Quantum Computation
Question & Answer Study Guide

Covering All 22 Lectures

Based on *CPSC 519/619: Quantum Computation*
John Watrous, University of Calgary (2006)

Concept explanations, mathematical notation, worked examples and revision questions

Independent learning companion based on the user-supplied 139-page lecture notes.

Contents

How to use this guide **2**

1 Lecture 1: Overview of quantum information **3**

2 Lecture 2: Overview of quantum information (continued) **5**

3 Lecture 3: Superdense coding, quantum circuits, and partial measurements **7**

4 Lecture 4: Quantum Teleportation; Deutsch's Algorithm **9**

5 Lecture 5: A simple searching algorithm; the Deutsch-Jozsa algorithm **11**

6 Lecture 6: Simon's algorithm **12**

7 Lecture 7: Arithmetic/number-theoretic problems; reversible computation **14**

8 Lecture 8: Phase estimation **17**

9 Lecture 9: Phase estimation (continued); the quantum Fourier transform **19**

10 Lecture 10: Order finding **21**

11 Lecture 11: Order finding (continued); reducing factoring to order finding **23**

12 Lecture 12: Grover's Algorithm **25**

13 Lecture 13: Grover's Algorithm (continued) **27**

14 Lecture 14: Quantum information revisited **28**

15 Lecture 15: Quantum information revisited (continued) **30**

16 Lecture 16: Quantum error correction **32**

17 Lecture 17: General quantum errors; CSS codes **34**

18 Lecture 18: Quantum Key Distribution **36**

19 Lecture 19: Impossibility of Quantum Bit Commitment **38**

20 Lecture 20: Bell inequalities and nonlocality **40**

21 Lecture 21: Quantum communication complexity **42**

22 Lecture 22: Quantum computational complexity **44**

How to use this guide

This handbook covers all 22 lectures in the uploaded notes, preserving their order and main named topics. Questions are newly formulated to help study the lecture material. Answers include selected explanatory passages adapted from the source text and a key equation for each lecture. This is a *study guide*, not a line-by-line transcription, and it does not reproduce every proof or diagram from the original notes. Consult the cited lecture for full technical derivations.

Important: Mathematical displays are typeset independently in LaTeX; prose from the PDF is cleaned for readability. The source itself contains more examples, proofs, and diagrams than can fit into this Q&A format.

Lecture 1: Overview of quantum information

Core mathematical relation

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

Q1. What is a probabilistic model and how does it work?

Explanation. It is helpful to start classically, with a model that will probably seem completely simple to everyone. Imagine that we have some physical device, called X, that has some finite, non-empty set of possible states1 . For example, we might have = {0, 1}, in which case we would think of X as representing a bit. For the following discussion let us restrict ourselves to this example (but keep in mind that everything can easily be generalized to sets other than {0, 1}). Suppose that we do not necessarily have complete information about the state of X, but instead represent our knowledge of its state by assigning probabilities to the different states. For example, Mathematically we can represent this type of knowledge about the state of X with a probability vector, which is a column vector whose entries are all nonnegative real numbers that sum to 1. In the case at hand, the associated probability vector is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with A probabilistic model. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Q2. What is Quantum bits (qubits) and why is it important in quantum computation?

Explanation. The framework of quantum information works in a similar way to the simple probabilistic model we just saw, but with some key differences. Let us again imagine that we have a physical device called X. As before we imagine that there is some set of possible states of X, and we will again consider for now just the simple case = {0, 1}. At this point, to avoid confusion let us now refer to elements of as classical states rather than just states. Intuitively you can think of a classical state that you as a human can look at, touch, and recognize without ambiguity. The device X will represent the quantum analogue of a bit, which we call a qubit. We will still represent our knowledge2 of X with column vectors indexed by , but this time they will not be probability vectors. Instead of representing probability distributions, the vectors represent what we call a superposition or just a state (by which we mean a quantum state). For example, here are a few vectors representing superpositions: Notice that the entries in these vectors are not probabilities: they are not necessarily nonnegative (in fact they are not even necessarily real numbers), and they do not necessarily sum to 1. We call these numbers amplitudes instead of probabilities. The condition that replaces the probabilities summing to 1 in a probability vector is this: vec

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum bits (qubits). Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Q3. What is Stochastic matrices and why is it important in quantum computation?

Explanation. physically meaningful operations are described by stochastic matrices. It works the other way as well; any stochastic matrix describes some physically meaningful operation. As mentioned before, this entire picture is easily generalized to the case where is not nec- essarily {0, 1}. In general the dimension of the vectors and matrices will be equal to the size The framework of quantum information works in a similar way to the simple probabilistic model we just saw, but with some key differences. Let us again imagine that we have a physical device called X. As before we imagine that there is some set of possible states of X, and we will again consider for now just the simple case = {0, 1}. At this point, to avoid confusion let us now refer to elements of as classical states rather than just states. Intuitively you can think of a classical state that you as a human can look at, touch, and recognize without ambiguity. The device X will represent the quantum analogue of a bit, which we call a qubit. We will still represent our knowledge2 of X with column vectors indexed by , but this time they will not be probability vectors. Instead of representing probability distributions, the vectors represent what we call a superposition or just a state (by which we mean a quantum state). For example, here are a few vectors representing

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Stochastic matrices. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Q4. What is Unitary matrices and why is it important in quantum computation?

Explanation. tary if and only if it preserves the Euclidean norm. Fortunately there is a very simple condition to check this: a matrix U is unitary if and only if where U is the conjugate transpose of U (meaning that you take the transpose of U and then take the complex conjugate of each of the entries). For example, these are unitary matrices:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Unitary matrices. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Q5. What is Hadamard transform and why is it important in quantum computation?

Explanation. If you measured X at this point you would see outcome 0 or 1, each with probability 1/2. If you didn't measure and instead applied the Hadamard transform again, the superposition would

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Hadamard transform. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Q6. What is Measurement and why is it important in quantum computation?

Explanation. and a measurement is performed, the outcome is 0 or 1, with probabilities | |2 and | |2 , respectively. The superposition of the qubit becomes depending on whether the measurement outcome was 0 or 1. 2. Perform a unitary operation. For any unitary matrix U , the operation described by U trans- forms any superposition v into the superposition U v.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Measurement. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 1.

Lecture 1 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 2: Overview of quantum information (continued)

Core mathematical relation

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}, \qquad (A \otimes B)(C \otimes D) = AC \otimes BD.$$

Q7. What is Multiple qubits and why is it important in quantum computation?

Explanation. In order to talk about what happens when we have multiple qubits, it will be helpful to briefly return to the probabilistic model from before. Suppose that X and Y are devices that implement bits. Then there are 4 possible states of the pair (X, Y), namely 00, 01, 10, and 11. Thus, our set of states corresponding to this pair is now {00, 01, 10, 11}. In the probabilistic model we represent our knowledge of the state of the pair (X, Y) with a 4 dimensional probability vector. For example we could have the following probability vector:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Multiple qubits. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Q8. What is Tensor products and why is it important in quantum computation?

Explanation. Returning again briefly to the probabilistic model, let us suppose that as before X and Y are devices implementing bits, and the two devices are completely uncorrelated with one another-let us say that the probability vector corresponding to X is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Tensor products. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Q9. What is Dirac Notation and why is it important in quantum computation?

Explanation. Because the dimension of vectors representing superpositions and matrices representing unitary transformations grows exponentially in the number of qubits, it quickly becomes difficult to write these things down with the notation we have been using. One way to avoid this problem is to use the Dirac notation, named after its inventor Paul Dirac (who made many important contributions to quantum mechanics and mathematical physics). The notation is simple but very convenient. In the Dirac notation, column vectors are represented by "kets", such as We use symbols such as | i and | i to represent arbitrary vectors (even when the symbols and have been assigned no meaning by themselves). Any vector indexed by the set {0, 1} can be represented by a linear combination of |0i and |1i, because {|0i , |1i} is a basis for this space of

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Dirac Notation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Q10. What is Entanglement and why is it important in quantum computation?

Explanation. We use tensor products for independent (or uncorrelated) operations as well. For example, suppose that we have two single-qubit unitary transformations such as and we perform transformation U on X and V on Y. Then the effect of these two independent operations on any superposition of the pair (X, Y) is determined by the matrix

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Entanglement. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Q11. What is Inner products and why is it important in quantum computation?

Explanation. In order to talk about what happens when we have multiple qubits, it will be helpful to briefly return to the probabilistic model from before. Suppose that X and Y are devices that implement bits. Then there are 4 possible states of the pair (X, Y), namely 00, 01, 10, and 11. Thus, our set of states corresponding to this

pair is now {00, 01, 10, 11}. In the probabilistic model we represent our knowledge of the state of the pair (X, Y) with a 4 dimensional probability vector. For example we could have the following probability vector:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Inner products. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Q12. What is Partial operations and why is it important in quantum computation?

Explanation. In order to talk about what happens when we have multiple qubits, it will be helpful to briefly return to the probabilistic model from before. Suppose that X and Y are devices that implement bits. Then there are 4 possible states of the pair (X, Y), namely 00, 01, 10, and 11. Thus, our set of states corresponding to this pair is now {00, 01, 10, 11}. In the probabilistic model we represent our knowledge of the state of the pair (X, Y) with a 4 dimensional probability vector. For example we could have the following probability vector:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Partial operations. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 2.

Lecture 2 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 3: Superdense coding, quantum circuits, and partial measurements

Core mathematical relation

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}, \qquad \text{CNOT}|a,b\rangle = |a, a \oplus b\rangle.$$

Q13. What is Superdense Coding and why is it important in quantum computation?

Explanation. Imagine a situation where two people (named Alice and Bob) are in different parts of the world. Alice has two bits: a and b. She would like to communicate these two bits to Bob by sending him just a single qubit. It turns out that there is no way they can can accomplish this task without additional resources. This is not obvious, but it is true-Alice cannot encode two classical bits into a single qubit in any way that would give Bob more than just one bit of information about the pair However, let us imaging that a long time ago, before Alice even knew what a and b are, that the two of them prepared two qubits A and B in the superposition Alice took the qubit A and Bob took the qubit B. We say that Alice and Bob share an e-bit or share an EPR pair in this situation. It is natural to view entanglement as a resource as we will see; and when Alice and Bob each have a qubit and the two of them are in the state above, it is natural to imagine that Alice and Bob share one unit (i.e., one entangled bit, or e-bit for short) of Given the additional resource of a shared e-bit of entanglement, Alice will be able to transmit both a and b to Bob by sending just one qubit. Here is how:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Superdense Coding. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Q14. What are the steps and resources in the Superdense coding protocol?

