Sum of Squares Algorithm: Open Problems / Boaz Barak

Here are some open problems regarding the Sum-of-Squares algorithm. In most cases I phrased the problem as asking to show a particular statement, though of course showing the opposite statement would be very interesting as well. These are not meant to be a complete or definitive list, but could perhaps spark your imagination to think of those or other research problems of your own. The broader themes these questions are meant to explore are:

- Can we understand in what cases do SOS programs of intermediate degree (larger than 2 but much smaller than $n$) yield non-trivial guarantees?
- Can we give more evidence to, or perhaps refute, the intuition that the SOS algorithm is *optimal* in some broad domains?
- Can we understand the performance of SOS in *average-case* setting, and whether there are justifications to consider it optimal in this setting as well? This is of course interesting for both machine learning and cryptography.
- Can we understand the role of *noise* in the performance of the SOS algorithm? Is noise a way to distinguish between "combinatorial" and "algebraic" problems in the sense of `http://windowsontheory.org/2013/10/07/structure-vs-combinatorics-in-computational-complexity/`?

## Well posed problems

**Problem 1:** Show that for every constant $C$ there is some $\delta > 0$ and a quasipolynomial ($n^{polylog(n)}$) time algorithm that on input a subspace $V \subseteq \mathbb{R}^n$, can distinguish between the case that $V$ contains the characteristic vector of a set of measure at most $\delta$, and the case that $\mathbb{E}_i v_i^4 \leq C(\mathbb{E}_i v_i^2)^2$ for every $v \in V$. Extend this to a quasipolynomial time algorithm to solve the small-set expansion problem (and hence refute the small set expansion hypothesis). Extend this to a quasipolynomial time algorithm to solve the unique-games problem (and hence refute the unique games conjecture). If you think this cannot be done then even showing that the $d = \log^2 n$ (in fact, even $d = 10$) SOS program does not solve the unique-games problem (or the 4/2 norms ratio problem as defined above) would be very interesting.

**Problem 2:** Show that there is some constant $d$ such that the degree-$d$ SOS problem can distinguish between a random graph and a graph in which a clique of size $f(n)$ was planted for some $f(n) = o(\sqrt{n})$, or prove that this cannot be done. Even settling this question for $d = 4$ would be very interesting.

**Problem 3:** Show that the SOS algorithm is optimal in some sense for "pseudo-random" constraint satisfaction problems, by showing that for every predicate $P : \{0,1\}^k \to \{0,1\}$, $\epsilon > 0$ and pairwise independent distribution $\mu$ over $\{0,1\}^k$, it is NP hard to distinguish, given an instance of MAX-$P$ (i.e., a set of constraints each of which corresponds to applying $P$ to $k$ literals of some Boolean variables $x_1, \ldots, x_n$), between the case that one can satisfy $1 - \epsilon$ fraction of the constraints, and the case that one can satisfy at most $\mathbb{E}_{x \sim \mu} P(x) + \epsilon$ fraction of them. (In a recent work with Chan and Kothari, we show that small degree SOS programs cannot distinguish between these two cases.)

**Problem 4:** More generally, can we obtain a "UGC free Raghavendra Theorem"? For example, can we show (without relying on the UGC) that for every predicate $P : \{0,1\}^k \to \{0,1\}$, $c > s$

and $\epsilon > 0$, if there is an $n$-variable instance of MAX-$P$ whose value is at most $s$ but on which the $\Omega(n)$ degree SOS program outputs at least $c$, then distinguishing between the case that a CSP-$P$ instance as value at least $c - \epsilon$ and the case that it has value at most $s + \epsilon$ is NP-hard?

**Problem 5:** Show that there is some $\eta > 1/2$ and $\delta < 1$ such that for sufficiently small $\epsilon > 0$, the degree $n^\delta$ SOS program for Max-Cut can distinguish, given a graph $G$, between the case that $G$ has a cut of value $1 - \epsilon$ and the case that $G$ has a cut of value $1 - \epsilon^\eta$. (Note that Kelner and Parrilo have a conjectured approach to achieve this.) Can you do this with arbitrarily small $\delta > 0$?

