Approximating MAX-E3LIN is NP-Hard

Evan Chen

May 4, 2016

This lecture focuses on the MAX-E3LIN problem. We prove that approximating it is NP-hard by a reduction from LABEL-COVER.

1 Introducing MAX-E3LIN

In the MAX-E3LIN problem, our input is a series of linear equations $\pmod 2$ in n binary variables, each with three terms. Equivalently, one can think of this as ± 1 variables and ternary products. The objective is to maximize the fraction of satisfied equations.

Example 1 (Example of MAX-E3LIN instance)

$$\begin{aligned} x_1 + x_3 + x_4 &\equiv 1 \pmod 2 & \qquad x_1 x_3 x_4 &= -1 \\ x_1 + x_2 + x_4 &\equiv 0 \pmod 2 & \qquad x_1 x_2 x_4 &= +1 \\ x_1 + x_2 + x_5 &\equiv 1 \pmod 2 & \qquad x_1 x_2 x_5 &= -1 \\ x_1 + x_3 + x_5 &\equiv 1 \pmod 2 & \qquad x_1 x_3 x_5 &= -1 \end{aligned}$$

A diligent reader can check that we may obtain $\frac{3}{4}$ but not 1.

Remark 2. We immediately notice that

- If there's a solution with value 1, we can find it easily with $\mathbb{F}_2$ linear algebra.
- It is always possible to get at least $\frac{1}{2}$ by selecting all-zero or all-one.

The theorem we will prove today is that these "obvious" observations are essentially the best ones possible! Our main result is that improving the above constants to 51% and 99%, say, is NP-hard.

Theorem 3 (Hardness of MAX-E3LIN)
The $\frac{1}{2} + \varepsilon$ vs. $1 - \delta$ decision problem for MAX-E3LIN is NP-hard.

This means it is NP-hard to decide whether an MAX-E3LIN instance has value $\le \frac{1}{2} + \varepsilon$ or $\ge 1 - \delta$ (given it is one or the other).

A direct corollary of this is approximating MAX-SATis also NP-hard.

Corollary 4
The $\frac{7}{8} + \varepsilon$ vs. $1 - \delta$ decision problem for MAX-SAT is NP-hard.

Remark 5. The constant $\frac{7}{8}$ is optimal in light of a random assignment. In fact, one can replace $1-\delta$ with δ, but we don't do so here.

Proof. Given an equation $a+b+c=1$ in MAX-E3LIN, we consider four formulas $a \lor \neg b \lor \neg c$, $\neg a \lor b \lor \neg c$, $a \lor \neg b \lor \neg c$, $a \lor b \lor c$. Either three or four of them are satisfied, with four occurring exactly when $a+b+c=0$. One does a similar construction for $a+b+c=1$. □

The hardness of MAX-E3LIN is relevant to the PCP theorem: using MAX-E3LIN gadgets, Håstad was able to prove a very strong version of the PCP theorem, in which the verifier merely reads just *three bits* of a proof!

Theorem 6 (Håstad PCP)
Let $\varepsilon, \delta > 0$. We have

$$\mathbf{NP} \subseteq \mathbf{PCP}_{\frac{1}{2}+\varepsilon, 1-\delta}(3, O(\log n)).$$

In other words, any $L \in \mathbf{NP}$ has a (non-adaptive) verifier with the following properties.

- The verifier uses $O(\log n)$ random bits, and queries just three (!) bits.
- The acceptance condition is either $a+b+c=1$ or $a+b+c=0$.
- If $x \in L$, then there is a proof Π which is accepted with probability at least $1-\delta$.
- If $x \notin L$, then every proof is accepted with probability at most $\frac{1}{2}+\varepsilon$.

2 Label Cover

We will prove our main result by reducing from the LABEL-COVER. Recall LABEL-COVER is played as follows: we have a bipartite graph $G = U \cup V$, a set of keys K for vertices of U and a set of labels L for V. For every edge $e = \{u, v\}$ there is a function $\pi_e : L \to K$ specifying a key $k = \pi_e(\ell) \in K$ for every label $\ell \in L$. The goal is to label the graph G while maximizing the number of edges e with compatible key-label pairs.

Approximating LABEL-COVER is NP-hard:

Theorem 7 (Hardness of LABEL-COVER)
The η vs. 1 decision problem for LABEL-COVER is NP-hard for every $\eta > 0$, given $|K|$ and $|L|$ are sufficiently large in η.

So for any $\eta > 0$, it is NP-hard to decide whether one can satisfy all edges or fewer than η of them.