Explanation. Imagine a situation where two people (named Alice and Bob) are in different parts of the world. Alice has two bits: a and b. She would like to communicate these two bits to Bob by sending him just a single qubit. It turns out that there is no way they can can accomplish this task without additional resources. This is not obvious, but it is true-Alice cannot encode two classical bits into a single qubit in any way that would give Bob more than just one bit of information about the pair However, let us imaging that a long time ago, before Alice even knew what a and b are, that the two of them prepared two qubits A and B in the superposition Alice took the qubit A and Bob took the qubit B. We say that Alice and Bob share an e-bit or share an EPR pair in this situation. It is natural to view entanglement as a resource as we will see; and when Alice and Bob each have a qubit and the two of them are in the state above, it is natural to imagine that Alice and Bob share one unit (i.e., one entangled bit, or e-bit for short) of Given the additional resource of a shared e-bit of entanglement, Alice will be able to transmit both a and b to Bob by sending just one qubit. Here is how:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Superdense coding protocol. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Q15. What is Quantum circuits and why is it important in quantum computation?

Explanation. The previous discussion of superdense coding used a fairly inefficient way to describe the protocol; basically just plain English along with a few equations. We will need a more efficient way to describe sequences of quantum operations and measurements than this, particularly for when the complexity of the algorithms and protocols has increased. The most common way of doing this is to use quantum circuit diagrams, which is the way we will use. The basic idea is as follows:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum circuits. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Q16. What is Controlled-NOT and why is it important in quantum computation?

Explanation. 5. Bob applies a Hadamard transform to A. 6. Bob measures both qubits A and B. The output will be (a, b) with certainty.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Controlled-NOT. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Q17. What is Toffoli gate and why is it important in quantum computation?

Explanation. Imagine a situation where two people (named Alice and Bob) are in different parts of the world. Alice has two bits: a and b. She would like to communicate these two bits to Bob by sending him just a single qubit. It turns out that there is no way they can can accomplish this task without additional resources. This is not obvious, but it is true-Alice cannot encode two classical bits into a single qubit in any way that would give Bob more than just one bit of information about the pair However, let us imaging that a long time ago, before Alice even knew what a and b are, that the two of them prepared two qubits A and B in the superposition Alice took the qubit A and Bob took the qubit B. We say that Alice and Bob share an e-bit or share an EPR pair in this situation. It is natural to view entanglement as a resource as we will see; and when Alice and Bob each have a qubit and the two of them are in the state above, it is natural to imagine that Alice and Bob share one unit (i.e., one entangled bit, or e-bit for short) of Given the additional resource of a shared e-bit of entanglement, Alice will be able to transmit both a and b to Bob by sending just one qubit. Here is how:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Toffoli gate. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Q18. What is Partial measurements and why is it important in quantum computation?

Explanation. Suppose we have a system consisting of two or more qubits and we only measure one of them. For example, suppose the qubits are X and Y, and these qubits are in the state

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Partial measurements. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 3.

Lecture 3 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 4: Quantum Teleportation; Deutsch's Algorithm

Core mathematical relation

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \qquad \text{teleportation}: \ 1 \text{ ebit} + 2 \text{ classical bits}.$$

Q19. What is Quantum teleportation and why is it important in quantum computation?

Explanation. Suppose Alice has a qubit that she wants to send to Bob. Let us say that the state of the qubit is |0i + |1i. How many classical bits would be required to accomplish this task? There are two reasonable answers to the question. The first answer is that the number of classical bits depends on the desired precision: and are arbitrary complex numbers (with | |2 + | |2 = 1), but Alice could send approximations of and to Bob. The second answer, which in some sense is the better answer, is that no number of classical bits of communication will suffice. Alice may not know and , and there is no way for her to perform measurements on her qubit that would reveal these numbers. Thus, she could not communicate this information to Bob because she could not even determine it for herself. There is also the possibility that Alice's qubit is entangled with one or more other qubits (in which case of course we would not be able to describe the state of the qubit as |0i + |1i). Classical communication from Alice to Bob would not be able to somehow transmit this entanglement. However, if we give Alice and Bob the additional resource of sharing an e-bit, just as in su- perdense coding, then it becomes possible for Alice to transmit a qubit to Bob using classical communication by means of quantum teleportation. Specifically, two bit

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum teleportation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 4.

Q20. What are the steps and resources in the Quantum teleportation protocol?

Explanation. Suppose Alice has a qubit that she wants to send to Bob. Let us say that the state of the qubit is |0i + |1i. How many classical bits would be required to accomplish this task? There are two reasonable answers to the question. The first answer is that the number of classical bits depends on the desired precision: and are arbitrary complex numbers (with | |2 + | |2 = 1), but Alice could send approximations of and to Bob. The second answer, which in some sense is the better answer, is that no number of classical bits of communication will suffice. Alice may not know and , and there is no way for her to perform measurements on her qubit that would reveal these numbers. Thus, she could not communicate this information to Bob because she could not even determine it for herself. There is also the possibility that Alice's qubit is entangled with one or more other qubits (in which case of course we would not be able to describe the state of the qubit as |0i + |1i). Classical communication from Alice to Bob would not be able to somehow transmit this entanglement. However, if we give Alice and Bob the additional resource of sharing an e-bit, just as in su- perdense coding, then it becomes possible for Alice to transmit a qubit to Bob using classical communication by means of quantum teleportation. Specifically, two bit

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum teleportation protocol. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 4.

Q21. How does Deutsch's Algorithm work, step by step?

Explanation. We have seen two interesting protocols that can be performed using quantum information: super- dense coding and teleportation. Next, we will discuss various quantum algorithms, starting with a very simple one: Deutsch's Algorithm. This is a very different setting in which some advantage is gained by using quantum information over classical information. Suppose that we have a device that computes some function f : {0, 1} {0, 1}. It is useful for the purposes of the present investigation to think of this device as a black box. This means that we cannot look inside the device to see how it works-the only way to gain information about the function f computed by the device is to give some input a {0, 1} and allow the device to output f (a) {0, 1}. There are four possible functions from {0, 1} to {0, 1}; let us call them f 0 , f1 , f2 ,

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Deutsch's Algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 4.

Q22. What is Entanglement as a resource and why is it important in quantum computation?

Explanation. Suppose Alice has a qubit that she wants to send to Bob. Let us say that the state of the qubit is |0i + |1i. How many classical bits would be required to accomplish this task? There are two reasonable answers to the question. The first answer is that the number of classical bits depends on the desired precision: and are arbitrary complex numbers (with | |2 + | |2 = 1), but Alice could send approximations of and to Bob. The second answer, which in some sense is the better answer, is that no number of classical bits of communication will suffice. Alice may not know and , and there is no way for her to perform measurements on her qubit that would reveal these numbers. Thus, she could not communicate this information to Bob because she could not even determine it for herself. There is also the possibility that Alice's qubit is entangled with one or more other qubits (in which case of course we would not be able to describe the state of the qubit as |0i + |1i). Classical communication from Alice to Bob would not be able to somehow transmit this entanglement. However, if we give Alice and Bob the additional resource of sharing an e-bit, just as in su- perdense coding, then it becomes possible for Alice to transmit a qubit to Bob using classical communication by means of quantum teleportation. Specifically, two bit

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Entanglement as a resource. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 4.

Q23. What is Oracle queries and why is it important in quantum computation?

Explanation. Suppose Alice has a qubit that she wants to send to Bob. Let us say that the state of the qubit is |0i + |1i. How many classical bits would be required to accomplish this task? There are two reasonable answers to the question. The first answer is that the number of classical bits depends on the desired precision: and are arbitrary complex numbers (with | |2 + | |2 = 1), but Alice could send approximations of and to Bob. The second answer, which in some sense is the better answer, is that no number of classical bits of communication will suffice. Alice may not know and , and there is no way for her to perform measurements on her qubit that would reveal these numbers. Thus, she could not communicate this information to Bob because she could not even determine it for herself. There is also the possibility that Alice's qubit is entangled with one or more other qubits (in which case of course we would not be able to describe the state of the qubit as |0i + |1i). Classical communication from Alice to Bob would not be able to somehow transmit this entanglement. However, if we give Alice and Bob the additional resource of sharing an e-bit, just as in su- perdense coding, then it becomes possible for Alice to transmit a qubit to Bob using classical communication by means of quantum teleportation. Specifically, two bit

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Oracle queries. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 4.

Lecture 4 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 5: A simple searching algorithm; the Deutsch-Jozsa algorithm

Core mathematical relation

$$H^{\otimes n}|0^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle.$$

Q24. What is a simple searching problem and how does it work?

Explanation. Suppose that we are given a function of the form f : {0, 1}2 {0, 1}, but this time we are promised that it is one of these four functions:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with A simple searching problem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 5.

Q25. What is the Deutsch-Jozsa Algorithm and how does it work?

Explanation. Our next algorithm is a generalization of Deutsch's Algorithm called the Deutsch-Jozsa Algorithm. Although the computational problem being solved is still artificial and perhaps not particularly impressive, we are moving toward more interesting problems. This time we assume that we are given a function f : {0, 1}n {0, 1}, where n is some arbitrary positive integer, and we are promised that one of two possibilities holds: 1. f is constant. In other words, either f (x) = 0 for all x {0, 1}n or f (x) = 1 for all 2. f is balanced. This means that the number of inputs x {0, 1}n for which the function takes

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The Deutsch-Jozsa Algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 5.

Q26. What is Promise problems and why is it important in quantum computation?

Explanation. Suppose that we are given a function of the form f : {0, 1}2 {0, 1}, but this time we are promised that it is one of these four functions:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Promise problems. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 5.

Q27. What is Quantum interference and why is it important in quantum computation?

Explanation. Suppose that we are given a function of the form f : {0, 1}2 {0, 1}, but this time we are promised that it is one of these four functions:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum interference. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 5.

Q28. What does Query complexity mean and why does it matter?

Explanation. Suppose that we are given a function of the form f : {0, 1}2 {0, 1}, but this time we are promised that it is one of these four functions:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Query complexity. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 5.

Lecture 5 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 6: Simon's algorithm

Core mathematical relation

$$f(x) = f(x \oplus s), \qquad y \cdot s = 0 \pmod 2.$$

Q29. What is Simon's problem and why is it important in quantum computation?

