

Pseudorandom Generators and the BQP vs PH Problem

Bill Fefferman (IQI, Caltech)

Joint with Chris Umans

Agenda

- I. A simplistic primer on computational pseudorandomness
- II. How (classically) powerful are quantum computers?
- III. A new approach to an old problem in quantum complexity

I. A simplistic primer on Computational Pseudorandomness

- We say a distribution X over $\{0,1\}^t$ is ϵ -indistinguishable with respect to a circuit class C' iff for all $C \in C'$:
 - $|Pr[C(X) = 1] - Pr[C(U_t) = 1]| \leq \epsilon$
- Suppose $t > s$, then:
 - PRG: $\{0,1\}^s \rightarrow \{0,1\}^t$ is a (C', ϵ) -Pseudorandom Generator iff
 - PRG(U_s) is ϵ -indistinguishable wrt C'

I. Explicit constructions of PRGs

- [Nisan & Wigderson '94]
 - Suppose we have a function f which is hard (on average) for C' to compute.
 - Then we can construct an explicit PRG using that function f !
 - [NW '94] show a combinatorial construction of a (C', ϵ) -PRG, NW^f
 - i.e., for all $C \in C'$
$$|\Pr[C(NW^f(U_s)) = 1] - \Pr[C(U_t) = 1]| \leq \epsilon$$

I. Who cares?

- Strong PRGs (e.g., $t \gg s$) have *derandomization* consequences
 - Consider a probabilistic algorithm $A(x,r)$ with $|r|=t$, which can be computed by some circuit in C'
 - Let f be a function hard (on average) for C'
 - Then a circuit powerful enough to compute f can *derandomize* A by:
 - Run A on x with “randomness” $NW^f(z)$
 - Do this once for each of 2^s binary seeds z and take a majority vote!
 - High-level reason we think **BPP=P...**

I. Can quantum computers break a classical PRG?

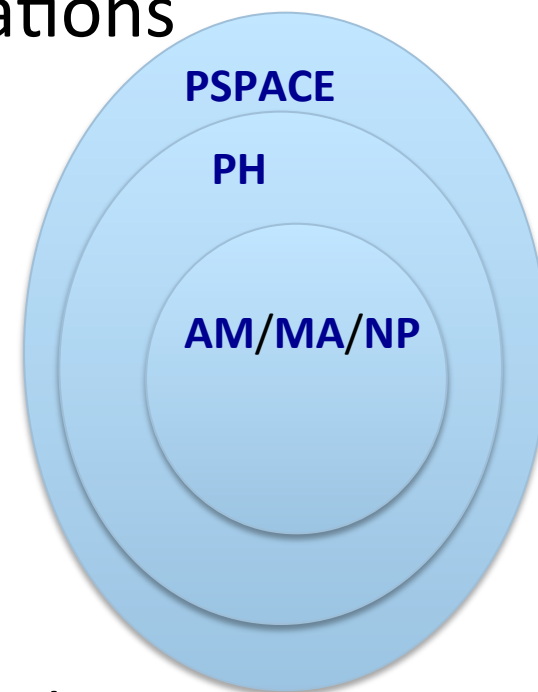
- Will show that quantum computers can “break an instantiation of a NW PRG”
 - i.e., for a specific f which is hard classically will show that there is a quantum algorithm which can *distinguish* $NW^f(U_s)$ from U_t
 - This will have applications to quantum complexity theory!

II. How (classically) powerful are quantum computers?

- **BQP** – Class of languages that can be decided efficiently by a quantum computer
- Where is **BQP** relative to **NP**?
 - Is there a problem that can be solved with a quantum computer that can't be verified efficiently (**BQP** $\not\subseteq$ **NP**?)
 - Can we give evidence?
 - Oracle separations

II. Is **BQP** $\not\subseteq$ **PH**?

- History: Towards stronger oracle separations
 - [Bernstein & Vazirani '93]
 - Recursive Fourier Sampling?
 - [Aaronson '09]
 - Conjecture: “Fourier Checking” not in **PH**
 - Assuming GLN
 - [Aaronson '10] (counterexample!)
 - GLN false (depth 3)
- Why is it so hard?
 - Cannot rely on crude arguments about low degree approximating polynomials (both classes have such approximations... see [RS '87], [Beals et al '01])



II. What is the **PH**?

- Formally:
 - **NP^{NP^{NP}}**... constant number of oracles...
 - Alternatively, any constant number of alternations
 - $\exists \pi_1 \forall \pi_2, \dots, \bigoplus_k \pi_k V_L(x, \pi_1, \pi_2, \dots, \pi_k) = 1$
 - Attractive for quantum computing:
 - (trivial upper bound) **PH** \subseteq **PSPACE**: n alternations
 - (trivial lower bound) **NP** \subseteq **PH**: 1 existential alternation
 - These are probably strict!
 - **Our question**: Do we need superconstant but sublinear alternations to simulate quantum computation?