3 Setup

We are going to make a reduction of the following shape:

$$\begin{array}{ccc} \boxed{\text{LABEL-COVER}} & \rightsquigarrow & \boxed{\text{MAX-E3LIN}} \\ < \eta & \xrightarrow{\text{Soundness}} & < \frac{1}{2} + \varepsilon \\ = 1 & \xrightarrow{\text{Completeness}} & \geq 1 - \delta \end{array}$$

In words this means that

- "Completeness": If the LABEL-COVER instance is completely satisfiable, then we get a solution of value $\ge 1-\delta$ in the resulting MAX-E3LIN.
- "Soundness": If the LABEL-COVER instance has value $\le \eta$, then we get a solution of value $\le \frac{1}{2}+\varepsilon$ in the resulting MAX-E3LIN.

Thus given an oracle for MAX-E3LIN decision, we can obtain η vs. 1 decision for LABEL-COVER, which we know is hard.

The setup for this is quite involved, using a huge number of variables. Just to agree on some conventions:

Definition 8 ("Long Code")**.** A K-indexed binary string $x = (x_k)_k$ is a ± 1 sequence indexed by K. We can think of it as an element of $\{\pm 1\}^K$. An L-binary string $y = (y_\ell)_\ell$ is defined similarly.

Now we initialize $|U| \cdot 2^{|K|} + |V| \cdot 2^{|L|}$ variables:

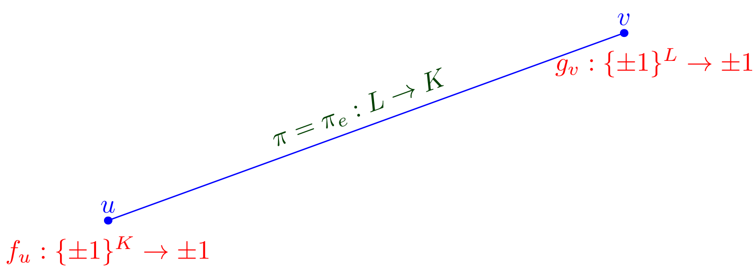
- At every vertex $u \in U$, we will create $2^{|K|}$ binary variables, one for every K-indexed binary string. It is better to collect these variables into a function

$$f_u : \{\pm 1\}^K \to \{\pm 1\}.$$

- Similarly, at every vertex $v \in V$, we will create $2^{|L|}$ binary variables, one for every L-indexed binary string, and collect these into a function

$$g_v : \{\pm 1\}^L \to \{\pm 1\}.$$

Picture:



Next we generate the equations. Here's the motivation: we want to do this in such a way that given a satisfying labelling for LABEL-COVER, nearly all the MAX-E3LIN equations can be satisfied. One idea is as follows: for every edge e, letting $\pi = \pi_e$,

- Take a K-indexed binary string $x = (x_k)_k$ at random. Take an L-indexed binary string $y = (y_\ell)_\ell$ at random.
- Define the L-indexed binary $z = (z_\ell)_\ell$ string by $z = \left(x_{\pi(\ell)} y_\ell\right)$.
- Write down the equation $f_u(x) g_v(y) g_v(z) = +1$ for the MAX-E3LIN instance.

Thus, assuming we had a valid coloring of the graph, we could let f_u and g_v be the dictator functions for the colorings. In that case, $f_u(x) = x_{\pi(\ell)}$, $g_v(y) = y_\ell$, and $g_v(z) = x_{\pi(\ell)} y_\ell$, so the product is always $+1$.

Unfortunately, this has two fatal flaws:

1. This means a 1 instance of LABEL-COVER gives a 1 instance of MAX-E3LIN, but we need $1-\delta$ to have a hope of working.

2. Right now we could also just set all variables to be $+1$.

We fix this as follows, by using the following equations.

Definition 9 (Equations of reduction)**.** For every edge e, with $\pi = \pi_e$, we alter the construction and say

- Let $x = (x_k)_k$ be and $y = (y_\ell)_\ell$ be random as before.
- Let $n = (n_\ell)_\ell$ be a random L-indexed binary string, drawn from a δ-biased distribution (-1 with probability δ). And now define $z = (z_\ell)_\ell$ by

$$z_\ell = x_{\pi(\ell)} y_\ell n_\ell.$$

  The n_ℓ represents "noise" bits, which resolve the first problem by corrupting a bit of z with probability δ.

- Write down one of the following two equations with $\frac{1}{2}$ probability each:

$$f_u(x)g_v(y)g_v(z) = +1$$
$$f_u(x)g_v(y)g_v(-z) = -1.$$

  This resolves the second issue.

This gives a set of $O(|E|)$ equations.

I claim this reduction works. So we need to prove the "completeness" and "soundness" claims above.

4 Proof of Completeness

Given a labeling of G with value 1, as described we simply let f_u and g_v be dictator functions corresponding to this valid labelling. Then as we've seen, we will pass $1-\delta$ of the equations.

