SOS Lecture 8: Semidefinite extension complexity lower bounds from SOS lower bounds

Boaz Barak

In this course we have alluded to an intuition that, at least in some domains, the SOS algorithm is *optimal*, in the sense that no other efficient algorithm could beat it. There are several ways to try to justify this intuition:

Ideally, we would want to simply *prove* this, under assumptions such as $\mathbf{P} \neq \mathbf{NP}$. There are two main results along those lines:

- Siu-On Chan showed that for every $\epsilon > 0$ and predicate $P : \mathbb{F}_q^k \to \{0,1\}$ of the form 1_V where where V is an affine subspace of $\mathbb{F}_q^k$ such that the uniform distribution on V is pairwise independent, it is NP-hard to distinguish between the case that an CSP-P instance is $1-\epsilon$ satisfiable and the case that it is $|P^{-1}(1)|/q^k + \epsilon$ satisfiable. Up to the ϵ, this exactly matches the SOS lower bound of Tusliani (which itself is a natural extension of Grigoriev's 3XOR lower bound that we saw in class).

- Prasad Raghavendra showed that if the Unique Games Conjecture is true, then for every $\epsilon > 0$ and predicate P, beating the performance of the degree 2 SOS algorithm on Max-P by $\epsilon > 0$ is NP-hard. Thus, if the UGC is true, that would be very strong evidence for optimality of SOS. Even if the UGC is false, but it is refuted by the SOS algorithm, one could hope that there would be a "modified Raghavendra theorem" showing that the SOS algorithm is optimal. The ideal version would translate a degree d SOS lower bound into a reduction that maps a (suitable variant of) label cover instance of n variables into an instance of the target problem of size $N = poly(n)2^{O(n/d)}$, hence ruling out an $N^{o(d)}$-time algorithm under the exponential time hypothesis.

However, given current knowledge in complexity, such proofs are always conditional on some assumption. Even if we are not too concerned with taking $\mathbf{P} \neq \mathbf{NP}$, or even the ETH, as an axiom (we cannot take such a blasé attitude towards the UGC), these assumptions are inherently limited in the sense that they don't apply (again, based on current knowledge) to *average-case* complexity. Therefore, for several reasons it is also interesting to try to prove that some natural algorithms do not beat the SOS algorithm in some interesting domains.

Perhaps the first question to ask is whether one can beat the SOS algorithm by simply using stronger semidefinite programs. The degree d SOS algorithm can be thought of as obtaining a tighter SDP by adding a set of very specific n^d constraints to the original basic SDP, but perhaps it is possible to add a different set of n^d constraints that would give better performance. The formal way to phrase this question is *extension complexity*. In this lecture we will discuss a very recent result of Lee, Raghavendra and Steurer giving a "Raghavendra Theorem for Semidefinite extension complexity", by translating SOS lower bounds into semidefinite programming extension complexity lower bounds. This can be thought of as the analog of the prior work of Chan,Lee,Raghavendra and Steurer that showed a similar connection between Sherali-Adams lower bounds and linear programming extension complexity lower bounds.

For example using this translation they use Grigoriev's 3XOR lower bound to show the following theorem:

Theorem 1. *For every $\epsilon > 0$ and subspace U of the functions from $\{\pm 1\}^n$ to $\mathbb{R}$ of dimension less than $n^{o(\log n/\log\log n)}$, there is an instance I of $3XOR$ of value (i.e., maximum fraction of satisfied constraints) at most $1/2 + \epsilon$, such that there is no U-proof that the value of I is less than $1 = \epsilon$.*

The definition of a U *prood* that some function $f : \{\pm 1\}^n \to \mathbb{R}$ satisfies $f \leq \alpha$ is that there are functions $g_1, \ldots, g_t \in U$ such that

$$f(x) = \alpha - \sum_{i=1}^{t} g_i(x)^2$$

for every $x \in \{\pm 1\}^n$. Note that a degree-d SOS proof corresponds to a U-proof where U is the span of all monomials of degree at most d, and hence U proofs are generalization of degree $\log_n \dim(U)$-SOS proofs.

