1/2$. Consider a clause $C \in \mathcal{C}$. If $r(C) \ge |C|/4$, we satisfy C with probability at least

$$\frac{\min(\|z_C\|^2 |C| d/64, \mathrm{e}^{|C|/4})}{4d^{|C|}}.$$

If $r(C) \le |C|/4$, we satisfy C with probability at least $\mathrm{e}^{|C|/8}/(2d^{|C|})$. So we satisfy every clause C with probability at least

$$\frac{\min(\|z_C\|^2 |C| d/64,\ 2\mathrm{e}^{|C|/8})}{4d^{|C|}}.$$ □

5 Approximation algorithm for MAX-k-CSP$_d$

In this section we combine results from previous sections and prove the main theorem of the paper.

Theorem 5.1. *There is a polynomial-time randomized approximation algorithm for* MAX-k-CSP$_d$ *that given an instance $\mathcal{I}$ finds an assignment that satisfies at least $\Omega(\min(kd, \mathrm{e}^{k/8})\, OPT(\mathcal{I})/d^k)$ clauses with constant probability.*

Proof. If $d \le 113$, we run the algorithm of Charikar, Makarychev and Makarychev [4] and get $\Omega(k/d^k)$ approximation. So we assume below that $d \ge 113$. We also assume that $kd/d^k \ge 1/|\mathcal{C}|$, as otherwise we just choose one clause from $\mathcal{C}$ and find an assignment that satisfies it. Thus d^k is polynomial in the size of the input.

We solve the SDP relaxation for the problem and run the rounding scheme from Lemma 4.1 d^k times. We output the best of the obtained solutions. By Lemma 4.1, each time we run the rounding scheme we get a solution with expected value at least

$$\sum_{C\in\mathcal{C}} \frac{\min(\|z_C\|^2|C|d/64, 2e^{|C|/8})}{4d^{|C|}} \geq \sum_{C\in\mathcal{C}} \frac{\min(kd/64, 2e^{k/8})}{4d^k}\|z_C\|^2 \geq \frac{\min(kd/64, 2e^{k/8})}{4d^k} SDP(\mathcal{I}) \geq \frac{\min(kd/64, 2e^{k/8})}{4d^k} OPT(\mathcal{I}).$$

Denote $\alpha = \min(kd/64, 2e^{k/8})/4d^k$. Let Z be the random variable equal to the number of satisfied clauses. Then $\mathbb{E}[Z] \geq \alpha OPT(\mathcal{I})$, and $Z \leq OPT(\mathcal{I})$ (always). Let $p = \Pr(Z \leq \alpha OPT(\mathcal{I})/2)$. Then

$$p \cdot (\alpha OPT(\mathcal{I})/2) + (1-p)\cdot OPT(\mathcal{I}) \geq \mathbb{E}[Z] \geq \alpha OPT(\mathcal{I}).$$

So

$$p \leq \frac{1-\alpha}{1-\alpha/2} = 1 - \frac{\alpha}{2-\alpha}.$$

So with probability at least $1-p \geq \alpha/(2-\alpha)$, we find a solution of value at least $\alpha OPT(\mathcal{I})/2$ in one iteration. Since we perform $d^k > 1/\alpha$ iterations, we find a solution of value at least $\alpha OPT(\mathcal{I})/2$ with constant probability. □

6 Improved approximation factor for Boolean MAX-k-CSP

In this section we present an approximation algorithm for the Boolean Maximum k-CSP problem, MAX-k-CSP$_2$. The algorithm has approximation factor $0.626612k/2^k$ if k is sufficiently large. This bound improves the previously best known bound of $0.44k/2^k$ [4] (if k is sufficiently large).

Our algorithm is a slight modification of the algorithm for rounding uniform solutions of MAX-k-CSP$_d$. We use the SDP relaxation presented in Section 2. Without loss of generality, we will assume below that all clauses have length exactly k. If a clause C is shorter, we can introduce $k-|C|$ new variables and append them to C. This transformation will not change the value of the instance.

