Introduction

In this presentation, we will:

- Review classical complexity classes
- Introduce **QMA** (the quantum analogue of **NP**)
- Show that the local Hamiltonians problem is **QMA**-complete

Problems and Languages

- We will only consider decision problems (where the output is in $\{0, 1\}$)
- This can be formulated as testing if a string $x \in \{0, 1\}^*$ is in some language $L \subseteq \{0, 1\}^*$ which describes the problem we are considering
- Strings x for which the output is 0 are called no-instances and strings for which the output is 1 are called yes-instances
- We'll assume we're using a RAM machine; this is equivalent to using a Turing machine up to polynomial factors

Deterministic complexity classes I

- **P** denotes the class of all decision problems can be solved in deterministic polynomial-time
- **NP** is the class of problems for which yes-instances can be verified efficiently by a deterministic algorithm

Definition

$L \in \mathbf{NP}$ if there exists a deterministic polynomial-time algorithm A and a polynomial $p(n)$ such that
$x \in L \Leftrightarrow \exists w \; |w| \leq p(n) \wedge A(x, w) = 1$

Deterministic complexity classes II

- One can also think of **NP** in terms of the game where Arthur and Merlin are given an input x and Arthur must decide if $x \in L$
- Merlin has unlimited computational resources and must send a witness w to Arthur; his goal is to get Arthur to conclude that $x \in L$
- Arthur runs a polynomial-time computation on x, w
  - If $x \in L$, we require that it is possible for Merlin to convince Arthur that this is that case by sending some w
  - If $x \notin L$, we require that — no matter what w Merlin provides to Arthur — he cannot trick Arthur into concluding that $x \in L$

Reductions

- Reductions allow us to compare the hardness of different problems

Definition

L_1 is Karp-reducible to L_2 (denoted $L_1 \leq_P L_2$) if there exists a deterministic polynomial-time algorithm A such that $x \in L_1 \Leftrightarrow A(x) \in L_2$

- We'll only deal with Karp-reductions in this talk, so from now on we'll just refer to these as reductions

Definition

L is **NP**-hard if every language in **NP** is reducible to L

Definition

L is **NP**-complete if $L \in$ **NP** and it is **NP**-hard

The Cook-Levin Theorem

Theorem (Cook-Levin)

SAT is **NP***-complete*

- Many important problems such as SAT, independent set, subset sum, etc. are **NP**-complete
- One can reduce SAT to k-SAT when $k \geq 3$ so k-SAT is also **NP**-complete

Randomized complexity classes I

- **BPP** denotes the class of all problems can be solved in bounded-error probabilistic polynomial-time

Definition

$L \in$ **BPP** if there exists a randomized polynomial-time algorithm A such that

- $x \in L \Rightarrow \Pr(A(x) = 1) \geq 2/3$
- $x \notin L \Rightarrow \Pr(A(x) = 1) \leq 1/3$

Randomized complexity classes II

- **MA** is the class of problems for which yes-instances can be verified efficiently by a randomized algorithm

Definition

$L \in$ **MA** if there exists a randomized polynomial-time algorithm A and a polynomial $p(n)$ such that

- $x \in L \Rightarrow \exists w \ |w| \leq p(n) \wedge \Pr(A(x, w) = 1) \geq 2/3$
- $x \notin L \Rightarrow \forall w \ |w| \leq p(n) \wedge \Pr(A(x, w) = 1) \leq 1/3$

Randomized complexity classes III

- Similarly to **NP** , we can think of **MA** in terms a game where Merlin sends a witness to Arthur
- The only difference is that now we only require that Arthur gets the right answer with bounded-error
  - If $x \in L$, we require that Merlin can send some witness w which will convince Arthur that $x \in L$ with probability at least $2/3$
  - If $x \notin L$, we require that Merlin cannot trick Arthur into concluding that $x \in L$ with probability more than $1/3$

Quantum complexity classes I

- **BQP** denotes the class of all problems which can be solved in bounded-error quantum polynomial-time

Definition

$L \in$ **BQP** if there exists a quantum polynomial-time algorithm A such that

- $x \in L \Rightarrow \Pr(A(x) = 1) \geq 2/3$
- $x \notin L \Rightarrow \Pr(A(x) = 1) \leq 1/3$

Quantum complexity classes II

- **QMA** is the class of problems for which yes-instances can be verified efficiently by a quantum algorithm

Definition

$L \in$ **QMA** if there exists a quantum polynomial-time algorithm A and a polynomial $p(n)$ such that

- $x \in L \Rightarrow \exists |w\rangle \in \mathbb{C}^{2^{p(n)}} \Pr(A(x, |w\rangle) = 1) \geq 2/3$
- $x \notin L \Rightarrow \forall |w\rangle \in \mathbb{C}^{2^{p(n)}} \Pr(A(x, |w\rangle) = 1) \leq 1/3$

- Similarly to **MA** , we can think of **QMA** in terms a game where Merlin sends a witness to Arthur
- The only difference is that the witness is now a quantum state $|w\rangle$

