Complexity Theory

Lecture 12

Quantum Complexity

Mathematical abstraction of quantum computing

Quantum mechanics is an ℓ_2 probability theory

Fully captured by 4 postulates:

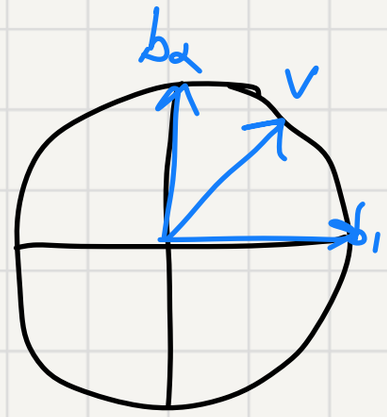
(I) Superposition

A quantum state is a vector $V \in \mathbb{C}^n$ s.t. $\|V\|_2^2 = 1$.

Ex: a qubit $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$, $\alpha^2 + \beta^2 = 1$

(II) Measurement

Measuring state $V \in \mathbb{C}^n$ in orthonormal basis $\{b_1, \ldots, b_n\}$ collapses to b_i w.p. $|\langle V, b_i \rangle|^2$



Mathematical abstraction of quantum computing

(III) Evolution

A quantum state $v \in \mathbb{C}^n$ evolves to $v' \in \mathbb{C}$ via a unitary map

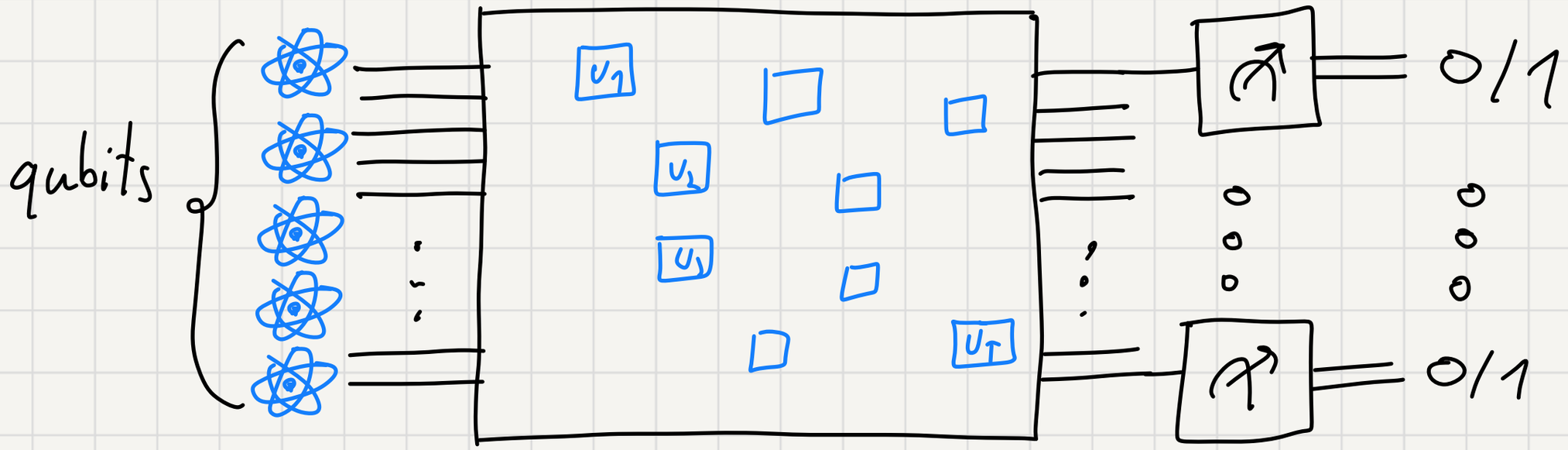
$$\begin{pmatrix} | \\ v' \\ | \end{pmatrix} = \begin{pmatrix} & \cup & \end{pmatrix} \cdot \begin{pmatrix} | \\ v \\ | \end{pmatrix}$$

(IV) Entanglement

States $u, v \in \mathbb{C}^n$ are composed via the tensor product $u \otimes v$.

Ex: $\begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 000 \\ 0 & 001 \\ 0 & 010 \\ 0 & 011 \\ 0 & 100 \\ 0 & 101 \\ 0 & 110 \\ 0 & 111 \end{pmatrix}$

Quantum algorithms



3 examples where quantum algorithms excel:

(I) Finding sub-group structure (Shor's factoring)