**Problem 6:** If you think the above cannot be done, even showing that the degree $d = 10$ (or even better, $d = \log^2 n$) SOS program cannot achieve this, even for the more general Max-2-LIN problem, would be quite interesting. As an intermediate step, prove or disprove the Khot-Moshkovitz conjecture that for an arbitrarily large constant $c$ the Max-2-LIN instance they construct where the degree $d$ (for some constant $d$) SOS value is $1 - \epsilon$, has actual value at most $1 - c\epsilon$. Some intermediate steps that could be significantly easier are: the Khot-Moshkovitz construction is a reduction from a $k$-CSP on $N$ variables that first considers all $n$-sized subsets of the $N$ original variables and then applies a certain encoding to each one of those $\binom{N}{n}$ "cloud". Prove that if they used a single cloud then the reduction would be "sound" in the sense that there would be no integral solution of value larger than $1 - c\epsilon$. (This should be significantly easier to prove than the Khot-Moshkovitz conjecture since it completely does away with their consistency test; still to my knowledge it is not proven in their paper. The reduction will not be "complete" in this case, since it will have more than exponential blowup and will not preserve SOS solutions; but I still view this as an interesting step. Also if this step is completed, perhaps one can think of other ways than the "cloud" approach of KM to reduce the blowup of this reduction to $2^{\delta N}$ for some small $\delta > 0$; perhaps a "biased" version of their code could work as well.) Another statement that can show the challenge in proving the KM conjecture: Recall that the KM boundary test takes a function $f : \mathbb{R}^n \to \{\pm 1\}$ and checks if $f(x) = f(y)$ where $x$ and $y$ have standard Gaussian coordinates that are each $1 - \alpha$ correlated for some $\alpha \ll 1/n$. Their intended solution $f(x) = (-1)^{\lfloor \langle a,x \rangle \rfloor}$ for $a \in \{\pm 1\}^n$ will fail the test with probability $O(\sqrt{\alpha n})$. Prove that there is a function $f$ that passes the test with $c\sqrt{\alpha n}$ for some $c$ but such that for every constant $d$ and function $g$ of the form $g(x) = (-1)^{\lfloor p(x) \rfloor}$ where $p$ a polynomial of degree at most $d$, $|\mathbb{E}p(x)f(x)| = o(1/n)$.

**Problem 7:** Show that there are some constant $\eta < 1/2$ and $d$, such that the degree $d$-SOS program yields an $O(\log^\eta n)$ approximation to the *Sparsest Cut* problem. If you think this can't be done, even showing that the $d = 8$ algorithm doesn't beat $O(\sqrt{\log n})$ would be very interesting.

**Problem 8:** Give a polynomial-time algorithm that for some sufficiently small $\epsilon > 0$, can (approximately) recover a planted $\epsilon n$-sparse vector $v_0$ inside a random subspace $V \subseteq \mathbb{R}^n$ of dimension $\ell = n^{0.6}$. That is, we choose $v_1, \ldots, v_\ell$ as random Gaussian vectors, and the algorithm gets an arbitrary basis for the span of $\{v_0, v_1, \ldots, v_\ell\}$. Can you extend this to larger dimensions? Can you give a quasipolynomial time algorithm that works when $V$ has dimension $\Omega(n)$? Can you give a quasipolynomial time algorithm for certifying the *Restricted Isometry Property* (RIP) of a random matrix?

**Problem 9:** Improve the dictionary learning algorithm of [Barak-Kelner-Steurer] (in the setting of constant sparsity) from *quasipolynomial* to *polynomial* time.

**Problem 10:** (Suggested by Prasad Raghavendra.) Can SDP relaxations simulate local search? While sum of squares SDP relaxations yield the best known approximations for CSPs, the same is not known for bounded degree CSPs. For instance, MAXCUT on bounded degree graphs can be approximated better than the Goemans-Willamson constant 0.878.. via a combination of SDP rounding and local search. Here local search refers to improving the value of the solution by locally modifying the values. Show that for every constant $\Delta$, there is some $\epsilon > 0, d \in \mathbb{N}$ such that $d$ rounds of SOS yield an $0.878.. + \epsilon$ approximation for MAXCUT on graphs of maximum degree $\Delta$. Another problem to consider is maximum matching in 3-uniform hypergraphs. This can be approximated to a 3/4 factor using only local search (no LP/SDP relaxations), and some natural relaxations have a 1/2 integrality gap for it. Show that for every $\epsilon > 0$, $O(1)$ rounds of SOS give a $3/4 - \epsilon$ approximation for this problem, or rule this out via an integrality gap.

**Problem 11:** (Suggested by Ryan O'Donnell) Let $G$ be the $n$ vertex graph on $\{0, 1 \ldots, n-1\}$ where we connect every two vertices $i, j$ such that their distance (mod $n$) is at most $\Delta$ for some constant $\Delta$. The set $S$ of $n/2$ vertices with east expansion is an arc. Can we prove this with an SOS proof of constant (independent of $\Delta$) degree? For every $\delta > 0$ there is a $c$ such that if we let $G$ be the graph with $n = 2^\ell$ vertices corresponding to $\{0,1\}^\ell$ where we connect vertices $x, y$ if their Hamming distance is at most $c\sqrt{n}$, then for every subsets $A, B$ of $\{0,1\}^\ell$ satisfying $|A|, |B| \geq \delta n$, there is an edge between $A$ and $B$. Can we prove this with an SOS proof of constant degree?