First, we describe a rounding scheme for an SDP solution $\{u_1, u_2\}_{u\in X} \cup \{z_C\}_{C\in\mathcal{C}}$.

Lemma 6.1. *There is a polynomial-time randomized rounding algorithm such that for every clause $C\in\mathcal{C}$ the probability that the algorithm satisfies C is at least*

$$\frac{1}{2^k\sqrt{2\pi/k}} \int_0^\infty h_\beta(t)^k dt, \text{ where } h_\beta(t) = 2\Phi(\beta t)\, e^{-t^2/2},$$

and $\beta = \sqrt{k}\,\|z_C\|_2$.

Proof. We round the SDP solution as described in Figure 2 below.

Consider a clause $C\in\mathcal{C}$. We assume without loss of generality that $C = \{(u,1) : u \in \operatorname{supp}(C)\}$. Let $\gamma_C = \langle z_C, g\rangle$ and $\gamma_u = \langle u_2 - u_1 + z_C, g\rangle$ for $u\in\operatorname{supp}(C)$. Note that all variables γ_C and γ_u are jointly

SDP Rounding Scheme for MAX-k-CSP$_2$

Input: an instance of MAX-k-CSP$_2$ and an SDP solution.
Output: an assignment x.

- Choose a random Gaussian vector g so that every component of g is distributed as a Gaussian variable with mean 0 and variance 1, and all components are independent.
- For every $u \in X$, let $x_u = \arg\max_i \langle u_i, g\rangle$.

Figure 2: SDP Rounding Scheme for MAX-k-CSP$_2$.

Gaussian. We have for $u \in \operatorname{supp}(C)$,

$$\begin{aligned}
\operatorname{Var}[\gamma_C] &= \|z_C\|^2 = \beta^2/k \quad (\text{where } \beta = \sqrt{k}\,\|z_C\|_2),\\
\operatorname{Var}[\gamma_u] &= \|u_2 - u_1 + z_C\|^2 = \|u_1\|^2 + \|u_2\|^2 + \|z_C\|^2 - 2\langle u_1, z_C\rangle = \|u_1\|^2 + \|u_2\|^2 - \|z_C\|^2 \le 1,\\
\mathbb{E}[\gamma_C\gamma_u] &= \langle z_C, u_2 - u_1 + z_C\rangle = \langle z_C, u_2\rangle - \langle z_C, u_1\rangle + \langle z_C, z_C\rangle = 0 - \|z_C\|^2 + \|z_C\|^2 = 0.
\end{aligned}$$

Therefore, all random variables γ_u, for $u \in \operatorname{supp}(C)$, are independent from γ_C. The probability that C is satisfied equals

$$\begin{aligned}
\Pr(C \text{ is satisfied}) &= \Pr(\langle u_1, g\rangle > \langle u_2, g\rangle \text{ for every } u \in \operatorname{supp}(C))\\
&= \Pr(\gamma_C > \gamma_u \text{ for every } u \in \operatorname{supp}(C)) \ge \Pr(|\gamma_u| < \gamma_C \text{ for every } u \in \operatorname{supp}(C))\\
&= \mathbb{E}_{\gamma_C}[\Pr(|\gamma_u| \le \gamma_C \text{ for every } u \in \operatorname{supp}(C) \mid \gamma_C)]\\
&\overset{\text{let } t = \gamma_C/\beta}{=} \frac{1}{\sqrt{2\pi/k}} \int_{t=0}^{\infty} \Pr(|\gamma_u| \le \beta t \text{ for every } u \in \operatorname{supp}(C))\, \mathrm{e}^{-kt^2/2}\, dt\,.
\end{aligned}$$

