The Power of Quantum Fourier Sampling

Bill Fefferman
QuICS, University of Maryland/NIST
Joint work with Chris Umans (Caltech)

Based on arxiv:1507.05592
I. Complexity Theory Basics
Classical Complexity Theory

- **P**
  - Class of problems efficiently solved on classical computer

- **NP**
  - Class of problems with efficiently checkable solutions
  - Characterized by SAT
    - Input: $\Psi: \{0,1\}^n \rightarrow \{0,1\}$
      - n-variable 3-CNF formula
        » E.g., $(x_1 \lor x_2 \lor x_3) \land (x_1 \lor \neg x_2 \lor x_6) \land \ldots$
    - Problem: $\exists x_1, x_2, \ldots, x_n$ so that $\Psi(x)=1$?
  - Could use a box solving SAT to solve any problem in NP
Beyond **NP**

- **Tautology**
  - Input: $\Psi: \{0,1\}^n \rightarrow \{0,1\}$
  - $\forall x \; \Psi(x) = 1$?
  - Complete for **coNP**

- **$\text{QSAT}_k$**
  - Generalizes **SAT** and **Tautology**
  - Input: $\Psi: \{0,1\}^n \rightarrow \{0,1\}$ & partitioning $S_1, S_2, \ldots, S_k \subseteq [n]$
    - Problem: $\exists x_{S_1} \forall x_{S_2}, \ldots, Q_k x_{S_k}$ so that $\Psi(x) = 1$?
    - Thought to be strictly harder with larger $k$'s (or else there is a collapse)

- $\Sigma_k$ is class of problems solvable with a $\text{QSAT}_k$ box
- **PH** is class of problems solvable with a $\text{QSAT}_{O(1)}$ box
- **PSPACE** is class of problems solvable with a $\text{QSAT}_n$ box
Complexity of Counting

- **#SAT**
  - Input: $\Psi: \{0,1\}^n \rightarrow \{0,1\}$
  - Problem: How many satisfying assignments to $\Psi$?
- **#SAT** is complete for **#P**
- **PH $\subseteq$ P** [Toda’91]
- Permanent:$[X] = \sum_{\sigma \in S_n} \prod_{i=1}^{n} X_{i,\sigma(i)}$ is **#P**-hard
Complexity of *Approximate* Counting

- Given efficiently computable \( f: \{0,1\}^n \rightarrow \{0,1\} \) and \( y \in \{0,1\} \)
  - Want to compute \( \Pr_{x \sim U}[f(x)=y] \) exactly
  - This is \#P-hard
    - Because \( \Pr_x[f(x)=1] = \# x \text{’s so that } f(x)=1)/2^n = \sum_x f(x)/2^n \)
    - This is as hard as counting number of satisfying assignments to formula \( \Psi \)
- However, *estimating* \( \Pr_{x \sim U}[f(x)=y] \) to within multiplicative error can be done in \( \Sigma_3 \), the third level of \( \text{PH} \) [Stockmeyer ’83]
  - So for input \( f: \{0,1\}^n \rightarrow \{0,1\} \) and \( \varepsilon>0 \) can output \( \alpha: \)
    \[
    (1 - \varepsilon) \sum_x f(x) \leq \alpha \leq (1 + \varepsilon) \sum_x f(x)
    \]
    in time \( \text{poly}(n,1/\varepsilon) \) with \( \Sigma_3 \) oracle
- But, situation is very different for \( g: \{0,1\}^n \rightarrow \{+1,-1\} \)
  - Computing \( \Sigma_x g(x) \) exactly is still \#P-hard
  - Estimating \( \Sigma_x g(x) \) to within \((1 \pm \varepsilon)\) *multiplicative* error is \#P-hard!
    - Binary search & Padding
    - Can generalize this hardness:
    - Estimating \((\Sigma_x g(x))^2\) to within \((1 \pm \varepsilon)\) *multiplicative* error is \#P-hard
  - Why is this so much harder than the \{0,1\}-valued case?
    - Cancellations
Today

• Want to show that quantum computers are capable of sampling from distributions that cannot be sampled by randomized classical algorithms
• Two constructions of hard distributions
  1. “Exact” construction
     • No classical algorithm can sample from exactly the same distribution as the quantum algorithm
  2. “Approximate” construction
     • Goal: Show no classical algorithm can sample from any distribution even close (in total variation distance) to quantum distribution
     • Why do we want to do this?
       – “To model error”
       – [Aaronson ‘11] has shown that such a result would imply a “function problem” complexity separation (i.e., $\text{fBQP} \not\subset \text{fBPP}$)...

