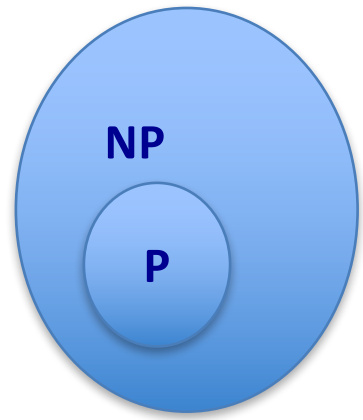
The Power of Quantum Fourier Sampling

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

Based on arxiv:1507.05592

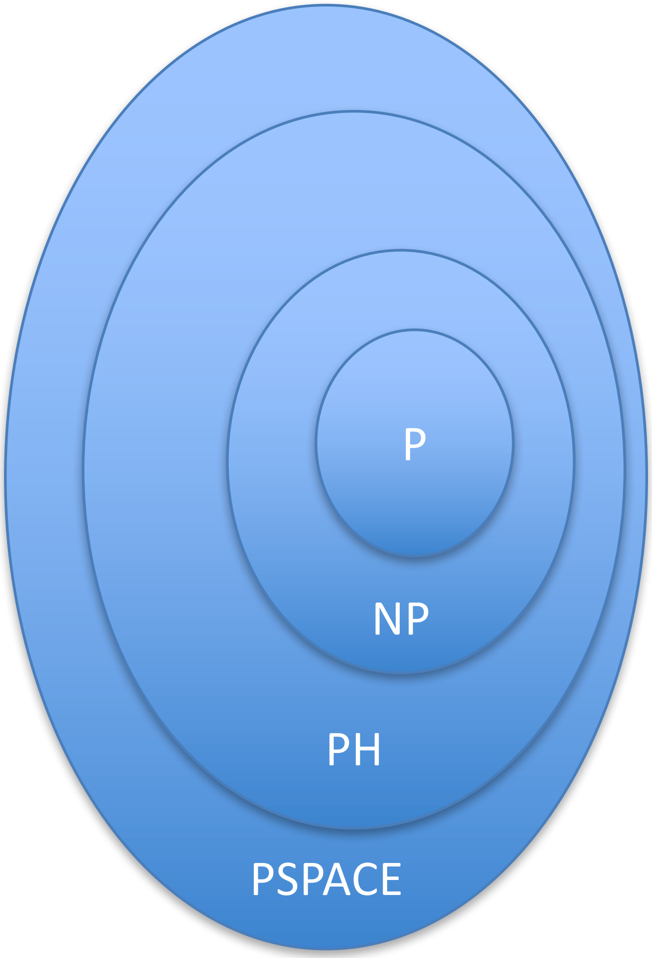
Classical Complexity Theory

- **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 boolean formula
        - » E.g., $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee - x_2 \vee x_6) \wedge \ldots$
    - Problem: $\exists\, x_1, x_2, \ldots, x_n$ so that $\Psi(x)=1$?
  - **SAT** is **NP**-complete