Explanation. for a positive integer n. Access to this function is restricted to queries to the black-box transformation Bf as before. Similar to the Deutsch-Jozsa problem, the function f is promised to obey a certain property, although the property is slightly more complicated than before. The property is that there exists a string s {0, 1}n such that for all x, y {0, 1}n. The goal of the problem is to find the string s. For example, if n = 3, then the following function is an example of a function that satisfies the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Simon's problem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 6.

Q30. How does Simon's algorithm work, step by step?

Explanation. A quantum algorithm (called Simon's Algorithm) for solving this task consists of iterating the following quantum circuit and doing some classical post-processing: Note that the Bf gate has a different (but very similar) form from before because the input and output of the function f are n-bit strings:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Simon's algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 6.

Q31. What is Classical post-processing and why is it important in quantum computation?

Explanation. When we run the circuit above, there are two cases: in the special case where s = 0 n , the measure- ment results in each string y {0, 1}n with probability

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Classical post-processing. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 6.

Q32. What is Hidden xor mask and why is it important in quantum computation?

Explanation. for a positive integer n. Access to this function is restricted to queries to the black-box transformation Bf as before. Similar to the Deutsch-Jozsa problem, the function f is promised to obey a certain property, although the property is slightly more complicated than before. The property is that there exists a string s {0, 1}n such that for all x, y {0, 1}n. The goal of the problem is to find the string s. For example, if n = 3, then the following function is an example of a function that satisfies the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Hidden xor mask. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 6.

Q33. What is Linear equations and why is it important in quantum computation?

Explanation. obtain s. All of the operations are modulo 2 operations, so it is not exactly the type of system of linear equations you were used to in linear algebra, but the system can be efficiently solved using similar methods. You will only get a unique non-zero solution s if you are lucky and y 1 , . . . , yn-1 are linearly independent. The probability that y1 , . . . , yn-1 are linearly independent is at least If you have linear independence, solve the system to get a candidate solution s 0 6= 0n , and test that f (0n) = f (s0). If f (0n) = f (s0), you know that s = s0 and the problem has been solved. If f (0n) 6= f (s0), it must be the case that s = 0n (because if this were not so, the unique nonzero solution to the linear equations would have been s). Either way, once you have linear independence, Now, repeat the entire process 4m times. The probability of not finding a linearly independent set during one of the iterations is less than

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Linear equations. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 6.

Lecture 6 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 7: Arithmetic/number-theoretic problems; reversible computation

Core mathematical relation

$$a \equiv b \pmod{N} \iff N \mid (a-b), \qquad a^{-1}a \equiv 1 \pmod{N}.$$

Q34. What is Arithmetic/number-theoretic problems and why is it important in quantum computation?

Explanation. Before we move on to Shor's Algorithm, we need to understand more about the topic with which it is concerned: computational number theory. Let us begin with an example-suppose we are given two integers x and y, represented as strings of bits using binary notation3 , and our goal is to determine x + y. A specification of this One can formalize the notion of an efficient algorithm for a given problem in various ways by describing precise models of computation. For the purposes of this discussion let us just think in terms of Boolean circuits and identify an algorithm with a collection of acyclic circuits (one for each possible input length) that produce the correct outputs for all possible input strings. The circuits should be composed of AND, OR, and NOT gates, along with FANOUT operations (which we often would not consider as gates at all), and in order to be considered efficient the total number of gates in each circuit should be polynomial in the number of input bits. The number of bits needed to write down the number x in binary notation is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Arithmetic/number-theoretic problems. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q35. What is Integer Addition and why is it important in quantum computation?

Explanation. One can formalize the notion of an efficient algorithm for a given problem in various ways by describing precise models of computation. For the purposes of this discussion let us just think in terms of Boolean circuits and identify an algorithm with a collection of acyclic circuits (one for each possible input length) that produce the correct outputs for all possible input strings. The circuits should be composed of AND, OR, and NOT gates, along with FANOUT operations (which we often would not consider as gates at all), and in order to be considered efficient the total number of gates in each circuit should be polynomial in the number of input bits. The number of bits needed to write down the number x in binary notation is assuming we keep a sign bit for each non-zero integer, and so the number of input bits of the above problem is lg(x) + lg(y). Now, if you consider the usual method for adding numbers that you probably learned in elementary school or before, but using base 2 instead of base 10, you could turn that method into a construction of Boolean circuits of size O(lg x + lg y) for solving this problem. This is in fact linear in the input size, and so this algorithm will be considered to be very By the way, I should mention that it is typical to place additional constraints on collections of Boolean circuits in order for t

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Integer Addition. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q36. What is Integer Multiplication and why is it important in quantum computation?

Explanation. The "elementary school" multiplication algorithm establishes that the bit complexity of multiplying x and y is O((lg x)(lg y)), or more simply that the bit complexity of multiplying two n bit numbers is O(n2). In fact, there are asymptotically better methods, such as the Scho nhage-Strassen Algorithm that has bit complexity O(n log n log log n). (Large constant factors in the running time make this particular method less efficient than other methods until n becomes quite large-several The cost is roughly the same as integer multiplication-the "elementary school" division method gives an algorithm with bit complexity O((lg x)(lg y)). (In fact it is actually somewhat better:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Integer Multiplication. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q37. What is Integer Division and why is it important in quantum computation?

Explanation. The cost is roughly the same as integer multiplication-the "elementary school" division method gives an algorithm with bit complexity O((lg x)(lg y)). (In fact it is actually somewhat better:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Integer Division. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q38. What is Greatest Common Divisor and why is it important in quantum computation?

Explanation. Euclid's Algorithm, which is over 2,000 years old, computes gcd(x, y) in O((lg x)(lg y)) bit oper-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Greatest Common Divisor. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q39. What is Modular Integer Addition and why is it important in quantum computation?

Explanation. Before we move on to Shor's Algorithm, we need to understand more about the topic with which it is concerned: computational number theory. Let us begin with an example-suppose we are given two integers x and y, represented as strings of bits using binary notation3 , and our goal is to determine x + y. A specification of this One can formalize the notion of an efficient algorithm for a given problem in various ways by describing precise models of computation. For the purposes of this discussion let us just think in terms of Boolean circuits and identify an algorithm with a collection of acyclic circuits (one for each possible input length) that produce the correct outputs for all possible input strings. The circuits should be composed of AND, OR, and NOT gates, along with FANOUT operations (which we often would not consider as gates at all), and in order to be considered efficient the total number of gates in each circuit should be polynomial in the number of input bits. The number of bits needed to write down the number x in binary notation is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Modular Integer Addition. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q40. What is Modular Integer Multiplication and why is it important in quantum computation?

Explanation. Here the bit complexity is O((lg N)2) by the elementary school algorithm, and again asymptoti-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Modular Integer Multiplication. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q41. What is Modular Inverse and why is it important in quantum computation?

Explanation. Input: A positive integer N and an integer x Z N = {a ZN : gcd(a, N) = 1}. Output: y ZN such that xy 1 (mod N).

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Modular Inverse. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q42. What is Modular Exponentiation and why is it important in quantum computation?

Explanation. Before we move on to Shor's Algorithm, we need to understand more about the topic with which it is concerned: computational number theory. Let us begin with an example-suppose we are given two integers x and y, represented as strings of bits using binary notation3 , and our goal is to determine x + y. A specification of this One can formalize the notion of an efficient algorithm for a given problem in various ways by describing precise models of computation. For the purposes of this discussion let us just think in terms of Boolean circuits and identify an algorithm with a collection of acyclic circuits (one for each possible input length) that produce the correct outputs for all possible input strings. The circuits should be composed of AND, OR, and NOT gates, along with FANOUT operations (which we often would not consider as gates at all), and in order to be considered

efficient the total number of gates in each circuit should be polynomial in the number of input bits. The number of bits needed to write down the number x in binary notation is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Modular Exponentiation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q43. What is Primality Testing and why is it important in quantum computation?

Explanation. Randomized algorithms having bit complexity O((lg N)3) that allow a probability 1/4, say, of making an error have been known for many years. In a breakthrough a few years ago, Agrawal, Kayal and Saxena gave a deterministic prime testing algorithm that gives (the last I heard) a O((lg N)6) deterministic algorithm for testing primality.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Primality Testing. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Q44. What is Reversible computation and why is it important in quantum computation?

Explanation. In order to implement Shor's algorithm, it is necessary to efficiently perform some of the arithmetic operations described above (modular exponentiation, in particular) with a quantum computer. In- tuitively this seems like it should not be a problem, given that there is an efficient way to implement these operations classically. But does an efficient classical implementation necessarily imply the existence of an efficient quantum implementation? The fact that classical algorithms can use non- unitary operations such as ANDs and ORs, while quantum computers are restricted to unitary operations, makes this question non-trivial to answer. Before we address the question of whether efficient classical algorithms for arithmetic prob- lems (or any other type of computational problem) imply efficient quantum algorithms for these problems, we need to briefly discuss the notion of quantum circuit complexity. What does it mean to have an efficient quantum algorithm for some task? Our answer to this question will be similar to the classical case. In order to completely specify a quantum algorithm, it is necessary to give a construction that produces a quantum circuit solving the problem at hand for any given input size. Also, as in the classical case, one assumes the quantum circuit is composed entirely of quantum gates acting on some smal

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Reversible computation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 7.

Lecture 7 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 8: Phase estimation

Core mathematical relation

$$U|u\rangle = e^{2\pi i\theta}|u\rangle, \qquad \theta \in [0,1).$$

Q45. What is the phase estimation problem and how does it work?

Explanation. Suppose that we have a description of some quantum circuit Q acting on n qubits. Associated with Q is some 2n 2n unitary matrix U . Obviously, if n is reasonably large such as n = 1, 000, it would be impossible to write down an explicit description of U because it is too large-all of the computers in the world could only store a tiny fraction of its entries. Even if you just wanted to compute a single entry of U from the description of Q, you might be faced with a computationally Because U is unitary, we know from linear algebra that it has a complete, orthonormal collec-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The phase estimation problem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 8.

Q46. What is the phase estimation procedure and how does it work?

