CMSC 657: Introduction to Quantum Information Processing
Lecture 29

Instructor: Daniel Gottesman

Fall 2024

1 Quantum Key Distribution

1.1 Attacks on QKD

1.1.1 Man-in-the-Middle Attack

One important aspect of BB84 or any QKD scheme is that the classical channel must be authenticated so that Alice knows that she is talking to Bob and vice-versa. Otherwise, Eve could intercept Alice's qubits and respond to Alice as if she were Bob. Alice would then generate a private key shared with Eve and not with Bob. Similarly, Eve could interact with Bob by sending him qubits and running the protocol just as if she were Alice. She would then also share a private key with Bob. When Alice sends a message encrypted with the one-time pad, Eve could decrypt it, read it, and re-encrypt it using Bob's key. Thus, neither Alice nor Bob would know that anything is wrong. However, all of this depends on Eve being able to impersonate Alice or Bob over the classical channel.

There are classical authentication protocols that are information-theoretically secure (i.e., do not rely on computational assumptions) but do require a pre-shared secret key. Luckily, the key required for authentication is much shorter than the message, about $\log N$ bits. Thus, if Alice and Bob already have a short shared key, QKD can be used to securely expand it to a much longer key.

Another way to use QKD is to use intermediaries to authenticate the classical messages. You must then trust them to handle this task, but you don't need to trust them with your private key. Finally, you could use some computationally secure public key digital signature protocol to authenticate messages. The result would only be computationally secure instead of information-theoretically secure, *but* in order to break the system, Eve would have to break the computational security of the digital signatures while the QKD protocol is running. She can't copy down the ciphertext and try to break it in her own time.

1.2 E91

There are many other QKD protocols. A different but closely related one, the second one discovered (independently of BB84) is the Ekert 91 protocol. It works as follows:

1. Alice generates N EPR pairs $|00\rangle + |11\rangle$.
2. Alice transmits one qubit from each pair to Bob over the quantum channel.
3. Alice and Bob choose a random subset of pairs to perform a test on. They do a Bell inequality test on these pairs. If the state violates the Bell inequality, they keep it and move on. If it satisfies the Bell inequality, then they abort the protocol, as Eve might have too much information.
4. Alice and Bob agree (over the classical channel) to measure each remaining EPR pair in either the horizontal or diagonal basis and do so, keeping the measurement result as a raw key bit.

5. Alice and Bob perform classical error correction and privacy amplification on the raw key to get the final key.

E91 is actually closely related to BB84. When Alice measures a perfect EPR pair in the horizontal or vertical polarization basis, Bob's state collapses to the same state, one of the BB84 states, just as if Alice had created in initially and sent it rhough the channel. The main difference is in the method to verify the error rate. In E91, Alice and Bob use Bell inequality violation as a sign on eavesdropping, as opposed to reserving some key bits to determine the error rate.

Why does E91 work? Well, suppose Eve interacted with the EPR pair in some way that let her predict Alice's and Bob's measurement results for either basis that he happens to measure in. That would be a hidden variable which would let her predict measurement outcomes. If the state violates Bell inequalities, we know that that is not possible and therefore the state is secure against Eve.

Moreover, notice that it does't really matter how Alice's and Bob's devices work for this protocol. They could even have Eve creating the EPR pairs and sending half of each to each player. Eve could even have created Alice's and Bob's detectors, provided that the detectors can't talk to each other or to Eve. Still, as long as the actual measurement results violate Bell inequalities, there is no hidden variable theories and the protocol is secure. This is now known as *device-independent QKD*.

1.3 Quantum Repeaters

QKD is usually implemented by sending photons either over optical fibers or through the air. However, there is a practical limit to the distance over which one can do this. For optical fibers, that limit is maybe 200-300 km, after which loss is prohibitive (and threatens security of the protocol through the photon-number-splitting attack). If you want to use QKD to establish a secret key with someone who is farther away than that, there are a few different solutions people have proposed.

One is to use trusted intermediaries. If you have a network of secure key exchange locations or just a network of outposts (e.g., branches of a bank), and each one is within range of another, making a connected graph, then you can exchange secret keys between neighboring locations and send secret messages in hops along the graph. The problem is these intermediaries really do need to be trusted completely, since they can read the messages you send. If even one is compromised, your security is broken. When there are multiple non-overlapping paths between Alice and Bob, you can reduce this risk by use the XOR of keys established along the different paths.