We use here that $\operatorname{Var}[\gamma_C/\beta] = 1/k$. By Šidák's Theorem (Theorem 2.5), we have

$$\begin{aligned}
\Pr(|\gamma_u| \le \beta t \text{ for every } u \in \operatorname{supp}(C)) &\ge \prod_{u \in \operatorname{supp}(C)} \Pr(|\gamma_u| \le \beta t) = \prod_{u \in \operatorname{supp}(C)} \Phi(\beta t/\sqrt{\operatorname{Var}[\gamma_u]})\\
&\ge \prod_{u \in \operatorname{supp}(C)} \Phi(\beta t) = \Phi(\beta t)^k\,.
\end{aligned}$$

We conclude that

$$\Pr(C \text{ is satisfied}) \ge \frac{1}{2^k\sqrt{2\pi/k}} \int_0^{\infty} h_\beta(t)^k dt. \qquad \square$$

Let $g(\beta) = \max_{t \in \mathbb{R}} h_\beta(t)$ ($h_\beta(t)$ attains its maximum since $h_\beta(t) \to 0$ as $t \to \infty$). Note that $g(\beta)$ is an increasing function since $h_\beta(t)$ is an increasing function of β for every fixed t. Additionally, $g(0) = 0$ and $\lim_{\beta \to \infty} g(\beta) = 2$ since $g(\beta) \ge h_\beta(\beta^{-1/2}) = 2\Phi(\sqrt{\beta})\mathrm{e}^{-1/(2\beta)} \to 2$ as $\beta \to \infty$, and for every β and t, $h_\beta(t) \le 2$. Therefore, g^{-1} is defined on $[0,2)$. Let $\beta_0 = g^{-1}(1)$. It is easy to check numerically that $\beta_0 \in (1.263282, 1.263283)$.

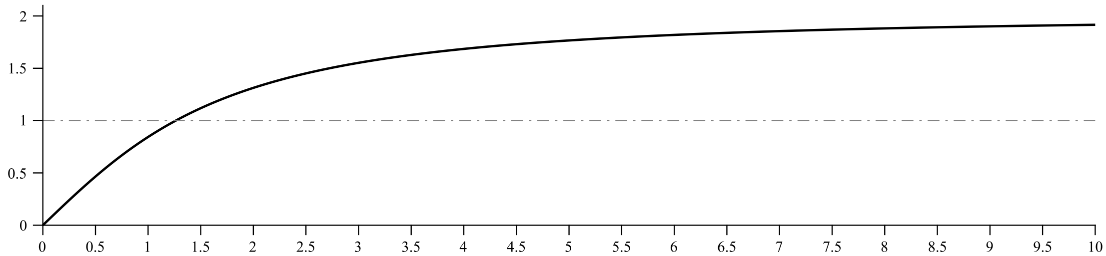

Figure 3: The figure shows the graph of $g(t)$. Note that $g(t) > 1$ when $t > \beta_0 \approx 1.263282$.

Claim 6.2. *For every $\beta > \beta_0$ there exists k_0 (which depends only on β) such that if $k \geq k_0$ and $\|z_C\| \geq \beta/\sqrt{k}$ then the probability that the algorithm from Lemma 6.1 returns an assignment that satisfies C is at least $k^2/2^k$.*

Proof. Let $\varepsilon_1 = (g(\beta)-1)/2 > 0$. Let ε_2 be the measure of the set $\{t : h_\beta(t) > 1+\varepsilon_1\}$. Since $h_\beta(t)$ is continuous, $\varepsilon_2 > 0$.

The probability that C is satisfied is at least

$$\frac{1}{2^k\sqrt{2\pi/k}}\int_0^\infty h_\beta(t)^k dt \geq \frac{\varepsilon_2(1+\varepsilon_1)^k}{2^k\sqrt{2\pi/k}}.$$

We choose k_0 so that for every $k \geq k_0$

$$\varepsilon_2(1+\varepsilon_1)^k \geq \sqrt{2\pi/k}\cdot k^2.$$

Then if $k \geq k_0$ the probability that the clause is satisfied is at least $k^2/2^k$. □

Now we are ready to describe our algorithm.