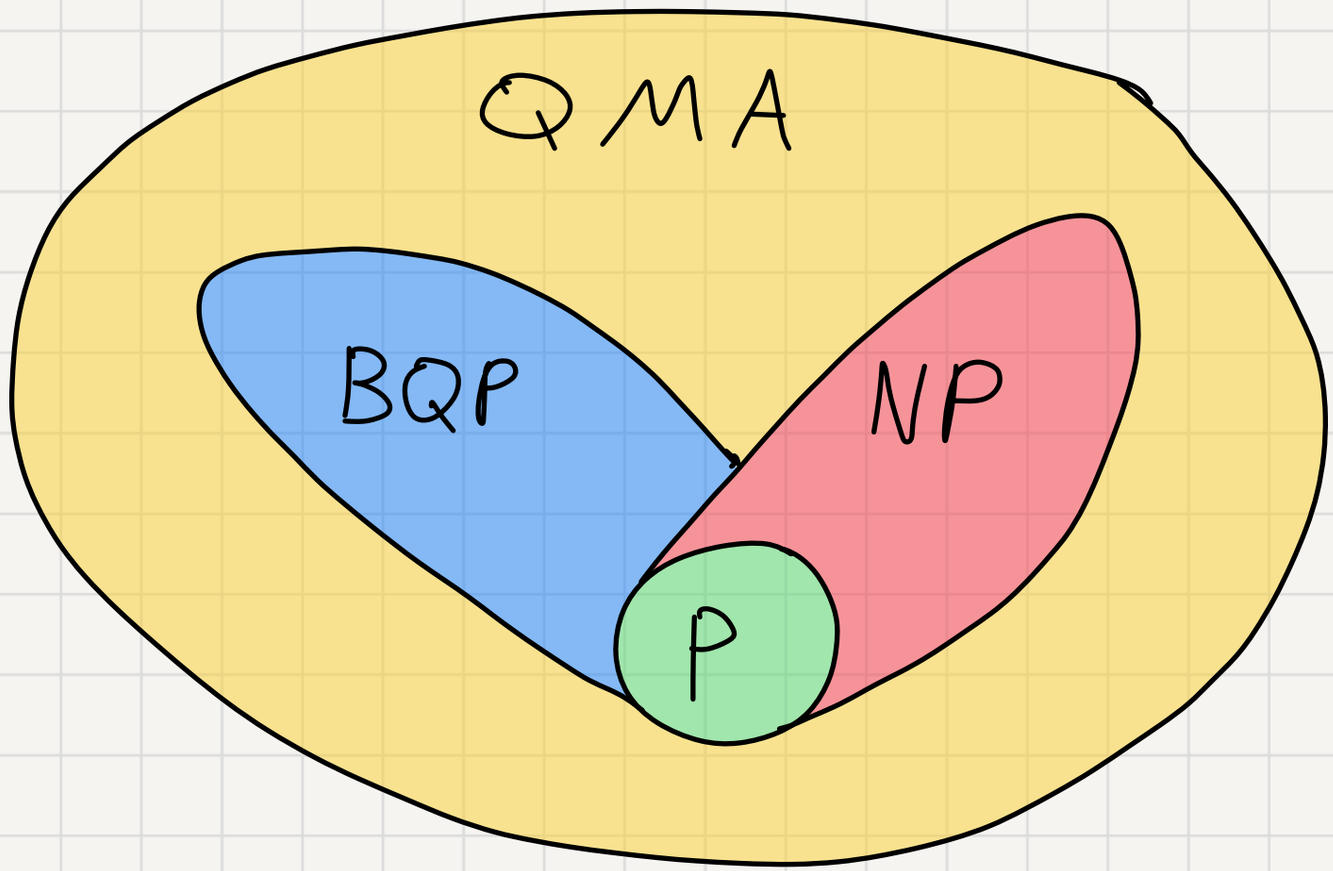
(II) Rapid mixing of Markov chains (Grover's search)

(III) Computing Fourier Transforms (QFT)

Quantum complexity classes

BQP = "quantum P"

QMA = "quantum NP"



BQP

The set of all problems solvable by a poly-time uniform quantum circuits $(C_n)_{n \in \mathbb{N}}$ of polynomial size, w.p. $\geq 2/3$

$T(n)$-uniformity: C_n can be generated in $T(n)$ time

Circuit size: # gates

Error reduction

Let A be a q-algo for computing f such that $Pr[A(x) = f(x)] \geq 2/3 \quad \forall x$.

$1/3$ error prob. can be reduced to ε!

Repeat A: $A_1, A_2, \ldots, A_{\underbrace{\log(\frac{1}{\varepsilon})}_{t}}$, rule by Maj.

Chernoff bound:

$$Pr\left[\left|\frac{\sum_{i \in [t]} A_i}{t} - \frac{2}{3}\right| \leq \frac{1}{6}\right] \leq \exp(-t)$$

Factoring

Given $n \in \mathbb{N}$, output primes $p_1, \ldots, p_n$ s.t.

$n = p_1 \cdot p_2 \cdot \ldots \cdot p_n$

Decision problem Factor$(n, k) = 1$

iff n has a prime factor $\leq k$

Shor's algorithm Factor $\in$ BQP

We know Factor $\in$ NP $\cap$ coNP.