Explanation. In order to describe the quantum procedure for phase estimation, let us introduce some notation. Suppose that U is a unitary transformation acting on n qubits, and suppose m is any positive integer. Then we let m (U) denote the unique unitary transformation on m + n qubits that satisfies for all choices of k {0, . . . , 2m - 1} and an arbitrary n-qubit vector | i. In words, the first m qubits specify the number of times that U is to be applied to the remaining n qubits. We sometimes denote this transformation as suggested in the following quantum circuit diagram:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The phase estimation procedure. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 8.

Q47. What is Controlled powers of a unitary and why is it important in quantum computation?

Explanation. Suppose that we have a description of some quantum circuit Q acting on n qubits. Associated with Q is some 2n 2n unitary matrix U . Obviously, if n is reasonably large such as n = 1, 000, it would be impossible to write down an explicit description of U because it is too large-all of the computers in the world could only store a tiny fraction of its entries. Even if you just wanted to compute a single entry of U from the description of Q, you might be faced with a computationally Because U is unitary, we know from linear algebra that it has a complete, orthonormal collec-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Controlled powers of a unitary. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 8.

Q48. What is Eigenvectors and eigenvalues and why is it important in quantum computation?

Explanation. Suppose that we have a description of some quantum circuit Q acting on n qubits. Associated with Q is some 2n 2n unitary matrix U . Obviously, if n is reasonably large such as n = 1, 000, it would be impossible to write down an explicit description of U because it is too large-all of the computers in the world could only store a tiny fraction of its entries. Even if you just wanted to compute a single entry of U from the description of Q, you might be faced with a computationally Because U is unitary, we know from linear algebra that it has a complete, orthonormal collec-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Eigenvectors and eigenvalues. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 8.

Q49. What is Precision and probability and why is it important in quantum computation?

Explanation. Suppose that we have a description of some quantum circuit Q acting on n qubits. Associated with Q is some 2n 2n unitary matrix U . Obviously, if n is reasonably large such as n = 1, 000, it would be impossible to write down an explicit description of U because it is too large-all of the computers in the world could only store a tiny fraction of its entries. Even if you just wanted to compute a single entry of U from the description of Q, you might be faced with a computationally Because U is unitary, we know from linear algebra that it has a complete, orthonormal collec-

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Precision and probability. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 8.

Lecture 8 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 9: Phase estimation (continued); the quantum Fourier transform

Core mathematical relation

$$\mathrm{QFT}_N|x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i x y/N}|y\rangle.$$

Q50. What is Phase estimation (continued) and why is it important in quantum computation?

Explanation. In the previous lecture we discussed phase estimation. Recall that the set-up was as follows. We have a quantum circuit for performing the transformation m (U), defined by for some unitary transformation U and positive integer m, along with a quantum state | i that is an eigenvector of U . The eigenvalue associated with | i is e2 i for [0, 1), and the goal is to We had devised the following procedure, which works perfectly when = j/2 m for some

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Phase estimation (continued). Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 9.

Q51. What is Efficient implementation of the quantum Fourier transform and why is it important in quantum computation?

Explanation. In the previous lecture we discussed phase estimation. Recall that the set-up was as follows. We have a quantum circuit for performing the transformation m (U), defined by for some unitary transformation U and positive integer m, along with a quantum state | i that is an eigenvector of U . The eigenvalue associated with | i is e2 i for [0, 1), and the goal is to We had devised the following procedure, which works perfectly when = j/2 m for some

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Efficient implementation of the quantum Fourier transform. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 9.

Q52. What is Quantum Fourier transform and why is it important in quantum computation?

Explanation. In the previous lecture we discussed phase estimation. Recall that the set-up was as follows. We have a quantum circuit for performing the transformation m (U), defined by for some unitary transformation U and positive integer m, along with a quantum state | i that is an eigenvector of U . The eigenvalue associated with | i is e2 i for [0, 1), and the goal is to We had devised the following procedure, which works perfectly when = j/2 m for some

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum Fourier transform. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 9.

Q53. What is Controlled phase shifts and why is it important in quantum computation?

Explanation. In the previous lecture we discussed phase estimation. Recall that the set-up was as follows. We have a quantum circuit for performing the transformation m (U), defined by for some unitary transformation U and positive integer m, along with a quantum state | i that is an eigenvector of U . The eigenvalue associated with | i is e2 i for [0, 1), and the goal is to We had devised the following procedure, which works perfectly when = j/2 m for some

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Controlled phase shifts. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 9.

Q54. What is Approximate QFT and why is it important in quantum computation?

Explanation. In the previous lecture we discussed phase estimation. Recall that the set-up was as follows. We have a quantum circuit for performing the transformation m (U), defined by for some unitary transformation U and positive integer m, along with a quantum state | i that is an eigenvector of U . The eigenvalue associated with | i is e2 i for [0, 1), and the goal is to We had devised the following procedure, which works perfectly when = j/2 m for some

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Approximate QFT. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 9.

Lecture 9 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 10: Order finding

Core mathematical relation

$$a^r \equiv 1 \pmod{N}, \qquad r = \mathrm{ord}_N(a).$$

Q55. What is the Order Finding problem and how does it work?

Explanation. Now that we have discussed the phase estimation technique, it is time to apply it to an interesting computational problem. The problem is called Order Finding, and as we will see in the next couple of lectures it is only one step away from integer factoring. In fact, this will be the quantum part of Shor's factoring algorithm-the rest is completely classical after this. Some additional notation will be helpful for discussing the Order Finding problem. If a, b, and to mean N | (a - b) (shorthand for N divides a - b). As I mentioned before, we let Z N denote the set ZN = {0, . . . , N - 1}. When we also associate with ZN the operations of addition and multiplication modulo N , ZN forms a ring. If in addition N is prime, then ZN forms a field. We

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The Order Finding problem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 10.

Q56. What is Solving order finding using phase estimation and why is it important in quantum computation?

Explanation. Our main goal for the remainder of the lecture and part of the next will be to show that the order finding problem can be solved using the method of phase estimation. Assume N 2 is given and let n be the number of bits needed to encode elements of Z N in binary (so n = blog2 (N - 1)c + 1). Given any a Z N , define an n-qubit transformation Ma as for every x ZN . This is not a complete specification of Ma because it does not specify Ma |xi for N x < 2n , but we will not care about its action on such states. For the sake of choosing a well-defined transformation, we may say for simplicity that Ma |xi = |xi for N x < 2n . The transformation Ma is reversible, and therefore unitary. This follows from the fact that it maps classical states to classical states, along with the observation that it has an inverse: M a-1 Ma = I, where a-1 is the inverse of a modulo N . Let us consider subjecting the transformation Ma to phase estimation. It will turn out that in doing this we will be able to determine the order of a modulo N . One can efficiently imple- ment a reversible circuit for performing Ma given that the functions f (x) = ax (mod N) and g(x) = a-1 x (mod N) are efficiently computable by Boolean circuits. If we wish to subject this transformation to phase estimation, however, recall that we will need to have an efficient

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Solving order finding using phase estimation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 10.

Q57. What is Modular multiplication and why is it important in quantum computation?

Explanation. Now that we have discussed the phase estimation technique, it is time to apply it to an interesting computational problem. The problem is called Order Finding, and as we will see in the next couple of lectures it is only one step away from integer factoring. In fact, this will be the quantum part of Shor's factoring algorithm-the rest is completely classical after this. Some additional notation will be helpful for discussing the Order Finding problem. If a, b, and to mean N | (a - b) (shorthand for N divides a - b). As I mentioned before, we let Z N denote the set ZN = {0, . . . , N - 1}. When we also associate with ZN the operations of addition and multiplication modulo N , ZN forms a ring. If in addition N is prime, then ZN forms a field. We

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Modular multiplication. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 10.

Q58. What is Continued fractions and why is it important in quantum computation?

Explanation. Now that we have discussed the phase estimation technique, it is time to apply it to an interesting computational problem. The problem is called Order Finding, and as we will see in the next couple of lectures it is only one step away from integer factoring. In fact, this will be the quantum part of Shor's factoring algorithm-the

rest is completely classical after this. Some additional notation will be helpful for discussing the Order Finding problem. If a, b, and to mean N | (a - b) (shorthand for N divides a - b). As I mentioned before, we let Z N denote the set ZN = {0, . . . , N - 1}. When we also associate with ZN the operations of addition and multiplication modulo N , ZN forms a ring. If in addition N is prime, then ZN forms a field. We

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Continued fractions. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 10.

Q59. What is Order-finding success probability and why is it important in quantum computation?

Explanation. Now that we have discussed the phase estimation technique, it is time to apply it to an interesting computational problem. The problem is called Order Finding, and as we will see in the next couple of lectures it is only one step away from integer factoring. In fact, this will be the quantum part of Shor's factoring algorithm-the rest is completely classical after this. Some additional notation will be helpful for discussing the Order Finding problem. If a, b, and to mean N | (a - b) (shorthand for N divides a - b). As I mentioned before, we let Z N denote the set ZN = {0, . . . , N - 1}. When we also associate with ZN the operations of addition and multiplication modulo N , ZN forms a ring. If in addition N is prime, then ZN forms a field. We

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Order-finding success probability. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 10.

Lecture 10 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 11: Order finding (continued); reducing factoring to order finding

Core mathematical relation

$$\gcd\left(a^{r/2}-1, N\right), \qquad \gcd\left(a^{r/2}+1, N\right).$$

Q60. What is Solving order finding using phase estimation (continued) and why is it important in quantum computation?

Explanation. We will begin this lecture by finishing the discussion of how the order finding problem can be Assume that we are given a Z N , and our goal is to calculate the order of a in Z N , which we suppose is r. Recall that we defined a unitary operation Ma so that Last time we considered what happens when we perform phase estimation on the state | 1 i. The eigenvalue associated with | 1 i is = e2 i(1/r) . Assuming we were to perform the phase estimation procedure several times and take the most commonly appearing result, with very high probability we will have an approximation j/2m that is close to 1/r. Our guess for r would therefore be 2m /j, rounded off to the nearest integer. We determined that if m = 2n and the phase estimation procedure succeeds in finding a best m bit approximation j/2 m to 1/r, then rounding 2m /j to the nearest integer will indeed give r. Unfortunately we do not know how to get our hands on | 1 i. In fact it seems like one would need to know r before preparing it, but if we know r there is no need to bother. There is no known way to prepare | 1 i without knowing r, so a different approach is required. Instead, we considered running the phase estimation procedure on the state

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Solving order finding using phase estimation (continued). Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 11.

Q61. What is Reduction of factoring to order finding and why is it important in quantum computation?