Theorem 6.3. *There is a randomized approximation algorithm for the Boolean* MAX-k-CSP*problem with approximation guarantee $\alpha_k k/2^k$ where $\alpha_k \to \alpha_0 \geq 0.626612$ as $k \to \infty$ and $\alpha_0 = 1/\beta_0^2$. (Here, as above, β_0 is the solution of the equation $g(\beta) = 1$ where $g(\beta) = \max_{t\in\mathbb{R}} 2\Phi(\beta t)e^{-t^2/2}$.)*

Proof. The algorithm with probability $p = 1/k$ rounds the SDP solution as described in Lemma 6.1, with probability $1-p$, it choses a completely random solution.

Let $\alpha < \alpha_0$. We will show that if k is large enough, every clause is satisfied with probability at least $\alpha k/2^k$. Let $\beta = (\beta_0 + \alpha^{-1/2})/2 \in (\beta_0, \alpha^{-1/2})$ (recall that $\alpha^{-1/2} > \alpha_0^{-1/2} = \beta_0$). Let k_0 be as in Claim 6.2. Suppose that $k \geq \max(k_0, (1-\alpha\beta^2)^{-1})$.

Consider a clause C. We show that the algorithm satisfies C with probability at least $\alpha\|z_C\|^2 k/2^k$. Indeed, we have:

- If $\|z_C\| < \beta/\sqrt{k}$, the clause is satisfied with probability at least

$$(1-p)/2^k \geq \frac{(1-p)k\|z_C\|^2}{\beta^2 2^k} \geq \alpha\|z_C\|^2 k/2^k.$$

- If $\|z_C\| \geq \beta/\sqrt{k}$, the clause is satisfied with probability at least $p \cdot k^2/2^k = k/2^k \geq k\|z_C\|^2/2^k$.

We conclude that the algorithm finds a solution that satisfies at least

$$\frac{\alpha k}{2^k} \sum_{C \in \mathcal{C}} \|z_C\|^2 = \frac{\alpha k}{2^k} \cdot SDP \geq \frac{\alpha k}{2^k} \cdot OPT$$

clauses in expectation. By running this algorithm polynomially many times (as we do in Theorem 5.1), we can find a solution of value at least $\alpha' k\, OPT/2^k$ for every constant $\alpha' < \alpha$ with high probability. □

7 Proof of Lemma 2.4

In this section we prove Lemma 2.4. We will use the following fact.

Lemma 7.1 (see, e. g., [3]). *For every $t > 0$,*

$$\frac{2t}{\sqrt{2\pi}\,(t^2+1)} e^{-\frac{t^2}{2}} < \bar{\Phi}(t) < \frac{2}{\sqrt{2\pi}\,t} e^{-\frac{t^2}{2}}.$$

Lemma 2.4. *For every $t > 0$ and $\beta \in (0,1]$, we have*

$$\bar{\Phi}(\beta t) \leq \bar{\Phi}(t)^{\beta^2}.$$

Proof. Rewrite the inequality we need to prove as follows: $(\bar{\Phi}(\beta t))^{1/\beta^2} \leq \bar{\Phi}(t)$. Denote the left hand side by $f(\beta,t)$:

$$f(\beta,t) = \bar{\Phi}(\beta t)^{1/\beta^2}.$$

We show that for every $t > 0$, $f(\beta,t)$ is a non-decreasing function as a function of $\beta \in (0,1]$. Then,

$$(\bar{\Phi}(\beta t))^{1/\beta^2} = f(\beta,t) \leq f(1,t) = \bar{\Phi}(t).$$

We first prove that $\frac{\partial f(1,t)}{\partial \beta} > 0$ for $t > 0$. Write,

$$\frac{\partial f(1,t)}{\partial \beta} = -2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) + t\bar{\Phi}'(t) = -2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) - \frac{2t\,e^{-t^2/2}}{\sqrt{2\pi}}.$$

Consider two cases.

