58th Annual IEEE Symposium on Foundations of Computer Science

The power of sum-of-squares for detecting hidden structures

Samuel B. Hopkins
Cornell
Ithaca, USA
samhop@cs.cornell.edu

Pravesh K. Kothari
IAS/Princeton
Princeton, USA
kothari@cs.princeton.edu

Aaron Potechin
IAS
Princeton, USA
aaronpotechin@gmail.com

Prasad Raghavendra
UC Berkeley
Berkeley, USA
prasad@cs.berkeley.edu

Tselil Schramm
UC Berkeley
Berkeley, USA
tschramm@cs.berkeley.edu

David Steurer
Cornell/IAS
Princeton, USA
dsteure@cs.cornell.edu

***Abstract*—We study planted problems—finding hidden structures in random noisy inputs—through the lens of the sum-of-squares semidefinite programming hierarchy (SoS). This family of powerful semidefinite programs has recently yielded many new algorithms for planted problems, often achieving the best known polynomial-time guarantees in terms of accuracy of recovered solutions and robustness to noise. One theme in recent work is the design of spectral algorithms which match the guarantees of SoS algorithms for planted problems. Classical spectral algorithms are often unable to accomplish this: the twist in these new spectral algorithms is the use of spectral structure of matrices whose entries are low-degree polynomials of the input variables.**

We prove that for a wide class of planted problems, including refuting random constraint satisfaction problems, tensor and sparse PCA, densest-k-subgraph, community detection in stochastic block models, planted clique, and others, eigenvalues of degree-d matrix polynomials are as powerful as SoS semidefinite programs of degree d. For such problems it is therefore always possible to match the guarantees of SoS without solving a large semidefinite program.

Using related ideas on SoS algorithms and low-degree matrix polynomials (and inspired by recent work on SoS and the planted clique problem [BHK$^+$16]), we prove a new SoS lower bound for the tensor PCA problem.

***Keywords*-sum of squares, semidefinite programming, average-case algorithms, average-case hardness, spectral algorithms**

I. Introduction

Recent years have seen a surge of progress in algorithm design via the sum-of-squares (SoS) semidefinite programming hierarchy. Initiated by the work of [BBH$^+$12], who showed that polynomial time algorithms in the hierarchy solve all known integrality gap instances for Unique Games and related problems, a steady stream of works have developed efficient algorithms for both worst-case [BKS14], [BKS15], [BKS17], [BGG$^+$16] and average-case problems [HSS15], [GM15], [BM16], [RRS16], [BGL16], [MSS16a], [PS17]. The insights from these works extend beyond individual algorithms to characterizations of broad classes of algorithmic techniques. In addition, for a large class of problems (including constraint satisfaction), the family of SoS semidefinite programs is now known to be as powerful as *any* semidefinite program (SDP) [LRS15].

In this paper we focus on recent progress in using Sum of Squares algorithms to solve average-case, and especially *planted* problems—problems that ask for the recovery of a planted *signal* perturbed by random *noise*. Key examples are finding solutions of random constraint satisfaction problems (CSPs) with planted assignments [RRS16] and finding planted optima of random polynomials over the n-dimensional unit sphere [RRS16], [BGL16]. The latter formulation captures a wide range of unsupervised learning problems, and has led to many unsupervised learning algorithms with the best-known polynomial time guarantees [BKS15], [BKS14], [MSS16b], [HSS15], [PS17], [BGG$^+$16].

In many cases, classical algorithms for such planted problems are *spectral* algorithms—i.e., using the top eigenvector of a natural matrix associ-

0272-5428/17 $31.00 © 2017 IEEE
DOI 10.1109/FOCS.2017.72

ated with the problem input to recover a planted solution. The canonical algorithms for the *planted clique* [AKS98], *principal components analysis* (PCA) [Pea01], and *tensor decomposition* (which is intimately connected to optimizaton of polynomials on the unit sphere) [Har70] are all based on this general scheme. In all of these cases, the algorithm employs the top eigenvector of a matrix which is either given as input (the adjacency matrix, for planted clique), or is a simple function of the input (the empirical covariance, for PCA).

Recent works have shown that one can often improve upon these basic spectral methods using SoS, yielding better accuracy and robustness guarantees against noise in recovering planted solutions. Furthermore, for worst case problems—as opposed to the average-case planted problems we consider here—semidefinite programs are strictly more powerful than spectral algorithms.[1] *A priori* one might therefore expect that these new SoS guarantees for planted problems would not be achievable via spectral algorithms. But curiously enough, in numerous cases these stronger guarantees for planted problems can be achieved by spectral methods! The twist is that the entries of these matrices are low-degree polynomials in the input to the algorithm . The result is a new family of low-degree spectral algorithms with guarantees matching SoS but requriring only eigenvector computations instead of general semidefinite programming [HSSS16], [RRS16], [AOW15a].

This leads to the following question which is the main focus of this work. *Are SoS algorithms equivalent to low-degree spectral methods for planted problems?*

We answer this question affirmatively for a wide class of distinguishing problems which includes refuting random CSPs, tensor and sparse PCA, densest-k-subgraph, community detection in stochastic block models, planted clique, and more. Our positive answer to this question implies that a light-weight algorithm—computing the top eigenvalue of a single matrix whose entries are low-degree polynomials in the input—can recover the performance guarantees of an often bulky semidefinite programming relaxation. This is related to the recent work of Fan and Montanari [FM16] who showed that for some planted problems on sparse random graphs, a class of simple procedures called as *local algorithms* performs as well as semidefinite programming relaxations.

[1]For example, consider the contrast between the SDP algorithm for Max-Cut of Goemans and Williamson, [GW94], and the spectral algorithm of Trevisan [Tre09]; or the SDP-based algorithms for coloring worst-case 3-colorable graphs [KT17] relative to the best spectral methods [AK97] which only work for random inputs.

A. SoS and spectral algorithms for robust inference

Many planted problems have several formulations: *search*, in which the goal is to recover a planted solution, *refutation*, in which the goal is to certify that no planted solution is present, and *distinguishing*, where the goal is to determine with good probability whether an instance contains a planted solution or not. Often an algorithm for one version can be parlayed into algorithms for the others, but distinguishing problems are often the easiest, and we focus on them here.

A distinguishing problem is specified by two distributions on instances: a *planted* distribution supported on instances with a hidden structure, and a *uniform* distribution, where samples w.h.p. contain no hidden structure. Given an instance drawn with equal probability from the planted or the uniform distribution, the goal is to determine with probability greater than $\frac{1}{2}$ whether or not the instance comes from the planted distribution. For example:

Planted clique *Uniform distribution:* $G(n, \frac{1}{2})$, the Erdős-Renyi distribution, which w.h.p. contains no clique of size $\omega(\log n)$. *Planted distribution:* The uniform distribution on graphs containing a n^{ε}-size clique, for some $\varepsilon > 0$. (The problem gets harder as ε gets smaller, since the distance between the distributions shrinks.)

Planted 3XOR *Uniform distribution:* a 3XOR instance on n variables and $m > n$ equations $x_i x_j x_k = a_{ijk}$, where all the triples (i, j, k) and the signs $a_{ijk} \in \{\pm 1\}$ are sampled uniformly and independently. No assignment to x will satisfy more than a 0.51-fraction of the equations, w.h.p. *Planted distribution:* The same, except the signs a_{ijk} are sampled to correlate with $b_i b_j b_k$ for a randomly chosen $b_i \in \{\pm 1\}$, so that the assignment $x = b$ satisfies a 0.9-fraction of the equations. (The problem gets easier as m/n gets larger, and the contradictions in the uniform case become more locally apparent.)