III. Today: A new approach

- Show oracle separation would follow from question studied in “pseudorandomness” literature [BSW '03]
- Quantum computers can break instantiation of the famous “Nisan-Wigderson” generator [NW '94]
- Unconditionally, gives another example of exponential quantum speedup over randomized classical computation

III. Equivalent Setup

- want a function $f:\{0,1\}^N \mapsto \{0,1\}$
 - in **BQLOGTIME**
 - $O(\log N)$ quantum steps
 - random access to N -bit input: $|i\rangle|z\rangle \mapsto |i\rangle|z \oplus f(i)\rangle$
 - accept with high probability iff $f(\text{input}) = 1$
 - but not in **AC₀**

III. Equivalent Setup

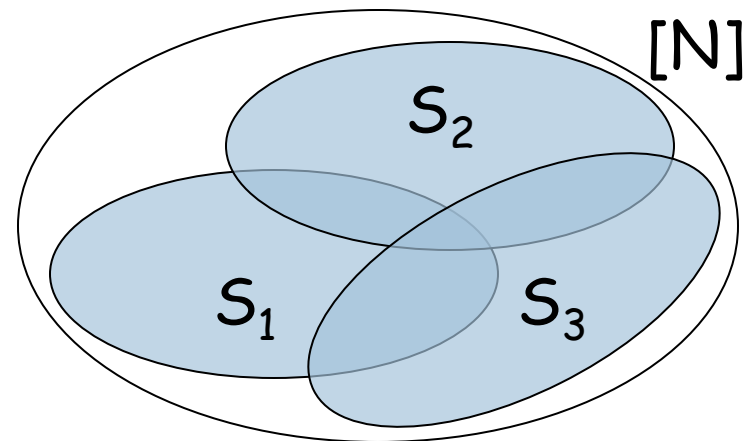
- More general (and transformable to previous setting):
 - two distributions on N bit strings D_1, D_2
 - **BQLOGTIME** algorithm that *distinguishes* them
i.e., there is some quantum circuit Q such that
$$|Pr[Q(D_1) = 1] - Pr[Q(D_2) = 1]| \geq \epsilon$$
 - proof no **AC₀** circuit can *distinguish* them
 - we will always take D_2 to be uniform

III. What can't AC_0 do?

- PARITY and MAJORITY not in AC_0 [FSS '84]
- AC_0 circuits can't *distinguish*:
 1. Bits distributed uniformly
 2. Bits drawn from “Nisan-Wigderson” distribution derived from:
 1. function hard (on average) for AC_0 to *compute*
 2. Nearly-disjoint “subset system”
- Our result: There exists a specific choice of these subsets, for which the resulting distribution generated by the MAJORITY function can be distinguished (from uniform) quantumly!

III. Formal: Nisan-Wigderson PRG

- $S_1, S_2, \dots, S_M \subset [N]$ is an (N', p) -design if
 - for all i , $|S_i| = N'$
 - for all $i \neq j$, $|S_i \cap S_j| \leq p$



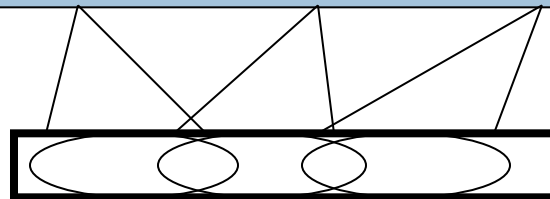
III. Nisan-Wigderson PRG

- $f: \{0,1\}^{N'} \rightarrow \{0,1\}$ is a hard function (e.g., MAJORITY)
- $S_1, \dots, S_M \subset [N]$ is an (N', p) -design

$$NW^f(x) = x \circ f(x|_{S_1}) \circ f(x|_{S_2}) \circ \dots \circ f(x|_{S_M})$$

truth table of f :

01010010111101010111001010



Seed $x \in \{0,1\}^N$

III. Proof of Classical Hardness: *Indistinguishability*

- Proof by contradiction:

- assume circuit C *distinguishes* from uniform:

$$|\Pr[C(U_{N+M}) = 1] - \Pr[C(NW^f(U_N)) = 1]| > \epsilon$$

loss from hybrid argument!

