Shamir's proof of PSPACE $\subseteq$ IP from sumcheck

Notes by Linus Tang.

These notes have not been thoroughly reviewed. Any errors below are my own responsibility.

Sources:

- The original paper by Adi Shamir
  - https://dl.acm.org/doi/pdf/10.1145/146585.146609
- BU CS 535 lecture notes by Mark Bun
  - https://cs-people.bu.edu/mbun/courses/535_F23/lectures/lec22.pdf

Background knowledge: Some familiarity with complexity theory is expected.

Read my notes on the sumcheck protocol first if you haven't learned about it already!

- https://www.mit.edu/~linust/files/Sumcheck.pdf

Complexity classes IP and PSPACE

If necessary, see the following for definitions of the two complexity classes:

- https://en.wikipedia.org/wiki/IP_(complexity)
- https://en.wikipedia.org/wiki/PSPACE

Proof of PSPACE $\subseteq$ IP using sumcheck

TQBF, a PSPACE-complete problem

We use without proof the fact that True Quantized Boolean Formula (TQBF) is PSPACE complete.

Specifically, a quantized boolean formula (QBF) is of the form

$$\exists x_1 \forall x_2 \exists x_3 \forall x_4 \cdots \phi(x_1, ..., x_n)$$

where all variables are quantified and ϕ is a boolean formula over n boolean variables.

TQBF is the language consisting of all true QBFs.

(Actually, the definition of QBF allows existential and universal quantifiers to appear in any order, but this distinction doesn't matter because we can just insert dummy variables to make the quantifiers alternate as above.)

Multilinear extensions and multilinearizations

Let c be an arbitrary $\{0,1\}^n \to \{0,1\}$ function and $\mathbb{F}$ be a field. Then the *multilinear extension* of c to $\mathbb{F}^n$ is the function $c' : \mathbb{F}^n \to \mathbb{F}$ given by

$$c'(x_1, ..., x_n) = \sum_{(b_1,...,b_n)\in\{0,1\}^n} c(b_1, ..., b_n) \prod_{i=1}^{n} \begin{cases} x_i & \text{if } b_i = 1 \\ 1 - x_i & \text{if } b_i = 0 \end{cases}.$$

It is the unique $\mathbb{F}^n \to \mathbb{F}$ function satisfying two properties:

- c' is multilinear (i.e. it is a polynomial with degree ≤ 1 in each input).
- c' extends c. That is, $c'(x_1, ..., x_n) = c(x_1, ..., x_n)$ for all $(x_1, ..., x_n) \in \{0,1\}^n$.

We also introduce a very similar concept called the multilinearization.

Let $\mathbb{F}$ be a field and d be an arbitrary $\mathbb{F}^n \to \mathbb{F}$ function. Then the *multilinearization* of d is the function d' given by

$$d'(x_1, ..., x_n) = \sum_{(b_1,...,b_n)\in\{0,1\}^n} d(b_1, ..., b_n) \prod_{i=1}^{n} \begin{cases} x_i & \text{if } b_i = 1 \\ 1 - x_i & \text{if } b_i = 0 \end{cases}.$$

It is the unique $\mathbb{F}^n \to \mathbb{F}$ function satisfying two properties:
- d' is multilinear
- $d'(x_1, ..., x_n) = d(x_1, ..., x_n)$ for all $(x_1, ..., x_n) \in \{0, 1\}^n$.

In our case, if we let "False = 0" and "True = 1", then ϕ is a $\{0, 1\}^n \to \{0, 1\}$ function, so we can take multilinear extensions.

Let $f_{\mathbb{R}}$ denote the multilinear extension of ϕ to $\mathbb{R}^n$. The following correspondence can be proven by inducting from the inside out:

$$\beta_{\mathbb{R}} = \sum_{x_1\in\{0,1\}} \prod_{x_2\in\{0,1\}} \sum_{x_3\in\{0,1\}} \prod_{x_4\in\{0,1\}} \cdots f_{\mathbb{R}}(x_1, ..., x_n)$$

is nonzero if and only if the original QBF

$$\exists x_1 \forall x_2 \exists x_3 \forall x_4 \cdots \phi(x_1, ..., x_n)$$

is true.

Comparison to sumcheck

The above is very similar to the sumcheck problem, except:
- We are working over $\mathbb{R}$ instead of a finite field $\mathbb{F}_p$
- Half of our quantifiers are products instead of sums.

Briefly speaking:
- The point can be handled by choosing a large prime p and working in $\mathbb{F}_p$ instead of $\mathbb{R}$.
- The second point can be handled by having the verifier check $g_i(0)g_i(1) = g_{i-1}(t_{i-1})$ with a product instead of a sum of the left side.
  - This would cause the polynomials g_i to grow exponentially in degree. It turns out that multilinearization is exactly what we need to keep the degree in check.

Modifying the sumcheck protocol to work for TQBF.

We have a prover trying to convince the verifier that a given QBF is true.

A word on notation: We use default typesetting (e.g. g and β) to denote the correct values that should be output by an honest prover and sans letters (e.g. g and $\boldsymbol{\beta}$) to denote what the prover actually outputs.