We now formally define a family of distinguishing problems, in order to give our main theorem. Let $\mathcal{I}$ be a set of instances corresponding to a product space (for concreteness one may think of $\mathcal{I}$ to be the set of graphs on n vertices, indexed by $\{0,1\}^{\binom{n}{2}}$, although the theorem applies more broadly). Let ν, our uniform distrbution, be a product distribution on $\mathcal{I}$.

With some decision problem $\mathcal{P}$ in mind (e.g. does G contain a clique of size $\geqslant n^{\varepsilon}$?), let $\mathcal{X}$ be a set of solutions to $\mathcal{P}$; again for concreteness one may think of $\mathcal{X}$ as being associated with cliques in a graph, so that $\mathcal{X} \subset \{0,1\}^n$ is the set of all indicator vectors on at least n^{ε} vertices.

For each solution $x \in \mathcal{X}$, let $\mu_{|x}$ be the uniform distribution over instances $I \in \mathcal{I}$ that contain x. For example, in the context of planted clique, if x is a clique on vertices $1, \dots, n^{\varepsilon}$, then $\mu_{|x}$ would be the uniform distribution on graphs containing the clique $1, \dots, n^{\varepsilon}$. We define the planted distribution μ to be the uniform mixture over μ_x, $\mu = U_{x \sim \mathcal{X}} \mu_{|x}$.

The following is our main theorem on the equivalence of sum of squares algorithms for distinguishing problems and spectral algorithms employing low-degree matrix polynomials.

Theorem (Informal)**.** *Let $N, n \in \mathcal{N}$, and let $\mathcal{A}, \mathcal{B}$ be sets of real numbers. Let $\mathcal{I}$ be a family of instances over $\mathcal{A}^N$, and let $\mathcal{P}$ be a decision problem over $\mathcal{I}$ with $\mathcal{X} = \mathcal{B}^n$ the set of possible solutions to $\mathcal{P}$ over $\mathcal{I}$. Let $\{g_j(x, I)\}$ be a system of $n^{O(d)}$ polynomials of degree at most d in the variables x and constant degree in the variables $\mathcal{I}$ that encodes $\mathcal{P}$, so that*

- *for $I \sim_{\nu} \mathcal{I}$, with high probability the system is unsatisfiable and admits a degree-d SoS refutation, and*
- *for $I \sim_{\mu} \mathcal{I}$, with high probability the system is satisfiable by some solution $x \in X$, and x remains feasible even if all but an $n^{-0.01}$-fraction of the coordinates of $\mathcal{I}$ are re-randomized according to ν.*

Then there exists a matrix whose entries are degree-$O(d)$ polynomials $Q : \mathcal{I} \to \mathbb{R}^{\binom{n}{\leqslant d} \times \binom{n}{\leqslant d}}$ such that

$$\mathop{\mathbb{E}}_{I \sim \nu} \left[\lambda^{+}_{max}(Q(I))\right] \leqslant 1, \text{ while } \mathop{\mathbb{E}}_{I \sim \mu} \left[\lambda^{+}_{max}(Q(I))\right] \geqslant n^{10d},$$

where $\lambda^{+}_{\max}$ denotes the maximum non-negative eigenvalue.

The condition that a solution x remain feasible if all but a fraction of the coordinates of $I \sim \mu_{|x}$ are re-randomized should be interpreted as a noise-robustness condition. To see an example, in the context of planted clique, suppose we start with a planted distribution over graphs with a clique x of size $n^{\varepsilon+0.01}$. If a random subset of $n^{0.99}$ vertices are chosen, and all edges not entirely contained in that subset are re-randomized according to the $G(n, 1/2)$ distribution, then with high probability at least n^{ε} of the vertices in x remain in a clique, and so x remains feasible for the problem $\mathcal{P}$: G has a clique of size $\geqslant n^{\varepsilon}$?

B. SoS and information-computation gaps

Computational complexity of planted problems has become a rich area of study. The goal is to understand which planted problems admit efficient (polynomial time) algorithms, and to study the *information-computation gap* phenomenon: many problems have noisy regimes in which planted structures can be found by inefficient algorithms, but (conjecturally) not by polynomial time algorithms. One example is the *planted clique* problem, where the goal find a large clique in a sample from the uniform distribution over graphs containing a clique of size n^{ε} for a small constant $\varepsilon > 0$. While the problem is solvable for any $\varepsilon > 0$ by a brute-force algorithm requiring $n^{\Omega(\log n)}$ time, polynomial time algorithms are conjectured to require $\varepsilon \geqslant \frac{1}{2}$.

A common strategy to provide evidence for such a gap is to prove that powerful classes of efficient algorithms are unable to solve the planted problem in the (conjecturally) hard regime. SoS algorithms are particularly attractive targets for such lower bounds because of their broad applicability and strong guarantees.

In a recent work, Barak et al. [BHK+16] show an SoS lower bound for the planted clique problem, demonstrating that when $\varepsilon < \frac{1}{2}$, SoS algorithms require $n^{\Omega(\log n)}$ time to solve planted clique. Intriguingly, they show that in the case of planted clique that SoS algorithms requiring $\approx n^d$ time can distinguish planted from random graphs only when there is a *scalar-valued* degree $\approx d \cdot \log n$ polynomial $p(A) : \mathbb{R}^{n \times n} \to \mathbb{R}$ (here A is the adjacency matrix of a graph) with

$$\mathop{\mathbb{E}}_{G(n,1/2)} p(A) = 0$$

$$\mathop{\mathbb{E}}_{\text{planted}} p(A) \geqslant n^{\Omega(1)} \cdot \left(\mathop{\mathbb{V}}_{G(n,1/2)} p(A) \right)^{1/2}.$$

That is, such a polynomial p has much larger expectation in under the planted distribution than its standard deviation in uniform distribution. (The choice of $n^{\Omega(1)}$ is somewhat arbitrary, and could be replaced with $\Omega(1)$ or $n^{\Omega(d)}$ with small changes in the parameters.) By showing that as long as $\varepsilon < \frac{1}{2}$ any such polynomial p must have degree $\Omega(\log n)^2$, they rule out efficient SoS algorithms when $\varepsilon < \frac{1}{2}$. Interestingly, this matches the spectral distinguishing threshold—the spectral algorithm of [AKS98] is known to work when $\varepsilon \geqslant \frac{1}{2}$.

This stronger characterization of SoS for the planted clique problem, in terms of *scalar* distinguishing algorithms rather than *spectral* distinguishing algorihtms, may at first seem insignificant. To see why the scalar characterization is more powerful, we point out that if the degree-d moments of the planted and uniform distributions are known, determining the optimal scalar distinguishing polynomial is easy: given a planted distribution μ and a random distribution ν over instances $\mathcal{I}$, one just solves a linear algebra problem in the $n^{d \log n}$ coefficients of p to maximize the expectation over μ relative to ν:

$$\max_{p} \mathop{\mathbb{E}}_{\mathcal{I} \sim \mu} [p^2(\mathcal{I})] \quad s.t. \quad \mathop{\mathbb{E}}_{\mathcal{I} \sim \nu} [p^2(\mathcal{I})] = 1 \,.$$

It is not difficult to show that the optimal solution to the above program has a simple form: it is the projection of the *relative density of ν with respect to μ* projected to the degree-$d \log n$ polynomials. So given a pair of distributions μ, ν, in $n^{O(d \log n)}$ time, it is possible to determine whether there exists a degree-$d \log n$ scalar distinguishing polynomial. Answering the same question about the existence of a spectral distinguisher is more complex, and to the best of our knowledge cannot be done efficiently.