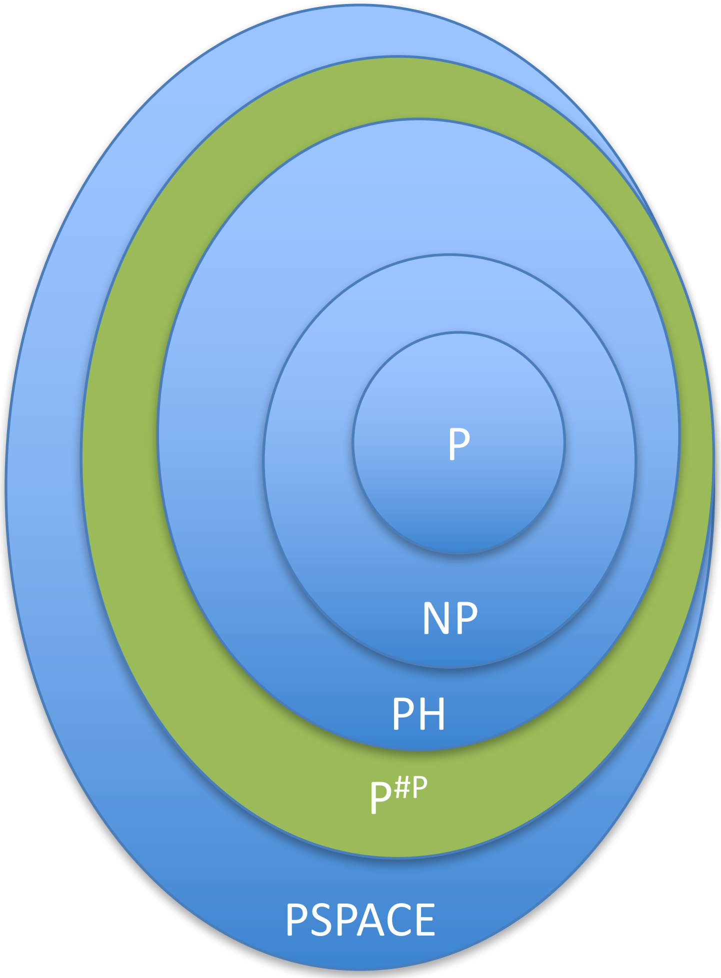
Beyond **NP**

- **Tautology**
  - Input: $\Psi:\{0,1\}^n \rightarrow \{0,1\}$
  - $\forall x\ \Psi(x)=1$?
  - Complete for **coNP**
  - Don't believe that **coNP**=**NP**
- Generalize **SAT** and **Tautology** by adding quantifiers:
  - **QSAT$_2$** is the version of the **SAT** problem with 2 quantifiers
  - E.g., $\exists x_1 x_2 x_3 \ldots x_{n/2} \forall x_{n/2+1} x_{n/2+2}, \ldots, x_n$ so that $\Psi(x)=1$?
  - Consider problems **QSAT$_3$,QSAT$_4$,QSAT$_5$...QSAT$_n$**
  - Conjectured to get strictly harder with increasing number of quantifiers (or else there's a *collapse*!)
- **Σ_k** is class of problems solvable with a **QSAT$_k$** box
- **PH** is class of problems solvable with a **QSAT$_{O(1)}$** box
- **PSPACE** is class of problems solvable with a **QSAT$_n$** box



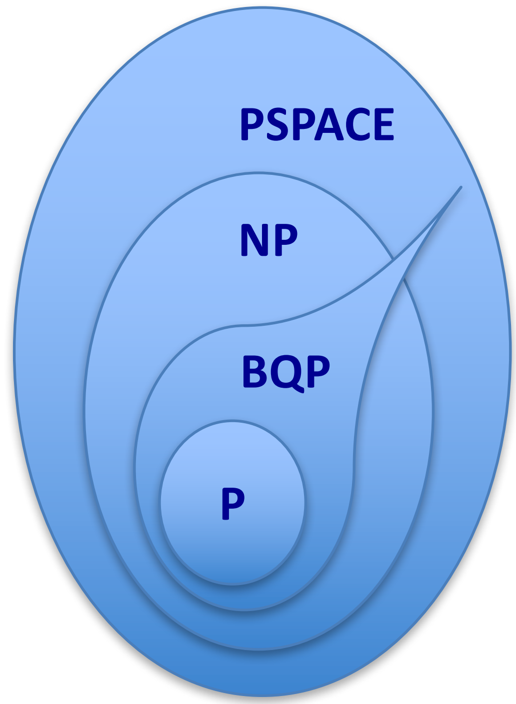
Complexity of *Counting*

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



How powerful are quantum computers?

- **BQP**: The class of *decision* problems solvable by quantum computers in polynomial time
- Certainly **P**$\subseteq$**BQP**
- But why should **BQP**$\not\subseteq$**P** (or **NP** or **PH**)?
  - Shor's algorithm: Factoring$\in$**BQP**
    - But little reason to believe Factoring is not in **P**
    - In fact, if Factoring is **NP**-hard then **PH** collapses
  - Oracle separations, see [e.g., Aaronson'10, F., Umans'11]
  - In short, not much is known!



Separations from sampling problems

- Starting with [DT'02][BJS'10] we know that there are *distributions* that can be sampled quantumly that cannot be sampled *exactly* classically (unless **PH** collapse)
  - *Quantumly*: Efficiently prepare a quantum state on n qubits and measure in standard basis
    - Distribution is over measurement outcomes
  - *Classically*: No efficient classical randomized algorithm can sample from *exactly* the same distribution
- *Our focus*: "Approximate sampling" hardness result
  - Want a hardness result even if the classical sampler samples from distribution 1/poly(n) close in total variation distance from quantum distribution
  - Why are we interested in this?
    - "To model experimental error"
    - Other complexity separations would follow (i.e., **fBQP**$\not\subset$**fBPP** [Aaronson'10])

Construction of quantumly sampleable distribution $\mathbf{D}_{\mathbf{PER}}$

- *Goal*: efficiently prepare a quantum state in which *each* amplitude is proportional to the **Permanent** of a different matrix
- *Sketch of procedure*:
  1. Prepare the "permutation matrix state"
     - Quantum state on n^2 qubits uniformly supported only on those n! permutation matrices
  2. Apply a quantum Fourier transform $H^{\otimes n^2}$
     - i.e., apply Hadamard on each of n^2 qubits
  3. Measure in standard basis to sample
- *Claim*: Each amplitude is proportional to the Permanent of a different $\{\pm 1\}^{n \times n}$ matrix

What's happening?

