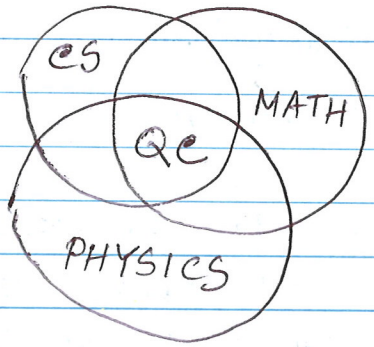
Lecture 1 (10/2/2019)

Quantum computation (QC) sits at intersection of CS + MATH + PHYSICS.



A lot of interest in last ~5 years (and growing!).

* A fun resource on QC (Scott Aaronson's blog)
  scottaaronson.com/blog/
  "Mention Scott's quote".

>>> History of QC

1. "Church-Turing thesis" (Alan Turing, Alonzo Church) (1936): UTM (universal Turing machine).
   If an algorithm can be performed on any piece of hardware, then there is an equivalent algorithm for UTM.

2. "Strong Church-Turing thesis" (1960-1970)
   A problem that has an efficient solution in some model of computation, has an efficient solution using UTMs.
   — analog computation seem to pose a challenge to Strong Church-Turing thesis.
   — advantage disappears if you consider noise.

3. Other challenges to Strong Church Turing thesis

"Randomized Algorithms" (1970s).

- Soloray-Strassen primality testing (outputs if a number is probably prime or composite with certainty).
  - Amplify using repetition.

Question: Suppose an algorithm succeeds with probability $p > 0$, & has polynomial runtime. Can you design an algorithm that succeeds with probability $1 - \varepsilon$? ($\varepsilon > 0$, fixed).

YES, by repeating the algorithm.

Q. How many times do you need to repeat? (HW).

Resolution: "Any algorithmic process can be simulated efficiently using a probabilistic Turing machine." (Non-deterministic Turing machine).

Note: Not all problems can be solved.

eg. Halting problem (undecidable). (Russel's Paradox) (HW)

Actual proof is complicated. on UTM

Basic idea.

HALT(P, Q)
- ▶ YES IF P HALTS ON Q
- ▶ NO otherwise.

NEW HALT(P)
if HALT(P, P) output YES
else output NO

REVERSE(P)
if NEW HALT(P) output YES, then loop forever
else halt

"Diagonalization arguments"

4. Primality testing

Randomized (compositeness test).

- Miller Rabin.
- Solovay-Strassen (HW).
- Frobenius primality test. (3 times more expensive than M-R, but achieves prob. bound eqv. to 7 rounds of M-R).
- Ballie-PSW primality test.
  (no known counter examples for $n < 2^{64}$).

Deterministic tests

AKS Test (2002) (Agrawal, Kayal, Saxena).

$\tilde{O}((\log n)^6)$ → currently the best version (Lenstra, Pomerance, 2005).

- original $\tilde{O}((\log n)^{12})$
- published $\tilde{O}((\log n)^{7.5})$
- $\hat{O}((\log n)^6)$ if Sophie-Germain conjecture is true.
  (if for p prime, $2p+1$ also prime, then it is SG prime. Conjecture is there are infinitely many SG primes).

Miller-Rabin

$n-1 = 2^s d$ (d odd).

- Pick $a < n$. ($\geq 3/4$ numbers are witnesses).
- If both
  (i) $a^d \not\equiv 1 \pmod n$
  (ii) $a^{2^r d} \not\equiv -1 \pmod n$ $\forall$ $0 \leq r \leq s-1$

then n is composite & a is a witness.

Solovay-Strassen

- Pick $a < n$
- If $a^{\frac{n-1}{2}} \not\equiv \left(\frac{a}{n}\right) \bmod n$ ← ~~Legend~~ Jacobi symbol.

then n is composite, and a is witness ($\geq 1/2$ numbers are witnesses).

Jacobi symbol

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{\alpha_1} \cdots \left(\frac{a}{p_k}\right)^{\alpha_k}$$ ← Legendre symbol.

where $n = p_1^{\alpha_1} \cdots p_k^{\alpha_k}$

Legendre symbol

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } a \equiv 0 \pmod p \\ 1 & \text{if } a \not\equiv 0 \pmod p \text{ \& } a \equiv x^2 \pmod p \text{ for some } x \\ -1 & \text{if } a \not\equiv 0 \pmod p \text{ \& not a quadratic residue mod } p. \end{cases}$$

Jacobi symbol computable in time $O((\log n)^2)$ using "Jacobi's generalization of law of quadratic reciprocity".

HW. Find out about law of quadratic reciprocity.

5. Factoring [Link to Sarnak's Golden Gate talk].

- No polynomial time algorithm
- Not known currently if it is NP-HARD
- It is in NP.
- Widely believed to be not in NP-HARD.
- Shor's algorithm can solve it in polynomial time on a quantum computer.
- Best known classical algo. is sub-exponential i.e. faster than $O((1+\varepsilon)^b)$ $\forall\ \varepsilon > 0$, $b = \#$ bits in representation of n.

GNFS: $\exp\left(\left(\sqrt[3]{\frac{64}{9}} + o(1)\right)(\ln n)^{1/3}(\ln \ln n)^{2/3}\right)$.