Given this powerful theorem for the case of the planted clique problem, one may be tempted to conjecture that this stronger, *scalar* distinguisher characterization of the SoS algorithm applies more broadly than just to the planted clique problem, and perhaps as broadly as Theorem I-A. If this conjecture is true, given a pair of distributions ν and μ with known moments, we would be able to efficiently determine whether polynomial-time SoS distinguishing algorithms exist!

Conjecture I.1. *In the setting of Theorem I-A, the conclusion may be replaced with the conclusion that there exists a scalar-valued polynomial $p : \mathcal{I} \to \mathbb{R}$ of degree $O(d \cdot \log n)$ so that*

$$\mathop{\mathbb{E}}_{uniform} p(I) = 0 \text{ and } \mathop{\mathbb{E}}_{planted} p(I) \geqslant n^{\Omega(1)} \left(\mathop{\mathbb{E}}_{uniform} p(I)^2 \right)^{1/2}$$

To illustrate the power of this conjecture, in the beginning of Section **??** we give a short and self-contained explanation of how this predicts, via simple linear algebra, our $n^{\Omega(1)}$-degree SoS lower bound for tensor PCA. As evidence for the conjecture, we verify this prediction by proving such a lower bound unconditionally.

C. Lower bounds for Tensor PCA

The second main result of this paper is a strong exponential lower bound on the sum-of-squares method (specifically, against $2^{n^{o(1)}}$ time or $n^{o(1)}$ degree algorithms) for the tensor *principle component analysis* problem introduced by [RM14].

Problem I.2 (Tensor PCA)**.** Given an order-k tensor in $(\mathbb{R}^n)^{\otimes k}$, determine whether it comes from:

- **Uniform Distribution**: each entry of the tensor sampled independently from $\mathcal{N}(0, 1)$.
- **Planted Distribution**: a spiked tensor, $\mathbf{T} = \lambda \cdot v^{\otimes k} + G$ where v is sampled uniformly from $\mathbb{S}^{n-1}$, and where G is a random tensor with each entry sampled independently from $\mathcal{N}(0, 1)$.

Here, v can be thought of as "signal" surrounded by random gaussian noise. The parameter λ can be thought of as controlling the strength of the signal. In particular, as λ grows, we expect the distinguishing problem above to get easier.

Tensor PCA (we restrict to the special case of $k = 3$ for this discussion for simplicity) is a natural generalization of the PCA problem in machine learning: given a $n \times n$ matrix M, distinguish between the case where every entry of M is independently drawn from the standard gaussian distribution $\mathcal{N}(0, 1)$ and the case when M is drawn from a distritbution as above with an added rank 1 shift $\lambda v v^\top$ in a uniformly random direction v. A natural and well-studied distinguisher here is the largest singular value/spectral norm of the input matrix. Equivalently, it is the maximizer of the degree two polynomial $\langle x, Mx \rangle$ in $x \in \mathbb{S}^{n-1}$.

A natural generalization of the above distinguisher for tensor PCA is the maximum of the

degree 3 polynomial $\langle T, x^{\otimes 3}\rangle$ over the unit sphere - or equivalently, the (symmetric) injective tensor norm of T. This maximum can be shown to be much larger in case of the planted distribution so long as $\lambda \gg \sqrt{n}$. Tensor PCA can thus be thought of as an instance of the problem of optimizing a random polynomial over the unit sphere.

While the PCA problem—maximizing a degree-2 polynomial over the unit sphere—has efficient algorithms, the corresponding problem for 3-tensors is NP-hard, even to approximate [HL09], [BBH$^+$12], so it is often studied in an average-case context and known algorithms show efficiently computable relaxations for the associated degree 3 polynomial optimization problem. Sum-of-Squares method is a natural algorithm in this context and yields state of the art algorithms for the problem. Specifically, polynomial-time SoS algorithm is known to recover the vector v and whenever $\lambda \gg n^{3/4}$ [HSS15]. A major open question in this direction is to understand the complexity of the problem for $\lambda \leqslant n^{3/4-\varepsilon}$. Algorithms (captured by SoS) are known which run in $2^{n^{O(\varepsilon)}}$ time [RRS16], [BGG$^+$16].

We show the following theorem which shows that the sub-exponential algorithm above is in fact nearly optimal for SoS algorithm.

Theorem I.3. *For a tensor T, let*

$$SoS_d(T) = \max_{\widetilde{\mathbb{E}}} \widetilde{\mathbb{E}}[\langle T, x^{\otimes k}\rangle]$$

where the maximum is taken over degree d pseudoexpectations $\widetilde{\mathbb{E}}$ satisfying $\{\|x\|^2 = 1\}$.[2] For every small enough constant $\varepsilon > 0$, if $T \in \mathbb{R}^{n\times n\times n}$ has iid Gaussian or $\{\pm 1\}$ entries, $SoS_d(T) \geqslant n^{k/4-\varepsilon}$ with probability $1 - o(1)$, for every $d \leqslant n^{c\cdot\varepsilon}$ for some universal $c > 0$.

In particular for third order tensors (i.e $k = 3$), since degree $n^{\Omega(\varepsilon)}$ SoS is unable to certify that a random 3-tensor has maximum value much less than $n^{3/4-\varepsilon}$, this SoS relaxation cannot be used to distinguish the planted and random distributions above when $\lambda \ll n^{3/4-\varepsilon}$.[3]

Our proof of Theorem I.3 generalizes and abstracts out the machinery developed in the recent work on proving a tight lower bound for planted clique problem [BHK$^+$16]. In order to demonstrate the power of Conjecture I.1 by observing that it implies the precise thresholds obtained in the theorem above.

D. Related work

By now, there's a large body of work that establishes lower bounds on SoS SDP for various average case problems. Beginning with the work of Grigoriev [Gri01a], a long line work have established tight lower bounds for random constraint satisfaction problems [Sch08], [BCK15], [KMOW17] and planted clique [MPW15], [DM15], [HKP15], [RS15], [BHK$^+$16]. The recent SoS lower bound for planted clique of [BHK$^+$16] was particularly influential to this work, setting the stage for our main line of inquiry. We also draw attention to previous work on lower bounds for the tensor PCA and sparse PCA problems in the degree-4 SoS relaxation [HSS15], [MW15]—our paper improves on this and extends our understanding of lower bounds for tensor and sparse PCA to any degree.