It turns out that the notion of *SDP rank* plays here the same role that the notion of *non-negative rank* plays for linear programming extension complexity. We say that a non-negative $p \times q$ matrix M has *psd-rank* at most r if there exist $r \times r$ psd matrices $\{A_i\}_{i=1}^p$ and $\{B_j\}_{j=1}^q$ such that $M_{i,j} = \mathrm{Tr}(A_i B_j)$. **Exercise 1:** Prove that M has *non-negative rank* at most r if and only if it has a decomposition such as the one above where the A_i's and B_j's are diagonal .

At the heart of their work is the following theorem:

Theorem 2. *For $d < m < n/2$, $f : \{\pm 1\}^m \to [0, \infty)$, let $M = M_n^f$ be the $\binom{n}{m} \times 2^n$ matrix such that $M(S, x) = f(x_S)$ for every $S \in \binom{[n]}{m}$ and $x \in \{\pm 1\}^n$. If there is no degree d SOS proof that $f \geq 0$ then*

$$\mathrm{rank}_{\mathsf{psd}}(M) \geq n^{\Omega(d)}$$

The rest of this lecture will be devoted to outlining the proof of Theorem 2. To make things concrete, we will consider the particular $f : \{\pm 1\}^m \to \mathbb{R}$ that one obtains by taking the instance of 3XOR I arising from Grigoriev's 3XOR lower bond. That is, we let $f(x)$ equal 0.7 minus the fraction of I's constraints that are satisfied by the assignment x. As we saw in class, for every x, $f(x) \geq 0.1$ but there is a degree (say) $d = m/1000$ pseudo-distribution D over $\{\pm 1\}^m$ that satisfies the constraint $\{f(x) = -0.3\}$.

There are several ways to represent such a pseudo-distribution. One representation (which is the one we used in class) is to simply use the pseudo-expectation operator mapping a polynomial P to $\tilde{\mathbb{E}}_{x \sim D} P$, but another one is to define for every $x \in \{\pm 1\}^x$, $D(x) \in \mathbb{R}$ to be a number such that

$$\tilde{\mathbb{E}}_{x \sim D} P = \mathbb{E}_{x \in \{\pm 1\}^n} D(x) P(x) \tag{1}$$

we shall follow the language of LRS and call this a "pseudo-density" operator. To move from the previous representation to (1), we can simply define

$$D(x) = \sum_{|\alpha| \leq d} \left(\tilde{\mathbb{E}}_{x' \sim D} \, \chi_\alpha(x') \right) \chi_\alpha(x)$$

where for $\alpha \subseteq [m]$, we define $\chi_\alpha(x) = \prod_{i \in \alpha} x_i$.

We suppose, toward a contradiction that $\mathrm{rank}_{\mathsf{psd}}(M_n^f) = r$ for some $r = n^{o(1)}$ demonstrated by some decomposition $\{P(S)\}_{S \in \binom{[n]}{m}}, \{Q(x)\}_{x \in \{\pm 1\}^n}$. We will consider the following quantity:

$$\mathbb{E}_{S \in \binom{[n]}{m}} \mathbb{E}_{x \in \{\pm 1\}^n} D(x_S) f(x_S) \quad (*) \tag{2}$$

On one hand for every fixed S and fixing of $x_{\overline{S}}$, (*) equals to $\mathbb{E}_{x' \in \{\pm 1\}^m} D(x') f(x') = -0.3$.