Case 1: $t \geq \sqrt{2e/\pi}$. By Lemma 7.1,

$$\bar{\Phi}(t) < \frac{2}{\sqrt{2\pi}t} e^{-t^2/2} \leq e^{-1/2} e^{-t^2/2} = e^{-(t^2+1)/2}.$$

Hence, $-2\log(\bar{\Phi}(t)) > (t^2+1)$, and by Lemma 7.1,

$$-2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) > (t^2+1)\,\bar{\Phi}(t) > \frac{2t\,e^{-t^2/2}}{\sqrt{2\pi}}.$$

Thus $\frac{\partial f(1,t)}{\partial \beta} > 0$.

Case 2: $t < \sqrt{2e/\pi}$. Let $\rho(x) = -\log x/(1-x)$ for $x \in (0,1)$ and write,

$$-\log \bar{\Phi}(t) = \rho(\bar{\Phi}(t)) \cdot (1-\bar{\Phi}(t)) = \frac{\rho(\bar{\Phi}(t))}{\sqrt{2\pi}} \int_{-t}^{t} \mathrm{e}^{-x^2/2} dx \geq \frac{2\rho(\bar{\Phi}(t)) t \mathrm{e}^{-t^2/2}}{\sqrt{2\pi}}.$$

Hence,

$$\frac{\partial f(1,t)}{\partial \beta} = -2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) - \frac{2t\,\mathrm{e}^{-t^2/2}}{\sqrt{2\pi}} \geq \frac{2t\mathrm{e}^{-t^2/2}}{\sqrt{2\pi}} \times (2\rho(\bar{\Phi}(t))\bar{\Phi}(t) - 1).$$

Assume first that $\bar{\Phi}(t) \geq 1/3$. Note that $2x\rho(x) > 1$ for $x \in [1/3, 1]$ since the function $x\rho(x)$ is increasing and $\rho(1/3) > 3/2$. Hence $2\bar{\Phi}(t)\rho(\bar{\Phi}(t)) > 1$ and thus $\frac{\partial f(1,t)}{\partial \beta} > 0$.

Assume now that $\bar{\Phi}(t) < 1/3$ (we still consider the case $t < \sqrt{2e/\pi}$). Then, $\bar{\Phi}(t) \geq \bar{\Phi}(\sqrt{2e/\pi}) > 1/6$ and hence $\bar{\Phi}(t) \in (1/6, 1/3)$. Since the function $-x\log x$ is increasing on the interval $(0, \mathrm{e}^{-1})$,

$$-2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) > -2\log(1/6) \cdot \frac{1}{6} > \frac{1}{2}.$$

The function $t\mathrm{e}^{-t^2/2}$ attains its maximum at $t = 1$, thus

$$\frac{2t\,\mathrm{e}^{-t^2/2}}{\sqrt{2\pi}} \leq \frac{2\mathrm{e}^{-1/2}}{\sqrt{2\pi}} < \frac{1}{2}.$$

We get

$$\frac{\partial f(1,t)}{\partial \beta} = -2\log(\bar{\Phi}(t))\,\bar{\Phi}(t) - \frac{2t\,\mathrm{e}^{-t^2/2}}{\sqrt{2\pi}} > 0.$$

Since $\frac{\partial f(1,t)}{\partial \beta} > 0$, for every $t' > 0$ there exists $\varepsilon_0 > 0$ such that for all $\varepsilon \in (0, \varepsilon_0)$, $f(1-\varepsilon, t') < f(1, t')$. Particularly, for $t' = \beta t$, some $\varepsilon_0 > 0$ and every $\varepsilon \in (0, \varepsilon_0)$, we have

$$f(\beta, t) = f(1, t')^{1/\beta^2} \geq f(1-\varepsilon, t')^{1/\beta^2} \geq f((1-\varepsilon)\beta, t).$$

Therefore, $f(\beta, t)$ is a non-decreasing function of β. □

8 Hardness of MAX-k-CSP$_d$

In this section we present Håstad's hardness result for MAX-k-CSP$_d$.

Definition 8.1. Let $f(k,d)$ be the infimum of all C such that there is a C/d^k approximation algorithm for MAX-k-CSP$_d$.

