Cosc 312/412: Cryptography and complexity
Lecture 4 (5/8/2024)
Quantum computation (some highlights)

Michael Albert
michael.albert@cs.otago.ac.nz

Quantum computation without (much) maths

- The universe solves some apparently very hard computational problems all the time.
- For a quantum system of n particles, each having two possible states, quantum mechanics gives a system of 2^n partial differential equations.
- Solving these classically is infeasible for any but the smallest values of n.
- And yet, the universe “solves” them in real time.
- Can this power be harnessed?
- What would that even mean?

Bits and qubits

- The fundamental object of classical computing is the bit - a system with two possible values, conventionally denoted 0 or 1.
- In quantum computing the analogous concept is a *qubit* denoted $|0\rangle$ or $|1\rangle$.
- It represents a system, e.g., a photon, that can be in two possible states (vertically or horizontally polarised).
- Unlike a classical bit, it can also be in a *superposition* of states:

  $$\alpha_0 |0\rangle + \alpha_1 |1\rangle$$

  where $\alpha_0, \alpha_1 \in \mathbb{C}$ and $|\alpha_0|^2 + |\alpha_1|^2 = 1$.
- In such a superposition, a measurement might indicate 0 or it might indicate 1 (with probability equal to the respective amplitudes $|\alpha_0|^2$ and $|\alpha_1|^2$) – call this *sneaky parallelism*.

Quantum circuits

- We introduce *gates* that apply operations to qubits, collections of qubits, and superpositions of collections of qubits.
- A computer is then a (normal) Turing machine that describes how to build the circuits that are needed to solve a problem.
- But there's a hitch.

Restrictions on gates

- Quantum gates can only apply *unitary* operations to their input.
- There are two key consequences of this:

reversibility All computations could be run in reverse – in the absence of measurement, we could take the output state apply the inverses of all the gates, and recover the input.

no cloning It is impossible to duplicate data/inputs.

- So how could we implement even something as simple as an and-gate?

Quantum and-gate

- Suppose that we have two registers, i.e., qubits in states $|x\rangle$ and $|y\rangle$.
- We want to produce a qubit in state $|x \wedge y\rangle$.
- Add a third input, and make sure it's initialized to $|0\rangle$.
- Now look at the operation:

$$(x, y, z) \mapsto (x, y, z \oplus (x \wedge y))$$

- This one is unitary!
- So most quantum gates have *control bits* (x and y) and *target bits* (z).
- All classical logic gates can be simulated in this way.

Quantum cryptography

- Basic quantum cryptography is really about key distribution.
- Alice and Bob use a quantum channel (details to come) to agree on a private key of whatever length they like.
- Any attempt by Eve to eavesdrop is detectable, and the protocol can be restarted, or mitigation techniques can be employed.
- In fact, there will be some error regardless of interception so in any case there are some technical mitigation efforts needed.
- A technical part: information reconciliation and privacy amplification.
- A cool part: the key exchange protocol itself.

Information reconciliation

- Alice and Bob think they share a secret key k but worry that some bits may not match (due to mistakes in transmission, or tampering by Eve).
- They can use standard error correction techniques (checksums for blocks etc.) to find and correct some errors.
- The length of the resulting string which they are sure to agree on may be somewhat shorter than the original key.
- The communications are in the clear, so some information about the key (some number of bits, parities etc.) leaks to Eve.

Privacy amplification

- For privacy amplification we imagine that Alice and Bob share a random key $k \in \mathbf{2}^n$.
- They worry that Eve has acquired some knowledge of k, i.e, a random variable that is somehow correlated with k.
- If this correlation is not too strong, they can use *universal hash functions* to map k to a shorter key in such a way that any (weak) correlation becomes much much weaker (to the point of being useless).
- This is all classical stuff from *information theory*.

The BB84 protocol (Bennet and Brassard)

- The underlying signal from Alice to Bob is a sequence of photons. They wish to end up with an *m* bit key.
- Alice generates photons in a polarized basis, either vertically-horizontally or diagonally. She and Bob have the following correspondence in mind:

Bit	Vert-Horiz	Diagonal
0	$\|0\rangle$	$\|0\rangle + \|1\rangle$
1	$\|1\rangle$	$\|0\rangle - \|1\rangle$

- These are chosen to have the following property: if Bob measures a photon in the same basis that it was generated, then he gets its value. If he measures in the other basis he gets a fair coin toss.

BB84 continued

- Alice generates two random bit strings *a* and *b* of length $(4 + \delta)n$.
- She uses *b* and the correspondence to generate photons encoding *a* (if a bit of *b* is 0 she uses VH encoding for the corresponding bit of *a*, if it is 1 she uses D encoding)
- Bob receives the photons, tells the world he did, and chooses his own random bit string b' to try and decode them.
- Alice announces *b* to the world. Bob compares *b* and b' and announces to the world a set of $2n$ bit-indices where he and Alice used the same basis (if unluckily there aren't enough, then restart)

BB84 concluded

- Alice chooses randomly and announces n of the $2n$ bit-indices.
- Alice and Bob publicly compare those n bits. If they disagree too often (due to Eve’s actions or transmission errors) they abort and retry.
- If not, then they are confident that the error level in the remaining n bits is sufficiently low to allow information reconciliation and privacy amplification obtaining an m bit key.
- Note that all parts of this process can be automated, so in effect “Alice presses a button and enters the number of bits she wants to have in common with Bob” is what happens at the end of the day.
- See wikipedia on quantum key distribution.