Explanation. I've mentioned several times that we can factor integers efficiently using quantum computers, and that this is the problem solved by Shor's algorithm. Really, it turns out that "Shor's algorithm" is more of a technique than a specific algorithm, and it can be used to solve an interesting collection of problems. (Order finding is one of the simplest to describe.) Now let us see that the integer factoring problem can be efficiently solved given an algorithm for order-finding. In other words, factoring reduces to order-finding. To be specific, it is a ran- domized polynomial-time Turing reduction, which is probably different from (but similar to) the types of reductions you might have seen in CPSC 313 and 413. The idea, however, is essentially the same-if you have an efficient algorithm for order finding, then the reduction gives you an Let us note a few facts, which we will not prove or discuss in any detail. First, if N is a prime num- ber or a prime power (i.e., pk for some prime p and integer k 1), then there are efficient classical algorithms for solving the integer factorization problem in these cases. So, we can imagine that we first run such an algorithm on the input N ; if it succeeds then we are done, otherwise we continue on under the assumption that N has at least two distinct prime factors (i.e., m 2). Next, it is

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Reduction of factoring to order finding. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 11.

Q62. What is Integer Factorization and why is it important in quantum computation?

Explanation. Let us note a few facts, which we will not prove or discuss in any detail. First, if N is a prime number or a prime power (i.e., pk for some prime p and integer k 1), then there are efficient classical algorithms for solving the integer factorization problem in these cases. So, we can imagine that we first run such an algorithm on the input N ; if it succeeds then we are done, otherwise we continue on under the assumption that N has at least two distinct prime factors (i.e., m 2). Next, it is enough to have an algorithm that takes a composite N as input and just finds two integers u, v 2 such that N = uv. If we have such an algorithm, we can run it recursively (interleaved with the classical algorithm for prime powers) to find a complete prime factorization Finally, if our goal is to find integers u, v 2 such that N = uv for N an even composite number, then we really don't need to work very hard; let u = 2 and v = N/2. Now, consider the process described in Figure 2. Why does it work? The basic idea is as follows. Suppose that the random choice of a is in Z N (which is very likely), and that the order r

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Integer Factorization. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 11.

Q63. What is Greatest common divisor and why is it important in quantum computation?

Explanation. We will begin this lecture by finishing the discussion of how the order finding problem can be Assume that we are given a Z N , and our goal is to calculate the order of a in Z N , which we suppose is r. Recall that we defined a unitary operation Ma so that Last time we considered what happens when we perform phase estimation on the state | 1 i. The eigenvalue associated with | 1 i is = e2 i(1/r) . Assuming we were to perform the phase estimation procedure several times and take the most commonly appearing result, with very high probability we will have an approximation j/2m that is close to 1/r. Our guess for r would therefore be 2m /j, rounded off to the nearest integer. We determined that if m = 2n and the phase estimation procedure succeeds in finding a best m bit approximation j/2 m to 1/r, then rounding 2m /j to the nearest integer will indeed give r. Unfortunately we do not know how to get our hands on | 1 i. In fact it seems like one would need to know r before preparing it, but if we know r there is no need to bother. There is no known way to prepare | 1 i without knowing r, so a different approach is required. Instead, we considered running the phase estimation procedure on the state

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Greatest common divisor. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 11.

Q64. How does Shor's algorithm work, step by step?

Explanation. Reduction of factoring to order finding I've mentioned several times that we can factor integers efficiently using quantum computers, and that this is the problem solved by Shor's algorithm. Really, it turns out that "Shor's algorithm" is more of a technique than a specific algorithm, and it can be used to solve an interesting collection of problems. (Order finding is one of the simplest to describe.) Now let us see that the integer factoring problem can be efficiently solved given an algorithm for order-finding. In other words, factoring reduces to order-finding. To be specific, it is a ran- domized polynomial-time Turing reduction, which is probably different from (but similar to) the types of reductions you might have seen in CPSC 313 and 413. The idea, however, is essentially the same-if you have an efficient algorithm for order finding, then the reduction gives you an Let us note a few facts, which we will not prove or discuss in any detail. First, if N is a prime num- ber or a prime power (i.e., pk for some prime p and integer k 1), then there are efficient classical algorithms for solving the integer factorization problem in these cases. So, we can imagine that we first run such an algorithm on the input N ; if it succeeds then we are done, otherwise we continue on under the assumption that N has at least two distinct p

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Shor's algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 11.

Lecture 11 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 12: Grover's Algorithm

Core mathematical relation

$$|s\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle, \qquad O|x\rangle = (-1)^{f(x)}|x\rangle.$$

Q65. What is Unstructured search and why is it important in quantum computation?

Explanation. We will be returning to the black-box context that was used to describe Deutsch's algorithm, the Deutsch-Jozsa algorithm, and Simon's algorithm. (The fact that Grover's algorithm works in this very general context represents one of its strengths, because in general there will not need to be any promises on the black box.) Suppose that we have a function

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Unstructured search. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 12.

Q66. How does Description of Grover's algorithm work, step by step?

Explanation. Grover's algorithm is remarkably simple to describe. We will need to use the following two unitary

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Description of Grover's algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 12.

Q67. How does Analysis of Grover's algorithm work, step by step?

Explanation. As you would imagine, analyzing Grover's algorithm is more difficult than describing it. When analyzing it, it will help to think about reflections and rotations in the plane. Here is a reflection of a vector | i about the line L:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Analysis of Grover's algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 12.

Q68. What is Amplitude amplification and why is it important in quantum computation?

Explanation. We will be returning to the black-box context that was used to describe Deutsch's algorithm, the Deutsch-Jozsa algorithm, and Simon's algorithm. (The fact that Grover's algorithm works in this very general context represents one of its strengths, because in general there will not need to be any promises on the black box.) Suppose that we have a function

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Amplitude amplification. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 12.

Q69. What is Oracle phase flip and why is it important in quantum computation?

Explanation. We will be returning to the black-box context that was used to describe Deutsch's algorithm, the Deutsch-Jozsa algorithm, and Simon's algorithm. (The fact that Grover's algorithm works in this very general context represents one of its strengths, because in general there will not need to be any promises on the black box.) Suppose that we have a function

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Oracle phase flip. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 12.

Lecture 12 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 13: Grover's Algorithm (continued)

Core mathematical relation

$$G = (2|s\rangle\langle s| - I)O, \qquad T \approx \frac{\pi}{4}\sqrt{N/M}.$$

Q70. How does Grover's Algorithm work, step by step?

Explanation. 1. Let X be an n-qubit quantum register with initial state |0n i. Perform H n on X. 2. Apply to the register X the transformation

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Grover's Algorithm. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 13.

Q71. What is the operations Zf and Z0 are defined as and how does it work?

Explanation. The algorithm requires k queries to the black box Bf for whatever value of k we choose in step 2. For the analysis, we had defined sets A and B as follows:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The operations Zf and Z0 are defined as. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 13.

Q72. What is Geometric rotation and why is it important in quantum computation?

Explanation. 1. Let X be an n-qubit quantum register with initial state |0n i. Perform H n on X. 2. Apply to the register X the transformation

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Geometric rotation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 13.

Q73. What is Number of iterations and why is it important in quantum computation?

Explanation. 1. Let X be an n-qubit quantum register with initial state |0n i. Perform H n on X. 2. Apply to the register X the transformation

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Number of iterations. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 13.

Q74. What is Search success probability and why is it important in quantum computation?

Explanation. 1. Let X be an n-qubit quantum register with initial state |0n i. Perform H n on X. 2. Apply to the register X the transformation

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Search success probability. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 13.

Lecture 13 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 14: Quantum information revisited

Core mathematical relation

$$\rho = \sum_j p_j |\psi_j\rangle\langle\psi_j|, \quad \rho \succeq 0, \quad \operatorname{Tr}\rho = 1.$$

Q75. What is Density matrices and why is it important in quantum computation?

Explanation. So far nothing is new except for a little bit of notation. In order to describe what really is new, it is helpful to start with a special case. Suppose that in the "old" representation, a register X having classical state set is in a quantum state | i X for X = C( ). The "new" way of representing We have seen objects like this before (in the analysis of Grover's algorithm. It is effectively a matrix. For example, suppose = {0, 1} and | i = |0i + |1i. Then

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Density matrices. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Q76. What is Operations on density matrices and why is it important in quantum computation?

Explanation. In our "old" description of quantum information, a unitary operation U applied to a state | i resulted in the state U | i. In the density matrix description, a unitary operation U applied to a pure state | i h | results in the density matrix This is consistent with the observation that (U | i) = h | U . More generally, applying U to the mixed state results in state U U . You can easily check that the two required conditions of density matrices are necessarily met by this new matrix. We can, however, consider a much more general set of possible operations than just unitary operations. Any operation that can be written as

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Operations on density matrices. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Q77. What is Mixed states and why is it important in quantum computation?

Explanation. So far nothing is new except for a little bit of notation. In order to describe what really is new, it is helpful to start with a special case. Suppose that in the "old" representation, a register X having classical state set is in a quantum state | i X for X = C( ). The "new" way of representing We have seen objects like this before (in the analysis of Grover's algorithm. It is effectively a matrix. For example, suppose = {0, 1} and | i = |0i + |1i. Then

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Mixed states. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Q78. What is Pure states and why is it important in quantum computation?

Explanation. Example 9. Suppose Alice has a qubit A. She flips a fair coin: if the result is HEADS she prepares A in the state |0i, and if the result is TAILS she prepares A in the state |1i. She gives Bob the qubit without revealing the result of the coin-flip. Bob's knowledge of the qubit is described by the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Pure states. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Q79. What is Partial trace and why is it important in quantum computation?

Explanation. cause when extended by linearity to arbitrary matrices it satisfies Tr Y (X Y) = Tr(Y)X for all X L(X) and Y L(Y).) We also refer to the action corresponding to this operation as tracing out the space Y. To express TrY in the form of an admissible operation, let denote the set of

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Partial trace. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Q80. What is Quantum channels and why is it important in quantum computation?

Explanation. So far nothing is new except for a little bit of notation. In order to describe what really is new, it is helpful to start with a special case. Suppose that in the "old" representation, a register X having classical state set is in a quantum state | i X for X = C( ). The "new" way of representing We have seen objects like this before (in the analysis of Grover's algorithm. It is effectively a matrix. For example, suppose = {0, 1} and | i = |0i + |1i. Then

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum channels. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 14.