Håstad proves that $f(k,d) \leq 4kd$ for $k \geq d$, assuming the Unique Games Conjecture. His result is based on the following theorem of Austrin and Mossel [1].

Theorem 8.2 (Austrin and Mossel [1]). *Let $\chi_1, \dots, \chi_k$ be a family of k pairwise independent random variables defined on a sample space Ω that take values in a set of size d. Then $f(k,d) \le |\Omega|$ assuming the Unique Games Conjecture.*

Theorem 8.2 shows that in order to prove that $f(k,d) = O(kd)$, it suffices to construct a family of k pairwise independent random variables defined on a sample space of size $O(kd)$.

Lemma 8.3. *Let p be a prime number. Suppose that d and k are powers of p, and $p \le d \le k$. Then there exist k random variables defined on a sample space Ω such that*

- *each random variable is uniformly distributed in a set of size d.*
- *all variables are pairwise independent, and*
- $|\Omega| = kd$.

Proof. Let F be a finite field of size p and E be an extension of F of size k. Note that E is a linear space over F. Let L be a linear subspace of E of dimension $\log_p d$ (L is not necessarily a subfield of E). Then $|L| = d$. Let π be a projection of E on L. Consider the probability space $\Omega = E \times L$ with uniform measure. Define the following family of random variables indexed by elements of E:

$$\chi_e(a,b) = \pi(ae) + b \quad \text{where } (a,b) \in \Omega\,.$$

Note that $|\Omega| = |E \times L| = kd$, the domain size of each random variable is $|L| = d$, and the number of random of variable is $|E| = k$. We show now that random variables $\{\chi_e\}_{e \in E}$ are uniformly distributed in L and pairwise independent.

Consider a random variable χ_e. Note that $\chi_e = \pi(ae) + b$ is uniformly distributed in L when a is fixed and b is random. Therefore, χ_e is also uniformly distributed in E when both a and b are random.

Now consider two random variables χ_e and $\chi_{e'}$. Observe that $\chi_e - \chi_{e'} = \pi(a(e-e'))$. Since $e \ne e'$, we have that $a(e-e')$ is uniformly distributed in E, and thus $\chi_e - \chi_{e'}$ is uniformly distributed in L. For every $c, c' \in L$, we have

$$\begin{aligned}\Pr\left(\chi_e = c, \chi_{e'} = c'\right) &= \Pr\left(\chi_e = c, \chi_e - \chi_{e'} = c - c'\right) = \mathbb{E}_a[\Pr\left(\chi_e = c, \chi_e - \chi_{e'} = c - c' \mid a\right)] \\ &= \mathbb{E}_a[\Pr\left(\chi_e = c \mid a\right) \cdot \mathbf{1}_{\chi_e - \chi_{e'} = c - c'}] = \mathbb{E}_a[1/|L| \cdot \mathbf{1}_{\chi_e - \chi_{e'} = c - c'}] = 1/|L|^2\,,\end{aligned}$$

where $\mathbf{1}_{\chi_e - \chi_{e'} = c - c'}$ is the indicator of the event $\chi_e - \chi_{e'} = c - c'$. Therefore, χ_e and $\chi_{e'}$ are independent. □

Theorem 8.4. *For every, $k \ge d \ge 2$, we have $f(k,d) \le 4kd$.*

Proof. Let $k' = 2^{\lceil \log_2 k \rceil} \in [k, 2k)$, and $d' = 2^{\lceil \log_2 d \rceil} \in [d, 2d)$. We apply Lemma 8.3 with parameters $p = 2$, k' and d'. We get that there are k' pairwise independent random variables taking values in a set of size d' defined on a sample space Ω of size $k'd'$. We choose k among these k' random variables (arbitrarily).