Tensor principle component analysis was introduced by Montanari and Richard [RM14] who indentified information theoretic threshold for recovery of the planted component and analyzed the maximum likelihood estimator for the problem. The work of [HSS15] began the effort to analyze the sum of squares method for the problem and showed that it yields an efficient algorithm for recovering the planted component with strength $\tilde{\omega}(n^{3/4})$. They also established that this threshold is tight for the sum of squares relaxation of degree 4. Following this, Hopkins et al. [HSSS16] showed how to extract a linear time spectral algorithm from the above analysis. Tomioka and Suzuki derived tight information theoretic thresholds for detecting planted components by establishing tight bounds on the injective tensor norm of random tensors [TS14]. Finally, very recently, Raghavendra et. al. and Bhattiprolu et. al. independently showed sub-exponential time algorithms for tensor pca [RRS16], [BGL16]. Their algorithms are spectral and are captured by the sum of squares method.

As alluded to above, many prior works explore the connection between SoS relaxations and spectral algorithms, beginning with the work of [BBH$^+$12] and including the followup works [HSS15], [AOW15b], [BM16] (plus many more).

[2]For definitions of pseudoexpectations and related matters, see the survey [BS14].

[3]In fact, our proof for this theorem will show somewhat more: that a large family of constraints—any valid constraint which is itself a low-degree polynomial of T—could be added to this convex relaxation and the lower bound would still obtain.

Of particular interest are the papers [HSSS16], [MS16b], which use the SoS algorithms to obtain *fast* spectral algorithms, in some cases running in time linear in the input size (smaller even than the number of variables in the associated SoS SDP).

In light of our Theorem I-A, it is particularly interesting to note cases in which the known SoS lower bounds matching the known spectral algorithms—these problems include planted clique (upper bound: [AKS98], lower bound:[4] [BHK+16]), strong refutations for random CSPs (upper bound:[5] [AOW15b], [RRS16], lower bounds: [Gri01b], [Sch08], [KMOW17]), and tensor principal components analysis (upper bound: [HSS15], [RRS16], [BGG+16], lower bound: this paper).

We also remark that our work applies to several previously-considered distinguishing and average-case problems within the sum-of-squares algorithmic framework: block models [MS16a] , densest-k-subgraph [BCC+10]; for each of these problems, we have by Theorem I-A an equivalence between efficient sum-of-squares algorithms and efficient spectral algorithms, and it remains to establish exactly what the tradeoff is between efficiency of the algorithm and the difficulty of distinguishing, or the strength of the noise.

Notation.: For two matrices A, B, let $\langle A, B\rangle \stackrel{\text{def}}{=} \mathrm{Tr}(AB)$. Let $\|A\|_{Fr}$ denote the Frobenius norm, and $\|A\|$ its spectral norm. For matrix valued functions A, B over $\mathcal{I}$ and a distribution ν over $\mathcal{I} \sim \mathcal{I}$, we will denote $\langle A, B\rangle_\nu = \mathbb{E}_{\mathcal{I}\sim\nu}\langle A(\mathcal{I}), B(\mathcal{I})\rangle$ and by $\|A\|_{Fr,\nu} \stackrel{\text{def}}{=} (\mathbb{E}_{\mathcal{I}\sim\nu}\langle A(\mathcal{I}), A(\mathcal{I})\rangle)^{1/2}$.

For a vector of formal variables $x = (x_1, \ldots, x_n)$, we use $x^{\leqslant d}$ to denote the vector consisting of all monomials of degree at most d in these variables. Furthermore, let us denote $X^{\leqslant d} \stackrel{\text{def}}{=} (x^{\leqslant d})(x^{\leqslant d})^T$.

Organization.: The remainder of this proceedings version is devoted to stating formally and sketching parts of the proof of Theorem I-A. We defer the full proof, as well as the proof of Theorem I.3, to the full version of this paper.

[4]SDP lower bounds for the planted clique problem were known for smaller degrees of sum-of-squares relaxations and for other SDP relaxations before; see the references therein for details.

[5]There is a long line of work on algorithms for refuting random CSPs, and 3SAT in particular; the listed papers contain additional references.

II. Distinguishing Problems and Robust Inference

In this section, we set up the formal framework within which we will prove our main result.

Uniform vs. Planted Distinguishing Problems: We begin by describing a class of *distinguishing* problems. For $\mathcal{A}$ a set of real numbers, we will use $\mathcal{I} = \mathcal{A}^N$ denote a space of instances indexed by N variables—for the sake of concreteness, it will be useful to think of $\mathcal{I}$ as $\{0,1\}^N$; for example, we could have $N = \binom{n}{2}$ and $\mathcal{I}$ as the set of all graphs on n vertices. However, the results that we will show here continue to hold in other contexts, where the space of all instances is $\mathbb{R}^N$ or $[q]^N$.

Definition II.1 (Uniform Distinguishing Problem)**.** Suppose that $\mathcal{I}$ is the space of all instances, and suppose we have two distributions over $\mathcal{I}$, a product distribution ν (the "uniform" distribution), and an arbitrary distribution μ (the "planted" distribution).

In a *uniform distinguishing problem*, we are given an instance $\mathcal{I} \in \mathcal{I}$ which is sampled with probability $\frac{1}{2}$ from ν and with probability $\frac{1}{2}$ from μ, and the goal is to determine with probability greater than $\frac{1}{2} + \varepsilon$ which distribution $\mathcal{I}$ was sampled from, for any constant $\varepsilon > 0$.

Polynomial Systems: In the uniform distinguishing problems that we are interested in, the planted distribution μ will be a distribution over instances that obtain a large value for some optimization problem of interest (i.e. the max clique problem). We define polynomial systems in order to formally capture optimization problems.

Program II.2 (Polynomial System)**.** Let $\mathcal{A}, \mathcal{B}$ be sets of real numbers, let $n, N \in \mathbb{N}$, and let $\mathcal{I} = \mathcal{A}^N$ be a space of instances and $\mathcal{X} \subseteq \mathcal{B}^n$ be a space of solutions. A *polynomial system* is a set of polynomial equalities

$$g_j(x, \mathcal{I}) = 0 \quad \forall j \in [m],$$

where $\{g_j\}_{j=1}^m$ are polynomials in the *program variables* $\{x_i\}_{i\in[n]}$, representing $x \in \mathcal{X}$, and in the *instance variables* $\{\mathcal{I}_j\}_{j\in[N]}$, representing $\mathcal{I} \in \mathcal{I}$. We define $\deg_{\text{prog}}(g_j)$ to be the degree of g_j in the program variables, and $\deg_{\text{inst}}(g_j)$ to be the degree of g_j in the instance variables.

Remark II.3. For the sake of simplicity, the polynomial system Program II.2 has no inequalities.

Inequalities can be incorporated in to the program by converting each inequality in to an equality with an additional slack variable. Our main theorem still holds, but for some minor modifications of the proof, as outlined in the full version of this paper.

A polynomial system allows us to capture problem-specific objective functions as well as problem-specific constraints. For concreteness, consider a quadtratic program which checks if a graph on n vertices contains a clique of size k. We can express this with the polynomial system over program variables $x \in \mathbb{R}^n$ and instance variables $\mathcal{I} \in \{0,1\}^{\binom{n}{2}}$, where $\mathcal{I}_{ij} = 1$ iff there is an edge from i to j, as follows:

$$\begin{aligned}&\left\{\sum\nolimits_{i\in[n]} x_i - k = 0\right\}\\ &\cup \{x_i(x_i-1)=0\}_{i\in[n]}\\ &\cup \{(1-\mathcal{I}_{ij})x_i x_j = 0\}_{i,j\in\binom{[n]}{2}}.\end{aligned}$$

Planted Distributions: We will be concerned with planted distributions of a particular form; first, we fix a polynomial system of interest $\mathcal{S} = \{g_j(x,\mathcal{I})\}_{j\in[m]}$ and some set $\mathcal{X} \subseteq \mathcal{B}^n$ of feasible solutions for $\mathcal{S}$, so that the program variables x represent elements of $\mathcal{X}$. Again, for concreteness, if $\mathcal{I}$ is the set of graphs on n vertices, we can take $\mathcal{X} \subseteq \{0,1\}^n$ to be the set of indicators for subsets of at least n^ε vertices.