On the other hand, by the psd decomposition this equals

$$\mathbb{E}_S \mathbb{E}_{x \in \{\pm 1\}^n} D(x_S) \mathrm{Tr}(P(S) Q(x)) = \mathbb{E}_S \mathbb{E}_{x'' \in \{\pm 1\}^{\overline{S}}} \mathbb{E}_{x' \in \{\pm 1\}^S} D(x') \| \sqrt{P(S)} \sqrt{Q(x', x'')} \|_F^2 \tag{3}$$

(where the square roots of P and Q are defined since these are p.s.d matrices.) **Exercise 2:** Prove that for every psd matrices PQ, $\mathrm{Tr}(PQ) = \|\sqrt{PQ}\|_F^2 = \sum_{i,j} \sqrt{PQ}_{i,j}^2$, where for a psd matrix $A = \sum \lambda_i v_i v_i^\top$, $\sqrt{A} = \sum \sqrt{\lambda_i} v_i v_i^\top$.

Note that for every fixed S, x'', the function $x' \mapsto \|\sqrt{P(S)}\sqrt{Q(x,x'')}\|_F^2$ is the sum of the squares of the entries of these matrices, and so this function is some sum of squares $g(x')$. If by some luck the degree of g was smaller than $d < 2$ we would get our contradiction and be done since we know that

$$\mathbb{E}_{x'} D(x') g(x') = \tilde{\mathbb{E}} g(x') \geq 0$$

for every SOS g of degree at most $d/2$.

The heart of the LRS proof is therefore in showing that this function g can in fact be sufficiently well approximated by a low degree polynomial. They do so using two tools:

- *Quantum learning* — they use an instance of the following general principle that has turned out to be useful again and again in computer science:

  Simple tests can be fooled by (fairly) simple objects.

  At a high level this phenomena underlies the whole theory of pseudorandomness, but we will be focused on a particular set of manifestations of it, known as "Boosting", "Impagliazzo's Hardcore Lemma", "Dense Model Theorem" and more. LRS use a quantum version of this principle.

  Their idea is that because in our case we test the function $Q(x)$ against the degree m function $\mathbb{E}_S P(S) D(x_S)$, we can approximate it with some function of the form $h(x) = R^2(x)$ where R has degree at most $\tilde{O}(m)$.

- *Random restrictions* — the above is still not sufficient since we need to reduce the degree below $m/1000$. For this we use the other general tool

  Simple functions become even simpler if we fix most of their inputs at random.

  Specifically, using tools from Fourier analysis of Boolean functions, one can show that, once we established the degree of h is not too large, if we choose a random S and fix the values of x in $\overline{S}$ then we significantly reduce the degree of h further (up to some negligible error), and hence we can apply the result above.

1 Boosting, dense model, hardcore lemma, multiplicative weights, computational entropy, and their quantum/matrix/semidefinite cousins

There is a set of results that has been used across many areas in mathematics and computer science. Here are some examples of these results. (This is not meant to be comprehensive review of the related literature— these are the kind of ideas that seem to have been rediscovered again and again by people in different communities and for different purposes.)

- Suppose that you are trying to *learn* some unknown function $F : \Omega \to \{0,1\}$ and that for every distribution D over Ω, you can find a function $f_D : \Omega \to \{0,1\}$ that has $1/2 + \epsilon$ agreement

with F, then you can boost this to $1-\delta$ agreement by combining $\text{poly}(1/\epsilon, \log(1/\epsilon))$ of these functions.

An algorithm achieving such boosting was first put forward by Schapire in 1989, with some later improvements by Freund culminating in their 1995 *AdaBoost* algorithm.

- Suppose that you know that some function $F : \Omega \to \{0,1\}$ is *mildly hard* in the sense that every efficient algorithm A has at most $1-\delta$ agreement with it. It turns out that there is a not-too-small subset $H \subseteq \Omega$ (in fact, of measure 2δ) on which F is *extremely hard* in the sense that every efficient algorithm has at most $1/2+\epsilon$ (where the ϵ plays a part in the quantitative losses between our two instantiations of the word "efficient").