By Theorem 8.2, we have $f(k,d') \le |\Omega| = k'd' \le 4kd$. It was shown in [4] that the function $f(k,d)$ is monotone in d. Therefore, $f(k,d) \le f(k,d') \le 4kd$. □

9 Simple greedy algorithm for MAX-k-CSP$_d$

In this section we present a very simple approximation algorithm for MAX-k-CSP$_d$ with approximation guarantee $\Omega(d/d^k)$. This algorithm gives a better approximation than our algorithm from Theorem 5.1 when $k = O(\log d)$.

Theorem 9.1. *There is a* $(d/\mathrm{e})/d^k$ *approximation algorithm for* MAX-k-CSP$_d$.

Proof. Our algorithm consists of two steps. In the first step, for every $u \in V$,

- with probability $(k-1)/k$, we assign x_u a value $x'_u \in [d]$ uniformly at random;
- with probability $1/k$, we do not assign any value to x_u.

We get a partial assignment x'. In the second step, we assign values to unassigned variables. Let $\mathcal{P}'$ be the set of clauses $C \in \mathcal{P}$ such that exactly one variable in $\mathrm{supp}(C)$ is unassigned. Let $\mathcal{P}''$ be the subset of clauses in $\mathcal{P}'$ that are consistent with x'. Now we assign values to unassigned variables so as to maximize the number of satisfied clauses in $\mathcal{P}''$. Specifically, for every unassigned variable x_u, we find value i that maximizes $|\{C \in \mathcal{P}'' : (u,i) \in C\}|$ and assign $x'_u = i$. We obtain an assignment x'.

Let us lower bound the number of constraints satisfied by x'. Let x^* be an optimal assignment and $\mathcal{P}^*$ be the set of clauses x^* satisfies. Note that every clause C belongs to $\mathcal{P}'$ with probability $k \cdot 1/k \cdot (1-1/k)^{k-1} \geq 1/\mathrm{e}$. Every clause in $\mathcal{P}'$ belongs to $\mathcal{P}''$ with probability $1/d^{k-1}$. Therefore,

$$\mathbb{E}[|\mathcal{P}^* \cap \mathcal{P}''|] \geq \frac{1}{\mathrm{e}\, d^{k-1}} \cdot |\mathcal{P}^*| = \frac{1}{\mathrm{e}\, d^{k-1}} \cdot OPT.$$

Note that x^* satisfies at least $|\mathcal{P}^* \cap \mathcal{P}''|$ clauses in $\mathcal{P}''$ since it satisfies all clauses in $\mathcal{P}^* \cap \mathcal{P}''$. Since in the second step we assign values to x_u so as to maximize the number of satisfied clauses in $\mathcal{P}''$, we have that x' also satisfies at least $|\mathcal{P}^* \cap \mathcal{P}''|$ clauses in $\mathcal{P}''$. Thus in expectation x' satisfies at least

$$\mathbb{E}[|\mathcal{P}^* \cap \mathcal{P}''|] \geq \frac{1}{\mathrm{e}\, d^{k-1}} OPT$$

clauses. □

Acknowledgement

We would like to thank Johan Håstad for giving us a permission to present his hardness result for MAX-k-CSP$_d$ in this paper.

References

[1] PER AUSTRIN AND ELCHANAN MOSSEL: Approximation resistant predicates from pairwise independence. *Comput. Complexity*, 18(2):249–271, 2009. Preliminary version in CCC'08. [doi:10.1007/s00037-009-0272-6] 342, 354, 355

[2] SIU ON CHAN: Approximation resistance from pairwise independent subgroups. In *Proc. 45th STOC*, pp. 447–456. ACM Press, 2013. [doi:10.1145/2488608.2488665] 342

[3] MOSES CHARIKAR, KONSTANTIN MAKARYCHEV, AND YURY MAKARYCHEV: Near-optimal algorithms for unique games. In *Proc. 38th STOC*, pp. 205–214. ACM Press, 2006. [doi:10.1145/1132516.1132547] 343, 353

[4] MOSES CHARIKAR, KONSTANTIN MAKARYCHEV, AND YURY MAKARYCHEV: Near-optimal algorithms for maximum constraint satisfaction problems. *ACM Trans. Algorithms*, 5(32), 2009. Preliminary version in SODA'07. [doi:10.1145/1541885.1541893] 342, 343, 349, 350, 355