Lecture 14 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 15: Quantum information revisited (continued)

Core mathematical relation

$$|\psi\rangle = \sum_j \sqrt{\lambda_j}|u_j\rangle|v_j\rangle, \quad \rho_A = \mathrm{Tr}_B|\psi\rangle\langle\psi|.$$

Q81. What is More on the partial trace and why is it important in quantum computation?

Explanation. In the previous lecture we were discussing an important admissible operation called the partial trace. We will continue to discuss this operation in the first part of this lecture. Recall that if we have registers X and Y with corresponding spaces X and Y, then the admissi- ble operation that describes the action of throwing away Y and leaving X alone is the partial trace (over the space Y). Assuming that Y = C( ) for some arbitrary classical state set , we can express the partial trace over Y as follows: You could replace the vectors {|ai : a } (and their corresponding bra vectors) by any other orthonormal basis of Y without changing the operation. Another way of expressing the partial trace, which makes this fact apparent, is TrY = IL(X) Tr (where IL(X) refers to the identity operator on L(X), which is the admissible operation that corresponds to doing nothing at all). One can define TrX , the partial trace over the space X , similarly to TrY . More generally, we could imagine any number of registers and consider the operation that corresponds to discarding Let us now go back to the simple case of two qubits, and consider what happens when the partial trace is applied to an entangled state. Suppose that the pair of qubits (X, Y) is in the state

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with More on the partial trace. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Q82. What is Purifications and why is it important in quantum computation?

Explanation. Suppose that X is a register with corresponding space X = C( ), and let D(X) be a mixed state of X. Sometimes it is helpful to view as being the reduced state of some pure state of the pair (X, Y), where Y is some other register. It is always possible to do this so long as the space Y corresponding to the register Y has dimension at least that of X . (Really, as long as the dimension of Y is as large as the rank of , this will be possible.) One way to see this is to consider a spectral where n = | |, (p1 , . . . , pn) is a probability vector, and {| 1 i , . . . , | n i} is an orthonormal basis of X . Every density matrix has such a decomposition. For Y = X , define a unit vector | i X Y

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Purifications. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Q83. What is the Schmidt decomposition and how does it work?

Explanation. By considering reduced states of bipartite systems, we can learn something interesting about pure Suppose that | i X Y is a pure quantum state of registers (X, Y), and let D(X) denote Like all density matrices, is positive semidefinite and has trace equal to 1. It is therefore possible

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The Schmidt decomposition. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Q84. What is Measurements and why is it important in quantum computation?

Explanation. Finally, the last remaining piece of the general formalism of quantum information is measurements. There are more general types of measurements than the simple type we have discussed previously. Mathematically speaking, they are described in a manner similar to admissible operations. Suppose that X is a quantum register with classical state set and corresponding vector space X = C( ). A measurement on X can have any finite, nonempty set of possible outcomes. Formally, a measurement is described by a collection If the register X is measured with respect to this measurement while it is in the state D(X), the outcome must be some element of . As before, the outcome may be random: for each a ,

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Measurements. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Q85. What is Reduced states and why is it important in quantum computation?

Explanation. These density matrices are different, so there is some measurement that Bob could make that would give him some information about which state they share. If Bob measures with respect to the standard basis, he could guess that they shared | 1 i if the measurement result was 1 and guess | 2 i if the measurement result was 0. He would be right 64% of the time, implying a gain of partial

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Reduced states. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Q86. What is Measurement operators and why is it important in quantum computation?

Explanation. In the previous lecture we were discussing an important admissible operation called the partial trace. We will continue to discuss this operation in the first part of this lecture. Recall that if we have registers X and Y with corresponding spaces X and Y, then the admissi- ble operation that describes the action of throwing away Y and leaving X alone is the partial trace (over the space Y). Assuming that Y = C( ) for some arbitrary classical state set , we can express the partial trace over Y as follows: You could replace the vectors {|ai : a } (and their corresponding bra vectors) by any other orthonormal basis of Y without changing the operation. Another way of expressing the partial trace, which makes this fact apparent, is TrY = IL(X) Tr (where IL(X) refers to the identity operator on L(X), which is the admissible operation that corresponds to doing nothing at all). One can define TrX , the partial trace over the space X , similarly to TrY . More generally, we could imagine any number of registers and consider the operation that corresponds to discarding Let us now go back to the simple case of two qubits, and consider what happens when the partial trace is applied to an entangled state. Suppose that the pair of qubits (X, Y) is in the state

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Measurement operators. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 15.

Lecture 15 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 16: Quantum error correction

Core mathematical relation

$$\alpha|0\rangle + \beta|1\rangle \longmapsto \alpha|000\rangle + \beta|111\rangle.$$

Q87. What is Classical repetition codes and why is it important in quantum computation?

Explanation. We will start with a very simple classical error correcting code, the 3 bit repetition code. Initially, just for the sake of having a concrete error model to think about, we will consider errors described by the binary symmetric channel. Here, we have a channel from Alice to Bob that carries 1 bit at a time. There is noise on the channel, so the bit sent by Alice is not always received correctly by Bob. Specifically, we suppose that the noise is parameterized by some real number p [0, 1] that represents the probability of an error: if Alice sends the bit b, then Bob receives b with probability Suppose that Alice needs to send a very important bit to Bob, and the two of them do not wish to accept that Bob will receive the wrong bit with probability p. What can they do? The answer is that they can use an error correcting code to decrease the probability of error. Possibly the simplest example of an error correcting code is the 3 bit repetition code. The encoding is simply to repeat the bit to be transmitted three times, and decoding is just taking the most frequently appearing bit (which is the same as computing the majority function).

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Classical repetition codes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Q88. What is Protecting quantum information against bit-flips and why is it important in quantum computation?

Explanation. Can we encode quantum information in a similar way to the previous example? For instance, suppose Alice has a qubit in the pure state | i = |0i + |1i that she wishes to send to Bob through a noisy quantum channel. We will have to discuss exactly what a noisy quantum channel is, but let us just think about this notion informally for now. Can something analogous to the First let us note that it is not acceptable to encode | i as | i | i | i. In case Alice does not know and , this would violate the no-cloning theorem. It is also not clear how Bob would A more sensible analogue of the repetition code is to perform this encoding:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Protecting quantum information against bit-flips. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Q89. What is Other quantum errors and why is it important in quantum computation?

Explanation. Although the above code protects against bit-flips, this is not good enough in general. The problem is that there are different types of quantum errors. For example, a phase-flip error could occur:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Other quantum errors. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Q90. What is Phase flips and why is it important in quantum computation?

Explanation. is yes, but we need to concatenate the codes. In other words, we first apply the code that protects against phase-flips, and then encode each of the three resulting qubits using the code that protects against bit-flips. In other words, we will encode |0i + |1i as This is known as Shor's 9 qubit code, because it was first discovered by Shor. A circuit for the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Phase flips. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Q91. What is Error syndromes and why is it important in quantum computation?

Explanation. We will start with a very simple classical error correcting code, the 3 bit repetition code. Initially, just for the sake of having a concrete error model to think about, we will consider errors described by the binary symmetric channel. Here, we have a channel from Alice to Bob that carries 1 bit at a time. There is noise on the channel, so the bit sent by Alice is not always received correctly by Bob. Specifically, we suppose that the noise is parameterized by some real number p [0, 1] that represents the probability of an error: if Alice sends the bit b, then Bob receives b with probability Suppose that Alice needs to send a very important bit to Bob, and the two of them do not wish to accept that Bob will receive the wrong bit with probability p. What can they do? The answer is that they can use an error correcting code to decrease the probability of error. Possibly the simplest example of an error correcting code is the 3 bit repetition code. The encoding is simply to repeat the bit to be transmitted three times, and decoding is just taking the most frequently appearing bit (which is the same as computing the majority function).

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Error syndromes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Q92. What is Quantum error correction and why is it important in quantum computation?

Explanation. We will start with a very simple classical error correcting code, the 3 bit repetition code. Initially, just for the sake of having a concrete error model to think about, we will consider errors described by the binary symmetric channel. Here, we have a channel from Alice to Bob that carries 1 bit at a time. There is noise on the channel, so the bit sent by Alice is not always received correctly by Bob. Specifically, we suppose that the noise is parameterized by some real number p [0, 1] that represents the probability of an error: if Alice sends the bit b, then Bob receives b with probability Suppose that Alice needs to send a very important bit to Bob, and the two of them do not wish to accept that Bob will receive the wrong bit with probability p. What can they do? The answer is that they can use an error correcting code to decrease the probability of error. Possibly the simplest example of an error correcting code is the 3 bit repetition code. The encoding is simply to repeat the bit to be transmitted three times, and decoding is just taking the most frequently appearing bit (which is the same as computing the majority function).

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum error correction. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 16.

Lecture 16 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 17: General quantum errors; CSS codes

Core mathematical relation

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad Y = iXZ.$$

Q93. What is General quantum errors and why is it important in quantum computation?

Explanation. In the previous lecture we discussed Shor's 9 qubit code, which can be viewed as an encoding of one qubit into three blocks of three qubits. It protects against up to 1 bit-flip error in each block and any number of phase-flip errors contained in any one block. In particular, it can protect against occurring on a single qubit. The code of course does not require that an error takes place to work correctly, so we can add the identity matrix to the above list. Finally, because y = -i x z and global phase factors are irrelevant, we may say that Shor's 9 qubit code protects against any of the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with General quantum errors. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Q94. What is CSS codes and why is it important in quantum computation?

Explanation. The last thing we will discuss on the topic of quantum error correction is CSS codes, named after their co-discoverers Robert Calderbank, Peter Shor, and Andrew Steane. These codes are interesting because they illustrate how classes of classical error correcting codes can sometimes be turned into classes of quantum codes. They are also useful for one of the simpler proofs of security of the BB84 quantum key-distribution protocol, which we will discuss next week. In order to discuss CSS codes, we will need to first learn a little bit about classical linear codes. Some terminology will be helpful when doing this. First, when we refer to an [n, k] code, we mean an error correcting code that encodes k bits into n (so we will always have n > k). Formally, we define an [n, k] code simply to be a subset of Zn2 of size 2k . Here, and throughout this discussion, we are identifying strings of length n with (column) vectors in Zn2 in the most straightforward way. The elements of such a code C Z n2 are An [n, k] code C is said to have distance d if the Hamming distance between any two distinct code words y, z C is at least d, and d is the largest positive integer that satisfies this property. The Hamming distance between strings y and z is the number of positions in which they disagree. An [n, k] code with minimum distance d is sometime

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with CSS codes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Q95. What is Classical linear codes and why is it important in quantum computation?