- transform C into a *predictor* circuit P

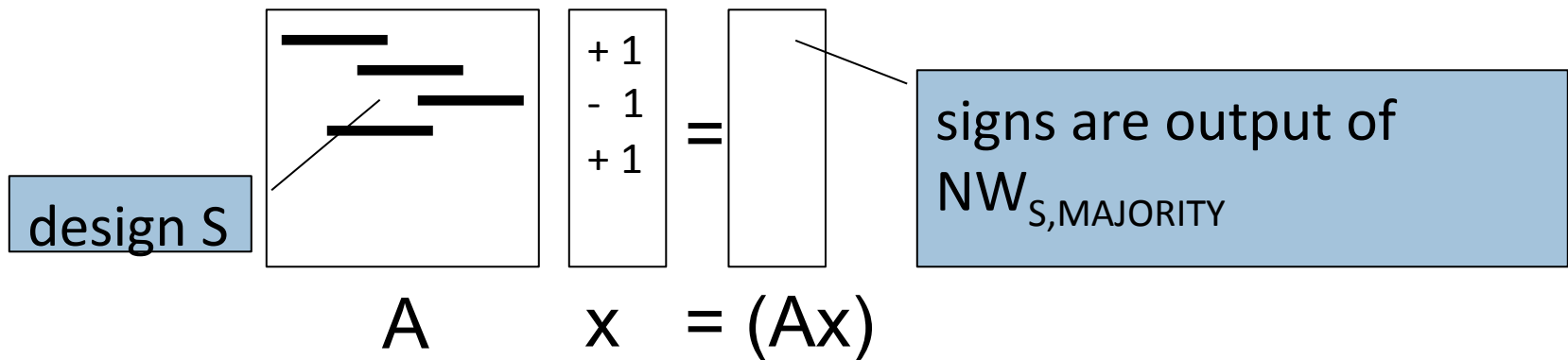
$$\Pr_{x \sim U}[P(NW^f(x)_{1 \dots i-1}) = NW^f(x)_i] > \frac{1}{2} + \epsilon/M$$

- derive similar sized circuit approximating hard function (using properties of subset system)
- Contradiction (assuming hard function cannot be approximated this well)

III. Distributions distinguishable from Uniform with a quantum computer

$D_A = (x, y)$: pick x uniformly from $\{1, -1\}^N$, set $y_i = \text{sgn}((Ax)_i)$

- Goal: Matrix A with rows that
 1. Have large supports
 2. Have supports with small pairwise intersection (form some (N', p) -design)
 3. Are pairwise orthogonal
 4. Should be an efficient quantum circuit (product of $\text{polylog}(N)$ local unitaries)



III. Quantum Algorithm

- We claim there is a quantum algorithm to distinguish oracle distributed over D_A from U_{2N}
- Quantum algorithm:

1. enter uniform superposition over $\log N$ qubits
2. query x and multiply into phases: $\sum_i x_i |i\rangle$
3. apply A : $\sum_i (Ax)_i |i\rangle$
4. query y and multiply into phases: $\sum_i y_i (Ax)_i |i\rangle$
5. measure in Hadamard basis, accept iff $(0,0,\dots,0)$

- Crucially, after step 4 we are back to all positive amplitudes in case oracle is D_A
- But in case oracle is U_{2N} with high prob. we have random mix of signs (low weight on $|0\dots 0\rangle$ after final Hadamard)