5 A Fourier Computation

Before proving soundness, we will first need to explicitly compute the probability an equation above is satisfied. Remember we generated an equation for e based on random strings x, y, λ.

For $T \subseteq L$, we define

$$\pi_e^{\text{odd}}(T) = \left\{ k \in K \mid \left| \pi_e^{-1}(k) \cap T \right| \text{ is odd} \right\}.$$

Thus T maps subsets of L to subsets of K.

Remark 10. Note that $|\pi^{\text{odd}}(T)| \le |T|$ and that $\pi^{\text{odd}}(T) \neq \varnothing$ if $|T|$ is odd.

Lemma 11 (Edge Probability)
The probability that an equation generated for $e = \{u, v\}$ is true is

$$\frac{1}{2} + \frac{1}{2} \sum_{\substack{T \subseteq L \\ |T| \text{ odd}}} (1-2\delta)^{|T|} \widehat{g}_v(T)^2 \widehat{f}_u(\pi_e^{\text{odd}}(T)).$$

Proof. Omitted for now... □

6 Proof of Soundness

We will go in the reverse direction and show (constructively) that if there is MAX-E3LIN instance has a solution with value $\geq \frac{1}{2} + 2\varepsilon$, then we can reconstruct a solution to LABEL-COVER with value $\geq \eta$. (The use of 2ε here will be clear in a moment). This process is called "decoding".

The idea is as follows: if S is a small set such that $\widehat{f}_u(S)$ is large, then we can pick a key from S at random for f_u; compare this with the dictator functions where $\widehat{f}_u(S) = 1$ and $|S| = 1$. We want to do something similar with T.

Here are the concrete details. Let $\Lambda = \frac{\log(1/\varepsilon)}{2\delta}$ and $\eta = \frac{\varepsilon^3}{\Lambda^2}$ be constants (the actual values arise later).

Definition 12. We say that a nonempty set $S \subseteq K$ of keys is **heavy** for u if

$$|S| \leq \Lambda \qquad \text{and} \qquad \widehat{f}_u(S) \geq \varepsilon^2.$$

Note that there are at most ε^{-2} heavy sets by Parseval.

Definition 13. We say that a nonempty set $T \subseteq L$ of labels is e-**excellent** for v if

$$|T| \leq \Lambda \qquad \text{and} \qquad S = \pi_e^{\text{odd}}(T) \text{ is heavy.}$$

In particular $S \neq \varnothing$ so at least one compatible key-label pair is in $S \times T$.

Notice that, unlike the case with S, the criteria for "good" in T actually depends on the edge e in question! This makes it easier to keys than to select labels. In order to pick labels, we will have to choose from a $\widehat{g}_v^2$ distribution.

Lemma 14 (At least ε of T are excellent)
For any edge $e = \{u, v\}$, at least ε of the possible T according to the distribution $\widehat{g}_v^2$ are e-excellent.

Proof. Applying an averaging argument to the inequality

$$\sum_{\substack{T \subseteq L \\ |T| \text{ odd}}} (1-2\delta)^{|T|} \widehat{g}_v(T)^2 \left| \widehat{f}_u(\pi^{\text{odd}}(T)) \right| \geq 2\varepsilon$$

shows there is at least ε chance that $|T|$ is odd and satisfies

$$(1-2\delta)^{|T|} \left| \widehat{f}_u(S) \right| \geq \varepsilon$$

where $S = \pi_e^{\text{odd}}(T)$. In particular, $(1-2\delta)^{|T|} \geq \varepsilon \iff |T| \leq \Lambda$. Finally by Remark 10, we see S is heavy. □

Now, use the following algorithm.

- For every vertex $u \in U$, take the union of all heavy sets, say

$$\mathcal{H} = \bigcup_{S \text{ heavy}} S.$$

  Pick a random key from $\mathcal{H}$. Note that $|\mathcal{H}| \le \Lambda\varepsilon^{-2}$, since there are at most ε^{-2} heavy sets (by Parseval) and each has at most Λ elements.

- For every vertex $v \in V$, select a random set T according to the distribution $\widehat{g}_v(T)^2$, and select a random element from T.

I claim that this works.

Fix an edge e. There is at least an ε chance that T is e-excellent. If it is, then there is at least one compatible pair in $\mathcal{H} \times T$. Hence we conclude probability of success is at least

$$\varepsilon \cdot \frac{1}{\Lambda\varepsilon^{-2}} \cdot \frac{1}{\Lambda} = \frac{\varepsilon^3}{\Lambda^2} = \eta.$$

(Addendum: it's pointed out to me this isn't quite right; the overall probability of the equation given by an edge e is $\ge \frac{1}{2} + \varepsilon$, but this doesn't imply it for every edge. Thus one likely needs to do another averaging argument.)