Today

Toda's Theorem: $\mathrm{PH} \subseteq \mathrm{P}^{\#\mathrm{P}}$

- Recall Operators.
- $\mathrm{PH} \subseteq \mathrm{BP} \cdot \oplus \cdot \mathsf{P}$: Operator calculus
- $\mathrm{BP} \cdot \oplus \cdot \mathsf{P} \subseteq \mathrm{P}^{\#\mathrm{P}}$: Polynomial magic.

Operators

- $\exists \cdot \mathcal{C}$, $\forall \cdot \mathcal{C}$, $\neg \cdot \mathcal{C}$: Definition obvious, Properties well-studied.
- $\oplus \cdot \mathcal{C}$:
  - Definition: $L \in \oplus \cdot \mathcal{C}$ if there exists M such that $L(M) \in \mathcal{C}$ and $L = \{x \mid |\{y \mid M(x,y) = 1\}|$ is odd $\}$.
  - Property: Can replace odd with even!
- $\mathrm{BP} \cdot \mathcal{C}$:
  - Definition: $L \in \mathrm{BP} \cdot \mathcal{C}$ if for every poly q there exists a machine M s.t. $L(M) \in \mathcal{C}$ and $\Pr_y[M(x,y) = L(x)] \geq 1 - 2^{-q(n)}$.
  - Note: Picking strong defn. This is needed.

Operator Calculus

$$\begin{aligned} Q \cdot \mathrm{BP} \cdot \oplus \cdot \mathsf{P} &\subseteq \mathrm{BP} \cdot \oplus \cdot \mathrm{BP} \cdot \oplus \cdot \mathsf{P} \\ &\subseteq \mathrm{BP} \cdot \mathrm{BP} \cdot \oplus \cdot \oplus \cdot \mathsf{P} \\ &\subseteq \mathrm{BP} \cdot \oplus \cdot \oplus \cdot \mathsf{P} \\ &\subseteq \mathrm{BP} \cdot \oplus \cdot \mathsf{P} \end{aligned}$$

(where $Q \in \{\exists, \forall\}$). All steps except first, trivial.

Steps in reverse order

Claim 1: $\oplus \cdot \oplus \cdot \mathcal{C} \subseteq \oplus \cdot \mathcal{C}$

Obvious reduction works.

Draw Picture.

Steps (contd.)

Claim 2: $\mathrm{BP} \cdot \mathrm{BP} \cdot \mathcal{C} \subseteq \mathrm{BP} \cdot \mathcal{C}$

If we wish probability of final error to be $2^{-q(n)}$, then pick LHS machines to have prob. of error at most $2 \cdot 2^{-q(n)}$.

Steps (contd.)

Claim 3: $\oplus \cdot \mathrm{BP} \cdot \mathcal{C} \subseteq \mathrm{BP} \cdot \oplus \cdot \mathcal{C}$

Draw picture.

Parity fanout 2^ℓ. BP fanout 2^m. Error 2^{-q}.

incorrect leaves before switch $2^{m+\ell-q}$.

incorrect parity gates after switch also at most same.

Error at most $2^{\ell-q}$.

Steps (contd.)

Claim 4: $\exists \cdot \mathcal{C} \subseteq \mathrm{BP} \cdot \oplus \cdot \mathcal{C}$

Pictorially: Replace OR gate by approx majority of parity. Familiar?

Razborov-Smolensky: Right error parameter. Wrong amount of randomness. (Exponential!).

Valiant-Vazirani: (Unique SAT in Parity SAT!): Right randomness; wrong error.

But error can be reduced by a cute amplification trick.

Overview

- Concludes Part 1 of Toda's Theorem.
- For Part 2, need to understand some arithmetic games one can play with # accepting paths.

Arithmetic games

- If non-deterministic machine M_1 on input w_1 has n_1 accepting paths, and M_2 on input w_2 has n_2 accepting paths, then can create machines + inputs that have n_1+n_2, or $n_1 \times n_2$ accepting paths.
- W.l.o.g. consider circuits. Have circuits C_1, C_2 $(C_i(\cdot) = M_i(w_i, \cdot))$ taking n-bit inputs and accepting n_1 and n_2 inputs respectively.
- Then, circuit C_3 given by $C_3(x,y) = C_1(x) \wedge C_2(x)$ accepts $n_1 n_2$ inputs.
- And, C_4 given by $C_4(x,b) = (b \wedge C_1(x)) \vee (\neg b \wedge C_2(x))$ has $n_1 + n_2$ accepting inputs.

More arithmetic

- Can also construction circuits with any fixed number of accepting inputs.
- So given any polynomial p with positive coefficients, and circuit C with N accepting inputs, can construct C' with $p(N)$ accepting inputs. Furthermore size of $C' = O(|p| \cdot |C|)$.
- If p is a constant degree polynomial with constant coefficients, can apply this process $O(\log n)$ times.

Will use the last parts later, but first show how to amplify.

"Boosting" modular counts

- Suppose $a = b(\text{mod } 2^{2^c})$ for $b \in \{0, -1\}$.
- Then for $h(a) = 3a^4 + 4a^3$ have $h(a) = b(\text{mod } 2^{2^{c+1}})$.
- Let $h^{(i)}(a) = h(h^{(i-1)}(a)$, where $h^{(0)}(a) = a$.
- Let $t = O(\log m)$. Let C' be the circuit with $h^{(t)}(\#_x C(x,y))$ accepting inputs. (Can construct such C' in polynomial time.).
- C' is what we need.

QED. (Done with Toda's theorem.)

Polynomial magic=?

How would we come up with the polynomial h?

- Requirements:
  - $h(a) = b(\mod 2^{2^{c+1}})$ for $b \in \{0, -1\}$.
  - Coefficients of h non-negative.
- First condition says $a^2 | h(a)$ and $(a+1)^2 | h(a) + 1$. Natural choice (to make coeff. of a^1 disappear), $h_1(a) + 1 = (a+1)^2(a-1)^2 = a^4 - 2a^2 + 1$. Now have $h_2(a) = a^4 - 2a^2$. Satisfies first condition, but violates second.
- To make coefficients positive, add a (large multiple of) polynomial with +ve

coefficients that is 0 on a^2 and $(a+1)^2$. Simple choice $= a^2(a+1)^2$.

- New candidate $h_2(a) = h_1(a) + 2 \cdot a^2(a+1)^2 = 3a^4 + 4a^3$.