       – Upshot: We’ll reach many of the same conclusions of the BosonSampling [AA’10] proposal with a (conceptually) much simpler setup. Our proposal also weakens the hardness conjectures needed by [AA’10], but as of yet does not resolve them....
II. “Exact” Construction [implicit in Aaronson ‘11]
Quantumly sampleable distribution

• **Recall:** For efficiently computable function $g : \{0,1\}^n \rightarrow \{\pm 1\}$, giving a $(1 \pm \varepsilon)$ mult error estimate to $(\sum g(x))^2$ is $\mathbf{#P}$-hard

• Consider the following quantum circuit:

  $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$

  $\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} g(x) |x\rangle$

  $\frac{1}{2^n} \sum_{y \in \{0,1\}^n} \sum_{x \in \{0,1\}^n} -1^{(x,y)} g(x) |y\rangle$

**Key point:** The probability of seeing 00...0 is $(\sum g(x))^2 / 2^{2n}$

Heilbronn/QALGO Quantum Algorithms Meeting, Cambridge University
Exact classical sampler collapses $\text{PH}$

• Suppose $C$ is a randomized algorithm that samples the outcome distribution so by definition:

$$\Pr_{r \sim U_{p(n)}}[C(r) = y] = \frac{1}{2^{2n}} \left( \sum_{x \in \{0,1\}^n} -1^{\langle x,y \rangle} g(x) \right)^2$$

• Note that $p = \Pr_r[C(r)=00...0] = (\sum_x g(x))^2/2^{2n}$ encodes a $\text{#P}$-hard quantity

• Use Stockmeyer’s algorithm to find a $(1 \pm \varepsilon)$ multiplicative error estimate to $p$

• Puts $\text{P}^\text{#P} \subseteq \Sigma_3$ (but Toda tells us that $\text{PH} \subseteq \text{P}^\text{#P}$)

• $\text{PH} \subseteq \Sigma_3$ (collapse!!)
How *robust* is this prior construction?

• Not very!!
  – Hardness based on a single exp. small probability
  – *Definition*: For distribution $X$ over $\{0,1\}^n$:

    
    Given as input $\varepsilon>0$, suppose a classical randomized algorithm samples from any distribution $Y$, with $|X-Y|_1<\varepsilon$, in time $\text{poly}(n,1/\varepsilon)$

    Call such a classical algorithm an “Approximate Sampler” for $X$

• **Our goal**: Find a quantumly sampleable $X$, where the existence of a classical “Approximate Sampler” would cause **PH** collapse.

• Prior construction doesn’t work! (Adversary just ”erases” probability we care about)
III. “Approximate” Construction using Quantum Fourier Sampling [F., Umans ‘15]
Construction of distribution $D_{\text{PER}}$

- Define an efficiently computable function $h:[n!]\rightarrow\{0,1\}^{n^2}$
  - Takes a permutation in $S_n$ to its trivial encoding as an $n \times n$ permutation matrix
  - Can be computed efficiently using e.g., Lehmer codes
  - Note $h$ is 1-to-1 and $h^{-1}$ also efficiently computable

- Quantum sampler:
  - Two steps:
    1. Prepare uniform superposition over $n \times n$ permutation matrices
      - Prepare uniform superposition over $S_n$
      - Apply $h$, followed by $h^{-1}$
    2. Hit with Hadamard on each of $n^2$ qubits
  - Measure in standard basis

\[
\frac{1}{\sqrt{n!}} \sum_{\sigma \in S_n} |\sigma\rangle |00\ldots0\rangle \\
\frac{1}{\sqrt{n!}} \sum_{\sigma \in S_n} |\sigma\rangle |h(\sigma)\rangle \\
\frac{1}{\sqrt{n!}} \sum_{\sigma \in S_n} |\sigma \oplus h^{-1}(h(\sigma))\rangle |h(\sigma)\rangle \\
\frac{1}{\sqrt{n!}} \sum_{\sigma \in S_n} |h(\sigma)\rangle \\
\frac{1}{\sqrt{n!2^{n^2}}} \sum_{w \in \{0,1\}^{n^2}} \sum_{\sigma \in S_n} (-1)^{\langle w, h(\sigma) \rangle} |w\rangle
\]

This is the permanent of $\{\pm 1\}^{n \times n}$ matrix encoded by the string $w$
What’s happening?

- Recall, \textbf{Permanent}(x_1, x_2, \ldots, x_{n^2}) is a multilinear polynomial of degree n.
- Our quantum sampling algorithm (\textit{omitting normalization}):

  All possible multilinear monomials over \(n^2\) variables \(M_1, \ldots, M_{2^{n^2}}\)

\[
\begin{pmatrix}
1 \\
0 \\
\vdots \\
0 \\
1 \\
0 \\
\end{pmatrix}
= 
\begin{pmatrix}
\Per[X_1] \\
\Per[X_2] \\
\vdots \\
\Per[X_{2^{n^2}}]
\end{pmatrix}
\]

This is supported on the monomials in the \textbf{Permanent}.
Approximate sampler consequences

- Each $Y \in \{-1,+1\}^n \times \{1\}$, the probability of outcome $Y$ according to $D_{\text{PER}}$:

  - Suppose we had an approximate sampler, $A$, for $D_{\text{PER}}$—Unlike "exact case" can't trust the sampler on any single probability
  - But, if $A$ samples from distribution $\epsilon \delta$-far from $D_{\text{PER}}$ we know:
    - "Most probabilities in $A$'s distribution must be close to probabilities in $D_{\text{PER}}"$
    - At least $(1-\delta)$-fraction of probabilities must be within $\epsilon/2^{n^2}$ of true probability
  - **Strategy**: Choose a $Z \in \{-1,+1\}^{n^2}$ matrix with iid uniformly distributed entries and approximate its probability using Stockmeyer’s algorithm

- We’d obtain solution that “solves Per$^2(X)$” in $\Sigma_3$ with two major caveats:
  - Only “works” with probability $1-\delta$ over choice of matrix
  - “Works” means approximating within additive error $\pm \epsilon n$!