The RSA trapdoor

Given a positive integer $N = pq$, which is the product of two large primes, p and q, Eve cannot find its prime factorisation.

A keyhole?

If P is a prime and

$$a^2 \equiv 1 \pmod{P}$$

then,

$$a \equiv \pm 1 \pmod{P}.$$

But, if N is composite then there are other solutions of

$$b^2 \equiv 1 \pmod{N}.$$

Given such a b:

$$\begin{aligned} b^2 &= kN + 1 \\ b^2 - 1 &= kN \\ (b-1)(b+1) &= kN. \end{aligned}$$

Then:

$$1 < \gcd(b-1, N), \gcd(b+1, N) < N$$

and so we would know some non-trivial factors of N.

Building a key

For composite odd N with at least two distinct prime factors, and x with $\gcd(x, N) = 1$ the following hold with probability at least $3/4$:

- The least positive integer r such that $x^r \equiv 1 \pmod{N}$ is even (called the *order* of x), and
- $x^{r/2} \not\equiv -1 \pmod{N}$.

So, if Eve had access to an *order finding algorithm* then she would have the ability to break RSA encryption.

Finding a factor of N

1. If N is even, return 2.
2. If $N = a^b$ for some $a, b \geq 3$ return a (see Note 1)
3. Randomly choose $2 < x < N - 1$. If $\gcd(x, N) > 1$ return it.
4. Compute the order, r, of x modulo N.
   - If r is odd, or $x^{r/2} \equiv -1 \pmod{N}$. Fail (see Note 2)
   - Otherwise, return $\gcd(x^{r/2} - 1, N)$.

Note 1: $b < \log_2 N$ in this case so this is easy to check classically.
Note 2: This occurs with probability at most 1/4, so actually just restart from (3).

Order finding (Shor's algorithm) on a quantum computer

- Let N be given, and choose n with $N < 2^n$.
- Consider a pure n-qubit state, $|b\rangle$ as representing an integer in the range $[0, 2^n)$.
- For $1 < x < N$ consider the map, defined as follows:

$$U|b\rangle = \begin{cases} |xb \pmod{N}\rangle & \text{if } b < N, \\ |b\rangle & \text{if } N \leq b \end{cases}$$

- Then U is unitary (easy to check) and we can compute it with polynomially many gates (a bit harder).

Eigenvalues and all that

Suppose that the order of x is r. Let $\omega = \exp(2\pi i/r)$. For $0 \leq s < r$ define:

$$|u_s\rangle = |1\rangle + \omega^s |x\rangle + \omega^{2s} |x^2\rangle + \cdots + \omega^{(r-1)s} |x^{r-1}\rangle .$$

Then

$$U |u_s\rangle = \omega^{-s} |u_s\rangle .$$

and

$$|1\rangle = |u_0\rangle + |u_1\rangle + \cdots + |u_{r-1}\rangle .$$

(Magic happens) Using a technique called *quantum phase estimation* we can arrange to get an output (on some new lines) that, with probability as close to 1 as we like, is a t-bit approximation to some s/r.

Back to the classical world

- The quantum computer has allowed us to compute a t-bit approximation to s/r for some s (randomly caused by quantum measurement stuff), but t under our control.
- Given a binary number, we can compute its “best” rational approximations using a technique called *continued fractions* – if $2r^2 \leq 2^t$ (i.e., basically if $t \geq 2n$) then we can *guarantee* that s/r will occur as one of them.
- After a bit of cleaning up (s/r might not be in lowest terms), we’ll have r (the cleaning up can be done in a way that requires us to run the algorithm only a constant number of times).

The practicality of quantum computing?

There are two fundamental issues that cause a gap between the ideal view of (gate-based) quantum computing and current experimental results:

fidelity The error rate of the various components of a QC. At present each two-qubit gate has an error rate of “a few percent”.

decoherence Since the QC cannot be completely isolated from the universe at large there will always be some interactions with the environment – effectively this can be thought of as the environment making small random measurements on the system which can (will) cause a partial collapse of the wave function.

Error correction

- Based on classical experience the obvious approach to the problems of (lack of) fidelity and decoherence would be to use *error correction*.
- But quantum error correction is a rather trickier beast! The error correction overheads are very high.
- With a 0.1% error rate per physical qubit, it would require approximately 15,000 physical qubits to provide *one* error-corrected logical qubit.
- Shor's algorithm appears to require (several hundred)n^3 gates and $2n + 2$ qubits to operate on n-bit input.

Opinions and speculations

Not for the record.