We describe the protocol in terms of an honest prover below, so $\mathsf{g} = g$ and $\boldsymbol{\beta} = \beta$, but we choose to use default typesetting for prover operations and sans letters when the verifier is performing computations over untrusted values sent by the prover. This notational distinction is mainly meant to improve the clarity of the soundness analysis later.

Let f be the multilinear extension of φ to $\mathbb{F}_p^n$, for some prime $p \gg n$. We will see later that we want p to be chosen by the prover.

Define

$$
\begin{aligned}
\beta_0 &= \sum_{x_1\in\{0,1\}} \prod_{x_2\in\{0,1\}} \sum_{x_3\in\{0,1\}} \prod_{x_4\in\{0,1\}} \cdots f(x_1,...,x_n) \\
\beta_1(x_1) &= \prod_{x_2\in\{0,1\}} \sum_{x_3\in\{0,1\}} \prod_{x_4\in\{0,1\}} \cdots f(x_1,...,x_n) \\
\beta_2(x_1,x_2) &= \sum_{x_3\in\{0,1\}} \prod_{x_4\in\{0,1\}} \cdots f(x_1,...,x_n) \\
&\vdots \\
\beta_n(x_1,...,x_n) &= f(x_1,...,x_n).
\end{aligned}
$$

The protocol is as follows:

- The prover evaluates β_0 and sends it to the verifier.
- The prover computes evaluates the multilinearization $g_1(\cdot)$ of $\beta_1(\cdot)$ and sends its two coefficients to the verifier.
- The verifier uses the coefficients to check that $g_1(0) + g_1(0) = \beta$.
- For $i = 2, ..., n$:
  - The verifier sends uniformly sampled t_{i-1} from $\mathbb{F}_p$.
  - The prover computes the multilinearization $g_i(\cdot)$ of $\beta_i(t_1, ..., t_{i-1}, \cdot)$, and again sends its two coefficients as a degree $\leq d$ polynomial in x.
  - If i is odd, the verifier checks $g_i(0) + g_i(1) = g_{i-1}(t_{i-1})$.
  - If i is even, the verifier checks $g_i(0)g_i(1) = g_{i-1}(t_{i-1})$.
- Finally, the verifier uniformly samples t_m from $\mathbb{F}_p$ and checks that $g_m(t_m) = f(t_1, ..., t_m)$ using its oracle access to f. The verifier accepts if and only if all checks have passed and $\beta_0 \neq 0$.

Soundness analysis

The soundness analysis is essentially the same as that for the original sumcheck. In this case, the degree of f in each variable is at most 1, a dishonest verifier passes with probability at most $\frac{n}{p}$. Hence we require that $p \gg n$.

Completeness analysis

By following the above protocol honestly, the prover is guaranteed to pass all checks other than the $\beta_0 \neq 0$. To pass this check, the prover needs to choose p carefully.

- Recall the previous claim that

$$
\begin{gathered}
\beta_{\mathbb{R}} = \sum_{x_1\in\{0,1\}} \prod_{x_2\in\{0,1\}} \sum_{x_3\in\{0,1\}} \prod_{x_4\in\{0,1\}} \cdots f_{\mathbb{R}}(x_1,...,x_n) \neq 0 \\
\Leftrightarrow \\
\exists x_1 \forall x_2 \exists x_3 \forall x_4 \cdots \phi(x_1, ..., x_n) \text{ is a true QBF.}
\end{gathered}
$$

- Consider the case where the prover is honest, and the QBF is true, so $\beta_{\mathbb{R}} \neq 0$. Note that $\beta_{\mathbb{R}}$ is an integer and that $\beta_0 \in \mathbb{F}_p$ is the residue class of $\beta_{\mathbb{R}}$ mod p.
- Therefore, the prover sets p to be a large prime number at the begnning of the protocol, such that p does not divide $\beta_{\mathbb{R}}$. (Thus we avoid the situation where $\beta_0 = 0$ even though the QBF is true).

Note that p cannot be too large; it has to be $\leq 2^{\text{poly}(n)}$. We now want to show that some large prime $p \leq 2^{\text{poly}(n)}$ does not divide $\beta_{\mathbb{R}}$. While we don't give a full proof of this, here's a sketch:

- $\beta_{\mathbb{R}}$ can't be too large. Specifically, $\beta_{\mathbb{R}} = 2^{2^{O(n)}}$.
- We now want to show that $\beta_{\mathbb{R}}$ cannot be divisible be divisible by every prime $n \ll p \leq 2^{\text{poly}(n)}$.

- If $\beta_{\mathbb{R}}$ were divisible by all of these primes, then it would be divisible by their product, which we can estimate using the Prime Number Theorem. Said product is not $2^{2^{O(n)}}$, so we can conclude that there is a prime p the prover can choose that does not divide $\beta_{\mathbb{R}}$.

So, $\beta_0 \neq 0$ and the honest prover passes this check as well.

We have constructed an interactive proof for a PSPACE-complete problem TQBF, proving that PSPACE $\subseteq$ IP!

Remark

It is not too difficult to show the other direction IP $\subseteq$ PSPACE. We don't cover the proof in these notes but direct the reader to section 1 of https://cs-people.bu.edu/mbun/courses/535_F23/lectures/lec22.pdf.

Thus, putting these results together we get that IP $=$ PSPACE. Nice!