For each fixed $x \in \mathcal{X}$, let $\mu_{|x}$ denote the uniform distribution over $\mathcal{I} \in \mathcal{I}$ for which the polynomial system $\{g_j(x,\mathcal{I})\}_{j\in[m]}$ is feasible. The planted distribution μ is given by taking the uniform mixture over the $\mu_{|x}$, i.e., $\mu \sim U_{x\sim\mathcal{X}}[\mu_{|x}]$.

SoS Relaxations: If we have a polynomial system $\{g_j\}_{j\in[m]}$ where $\deg_{\text{prog}}(g_j) \leqslant 2d$ for every $j \in [m]$, then the degree-$2d$ sum-of-squares SDP relaxation for the polynomial system Program II.2 can be written as,

Program II.4 (SoS Relaxation for Polynomial System)**.** Let $\mathcal{S} = \{g_j(x,\mathcal{I})\}_{j\in[m]}$ be a polynomial system in instance variables $\mathcal{I} \in \mathcal{I}$ and program variables $x \in \mathcal{X}$. If $\deg_{\text{prog}}(g_j) \leqslant 2d$ for all $j \in [m]$, then an *SoS relaxation* for $\mathcal{S}$ is

$$\begin{aligned}&\langle G_j(\mathcal{I}), X\rangle = 0 \quad \forall j \in [m]\\ &X \succeq 0\end{aligned}$$

where X is an $[n]^{\leqslant d} \times [n]^{\leqslant d}$ matrix containing the variables of the SDP and $G_j : \mathcal{I} \to \mathbb{R}^{[n]^{\leqslant d}\times[n]^{\leqslant d}}$ are matrices containing the coefficients of $g_j(x,\mathcal{I})$ in x, so that the constraint $\langle G_j(\mathcal{I}), X\rangle = 0$ encodes the constraint $g_j(x,\mathcal{I}) = 0$ in the SDP variables. Note that the entries of G_j are polynomials of degree at most $\deg_{\text{inst}}(g_j)$ in the instance variables.

Sub-instances: Suppose that $\mathcal{I} = \mathcal{A}^N$ is a family of instances; then given an instance $\mathcal{I} \in \mathcal{I}$ and a subset $S \subseteq [N]$, let $\mathcal{I}_S$ denote the sub-instance consisting of coordinates within S. Further, for a distribution Θ over subsets of $[N]$, let $\mathcal{I}_S \sim_\Theta \mathcal{I}$ denote a subinstance generated by sampling $S \sim \Theta$. Let $\mathcal{I}_\downarrow$ denote the set of all sub-instances of an instance $\mathcal{I}$, and let $\mathcal{I}_\downarrow$ denote the set of all sub-instances of all instances.

Robust Inference: Our result will pertain to polynomial systems that define planted distributions whose solutions to sub-instances generalize to feasible solutions over the entire instance. We call this property "robust inference."

Definition II.5. Let $\mathcal{I} = \mathcal{A}^N$ be a family of instances, let Θ be a distribution over subsets of $[N]$, let $\mathcal{S}$ be a polynomial system as in Program II.2, and let μ be a planted distribution over instances feasible for $\mathcal{S}$. Then the polynomial system $\mathcal{S}$ is said to satisfy the *robust inference property for probability distribution μ on $\mathcal{I}$ and subsampling distribution* Θ, if given a subsampling $\mathcal{I}_S$ of an instance $\mathcal{I}$ from μ, one can infer a setting of the program variables x^* that remains feasible to $\mathcal{S}$ for most settings of $\mathcal{I}_{\overline{S}}$.

Formally, there exists a map $x : \mathcal{I}_\downarrow \to \mathbb{R}^n$ such that

$$\begin{aligned}&\mathbb{P}_{\mathcal{I}\sim\mu, S\sim\Theta, \tilde{\mathcal{I}}\sim\nu_{|\mathcal{I}_S}}[x(\mathcal{I}_S) \text{ is a feasible for } \mathcal{S} \text{ on } \mathcal{I}_S \circ \tilde{\mathcal{I}}]\\ &\geqslant 1 - \varepsilon(n,d)\end{aligned}$$

for some negligible function $\varepsilon(n,d)$. To specify the error probability, we will say that polynomial system is $\varepsilon(n,d)$*-robustly inferable*.

Main Theorem: We are now ready to state our main theorem.

Theorem II.6. *Suppose that $\mathcal{S}$ is a polynomial system as defined in Program II.2, of degree at most $2d$ in the program variables and degree at most k in the instance variables. Let $B > d \cdot k \in \mathbb{N}$ such that*

1) *The polynpomial system $\mathcal{S}$ is $\frac{1}{n^{8B}}$-robustly inferable with respect to the planted distribution μ and the sub-sampling distribution Θ.*

2) *For $\mathcal{I} \sim \nu$, the polynomial system $\mathcal{S}$ admits a degree-d SoS refutation with numbers bounded by n^B with probability at least $1 - \frac{1}{n^{8B}}$.*

Let $D \in \mathbb{N}$ be such that for any subset $\alpha \subseteq [N]$ with $|\alpha| \geqslant D - 2dk$,

$$\mathop{\mathbb{P}}_{S\sim\Theta}[\alpha \subseteq S] \leqslant \frac{1}{n^{8B}}$$

There exists a degree $2D$ matrix polynomial $Q : \mathcal{I} \to \mathbb{R}^{[n]^{\leqslant d}\times[n]^{\leqslant d}}$ such that,

$$\frac{\mathbb{E}_{\mathcal{I}\sim\mu}[\lambda^+_{max}(Q(\mathcal{I}))]}{\mathbb{E}_{\mathcal{I}\sim\nu}[\lambda^+_{max}(Q(\mathcal{I}))]} \geqslant n^{B/2}$$

Remark II.7. Our argument implies a stronger result that can be stated in terms of the eigenspaces of the subsampling operator. Specifically, suppose we define

$$\mathcal{S}_\varepsilon \stackrel{\text{def}}{=} \left\{\alpha \mid \mathop{\mathbb{P}}_{S\sim\Theta}\{\alpha \subseteq S\} \leqslant \varepsilon\right\}$$

Then, the distinguishing polynomial exhibited by Theorem II.6 satisfies $Q \in \text{span}\{\text{ monomials } \mathcal{I}_\alpha | \alpha \in \mathcal{S}_\varepsilon\}$. This refinement can yield tighter bounds in cases where all monomials of a certain degree are not equivalent to each other. For example, in the Planted Clique problem, each monomial consists of a subgraph and the right measure of the degree of a sub-graph is the number of vertices in it, as opposed to the number of edges in it.