A related approach is to use satellites. A satellite in Low Earth Orbit is within range of a free-space key distribution protocol (particularly since most of the loss is near the ground, where the air is thickest). China has a quantum satellite which has demonstrated ground-to-space QKD. If the satellite is trusted, Alice can establish key with it, wait until it orbits to a point above Bob, who then also establishes key with the satellite. Then there is only the single intermediary when they send messages. This approach can also be used to communicate securely with the satellite itself. A satellite can also, in principle, be upgraded so that it does not have to be trusted. By sending halves of an entangled pair down to two separate ground locations, those two locations can establish key even though they might be too far apart to do so directly.

Finally, we can use a quantum repeater. The classical solution to loss is an amplifier, which essentially measures the light and re-prepares a new stronger version of it. That will look like an eavesdropper to a QKD protocol, but we can use a quantum error-correcting code (configured to deal primarily with loss of photons instead of other kinds of errors). We need repeater stations spaced at appropriate intervals to correct errors, but they don't need to be trusted. Why? Because either they succeed or fail at correcting errors. If they succeed, the states are just what Alice sent and the repeater doesn't learn anything about the encoded state. If they fail, that will look like errors or eavesdropping and Alice and Bob will discard the key.

It is actually better to use a two-way entanglement distillation protocol than a QECC. In a QECC, Alice sends qubits to Bob but not vice-versa. If Alice and Bob try to establish an EPR pair, Bob can tell Alice if he didn't receive a photon. (Actually we want to do this between adjacent repeater stations, not Alice and Bob directly.) That allows the protocol to be more efficient than a QECC, which needs to be prepared for the possible loss of any photon in the code. Once adjacent repeater stations have EPR pairs, you can

use entanglement swapping (basically teleportation, as you saw in an old homework) to create an EPR pair between Alice and Bob.

2 Other Types of Quantum Cryptography

There are other applications for cryptography. Broadly speaking, cryptography is the study of how to protect information against an adversary. The adversary might be trying to do a range of things, from gaining unauthorized information, to altering the information, to simply trying to confuse things and make it hard for you to operate. There are a variety of different kinds of cryptographic protocols dealing with different kinds of tasks and adversaries. Quantum cryptography falls into two categories, either using quantum information to somehow improve the protection of classical information, or using cryptographic tools to protect quantum information from an adversary. QKD is an example of the first.

You can also, for instance, use classical keys to encrypt quantum information in a quantum one-time pad: Use 2 bits (x, z) of classical key to specify one of the four Paulis P_{xz}. Then to encrypt a one-qubit message, Alice applies P_{xz} to her qubit and sends it to Bob, who decrypts by again applying P_{xz}. Since $P_{xz}^2 = I$, Bob decodes exactly Alice's qubit (in the absence of noise). But Eve doesn't know the key, so for her the state is $\sum_{x,z} P_{xz}\rho P_{xz} = I$, regardless of the state of the qubit: she has no information about the message, just like in the classical one-time pad. To encrypt an n-qubit message, Alice and Bob can just do this on every qubit with a different pair of classical key bits, using up a total of $2n$ key bits.

2.1 Secure Function Evaluation

Another example of using quantum information to (potentially) improve classical cryptography is for the task of secure function evaluation. In the simplest version of this, there are two people, Alice and Bob, and each has an input, x for Alice and y for Bob, that they wish to keep secret from the other. However, they wish to compute a function $f(x, y)$ and both learn the answer. Of course, the value of $f(x, y)$ reveals some information about the other input, but the cryptographic goal here is to learn nothing *else* about the other person's input other than the value of $f(x, y)$. One crucial difference here from QKD is that Alice and Bob don't trust each other, whereas in QKD they were working together to try to defeat an external adversary Eve. An example might be as follows: Alice wishes to buy a unique item from Bob and is willing to pay up to x dollars; Bob is willing to sell for y dollars. They want to find out if a deal is possible, i.e., if $y \leq x$, without letting Alice find out the actual value of y or Bob find out the actual value of x.

There are classical cryptographic techniques that can accomplish this task with computational security. Naturally, given the success of QKD, it makes sense to ask if quantum information can achieve two-party secure function evaluation with *information-theoretic security*.

There are classical cryptographic reductions which show that you can compute arbitrary functions securely given the ability to implement (somehow) certain cryptographic *primitives* (building blocks of protocols), in particular something called *oblivious transfer*. These reductions still work in the quantum case, and actually the situation is even better because in the quantum case, it is sufficient to be able to do a weaker protocol called *bit commitment*. Classical bit commitment is too weak to allow all two-party secure functions to be computed, but quantumly, it is enough.