  This theorem, which turns out to be extremely useful in the theory of pseudo-randomness, is known as "Impagliazzo's Hardcore Lemma", proven by Russell Impagliazzo in 1995 (with an important quantitative improvement by Holenstein in 2005. (The connection for Boosting was noted by Klivans and Servedio in 2003; see also my paper with Hardt and Kale.)

- Suppose that $S \subseteq \Omega$ is pseudorandom in the sense that one cannot distinguish a uniform element of S from a uniform element of Ω via efficient algorithm, and suppose that $P \subseteq S$ satisfies $|P| \geq \delta|S|$. Then you can find some $M \subseteq \Omega$ with $|M| \geq \delta|\Omega|$ such that one cannot distinguish a uniform element of P from a uniform element of M.

  This theorem is known as the *dense model theorem*, and was first shown by Green and Tao in the context of their 2004 work establishing that the set of primes contains arbitrarily long arithmetic progressions. A more general explicit version was later given by Tao and Zeigler. Some simplified proofs were later found by Gowers and (independently) Reingold, Trevisan and Vadhan. Roughly speaking, Green and Tao used in their theorem number theoretic results showing that the set S of *pseudo primes* (integers having only few large divisors) are pseudorandom. Since the set P of primes has constant density in S, the dense model theorem shows that it is indistinguishable from a set M dense in the set of all integers, but such sets contain large arithmetic progressions by Szemeredi's Theorem.

All these results share some the following properties:

- They are counterintuitive when you (or at least I) first hear about them.
- They are incredible useful.
- They are proven via the multiplicative weights algorithm or von Neumann's min-max theorem (aka linear programming duality or Hahn-Banach theorem).
- They are actually not that hard to prove once you have the nerve to guess that the result might be true.

It turns out that they are all essentially equivalent. Let me sketch how you might prove the Boosting result. You can think of this as a game between two players— Player I comes up with a distribution D, and Player II responds with an algorithm f_D that has $1/2+\epsilon$ agreement with f. We know that by the min-max theorem that Player II could come up with a single distribution over algorithms (or equivalently a probabilistic algorithm) A that would have such agreement with *every* distribution, which means that it succeeds in solving F on any input with probability $1/2+\epsilon$— probability that can be boosted to $1-\delta$ via $O(\log 1/\delta)$ repetitions. Now to converge to this algorithm we can use a fairly simple process of back and forth between the distribution player and

the algorithm player. The distributions would be updated according to a multiplicate update rule, and the final algorithm would be some weighted average of the algorithms obtained in each round. So, if these intermediate algorithms are simple, then so will be the final one.

Further reading on the "classical" versions. Luca Trevisan had several blog posts related to this, see `https://lucatrevisan.wordpress.com/2008/12/07/applications-of-low-complexity-approximat` and also a survey in the 2011 theory of cryptography conference. Sitanshu Gakkhar's Master's thesis `http://summit.sfu.ca/item/12349`, Russell Impagliazzo's talk `https://video.ias.edu/csdm/densemodelthm` and scribe notes `https://www.math.ias.edu/files/russell_scribe.pdf`, paper of Trevisan, Tulsiani and Vadhan `http://ttic.uchicago.edu/~madhurt/Papers/regularity-full.pdf`. These results can also be phrased in the language of computational entropy— intuitively a set P as in the dense model theorem, has "pseudo-entropy" at least $\log|\Omega| - \log(1/\delta)$. This was first explored in my 2003 paper with Shaltiel and Wigderson (see also Dziembowski and Pietrzak 2008). (Note that our paper had a bug and proved a weaker result than originally claimed; see the 2011 paper of Benjamin Fuller and Leonid Reyzin for discussion.)

Semidefinite/quantum extensions People have looked at extension of these results to the *quantum* setting. One can think of this as extending results from classical to quantum, from numbers to matrices, or from linear programming to semidefinite programming. In any case one obtains similar results, see the LRS paper. Note that (as we will see) LRS uses a very restricted special case of this general principle focusing on a single test.