[5] EDEN CHLAMTÁČ, KONSTANTIN MAKARYCHEV, AND YURY MAKARYCHEV: How to play unique games using embeddings. In *Proc. 47th FOCS*, pp. 687–696. IEEE Comp. Soc. Press, 2006. [doi:10.1109/FOCS.2006.36] 343

[6] LARS ENGEBRETSEN: The nonapproximability of non-boolean predicates. *SIAM J. Discrete Math.*, 18(1):114–129, 2004. Preliminary version in RANDOM'01. [doi:10.1137/S0895480100380458] 342

[7] LARS ENGEBRETSEN AND JONAS HOLMERIN: More efficient queries in PCPs for NP and improved approximation hardness of maximum CSP. *Random Struct. Algorithms*, 33(4):497–514, 2008. Preliminary version in STACS'05. [doi:10.1002/rsa.20226] 342

[8] VENKATESAN GURUSWAMI AND PRASAD RAGHAVENDRA: Constraint satisfaction over a non-Boolean domain: Approximation algorithms and Unique-Games hardness. In *Proc. 12th Internat. Workshop on Randomization and Computation (RANDOM'08)*, pp. 77–90. Springer, 2008. [doi:10.1007/978-3-540-85363-3_7] 342

[9] GUSTAV HAST: Approximating Max kCSP — outperforming a random assignment with almost a linear factor. In *Proc. 32th Internat. Colloq. on Automata, Languages and Programming (ICALP'05)*, pp. 956–968. Springer, 2005. [doi:10.1007/11523468_77] 342

[10] SUBHASH KHOT: On the power of unique 2-prover 1-round games. In *Proc. 34th STOC*, pp. 767–775. ACM Press, 2002. [doi:10.1145/509907.510017] 343

[11] PRASAD RAGHAVENDRA: Optimal algorithms and inapproximability results for every CSP? In *Proc. 40th STOC*, pp. 245–254. ACM Press, 2008. [doi:10.1145/1374376.1374414] 342

[12] ALEX SAMORODNITSKY AND LUCA TREVISAN: A PCP characterization of NP with optimal amortized query complexity. In *Proc. 32nd STOC*, pp. 191–199. ACM Press, 2000. [doi:10.1145/335305.335329] 342

[13] ALEX SAMORODNITSKY AND LUCA TREVISAN: Gowers uniformity, influence of variables, and PCPs. In *Proc. 38th STOC*, pp. 11–20. ACM Press, 2006. [doi:10.1145/1132516.1132519] 342

[14] LUCA TREVISAN: Parallel approximation algorithms by positive linear programming. *Algorithmica*, 21(1):72–88, 1998. Preliminary version in ESA'96. [doi:10.1007/PL00009209] 342, 343

[15] ZBYNĚK ŠIDÁK: Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62(318):626–633, 1967. [doi:10.2307/2283989] 345

AUTHORS

Konstantin Makarychev
researcher
Microsoft Research, Redmond, WA
komakary@microsoft.com
http://www.cs.princeton.edu/~kmakaryc/

Yury Makarychev
assistant professor
Toyota Technological Institute at Chicago
Chicago, IL
yury@ttic.edu
http://ttic.uchicago.edu/~yury/

ABOUT THE AUTHORS

KONSTANTIN MAKARYCHEV, known as "Kostya" to his friends, is a Researcher at Microsoft Research. He received his Ph. D. from Princeton University under the supervision of Moses Charikar in 2008. From 2007 to 2012, he was a Research Staff Member at IBM Research. In 2012, Kostya joined Microsoft Research in Redmond, WA.

YURY MAKARYCHEV is an Assistant Professor at the Toyota Technological Institute at Chicago. He received his Ph. D. from Princeton University under the supervision of Moses Charikar in 2008. Yury did a postdoctoral fellowship at Microsoft Research and then joined TTIC in 2009.