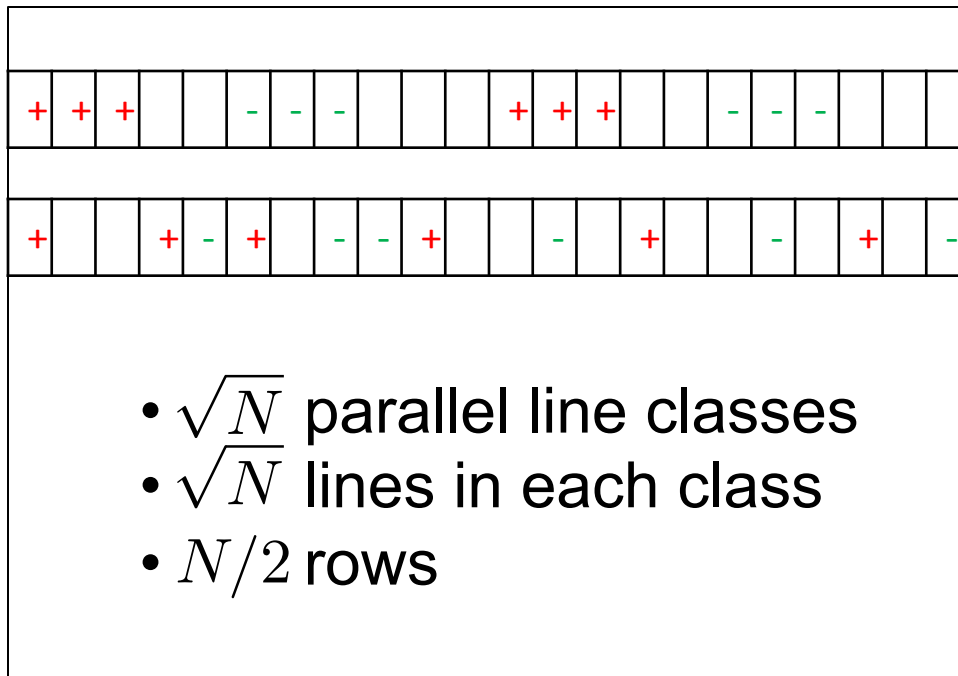
III. Constructing A using “Paired-Lines”

- Will describe $N/2$ pairwise-orthogonal vectors in $\{0, \pm 1\}^N$
- Identify N with the affine plane $\mathbb{F}_{\sqrt{N}} \times \mathbb{F}_{\sqrt{N}}$
- Let B_1, B_2 be an equipartition of $\mathbb{F}_{\sqrt{N}}$
- Take some $\phi : B_1 \rightarrow B_2$ (an arbitrary bijection). Then the vectors are:

$$v_{a,b}[x, y] = \begin{cases} -1 & y = ax + b \\ +1 & y = ax + \phi(b) \\ 0 & \textit{otherwise} \end{cases}$$

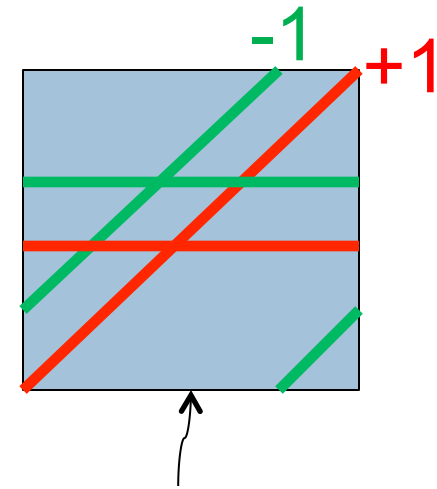
III. Construction

- Each row will be $v_{a,b}$ (supported on two parallel, “paired-lines” with slope a)
- Identify columns with affine plane $\mathbb{F}_{\sqrt{N}} \times \mathbb{F}_{\sqrt{N}}$