- **Our question**: How hard is this?
  - If it’s $\#P$-hard, by Toda’s theorem, an approximate sampler for $D_{\text{PER}}$ would imply a $\text{PH}$ collapse (as in the exact case)
Relating Additive to Multiplicative error

• Our procedure computes:
  • \( \text{Per}^2[X] \pm \varepsilon n! \) with probability \( 1 - \delta \) in \( \Sigma_3 \)-time \( \text{poly}(n, 1/\varepsilon, 1/\delta) \) time

• This is unnatural! Would like multiplicative error:
  • \((1 - \varepsilon)\text{Per}^2[X] \leq \alpha \leq (1 + \varepsilon)\text{Per}^2[X] \) with probability \( 1 - \delta \) in \( \Sigma_3 \)-time \( \text{poly}(n, 1/\varepsilon, 1/\delta) \) time

• Can we get multiplicative error using our procedure?
  • “Permanent Anti-concentration conjecture” [AA’11]
    • Need: exists polynomial \( p \) so that for all \( n \) and \( \delta \)
      – \( \Pr[X]|\text{Per}(X)| < \sqrt{(n!)/p(n, 1/\delta)} < \delta \)
    • This may actually be true!!
    • For Bernoulli distributed \( \{-1, +1\}^{n \times n} \) matrices:
      • \( \forall \varepsilon > 0 \Pr[X]|\text{Per}[X]|^2 < n!/n^{en} < 1/n^{0.1} \) [Tao &Vu ’08]
How hard is “Approximating” the Permanent?

• **Scenario 1:**
  – Suppose I had a box that:
    • “Solves all the Permanents approximately”
    • Input: \( \epsilon > 0 \) and matrix \( X \in \{-1,+1\}^{n \times n} \)
    • Output: \( \alpha \) so that:
      \[
      (1 - \epsilon) \text{Per}^2(X) \leq \alpha \leq (1 + \epsilon) \text{Per}^2(X)
      \]
    • In time \( \text{poly}(n,1/\epsilon) \)
  – This is \#P-hard!
    • Proof: “Padding and binary search!”

• **Scenario 2:**
  – Suppose I had a box that:
    • “Solves most of the Permanents exactly”
      \[
      \Pr_X[\alpha = \text{Per}^2[X]] > 1 - \delta
      \]
    • For \( \delta = 1/\text{poly}(n) \)
  – This is \#P-hard!
    • Proof idea: Polynomial interpolation [Lipton ‘89 in finite field case...!]

• Our ”solution” has weakness of both Scenario 1 and 2
  – Hardness proofs break-down!
  – This is exactly the same reason other two “approximate” sampling results need conjectures...
Generalizations

• Entries of Matrix
  – Replace Quantum Fourier Transform over $\mathbb{Z}_2^{n^2}$ with Quantum Fourier Transform over $\mathbb{Z}_k^{n^2}$
    • Resulting amplitudes proportional to Permanents of matrices with entries of evenly-spaced points around unit circle

• Generalizing the distribution over matrices
  – Can recapture the Gaussian distributed entries of [AA’11]...

• “Hard Polynomial”
  – Generalize Permanent to any Efficiently Specifiable polynomial sampling
    • Multilinear, homogenous polynomials with $m$ monomials of the form:
      $$Q(X_1, X_2, \ldots, X_n) = \sum_{y \in [m]} X_1^{h(y)_1} X_2^{h(y)_2} \ldots X_n^{h(y)_n}$$
    • Where $h$ is efficiently computable map (and $h^{-1}$ is also)

  – Examples:
    • Permanent, Hamiltonian Cycle polynomial, many more...

Heilbronn/QALGO Quantum Algorithms
Meeting, Cambridge University
Relation to other work

• There are lots of “exact” sampling results
  – Starting with [DiVincenzo-Terhal’02] and [Bremner-Jozsa-Shepherd’10]
  – These distributions can often be sampled by restrictive classes of quantum samplers
    • Constant depth quantum circuits [DT’02]
    • Quantum computations with commuting gates [BJS’10]
    • One clean qubit [Morimae et. al. 2014]
    • Etc...

• “Approximate” sampling is far less understood...
  – “Boson Sampling” [Aaronson and Arkhipov ’11]
  – “IQP Sampling” [Bremner, Montanaro and Shepherd’15]
  – Quantum Fourier Sampling [F.,Umans ’15]

• All rely on similar non-standard hardness assumptions
  – Need to conjecture that computing “average-case approximate” solution to some polynomial is hard for the PH
    • Permanent [AA’11]
    • The partition function of a random instance of an Ising model [BMS’15]
    • Any Efficiently Specifiable polynomial [F., Umans ‘15]
Thanks!