2 Random restrictions

The idea of random restriction is to take a function $f : \{\pm 1\}^n \to \mathbb{R}$ and change it into the function $g : \{\pm 1\}^m \to \mathbb{R}$ obtain by picking an m-sized set $S \subseteq [n]$ at random, and $x'' \in \{\pm 1\}^{\overline{S}}$, and then define $g(x') = f(x', x'')$. The hope is that g is significantly simpler than f. This is not necessarily always the case. For example, if f is simply the parity function $x_1 \cdots x_n$, then g is a parity as well, but if f is "simpler" than the parity in some sense, then g could be significantly even simpler. This idea has been used in Hastad's switching lemma, where one can use random restriction to show that if f has a small constant depth circuit, then g has a circuit of even smaller depth. In the current context we need an even simpler statement about the Fourier degree. (See Ryan O'Donnel's book for a thorough treatment of random restrictions.) If you have a monomial $\prod_{i \in T} x_i$, then after restricting to a random set S you are left with the monomial, $\prod_{i \in S \cap T} x_i$ which is expected to have size about $|S||T|/n$ which would be much smaller than both $|S|$ and $|T|$ if they are both much smaller than n. Specifically, if we take a unit norm function $f : \{\pm 1\}^n \to \mathbb{R}$ of degree at most $\ell = \tilde{O}(m)$, and restrict it to a random m sized set S, then the norm squared of the part of f that has degree at least $m/10^4$ would be in expectation less than $\binom{\ell}{m/10^4}(m/n)^{m/10^4}$ which for $n \gg m$ would be $n^{-\Omega(m)}$. If the presumed PSD rank r satisfies $r = n^{o(m)}$ then it turns out that this is small enough to be treated as negligible.

3 Proof sketch

We want to obtain a contradiction to the statement

$$\mathbb{E}_{S \in \binom{[n]}{m}} \mathbb{E}_{x \in \{\pm 1\}^n} D(x_S) f(x_S) = -0.3 \tag{4}$$

under the assumption that $f(x_s) = \mathrm{Tr}(P(S)Q(x))$ where $P(S)$ and $Q(x)$ are $r \times r$ psd matrices for $r = m^{o(1)}$.

We first phrase the LHS of (4) as

$$\mathrm{Tr}(MQ)$$

where M is the block matrix with 2^n $r \times r$ blocks with the x^{th} block corresponding to $\mathbb{E}_S D(x_S)P(S)$, and the Q is the block matrix 2^n $r \times r$ blocks with the x^{th} block corresponding to $2^{-n}Q(x)$.

We will later show using a quantum learning type argument that there exists a matrix $R = p(M)$, for some polynomial p with degree $\tilde{O}(1)$, such that $\mathrm{Tr}(R^2) = \mathrm{Tr}(Q)$ and

$$\mathrm{Tr}(MR^2) \leq \mathrm{Tr}(MQ) + 0.1 = -0.2 \tag{5}$$

Let's defer the proof of this for a moment and see what it yields. Since every element of M is a polynomial in x of degree at most m, we can think of R as a matrix-valued $\tilde{O}(m)$ degree polynomial in x and rewrite (5) as

$$\mathbb{E}_{S \in \binom{[n]}{m}} \mathbb{E}_{x \in \{\pm 1\}^n} D(x_S)\mathrm{Tr}(P(S)R^2(x)) = \mathbb{E}_{S \in \binom{[n]}{m}} P(S) \mathbb{E}_{x' \in \{\pm 1\}^S} D(x') \mathbb{E}_{x'' \in \{\pm 1\}^{\overline{S}}} R^2(x', x'') \tag{6}$$