Explanation. In order to discuss CSS codes, we will need to first learn a little bit about classical linear codes. Some terminology will be helpful when doing this. First, when we refer to an [n, k] code, we mean an error correcting code that encodes k bits into n (so we will always have n > k). Formally, we define an [n, k] code simply to be a subset of Zn2 of size 2k . Here, and throughout this discussion, we are identifying strings of length n with (column) vectors in Zn2 in the most straightforward way. The elements of such a code C Z n2 are An [n, k] code C is said to have distance d if the Hamming distance between any two distinct code words y, z C is at least d, and d is the largest positive integer that satisfies this property. The Hamming distance between strings y and z is the number of positions in which they disagree. An [n, k] code with minimum distance d is sometimes referred to as an [n, k, d] code. Such a code can correct bit-flips on any number of bits that is strictly less than half of the code's minimum distance. An [n, k] linear code is a code that happens to form a subspace of Z n2 of dimension k, where all arithmetic takes place in Z2 . In the case of a linear code, the minimum distance is the same as the minimal Hamming weight (or number of 1s) of any nonzero codeword. There are two matrices that can be associated

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Classical linear codes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Q96. What is Dual codes and why is it important in quantum computation?

Explanation. Suppose that we have an [n, k] linear code C. The dual code to C is denoted C , and contains all strings that have modulo 2 inner product 0 with all codewords of C. In symbols: Therefore, C is an [n, n - k] linear code. If we have a generator matrix G and a parity check matrix K for C, then the generator matrix for C is K T and the parity check matrix is GT . A code C is weakly self dual if C C and is strictly self dual (or just self dual) if C = C .

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Dual codes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Q97. What is Pauli errors and why is it important in quantum computation?

Explanation. In the previous lecture we discussed Shor's 9 qubit code, which can be viewed as an encoding of one qubit into three blocks of three qubits. It protects against up to 1 bit-flip error in each block and any number of phase-flip errors contained in any one block. In particular, it can protect against occurring on a single qubit. The code of course does not require that an error takes place to work correctly, so we can add the identity matrix to the above list. Finally, because y = -i x z and global phase factors are irrelevant, we may say that Shor's 9 qubit code protects against any of the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Pauli errors. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Q98. What is Syndrome extraction and why is it important in quantum computation?

Explanation. In the previous lecture we discussed Shor's 9 qubit code, which can be viewed as an encoding of one qubit into three blocks of three qubits. It protects against up to 1 bit-flip error in each block and any number of phase-flip errors contained in any one block. In particular, it can protect against occurring on a single qubit. The code of course does not require that an error takes place to work correctly, so we can add the identity matrix to the above list. Finally, because y = -i x z and global phase factors are irrelevant, we may say that Shor's 9 qubit code protects against any of the

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Syndrome extraction. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 17.

Lecture 17 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 18: Quantum Key Distribution

Core mathematical relation

$$c = m \oplus k, \qquad m = c \oplus k.$$

Q99. What is a perfectly secure cryptosystem and how does it work?

Explanation. Let us first consider a purely classical situation. Suppose that Alice and Bob can communicate through a classical channel, and a third party Eve is able to monitor the communications on that channel. Alice wishes to transmit a particular message to Bob, and wants Eve to learn as little as possible about it. Eve wants to learn as much as possible about the message. In cryptography we always need to be precise about our assumptions and model. In this exam- ple, we will assume that the channel Alice and Bob are communicate through is an authenticated channel: when Bob receives a message, he can verify that it came from Alice (and vice versa). We will also assume that Eve cannot jam the channel or modify messages sent over it, meaning that Alice and Bob always receive one another's messages without error. There exists a provably secure classical scheme, known as the one-time pad, for accomplishing the task at hand. It requires that Alice and Bob share a sequence of random bits, called a private key, that is secret from Eve. This private key could have been created a long time ago, and is completely independent of the message Alice wishes to send. The key must be of the same length as the message. The scheme works as follows.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with A perfectly secure cryptosystem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Q100. What is One-time Pad and why is it important in quantum computation?

Explanation. Alice and Bob share a private key K {0, 1}n . Alice wishes to send Bob a message M {0, 1}n . She computes E = K M (bitwise exclusive OR) and sends E (the encrypted message) to Bob. Bob receives E, and computes E K. The result is E K = M K K = M . Eve can learn nothing (aside from the length of the message) by looking at E. If K is uniformly One very important note is that the key K can only be used once (hence the name "one-time pad"). If they were to use it twice with two different messages, the encrypted texts would be

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with One-time Pad. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Q101. What is Quantum key distribution and why is it important in quantum computation?

Explanation. The aim of quantum key distribution is to allow Alice and Bob to generate a secure private key that can be used for the one-time pad without having to meet privately. They will be able to accomplish this task by using quantum information. There are a few different schemes for doing 1. BB84. Easy to implement, relatively hard to prove security. 3. Lo-Chau. Hard to implement, easy to prove security.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum key distribution. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Q102. What is the BB84 Key Exchange Protocol (first stage) and how does it work?

Explanation. Alice randomly generates two strings of bits x, y {0, 1}m . Define | 00 i = |0i, | 10 i = |1i, | 01 i = |+i, and | 11 i = |-i. These four states may be pictured and sends these m qubits over a quantum channel to Bob. Bob receives m qubits, although they may not longer be in the state | x,y i because Eve may have tampered with them, or possibly the channel is noisy. (To understand how the protocol works, it is helpful to imagine first that Eve is not present and the channel is not noisy, so Bob receives

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with The BB84 Key Exchange Protocol (first stage). Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Q103. What is Basis reconciliation and why is it important in quantum computation?

Explanation. Let us first consider a purely classical situation. Suppose that Alice and Bob can communicate through a classical channel, and a third party Eve is able to monitor the communications on that channel. Alice wishes to transmit a particular message to Bob, and wants Eve to learn as little as possible about it. Eve wants to learn as much as possible about the message. In cryptography we always need to be precise about our assumptions and model. In this exam- ple, we will assume that the channel Alice and Bob are communicate through is an authenticated channel: when Bob receives a message, he can verify that it came from Alice (and vice versa). We will also assume that Eve cannot jam the channel or modify messages sent over it, meaning that Alice and Bob always receive one another's messages without error. There exists a provably secure classical scheme, known as the one-time pad, for accomplishing the task at hand. It requires that Alice and Bob share a sequence of random bits, called a private key, that is secret from Eve. This private key could have been created a long time ago, and is completely independent of the message Alice wishes to send. The key must be of the same length as the message. The scheme works as follows.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Basis reconciliation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Q104. What is Eavesdropping detection and why is it important in quantum computation?

Explanation. Let us first consider a purely classical situation. Suppose that Alice and Bob can communicate through a classical channel, and a third party Eve is able to monitor the communications on that channel. Alice wishes to transmit a particular message to Bob, and wants Eve to learn as little as possible about it. Eve wants to learn as much as possible about the message. In cryptography we always need to be precise about our assumptions and model. In this exam- ple, we will assume that the channel Alice and Bob are communicate through is an authenticated channel: when Bob receives a message, he can verify that it came from Alice (and vice versa). We will also assume that Eve cannot jam the channel or modify messages sent over it, meaning that Alice and Bob always receive one another's messages without error. There exists a provably secure classical scheme, known as the one-time pad, for accomplishing the task at hand. It requires that Alice and Bob share a sequence of random bits, called a private key, that is secret from Eve. This private key could have been created a long time ago, and is completely independent of the message Alice wishes to send. The key must be of the same length as the message. The scheme works as follows.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Eavesdropping detection. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 18.

Lecture 18 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 19: Impossibility of Quantum Bit Commitment

Core mathematical relation

$$\rho_B^{(0)} = \rho_B^{(1)} \quad \text{(perfect concealment)}.$$

Q105. What is Bit commitment and why is it important in quantum computation?

Explanation. First let us define what bit commitment is. Alice has a bit b that she wishes to commit to Bob, but she doesn't want Bob to know what it is until she chooses to reveal it. Although Bob should not be able to determine b before Alice reveals, he should be sure that Alice cannot change the bit after it is committed. The two key properties of any bit commitment protocol are therefore that it Binding. Alice should not be able to change the bit she committed. Concealing. Bob should not be able to identify the bit that Alice committed until she reveals it.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Bit commitment. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 19.

Q106. What is Sketch of impossibility proof and why is it important in quantum computation?

Explanation. The impossibility of quantum bit commitment relies on the following fact, which we proved in Fact. Suppose | i , | i A B satisfy TrA | i h | = TrA | i h |. Then there exists a unitary operator U L(A) such that (U I) | i = | i. It happens to be the case that there are approximate versions of the above fact, but because we have not discussed meaningful distance measures for quantum states it will not be possible to go into greater detail about this. Expressed informally, the approximate versions are of this form: if then there exists a unitary operator U L(A) such that (U I) | i | i. Now let us apply the above fact to the problem of bit commitment. To begin, suppose that Alice and Bob use a "purely quantum" protocol to supposedly implement bit commitment. What we mean by this is that Alice and Bob apply only unitary operations and send quantum information back and forth-so assuming Alice and Bob are both being honest there are no measurements made until the end of the protocol, and there is no noise or other non-unitary transformations during the There are necessarily two phases of any bit commitment protocol: the commit phase and the reveal phase. During the commit phase, Alice and Bob may perform some sequence of unitary operations and send qubits back and forth to one another any number of times. Assume that Alice

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Sketch of impossibility proof. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 19.

Q107. What is Concealing versus binding and why is it important in quantum computation?

Explanation. First let us define what bit commitment is. Alice has a bit b that she wishes to commit to Bob, but she doesn't want Bob to know what it is until she chooses to reveal it. Although Bob should not be able to determine b before Alice reveals, he should be sure that Alice cannot change the bit after it is committed. The two key properties of any bit commitment protocol are therefore that it Binding. Alice should not be able to change the bit she committed. Concealing. Bob should not be able to identify the bit that Alice committed until she reveals it.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Concealing versus binding. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 19.

Q108. What is Entanglement attacks and why is it important in quantum computation?