In the full version of this paper, we make the routine verifications that the conditions of this theorem hold for a variety of distinguishing problems: planted clique, refuting random CSPs, stochastic block models, densest-k-subgraph, tensor PCA, and sparse PCA. Now we will proceed to prove the theorem.

III. Moment-Matching Pseudodistributions

We assume the setup from Section II: we have a family of instances $\mathcal{I} = \mathcal{A}^N$, a polynomial system $\mathcal{S} = \{g_j(x, \mathcal{I})\}_{j\in[m]}$ with a family of solutions $\mathcal{X} = \mathcal{B}^n$, a "uniform" distribution ν which is a product distribution over $\mathcal{I}$, and a "planted" distribution μ over $\mathcal{I}$ defied by the polynomial system $\mathcal{S}$ as described in Section II.

The contrapositive of Theorem II.6 is that if $\mathcal{S}$ is robustly inferable with respect to μ and a distribution over sub-instances Θ, and if there is no spectral algorithm for distinguishing μ and ν, then with high probability there is no degree-d SoS refutation for the polynomial system $\mathcal{S}$ (as defined in Program II.4). To prove the theorem, we will use duality to argue that if no spectral algorithm exists, then there must exist an object which is in some sense close to a feasible solution to the SoS SDP relaxation.

Since each $\mathcal{I}$ in the support of μ is feasible for $\mathcal{S}$ by definition, a natural starting point is the SoS SDP solution for instances $\mathcal{I} \sim_\mu \mathcal{I}$. With this in mind, we let $\Lambda : \mathcal{I} \to (\mathbb{R}^{[n]^{\leqslant d}\times[n]^{\leqslant d}})_+$ be an arbitrary function from the support of μ over $\mathcal{I}$ to PSD matrices. In other words, we take

$$\Lambda(\mathcal{I}) = \hat{\mu}(\mathcal{I}) \cdot M(\mathcal{I})$$

where $\hat{\mu}$ is the relative density of μ with respect to ν, so that $\hat{\mu}(\mathcal{I}) = \mu(\mathcal{I})/\nu(\mathcal{I})$, and M is some matrix valued function such that $M(\mathcal{I}) \succeq 0$ and $\|M(\mathcal{I})\| \leqslant B$ for all $\mathcal{I} \in \mathcal{I}$. Our goal is to find a PSD matrix-valued function P that matches the low-degree moments of Λ in the variables $\mathcal{I}$, while being supported over most of $\mathcal{I}$ (rather than just over the support of μ).

The function $P : \mathcal{I} \to (\mathbb{R}^{[n]^{\leqslant d}\times[n]^{\leqslant d}})_+$ is given by the following exponentially large convex program over matrix-valued functions,

Program III.1 (Pseudodistribution Program)**.**

$$\min \quad \|P\|^2_{Fr,\nu} \tag{III.1}$$

$$s.t. \quad \langle Q, P\rangle_\nu = \langle Q, \Lambda'\rangle_\nu \tag{III.2}$$

$$\forall Q : \mathcal{I} \to \mathbb{R}^{[n]^{\leqslant d}\times[n]^{\leqslant d}},\ \deg_{\text{inst}}(Q) \leqslant D \tag{III.3}$$

$$P \succeq 0$$

$$\Lambda' = \Lambda + \eta \cdot \text{Id}, \quad 2^{-2^{2^n}} > \eta > 0 \tag{III.4}$$

The constraint (III.3) fixes $\mathbb{E}\,\text{Tr}(P)$, and so the objective function (III.1) can be viewied as minimizing $\mathbb{E}\,\text{Tr}(P^2)$, a proxy for the collision probability of the distribution, which is a measure of entropy.

Remark III.2. We have perturbed Λ in (III.4) so that we can easily show that strong duality holds in the proof of Claim III.4. For the remainder of the paper we ignore this perturbation, as we can accumulate the resulting error terms and set η to be small enough so that they can be neglected.

The dual of the above program will allow us to relate the existence of an SoS refutation to the existence of a spectral algorithm.

Program III.3 (Low-Degree Distinguisher)**.**

$$\begin{aligned} \max \quad & \langle \Lambda, Q \rangle_\nu \\ s.t. \quad & Q : \mathscr{I} \to \mathbb{R}^{[n]^{\leqslant d} \times [n]^{\leqslant d}},\ \deg_{\text{inst}}(Q) \leqslant D \\ & \|Q_+\|^2_{Fr,\nu} \leqslant 1, \end{aligned}$$

where Q_+ is the projection of Q to the PSD cone.

Claim III.4. Program III.3 is a manipulation of the dual of Program III.1, so that if Program III.1 has optimum $c > 1$, Program III.3 as optimum at least $\Omega(\sqrt{c})$.

Before we present the proof of the claim, we summarize its central consequence in the following theorem: if Program III.1 has a large objective value (and therefore does not provide a feasible SoS solution), then there is a spectral algorithm.

Theorem III.5. *Fix a function $M : \mathscr{I} \to \mathbb{R}_+^{[n]^{\leqslant d} \times [n]^{\leqslant d}}$ be such that* $\mathrm{Id} \succeq M \succeq 0$. *Let $\lambda^+_{\max}(\cdot)$ be the function that gives the largest non-negative eigenvalue of a matrix. Suppose $\Lambda = \mu \cdot M$ then the optimum of Program III.1 is equal to* $\mathrm{opt} > 1$ *only if there exists a low-degree matrix polynomial Q such that,*

$$\mathop{\mathbb{E}}_{\mathcal{I} \sim \mu} [\lambda^+_{max}(Q(\mathcal{I}))] \geqslant \Omega(\sqrt{\mathrm{opt}}/n^d)$$

while,

$$\mathop{\mathbb{E}}_{\mathcal{I} \sim \nu} [\lambda^+_{max}(Q(\mathcal{I}))] \leqslant 1\,.$$

Proof: By Claim III.4, if the value of Program III.1 is $\mathrm{opt} > 1$, then there is a polynomial Q achieves a value of $\Omega(\sqrt{\mathrm{opt}})$ for the dual. It follows that

$$\begin{aligned} & \mathop{\mathbb{E}}_{\mathcal{I} \sim \mu} [\lambda^+_{max}(Q(\mathcal{I}))] \\ & \geqslant \frac{1}{n^d} \mathop{\mathbb{E}}_{\mathcal{I} \sim \mu} [\langle \mathrm{Id}, Q(\mathcal{I}))\rangle] \\ & \geqslant \frac{1}{n^d} \langle \Lambda, Q \rangle_\nu \\ & = \Omega(\sqrt{\mathrm{opt}}/n^d), \end{aligned}$$

while

$$\begin{aligned} & \mathop{\mathbb{E}}_{\mathcal{I} \sim \nu} [\lambda^+_{max}(Q(\mathcal{I}))] \\ & \leqslant \sqrt{\mathop{\mathbb{E}}_{\mathcal{I} \sim \nu} [\lambda^+_{\max}(Q(\mathcal{I}))^2]} \\ & \leqslant \sqrt{\mathop{\mathbb{E}}_{\mathcal{I} \sim \nu} \|Q_+(\mathcal{I})\|^2_{Fr}} \leqslant 1. \end{aligned}$$

■

It is interesting to note that the specific structure of the PSD matrix valued function M plays no role in the above argument—since M serves as a proxy for monomials in the solution as represented by the program variables $x^{\otimes d}$, it follows that the choice of how to represent the planted solution is not critical. Although seemingly counterintuitive, this is natural because the property of being distinguishable by low-degre distinguishers or by SoS SDP relaxations is a property of ν and μ.