The k-local Hamiltonians problem

- Given: *classical* descriptions of r positive-semidefinite k-local Hamiltonians H_i of norm at most 1 and two positive real numbers a and b such that $b - a \geq 1/\mathrm{poly}(n)$
- Goal: determine if the smallest eigenvalue of $H = \sum_i H_i$ less than a or if all eigenvalues are greater than b
- All inputs are specified to $\mathrm{poly}(n)$ bits of precision
- We'll call this problem k-HAM from now on
- It's worth noting that 3-SAT can be reduced to 3-HAM by creating a 3-local projector for each clause in the 3-SAT formula which introduces a penalty whenever that clause is not satisfied

QMA-completeness of 5-HAM

- We will now show Kitaev's proof that 5-HAM is **QMA**-complete
- There are two steps. We must show that
  - 5-HAM $\in$ **QMA** and
  - 5-HAM is **QMA**-hard
- The first is fairly easy while the second is more involved

k-HAM $\in$ **QMA** I

- Since k is constant, we can compute each spectral decomposition $H_i = \sum_j w_j^i \left|\alpha_j^i\right\rangle \left\langle\alpha_j^i\right|$ in constant time
- Moreover, each state $\left|\alpha_j^i\right\rangle$ has support only on k qubits so it can be prepared by some unitary U_j^i in constant time
- This implies that we can control by this state by applying $U_j^{i\dagger}$ so that we can implement the operator defined by $T_i \left|\alpha_j^i\right\rangle |0\rangle = \left|\alpha_j^i\right\rangle \left(\sqrt{w_j^i}\,|0\rangle + \sqrt{1 - w_j^i}\,|1\rangle\right)$ in $\mathrm{poly}(r, n)$ time
- Consider any state $|\eta\rangle\,|0\rangle$ and suppose we apply T_i to this state and then measure the second register in the computational basis
- Using the Schmidt decomposition, one can show that this probability is $1 - \langle\eta|\, H_i\, |\eta\rangle$

k-HAM $\in$ **QMA** II

- The verification procedure consists of choosing an $i \in [r]$ uniformly at random and then applying the above procedure; the probability of observing 1 is $1 - \langle\eta| H |\eta\rangle / r$
- If H is a yes-instance and $|\eta\rangle$ is the ground state then $1 - \langle\eta| H |\eta\rangle / r \geq 1 - a/r$
- If H is a no-instance then $1 - \langle\eta| H |\eta\rangle / r \leq 1 - b/r$

Proof of the Cook-Levin Theorem

- The proof that 5-HAM is **QMA**-hard follows the proof of the Cook-Levin theorem which we will now review
- For a *fixed* input size n, any Turing machine that runs in $\mathrm{poly}(n)$ time can be simulated by a boolean circuit of size $\mathrm{poly}(n)$
- By constructing such a circuit for the verifier for a **NP** problem, we can show that CIRCUIT-SAT is **NP**-hard
- It's clear that CIRCUIT-SAT is in **NP** so this shows it is **NP** -complete
- Since we can also reduce CIRCUIT-SAT to 3-SAT, it follows that 3-SAT is also **NP**-complete
- To prove that 5-HAM is **QMA**-hard, we will construct a set of 5-local Hamiltonians which simulate the quantum circuit that serves as the verifier

5-HAM is **QMA**-hard I

- Consider $L \in \mathbf{QMA}$; our goal is to reduce L to 5-HAM
- We know that there exists a quantum circuit $Q = U_T \cdots U_1$ of size $T = \mathrm{poly}(n)$ which takes as input $|x\rangle\,|\xi\rangle$ and outputs 1 if $|\xi\rangle$ is a witness that $x \in L$; each U_i is a two-qubit gate
- We'll start by reducing L to $O(\log(n))$-HAM and then show how to make the resulting Hamiltonian 5-local
- Consider a state of the form $\frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} U_t \cdots U_1\,|x\rangle\,|\xi\rangle$; we will design a Hamiltonian with this as the ground state
- The term $H_{in} = \sum_i \Pi_i^{\neg x_i} \otimes |0\rangle\langle 0|$ (where Π_i^b is the projector onto the states where the i^{th} qubit is equal to b) creates an energy penalty whenever the input state is not $|x\rangle$
- The term $H_{out} = \Pi_1^0 \otimes |T\rangle\langle T|$ adds an energy penalty whenever the output is not 1 (i.e. when the computation did not accept)

5-HAM is **QMA**-hard II

- The term

$$H_{prop}(t) = \frac{1}{2}\left(I \otimes |t\rangle\langle t| - U_t \otimes |t\rangle\langle t-1| + I \otimes |t-1\rangle\langle t-1| - U_t^\dagger \otimes |t-1\rangle\langle t|\right)$$

  Adds a penalty unless the state at time t was obtained from the state a time $t-1$ by U_t
- Let $H_{prop} = \sum_{t=0}^{T} H_{prop}(t)$ and $H = H_{in} + H_{out} + H_{prop}$
- At this point, there is one problem left which is that H is $O(\log n)$-local
- We can make it 5-local by using a unary representation instead of a binary representation for the clock register $|t\rangle$
- The value 5 comes from using two qubit unitaries in the computation register and three qubit projectors in the clock register
- Note that formalizing the above proof sketch is non-trivial!