2.2 Quantum Bit Commitment

What is bit commitment? Bit commitment consists of two stages. In the first stage, the *commitment* phase, Alice sends an encrypted bit b to Bob. The commitment should have the property that Bob cannot learn any information about b. You can think of this stage as Alice sending b written on a piece of paper and put inside a locked box to which Bob does not have the key. In the second stage, the *opening* phase, Alice sends Bob additional information to enable Bob to read the bit she sent initially. This phase should have the security property that Alice cannot change the bit that she initially sent to Bob; she is *committed* to it. You can think of this stage as Alice sending the key to the box to Bob. Bob can now open the box and see

the bit b and Alice cannot now change her mind and cause Bob to instead read the bit as $b' \neq b$. Both the commitment phase and the opening phase can potentially involve many rounds of back-and-forth (quantum) communication between Alice and Bob.

This is again not a protocol that is generally of too much interest by itself, but is a useful primitive to put together into more general functions. You can come up with examples where it is directly relevant, though. For instance, suppose Alice wants to convince Bob she can predict the stock market, but doesn't want to reveal her predictions unless Bob pays her first, and Bob is not willing to pay until Alice proves her claim. In the commitment phase, Alice can commit a bit representing whether a particular stock will go up or down in the next day. Then, the following day, they perform the opening phase and Alice reveals her prediction. Here, it is important that Bob can't read the bit until Alice opens the commitment, otherwise he could invest according to the prediction without paying Alice, but it is also important that Alice can't change the prediction, since otherwise she could just change the bit Bob reads into whatever actually happened.

Classically, there are protocols that allow bit commitment with computational security using one-way functions, functions for which it is computationally easy to compute the output $f(x)$ given the input x but hard to compute an input x which gives a particular output $f(x)$. However, it should be clear that there is no information-theoretically secure classical bit commitment scheme: If whatever Alice sends to Bob depends on her committed bit, than Bob can, given enough computational power, distinguish the possibilities and learn Alice's bit before it is revealed. But if Alice's commitment doesn't depend on her bit, then she can just choose whichever bit value she wants when she opens the commitment.

Here's a protocol that looks like it might work to provide information-theoretically secure quantum bit commitment: In the commitment phase, Alice sends Bob one of the four BB84 states. If Alice wants to send a bit $b = 0$, she uses the Z basis, and sends either $|0\rangle$ or $|1\rangle$ at random. If Alice wants to send a bit $b = 1$, she uses the X basis, and sends either $|+\rangle$ or $|-\rangle$ at random. Because Bob doesn't know which state it is, his density matrix for either case is

$$\rho = I/2 = \frac{1}{2}(|0\rangle\langle 0| + |1\rangle\langle 1|) = \frac{1}{2}(|+\rangle\langle +| + |-\rangle\langle -|). \tag{1}$$

Therefore Bob can learn no information about b.

In the opening phase, Alice sends to Bob which basis she used and what her actual bit value is. Bob measures in that basis and verifies that the state is correct. If Alice lies about the basis, there is a 50% chance that Bob gets the other state than she sent. You can try to get better security by having Alice send not one qubit, but many, all in the same basis but using different random bit values. When Alice lies about the basis, the odds of getting all the bit values right is therefore small.

However, this protocol is actually not secure at all. It is true that Bob can learn nothing about Alice's bit after the commitment phase, but in the opening phase, Alice can open it to either value. How is that possible?

The key insight is that Alice doesn't have to send Bob a pure state. Instead, she could create a Bell state $|00\rangle + |11\rangle$ and send the second qubit to Bob for the commitment phase. It is still the case that Bob's density matrix at this point is $I/2$, so the situation looks to Bob just the same as before. In the opening phase, if Alice decides she wants to open with a value $b = 0$, she measures her remaining qubit in the Z basis to get 0 or 1. Bob's state then collapses to either $|0\rangle$ or $|1\rangle$, according to Alice's measurement result, just as if Alice had sent that state initially. She then sends Bob $b = 0$ along with her measurement result, and Bob measures and confirms that her claim is correct.

But note that

$$|00\rangle + |11\rangle = |++\rangle + |--\rangle. \tag{2}$$

(You can check this by computing yourself.) That means that if Alice measures in the X basis and gets $+$, Bob's state is $|+\rangle$, and if she gets $-$, Bob's state is $|-\rangle$, again as if that is the state Alice had sent initially. Again, Alice can send $b = 1$ and the measurement result, and Bob's measurement will confirm that she is correct. Since Alice never gets caught, repeating with many qubits won't help; whichever value of b she decides, she can open the commitment that way without Bob detecting any problem.

You can try to come up with alternative quantum bit commitment protocols, but all of them fail for precisely this reason. In general, it can be proven that informationally-secure quantum bit commitment is impossible.