## Fuzzier problems

The following problems are not as well-defined, but this does not mean they are less important.

**Problem 12:** Find more problems in the area of unsupervised learning where one can obtain an efficient algorithm by giving a proof of identifiability using low degree SOS.

**Problem 13:** The notion of pseudo-distributions gives rise to a computational analog of Bayesian reasoning about the knowledge of a computationally-bounded observer. Can we give any interesting applications of this? Perhaps in economics? Or cryptography?

**SOS, Cryptography, and NP $\cap$ coNP.** It sometimes seems as if in the context of combinatorial optimization it holds that "**NP** $\cap$ **coNP** = **P**", or in other words that all proof systems are automatizable. Can the SOS algorithm give any justification to this intuition? In contrast note that we do not believe that this assertion is actually true in general. Indeed, many of our candidates for public key encryption (though not all— see discussion in [Applebaum,Barak, Wigderson]) fall inside **NP** $\cap$ **coNP** (or **AM** $\cap$ **coAM**). Can SOS shed any light on this phenonmenon? A major issue in cryptography is (to quote Adi Shamir) the lack of diversity in the "gene pool" of problems that can be used as a basis for public key encryption. If quantum computers are built, then essentially the only well-tested candidates are based on a single problem— Regev's "Learning With Errors" (LWE) assumption (closely related to various problems on integer lattices). Some concrete questions along these lines are:

**Problem 14:** Find some evidence to the conjecture of Barak-Kindler-Steurer (or other similar conjectures) that the SOS algorithm might be optimal even in an *average case* setting. Can you find applications for this conjecture in cryptography?

**Problem 15:** Can we use a conjectured optimality of SOS to give *public key encryption schemes*? Perhaps to justify the security of LWE? One barrier for the latter could be that breaking LWE and related lattice problems is in fact in **NP** $\cap$ **coNP** or **AM** $\cap$ **coAM**.

**Problem 16:** Understand the role of *noise* in the performance of the SOS algorithm. The algorithm seems to be inherently noise robust, and it also seems that this is related to both its power and its weakness– as is demonstrated by cases such as solving linear equations where it cannot get close to the performance of the Gaussian elimination algorithm, but the latter is also extremely sensitive to noise. Can we get any formal justifications to this intuition? What is the right way to define noise robustness in general? If we believe that the SOS algorithm is optimal (even in some average case setting) for noisy problems, can we get any quantitative predictions to the amount of noise needed for this to hold? This may be related to the question above of getting *public key cryptography* from assuming the optimality of SOS in the average case (see Barak-Kindler-Steurer and Applebaum-Barak-Wigderson).

**Problem 17:** Related to this: is there a sense in which SOS is an optimal noise-robust algorithm or proof system? Are there natural stronger proof systems that are still automatizable (maybe corresponding to other convex programs such as hyperbolic programming, or maybe using a completely different paradigm)? Are there natural noise-robust algorithms for combinatorial optimizations that are *not* captured by the SOS framework? Are there natural stronger proof systems than SOS (even non automatizable ones) that are noise-robust and are stronger than SOS for natural combinatorial optimization problems? Can we understand better the role of the *feasible interpolation property* in this context?

**Problem 18:** I have suggested that the main reason that a ”robust” proof does not translate into an SOS proof is by use of the probabilistic method, but this is by no means a universal law and getting better intuition as to what types of arguments do and don’t translate into low degree SOS proofs is an important research direction. Ryan O’Donnell’s problems above present one challenge to this viewpoint. Another approach is to try to use techniques from derandomization such as use of additive combinatorics or the Zig-Zag product to obtain ”hard to SOS” proofs. In particular, is there an SOS proof that the graph constructed by Capalbo, Reingold, Vadhan and Wigderson (STOC 2002) is a ”lossless expander” (expansion larger than $degree/2$)? Are there SOS proofs for the pseudorandom properties of the condensers we construct in the work with Impagliazzo and Wigderson (FOCS 2004, SICOMP 2006) or other constructions using additive combinatorics? I would suspect the answer might be ”no”. (Indeed, this may be related to the planted clique question, as these tools were used to construct the best known Ramsey graphs.)