If we fix a "typical" value for S and x'', then as mentioned above, we can argue using random restrictions that the function $R'(x') = R(x', x'')$ equals $L(x') + H(x')$ where $L(x')$ is a degree $o(m)$ polynomial and $\|H(x')\| \leq r^{-\omega(1)}$. Since without loss of generality, using the fact that $P(S)$ is an $r \times r$ matrix, $\|P(S)\| \leq r^2$, the effect of $H(x)$ will be negligible and we get that

$$\mathbb{E}_{x' \in \{\pm 1\}^S} D(x')\mathrm{Tr}(P(S)L^2(x')) \leq -0.2 + o(1) \tag{7}$$

Now define $g(x')$ to be $\mathrm{Tr}(P(S)L^2(x')) = \|\sqrt{P(S)}L(x')\|_F^2$. Since for a fixed S, every entry of the matrix $\sqrt{P(S)}L(x')$ is a degree $o(m)$ polynomial in x', this quantity is a polynomial of degree $o(m)$ which is a sum of squares, and so we get

$$\tilde{\mathbb{E}}_D g(x') = \mathbb{E}_{x'} D(x')g(x') \leq -0.2 + o(1) < 0$$

contradicting the fact that D is a degree $m/1000$ pseudo distribution.

3.1 Quantum learning argument

We only need a rather restricted version of quantum learning (see the LRS paper for a much more general statement). We want to prove the following:

Lemma 3. *Let M, Q be $s \times s$ matrices such that Q is psd and $\mathrm{Tr}(Q) = 1$, then there exists a degree $\mathrm{poly}(\log \|Q\| s, 1/\epsilon, \|M\|)$ polynomial p such that*

$$\mathrm{Tr}(Mp(M)^2)/\mathrm{Tr}(p(M)^2) \leq \mathrm{Tr}(MQ) + \epsilon \tag{8}$$

Proof. We will show that this holds where instead of using a polynomial we write a matrix exponential $p(M) = \alpha e^{(\theta/2)M}$ for $\theta = poly(\|M\|, \log \|Q\| s, 1/\epsilon)$, and the result would then follow from a Taylor approximation. (Note that in our case $\|Q\| = poly(r)/s$; more generally LRS work with a lower bound on the von Neumann entropy of Q which is simply the Shannon entropy of the eigenvalues. If Q has trace 1 and $\|Q\| \leq \alpha/s$ then the von-Neumann entropy is at least $\log s - \log(1/\alpha)$ using known relations between min-entropy and Shannon entropy.)

By making the transformation $M \mapsto I - M/\|M\|$ we can change to the case that M is psd of norm at most 1 and that our goal is to show that for $\theta = O(\log \|Q\| s, poly(1/\epsilon))$

$$\mathrm{Tr}(Me^{\theta M}) \geq \mathrm{Tr}(MQ) - \epsilon \tag{9}$$

What is the maximum value a matrix Q can achieve for $\mathrm{Tr}(MQ)$ subject to being psd, trace 1, and having norm at most some value α/s? It's not hard to see that this would be obtained by having $Q = (\alpha/s)\sum_{i=1}^{s/\alpha} v_i v_i^\top$ where $v_1, v_2, v_3, \ldots$ are the eigenvectors of M sorted in descending order of the eigenvalues $\lambda_1, \lambda_2, \ldots$. In this case $\mathrm{Tr}(MQ)$ will simply be the average of the s/α top eigenvalues of M. Once again, one can see that the most extreme situation would be if all the top s/α eigenvalues would be equal to 1 while the rest are zero, since that would maximize $\mathrm{Tr}(MQ)$ under our conditions. Now, if we consider the matrix

$$e^{\theta M} = \sum e^{\theta \lambda_i} v_i v_i^\top$$

we can see that it gives weight 1 to the eigenvectors corresponding to zero, and weight e^θ to the eigenvectors corresponding to 1, since there are at most α times more of the former than the latter, if $\theta \gg \log \alpha$ then almost of all of the weight will be on the top eigenvectors, and we get that the matrix (after normalizing to trace 1) will give a value very close to $\mathrm{Tr}(MQ)$. □