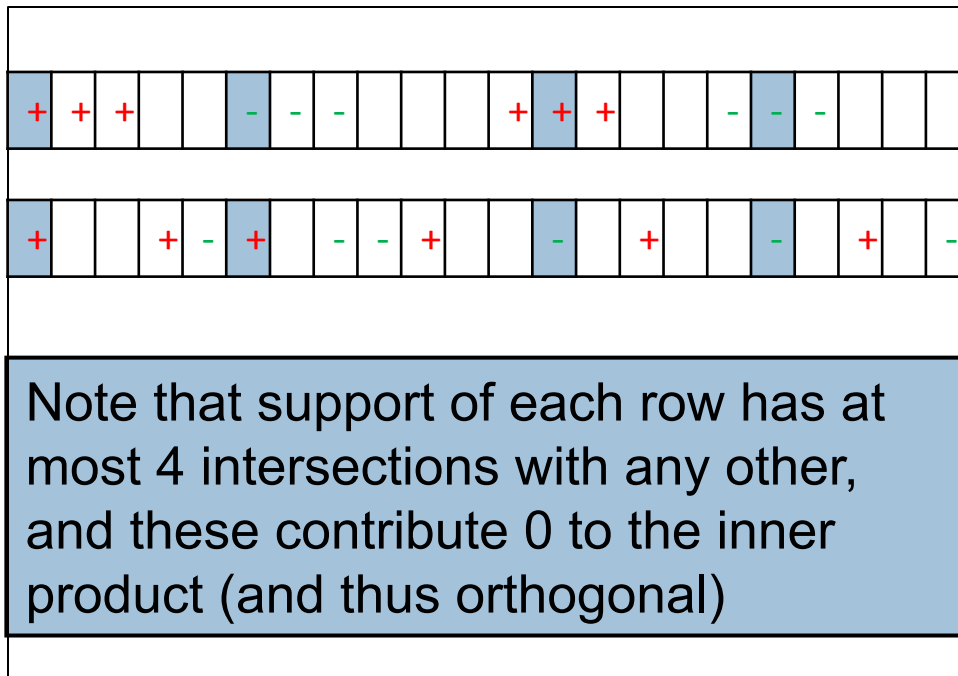
- \sqrt{N} parallel line classes
- \sqrt{N} lines in each class
- $N/2$ rows

A

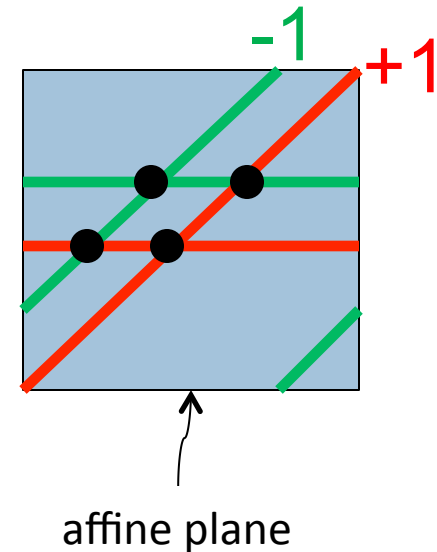


III. Construction

- Each row will be $v_{a,b}$ (supported on two parallel, “paired-lines” with slope a)
- Identify columns with affine plane $\mathbb{F}_{\sqrt{N}} \times \mathbb{F}_{\sqrt{N}}$



A



III. Putting it all together

- “Technical Core”: We construct an efficient quantum circuit realized by unitary whose (un-normalized) rows are vectors from a paired-lines construction wrt a specific bijection
 - $N \times N$
 - Half of the rows will correspond to the paired-lines vectors
- Note that we have a quantum algorithm, as described before, that uses this unitary A to distinguish between D_A and U_{2N}
- But distinguishing should be hard for AC_0 since Ax is instantiation of NW generator!

III. But why aren't we finished?

- Distribution on $(3/2)N$ bits that is the NW generator w.r.t. MAJORITY on $N^{1/2}$ bits, with output length $N/2$
- Suppose AC_0 can distinguish from uniform with constant gap ε
 - proof: distinguisher to predictor, and then circuit for majority w/ success $\frac{1}{2} + \varepsilon/(N/2)$
 - but already possible w/ success $\frac{1}{2} + \Omega(1/N^{1/4})$
 - ... no contradiction

III. Our Conjecture

- Distribution on $(3/2)N$ bits that is the NW generator w.r.t. MAJORITY on $N^{1/2}$ bits, with output length $N/2$
- Can AC_0 distinguish from uniform with constant gap ϵ ?

Conjecture: No.

III. Recent new work [with Shaltiel, Umans & Viola]

- (Non-trivial) simplification of conjecture:
 - Take M *completely* disjoint subsets
 - Distinguish:
 1. All bits distributed uniformly
 2. First half bits are uniform, second are majorities over disjoint subsets of first half
 - This is indeed hard for **AC₀**!

Conclusions

- Assuming conjecture, gives a quantum algorithm that can “break” a PRG
- Unitaries used are novel and don’t seem to resemble those used in other quantum algorithms
- Conjecture implies oracle relative to which **BQP** is not in **PH**