We wrap up the section by presenting a proof of the Claim III.4.

Proof of Claim III.4: We take the Lagrangian dual of Program III.1. Our dual variables will be some combination of low-degree matrix polynomials, Q, and a PSD matrix A:

$$\mathcal{L}(P, Q, A) = \|P\|^2_{Fr,\nu} - \langle Q, P - \Lambda' \rangle_\nu - \langle A, P \rangle_\nu \text{ s.t. } A \succeq 0.$$

It is easy to verify that if P is not PSD, then A can be chosen so that the value of $\mathcal{L}$ is ∞. Similarly if there exists a low-degree polynomial upon which P and Λ differ in expectation, Q can be chosen as a multiple of that polynomial so that the value of $\mathcal{L}$ is ∞.

Now, we argue that Slater's conditions are met for Program III.1, as $P = \Lambda'$ is strictly feasible. Thus strong duality holds, and therefore

$$\min_P \max_{A \geq 0, Q} \mathcal{L}(P, Q, A) \leqslant \max_{A \geq 0, Q} \min_P \mathcal{L}(P, Q, A).$$

Taking the partial derivative of $\mathcal{L}(P, Q, A)$ with respect to P, we have

$$\frac{\partial}{\partial P} \mathcal{L}(P, Q, A) = 2 \cdot P - Q - A.$$

where the first derivative is in the space of functions from $\mathscr{I} \to \mathbb{R}^{[n]^{\leqslant d} \times [n]^{\leqslant d}}$. By the convexity of $\mathcal{L}$ as a function of P, it follows that if we set $\frac{\partial}{\partial P} \mathcal{L} = 0$, we will have the minimizer. Substituting, it follows that

$$\begin{aligned} & \min_P \max_{A \geq 0, Q} \mathcal{L}(P, Q, A) && \text{(III.5)} \\ & \leqslant \max_{A \geq 0, Q} \frac{1}{4} \|A + Q\|^2_{Fr,\nu} - \frac{1}{2} \langle Q, A + Q - \Lambda' \rangle_\nu && \text{(III.6)} \\ & - \frac{1}{2} \langle A, A + Q \rangle_\nu \\ & = \max_{A \geq 0, Q} \langle Q, \Lambda' \rangle_\nu - \frac{1}{4} \|A + Q\|^2_{Fr,\nu} && \text{(III.7)} \end{aligned}$$

Now it is clear that the maximizing choice of A is to set $A = -Q_-$, the negation of the negative-semidefinite projection of Q. Thus (III.7) simplifies to

$$\min_P \max_{A \geq 0, Q} \mathcal{L}(P, Q, A) \qquad \text{(III.8)}$$

$$\leqslant \max_{Q} \langle Q, \Lambda' \rangle_v - \frac{1}{4}\|Q_+\|_{Fr,v}^2$$
$$\leqslant \max_{Q} \langle Q, \Lambda \rangle_v + \eta \operatorname{Tr}_v(Q_+) - \frac{1}{4}\|Q_+\|_{Fr,v}^2, \quad \text{(III.9)}$$

where we have used the shorthand $\operatorname{Tr}_v(Q_+) \stackrel{\text{def}}{=} \mathbb{E}_{\mathcal{I}\sim v} \operatorname{Tr}(Q(\mathcal{I})_+)$. Now suppose that the low-degree matrix polynomial Q^* achieves a right-hand-side value of

$$\langle Q^*, \Lambda \rangle_v + \eta \cdot \operatorname{Tr}_v(Q_+^*) - \frac{1}{4}\|Q_+^*\|_{Fr,v}^2 \geqslant c.$$

Consider $Q' = Q^*/\|Q_+^*\|_{Fr,v}$. Clearly $\|Q'_+\|_{Fr,v} = 1$. Now, multiplying the above inequality through by the scalar $1/\|Q_+^*\|_{Fr,v}$, we have that

$$\begin{aligned}\langle Q', \Lambda \rangle_v &\geqslant \frac{c}{\|Q_+^*\|_{Fr,v}} - \eta \cdot \frac{\operatorname{Tr}_v(Q_+^*)}{\|Q_+^*\|_{Fr,v}} + \frac{1}{4}\|Q_+^*\|_{Fr,v} \\ &\geqslant \frac{c}{\|Q_+^*\|_{Fr,v}} - \eta \cdot n^d + \frac{1}{4}\|Q_+^*\|_{Fr,v}.\end{aligned}$$

Therefore $\langle Q', \Lambda \rangle_v$ is at least $\Omega(c^{1/2})$, as if $\|Q_+^*\|_{Fr,v} \geqslant \sqrt{c}$ then the third term gives the lower bound, and otherwise the first term gives the lower bound.

Thus by substituting Q', the square root of the maximum of (III.9) within an additive ηn^d lower-bounds the maximum of the program

$$\begin{aligned} \max \quad & \langle Q, \Lambda \rangle_v \\ s.t. \quad & Q : \mathscr{J} \to \mathbb{R}^{[n]^{\leqslant d} \times [n]^{\leqslant d}}, \quad \deg_{\text{inst}}(Q) \leqslant D \\ & \|Q_+\|_{Fr,v}^2 \leqslant 1. \end{aligned}$$

This concludes the proof. ∎

Acknowledgment

The authors would like to thank the anonymous reviewers. SBH was partially supported by an NSF GRFP under grant no. 1144153, by a Microsoft Research Graduate Fellowship, and by David Steurer's NSF CAREER award. AP was supported by the National Science Foundation under agreement No. CCF-1412958 and by the Simons Collaboration on Algorithms and Geometry. TS was supported by an NSF Graduate Research Fellowship (1106400). DS was supported by a Microsoft Research Fellowship, a Alfred P. Sloan Fellowship, an NSF CAREER award, and the Simons Collaboration for Algorithms and Geometry.

References

[AK97] Noga Alon and Nabil Kahale, *A spectral technique for coloring random 3-colorable graphs*, SIAM J. Comput. **26** (1997), no. 6, 1733–1748. 2

[AKS98] Noga Alon, Michael Krivelevich, and Benny Sudakov, *Finding a large hidden clique in a random graph*, Random Struct. Algorithms **13** (1998), no. 3-4, 457–466. 2, 4, 6

[AOW15a] Sarah R. Allen, Ryan O'Donnell, and David Witmer, *How to refute a random CSP*, 2015 IEEE 56th Annual Symposium on Foundations of Computer Science—FOCS 2015, IEEE Computer Soc., Los Alamitos, CA, 2015, pp. 689–708. MR 3473335 2

[AOW15b] Sarah R. Allen, Ryan O'Donnell, and David Witmer, *How to refute a random CSP*, FOCS, IEEE Computer Society, 2015, pp. 689–708. 5, 6

[BBH+12] Boaz Barak, Fernando G. S. L. Brandão, Aram Wettroth Harrow, Jonathan A. Kelner, David Steurer, and Yuan Zhou, *Hypercontractivity, sum-of-squares proofs, and their applications*, STOC, ACM, 2012, pp. 307–326. 1, 5