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

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

All possible matrices $X_1,X_2,\ldots,X_{2^\wedge\{n^\wedge 2\}} \in \{\pm 1\}^{n^\wedge 2}$

$$\begin{array}{c} \overbrace{\qquad\qquad\qquad\qquad\qquad\qquad}^{\text{All possible multilinear monomials over } n^\wedge 2 \text{ variables } M_1,\ldots M_{2^\wedge\{n^\wedge 2\}}} \\ \left.\begin{matrix} M_1(X_1), M_2(X_1), \ldots, M_{2^\wedge\{n^\wedge 2\}}(X_1) \\ \\ H^{\otimes n^2} \\ \\ M_1(X_{2^\wedge\{n^\wedge 2\}}), \ldots, M_{2^\wedge\{n^\wedge 2\}}(X_{2^\wedge\{n^\wedge 2\}}) \end{matrix}\right. \end{array} \begin{pmatrix} 1 \\ 0 \\ 0 \\ \ldots \\ 0 \\ 1 \\ 0 \\ \ldots \end{pmatrix} = \begin{pmatrix} \mathbf{Per}[X_1] \\ \mathbf{Per}[X_2] \\ \\ \ldots \\ \\ \mathbf{Per}[X_{2^\wedge\{n^\wedge 2\}}] \end{pmatrix}$$

This is supported on the monomials in the **Permanent**

Classical hardness sketch

- *Recall*: $\mathbf{D_{PER}}$ is a distribution over all $\{\pm 1\}^{n \times n}$ matrices X with probabilities proportional to $\mathbf{Permanent^2}[X]$
- Assume there's a classical algorithm that samples from distribution close in total variation distance to $\mathbf{D_{PER}}$
- *Key tool*: Stockmeyer's algorithm
  - *Input*: Classical sampler and an outcome
  - *Output*: A $(1\pm\varepsilon)$-*multiplicative estimate* to the probability of this outcome in time $\text{poly}(n,1/\varepsilon)$ with an **NP** oracle
    - i.e., for $\varepsilon=1/\text{poly}(n)$, this is in $\mathbf{BPP^{NP}} \subseteq \mathbf{\Sigma_3}$
- *Our strategy*: Chose a random $\{\pm 1\}^{n \times n}$ matrix X and use Stockmeyer's algorithm to estimate outcome probability of X $\approx$ $\mathbf{Permanent^2}[X]$
  - Since our sampler is *approximate*, can't trust it on any single outcome probability
  - Markov inequality: *Most* of the probabilities must be *close* to the true probabilities
  - So we end with a $\mathbf{BPP^{NP}}$ algorithm for *estimating* the $\mathbf{Permanent^2}$ of *most* matrices
- Is estimation task **#P**-hard? If so then $\mathbf{P^{\#P}} \subseteq \mathbf{BPP^{NP}} \subseteq \mathbf{\Sigma_3}$
  - But we know that $\mathbf{PH} \subseteq \mathbf{P^{\#P}}$ by Toda's theorem
  - So $\mathbf{PH} \subseteq \mathbf{\Sigma_3}$ (Collapse!)

How hard is "Approximating" the Permanent?

- *Our result:* If there is an approximate sampler for $D_{\mathbf{PER}}$ then there's a **PH** algorithm that "computes Permanent" with two *caveats*:
  1. Only "works" with high probability (over choice of matrix)
  2. "Works" means obtains a multiplicative estimate
- We can show that either of these weaknesses alone would be **#P**-hard!
- Don't know how to prove #P-hardness for both of these weakness!
  – This is exactly the same reason other two "approximate" sampling results need conjectures [Aaronson and Arkhipov, Bremner, Montanaro and Shepherd]…

Generalizing the argument

- Unlike the results of [Aaronson & Arkhipov '12] and [Bremner, Montanaro & Shepherd '16] we can generalize our argument to rely on alternative hardness conjectures
  - Can generalize the **Permanent** to any "*efficiently specifiable* polynomial"
  - Can generalize the entries of the matrices and the distribution over matrices (e.g., iid Gaussian instead of random sign matrix)
- If any of these conjectures are true, we show the desired "approximate sampling" separation

Thanks!