Aside.
explain O, Θ, Ω, o notations (if class needs).

6. Quantum algorithm breakthroughs

a) Peter Shor (1994)

↙ factoring

↘ discrete log
(in a group G, if $a, b \in G$, find k s.t $a^k = b$).

Need aside on groups if needed

b) Grover search (Lov Grover, AT&T Bell Labs, 1995)

$\sqrt{N}$ speedup in searching an unstructured database.

c) Simulation of quantum systems (1982) (Feynman)

- Around 1990s people started to start showing that indeed it is possible to simulate systems on quantum computer, that cannot be efficiently simulated classically.

e.g. Simulation of fermionic systems (2000). (Bravyi, Kitaev) [post paper?]
- important for computational chemistry.

- These will be the first applications of

7. Quantum information theory

- Similar in principle to classical information theory.

Shannon's noiseless channel coding theorem
- quantifies physical resources needed to store output from an information source.

Shannon's noisy channel coding theorem.
- quantifies how much info is possible to be transmitted reliably through a noisy channel. (Error correcting codes).

- Analog of noiseless coding theorem known in quantum case (1995, Schumacher).
- No known analog of noisy channel coding theorem.
- How to protect quantum information?
  - Quantum error correcting codes.
    (eg. CSS codes (1996) — Calderbank, Shor, Steane.
    Superseded by: Stabilizer codes).

8. Quantum cryptography

- RSA can be broken by quantum computers due to Shor's algorithm.
  (~ million qubits needed due to error correction — not anytime soon).
- Lattice based encryption schemes exist which are resistant to quantum attacks.

- Read pages 10-11 (Mike & Ike).
- Will give examples later classes of QI applications.

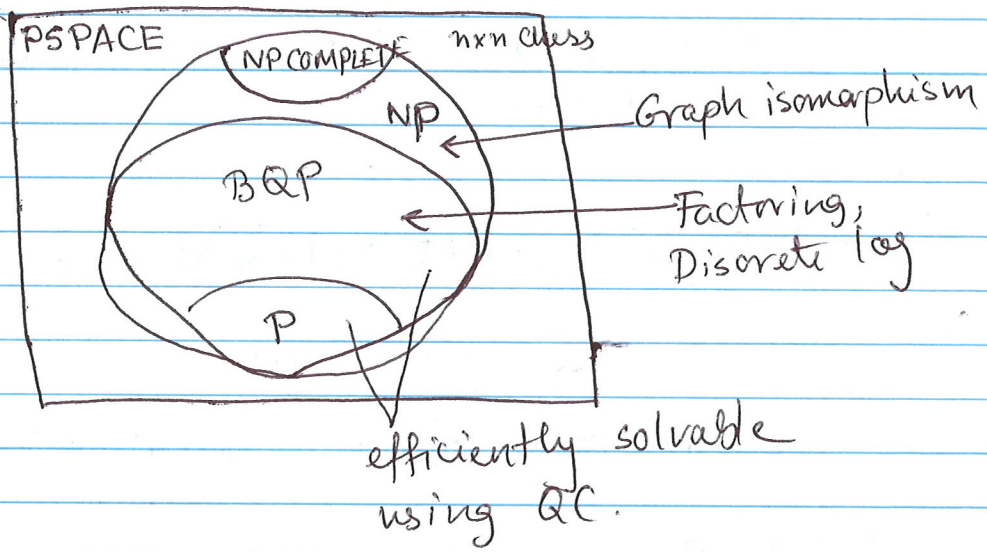
~~Qubits and Bra-ket notation~~

9. Power of quantum computation

[Link to Scott's article: The Limits of Quantum].

* surprising.

if you add a small non-linear term to QM, you can design machines that can efficiently solve NP-COMPLETE problems.



- for sure QC cannot efficiently solve problems outside PSPACE.
- for sure QC can efficiently solve problems in P.
- if QC strictly better (powerful) than classical comp. then $P \neq PSPACE$.
  - but $P \neq PSPACE$ still unproven
  - this is why some people think QC may not have any advantage over classical comp.
- Beliefs (unproven).
  - $P \neq NP$,
  - QC cannot efficiently solve NP-COMPLETE probls
  - it is believed some NP, PSPACE probs are efficiently solvable on QC, outside P, NP resp.

e.g.: QRS (Quantum Recommendation System) (Kerenidis, Prakash,) (2016).

— thought to be first example of an efficient quantum algorithm that provides exponential speedup over known classical algorithms (which were polynomial).

— Ewin Tang (2018) came up with a classical algorithm with similar exponential speed up.

>>> Qubits & Bra-Ket notation (Due to PAM Dirac).

$|0\rangle$ Ket, $\langle 0|$ bra.

$\langle 0|0\rangle$ inner prod, $|0\rangle\langle 0|$ outer prod.

Qubit: 2 level quantum system,

Qudit: d level quantum system.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1.$$

$|0\rangle$, $|1\rangle$ are "computational basis states".

Measurement: get either $|0\rangle$ or $|1\rangle$ with prob. $|\alpha|^2$ or $|\beta|^2$.

$$|\psi\rangle = e^{ir}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right)$$

↑ phase irrelevant

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle.$$