[BCC+10] Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan, *Detecting high log-densities—an $O(n^{1/4})$ approximation for densest k-subgraph*, STOC'10—Proceedings of the 2010 ACM International Symposium on Theory of Computing, ACM, New York, 2010, pp. 201–210. MR 2743268 6

[BCK15] Boaz Barak, Siu On Chan, and Pravesh K. Kothari, *Sum of squares lower bounds from pairwise independence [extended abstract]*, STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing, ACM, New York, 2015, pp. 97–106. MR 3388187 5

[BGG+16] Vijay V. S. P. Bhattiprolu, Mrinal Kanti Ghosh, Venkatesan Guruswami, Euiwoong Lee, and Madhur Tulsiani, *Multiplicative approximations for polynomial optimization over the unit sphere*, Electronic Colloquium on Computational Complexity (ECCC) **23** (2016), 185. 1, 5, 6

[BGL16] Vijay V. S. P. Bhattiprolu, Venkatesan Guruswami, and Euiwoong Lee, *Certifying random polynomials over the unit sphere via sum of squares hierarchy*, CoRR **abs/1605.00903** (2016). 1, 5

[BHK+16] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin, *A nearly tight sum-of-squares lower bound for the planted clique problem*, FOCS, IEEE Computer Society, 2016, pp. 428–437. 1, 3, 5, 6

[BKS14] Boaz Barak, Jonathan A. Kelner, and David Steurer, *Rounding sum-of-squares relaxations*, STOC, ACM, 2014, pp. 31–40. 1

[BKS15] _____, *Dictionary learning and tensor decomposition via the sum-of-squares method*, STOC, ACM, 2015, pp. 143–151. 1

[BKS17] Boaz Barak, Pravesh Kothari, and David Steurer, *Quantum entanglement, sum of squares, and the log rank conjecture*, CoRR **abs/1701.06321** (2017). 1

[BM16] Boaz Barak and Ankur Moitra, *Noisy tensor completion via the sum-of-squares hierarchy*, COLT, JMLR Workshop and Conference Proceedings, vol. 49, JMLR.org, 2016, pp. 417–445. 1, 5

[BS14] Boaz Barak and David Steurer, *Sum-of-squares proofs and the quest toward optimal algorithms*, CoRR **abs/1404.5236** (2014). 5

[DM15] Yash Deshpande and Andrea Montanari, *Improved sum-of-squares lower bounds for hidden clique and hidden submatrix problems*, COLT, JMLR Workshop and Conference Proceedings, vol. 40, JMLR.org, 2015, pp. 523–562. 5

[FM16] Zhou Fan and Andrea Montanari, *How well do local algorithms solve semidefinite programs?*, CoRR **abs/1610.05350** (2016). 2

[GM15] Rong Ge and Tengyu Ma, *Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms*, APPROX-RANDOM, LIPIcs, vol. 40, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015, pp. 829–849. 1

[Gri01a] Dima Grigoriev, *Complexity of positivstellensatz proofs for the knapsack*, Computational Complexity **10** (2001), no. 2, 139–154. 5

[Gri01b] _____, *Linear lower bound on degrees of positivstellensatz calculus proofs for the parity*, Theor. Comput. Sci. **259** (2001), no. 1-2, 613–622. 6

[GW94] Michel X. Goemans and David P. Williamson, *.879-approximation algorithms for MAX CUT and MAX 2sat*, STOC, ACM, 1994, pp. 422–431. 2

[Har70] Richard A Harshman, *Foundations of the parafac procedure: Models and conditions for an" explanatory" multi-modal factor analysis*. 2

[HKP15] Samuel B. Hopkins, Pravesh K. Kothari, and Aaron Potechin, *Sos and planted clique: Tight analysis of MPW moments at all degrees and an optimal lower bound at degree four*, CoRR **abs/1507.05230** (2015). 5

[HL09] Christopher J. Hillar and Lek-Heng Lim, *Most tensor problems are NP hard*, CoRR **abs/0911.1393** (2009). 5

[HSS15] Samuel B. Hopkins, Jonathan Shi, and David Steurer, *Tensor principal component analysis via sum-of-square proofs*, COLT, JMLR Workshop and Conference Proceedings, vol. 40, JMLR.org, 2015, pp. 956–1006. 1, 5, 6

[HSSS16] Samuel B. Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer, *Fast spectral algorithms from sum-of-squares proofs: tensor decomposition and planted sparse vectors*, STOC, ACM, 2016, pp. 178–191. 2, 5, 6

[KMOW17] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer, *Sum of squares lower bounds for refuting any CSP*, CoRR **abs/1701.04521** (2017). 5, 6

[KT17] Ken-ichi Kawarabayashi and Mikkel Thorup, *Coloring 3-colorable graphs with less than* $n^{1/5}$ *colors*, J. ACM **64** (2017), no. 1, 4:1–4:23. 2

[LRS15] James R. Lee, Prasad Raghavendra, and David Steurer, *Lower bounds on the size of semidefinite programming relaxations*, STOC, ACM, 2015, pp. 567–576. 1

[MPW15] Raghu Meka, Aaron Potechin, and Avi Wigderson, *Sum-of-squares lower bounds for planted clique [extended abstract]*, STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing, ACM, New York, 2015, pp. 87–96. MR 3388186 5

[MS16a] Andrea Montanari and Subhabrata Sen, *Semidefinite programs on sparse random graphs and their application to community detection*, STOC, ACM, 2016, pp. 814–827. 6

[MS16b] Andrea Montanari and Nike Sun, *Spectral algorithms for tensor completion*, CoRR **abs/1612.07866** (2016). 6

[MSS16a] Tengyu Ma, Jonathan Shi, and David Steurer, *Polynomial-time tensor decompositions with sum-of-squares*, CoRR **abs/1610.01980** (2016). 1

[MSS16b] ———, *Polynomial-time tensor decompositions with sum-of-squares*, FOCS, IEEE Computer Society, 2016, pp. 438–446. 1

[MW15] Tengyu Ma and Avi Wigderson, *Sum-of-squares lower bounds for sparse PCA*, CoRR **abs/1507.06370** (2015). 5

[Pea01] Karl Pearson, *On lines and planes of closes fit to systems of points in space*, Philosophical Magazine **2** (1901), 559–572. 2

[PS17] Aaron Potechin and David Steurer, *Exact tensor completion with sum-of-squares*, CoRR **abs/1702.06237** (2017). 1

[RM14] Emile Richard and Andrea Montanari, *A statistical model for tensor PCA*, NIPS, 2014, pp. 2897–2905. 4, 5

[RRS16] Prasad Raghavendra, Satish Rao, and Tselil Schramm, *Strongly refuting random csps below the spectral threshold*, CoRR **abs/1605.00058** (2016). 1, 2, 5, 6

[RS15] Prasad Raghavendra and Tselil Schramm, *Tight lower bounds for planted clique in the degree-4 SOS program*, CoRR **abs/1507.05136** (2015). 5

[Sch08] Grant Schoenebeck, *Linear level lasserre lower bounds for certain k-csps*, FOCS, IEEE Computer Society, 2008, pp. 593–602. 5, 6

[Tre09] Luca Trevisan, *Max cut and the smallest eigenvalue*, STOC, ACM, 2009, pp. 263–272. 2

[TS14] Ryota Tomioka and Taiji Suzuki, *Spectral norm of random tensors*, arXiv preprint arXiv:1407.1870 (2014). 5