Explanation. First let us define what bit commitment is. Alice has a bit b that she wishes to commit to Bob, but she doesn't want Bob to know what it is until she chooses to reveal it. Although Bob should not be able to determine b before Alice reveals, he should be sure that Alice cannot change the bit after it is committed. The two key properties of any bit commitment protocol are therefore that it Binding. Alice should not be able to change the bit she committed. Concealing. Bob should not be able to identify the bit that Alice committed until she reveals it.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Entanglement attacks. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 19.

Q109. What is No-go theorem and why is it important in quantum computation?

Explanation. First let us define what bit commitment is. Alice has a bit b that she wishes to commit to Bob, but she doesn't want Bob to know what it is until she chooses to reveal it. Although Bob should not be able to determine b before Alice reveals, he should be sure that Alice cannot change the bit after it is committed. The two key properties of any bit commitment protocol are therefore that it Binding. Alice should not be able to change the bit she committed. Concealing. Bob should not be able to identify the bit that Alice committed until she reveals it.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with No-go theorem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 19.

Lecture 19 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 20: Bell inequalities and nonlocality

Core mathematical relation

$$|E_{00} + E_{01} + E_{10} - E_{11}| \leq 2 \quad \text{(local hidden variables)}.$$

Q110. What is Nonlocal games and why is it important in quantum computation?

Explanation. We saw very early in the course that entanglement is a useful resource when combined with other resources such as in teleportation and super-dense coding. However, we know that entanglement alone cannot be used by directly for communication. For instance, if Alice and Bob share some entangled state | i A B, and Alice performs any measurement on her part of the state, the results are completely determined by her reduced state TrB | i h | and are independent of any operation that Bob could perform on his half of the state. The situation is similar for Bob. However, if both Alice and Bob perform measurements, then there can be interesting correla- tions between their measurement outcomes that would not be possible using classical information. These correlations can be studied by means of nonlocal games. In a nonlocal game there are two or more players (Alice, Bob, Charlie, Diane, . . .) who are viewed as cooperating with one another. A referee runs the game, and all communication in the game is between the players and the referee-no communication directly between any of the players is permitted. The referee randomly selects a question for each player and sends each question to the appropriate player. Each player sends an answer back to the referee, and based on all of the questions and answers the referee determines whether th

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Nonlocal games. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Q111. What is Bell Inequalities and why is it important in quantum computation?

Explanation. You may be wondering why the title of the lecture is "Bell inequalities and nonlocality". In the physical interpretation of nonlocal games, the bounds on classical strategies (or on local hidden variable theories) are typically known as Bell Inequalities, and quantum strategies (or experiments) that beat these bounds are said to violate a Bell inequality. Typically you would see the Bell Inequality (known as the CHSH inequality) associated with You can probably convince yourself easily that if A0 , A1 , B0 and B1 are treated as ordinary vari- ables taking values in [-1, 1], then the inequality must hold. This is equivalent to saying that there is no classical strategy that wins the CHSH game with probability exceeding 3/4. Because there exist observables A0 , A1 , B0 and B1 with eigenvalues in [-1, 1] and a state | i such that

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Bell Inequalities. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Q112. What is Tsirelson's bound and why is it important in quantum computation?

Explanation. You might wonder if it is possible to do better than cos 2 (/8) with a quantum strategy for the CHSH game. The answer is "no" due to Tsirelson's bound: for any choice of observables A0 , A1 , B0 and B1 with eigenvalues in [-1, 1] and any state | i. This includes the possibility that | i is a state on many qubits, not just two. This can be shown as fol- lows, using the fact that kA0 k , kA1 k , kB0 k , kB1 k 1 along with some other simple properties

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Tsirelson's bound. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Q113. What is CHSH game and why is it important in quantum computation?

Explanation. there are only two players: Alice and Bob. The referee chooses questions rs {00, 01, 10, 11} uniformly, and Alice and Bob must each answer a single bit: a for Alice, b for Bob. They win if a b = r s, i.e., the winning conditions are: By similar reasoning to the GHZ game, the maximum probability with which a classical strategy There are two ways to describe the quantum strategy that we will consider. The first way will be more familiar, but the second way is useful and this is a good opportunity to learn something

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with CHSH game. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Q114. What is Entangled strategies and why is it important in quantum computation?

Explanation. We saw very early in the course that entanglement is a useful resource when combined with other resources such as in teleportation and super-dense coding. However, we know that entanglement alone cannot be used by directly for communication. For instance, if Alice and Bob share some entangled state | i A B, and Alice performs any measurement on her part of the state, the results are completely determined by her reduced state TrB | i h | and are independent of any operation that Bob could perform on his half of the state. The situation is similar for Bob. However, if both Alice and Bob perform measurements, then there can be interesting correla- tions between their measurement outcomes that would not be possible using classical information. These correlations can be studied by means of nonlocal games. In a nonlocal game there are two or more players (Alice, Bob, Charlie, Diane, . . .) who are viewed as cooperating with one another. A referee runs the game, and all communication in the game is between the players and the referee-no communication directly between any of the players is permitted. The referee randomly selects a question for each player and sends each question to the appropriate player. Each player sends an answer back to the referee, and based on all of the questions and answers the referee determines whether th

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Entangled strategies. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Q115. What is Classical strategies and why is it important in quantum computation?

Explanation. strategies which allow the players to share an entangled state and to perform quantum measurements to determine their answers to the referees questions. What is interesting is that sometimes quantum strategies can allow the players to win with a much higher probability than they could Example 24 (The GHZ game). Consider the following game, where there are three players: Alice, Bob, and Charlie. Each question and each answer will be a single bit. The referee chooses a three bit string r s t uniformly from the set {000, 011, 101, 110} and sends r to Alice, s to Bob, and t to Charlie. Their answers must be bits: a from Alice, b from Bob, and c from Charlie. They win if a b c = r s t and lose otherwise. The following table lists the winning condition for each possible set of questions: What is the maximum probability of winning if Alice, Bob, and Charlie use a classical strategy? Fir s t consider a deterministic strategy, where each answer is a function of the question received and no randomness is used by the players. Let us write ar , bs and ct to denote the answers that would be given for each choice of r, s, and t. For example, if a 0 = 1 and a1 = 0, then Alice always answers the question 0 with 1 and the question 1 with 0. The winning conditions can be expressed

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Classical strategies. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 20.

Lecture 20 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 21: Quantum communication complexity

Core mathematical relation

$$I(X;Y) \leq \chi = S\left(\sum_x p_x \rho_x\right) - \sum_x p_x S(\rho_x).$$

Q116. What is Holevo's Theorem and why is it important in quantum computation?

Explanation. Let us begin with a simplified version of the theorem, which is sufficient to give the spirit of Holevo's Theorem, but not too much more. Holevo's Theorem (informal, simplified version). If Alice and Bob share no prior entanglement, then it is necessary for them to exchange at least n qubits (or bits) in order for Alice to communicate n classical bits of information to Bob. If they do share prior entanglement, then they must exchange

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Holevo's Theorem. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Q117. What does Communication complexity mean and why does it matter?

Explanation. Although quantum information cannot reduce the direct cost of communication as Holevo's The- orem implies, there are many distributed computational tasks that require communication but not direct communication in an information theoretic sense. Communication complexity studies such The general type of problem considered by communication complexity is as follows. Alice receives an input string x {0, 1}n and Bob receives an input string y {0, 1}n . Their goal is to compute f (x, y) for some function f : {0, 1}n {0, 1}n {0, 1}. We can also consider the more general case where Alice and Bob receive strings of different lengths and the output of the function f is longer than just one bit, but the simpler model will be sufficient for this lecture. They wish to accomplish this task using as little communication as possible. We can measure the amount of communication in the number of bits or qubits sent, depending on whether we are considering classical or quantum communication complexity. For simplicity let us be more precise and say that Bob should learn the value f (x, y)-if instead we required that both Alice and Bob learn f (x, y), the amount of communication required would differ by at most one bit. We are typically interested in the cost for the worst case choice of x and y, and will consider the minimum such cost over al

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Communication complexity. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Q118. What does Quantum communication complexity mean and why does it matter?

Explanation. It is a natural question to ask whether quantum information can reduce communication complexity. In light of Holevo's Theorem, it may be tempting to think not, but in fact significant reductions are Q(DISJ), satisfies Q(DISJ) = O(n log n). More specifically, Q(f) refers to quantum commu- turns out that Q(DISJ) = (n); the elimination of the logarithmic factor is clever optimization of the method we will see, while the lower bound requires a rather difficult proof. Before describing the protocol, let us consider the following problem. Suppose that the strings x and y are fixed, but unknown to us, and imagine that we have access to a black box B g for a function g : {0, 1}n {0, 1} satisfying

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum communication complexity. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Q119. What is Exponential separation and why is it important in quantum computation?

Explanation. It turns out that an exponential separation between quantum and classical communication complexity is possible, provided we allow for promises on the inputs. Ran Raz proved that the following Alice receives classical descriptions of (i) a unit vector | i Cn and (ii) a projective measure- ment { 0 , 1 } on Cn , and Bob receives a classical description of a unitary operator U L(C n). It is promised that one of k 0 U | ik2 or k 1 U | ik2 is close to 1 (and so the other is close to 0), and the goal is to output 0 or 1 accordingly. It is easy for Alice

and Bob to solve this problem using O(log n) qubits of communication: Alice sends | i to Bob, Bob applies U and sends it back to Alice, and Alice measures U | i with respect to { 0 , 1 }. The challenging part is to prove that no classical probabilistic protocol can achieve this-it is possible to prove a lower bound of (n1/4 / log n) for the classical communication complexity. The proof is very difficult. It is still an open question whether an exponential improvement of classical over quantum communication complexity is possible for a total function f (meaning that no promises are allowed Finally, I would like to mention a variant of the communication complexity model where it is possible to prove an exponential separation between quantum and classical complexities for a tota

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Exponential separation. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Q120. What is Quantum fingerprinting and why is it important in quantum computation?

Explanation. Finally, I would like to mention a variant of the communication complexity model where it is possible to prove an exponential separation between quantum and classical complexities for a total function (i.e., not a promise problem). It is a very weak model called the simultaneous message passing model. As before, Alice receives an input string x and Bob receives y, but now Alice and Bob cannot communicate with one another at all. Instead, they must both send a single message to a third party, called the referee. The referee does not receive any input, but is required to give the output of the function f based on the messages received from Alice and Bob. One can consider several variants of this model-we will compare the classical probabilistic model with the quantum model, where the referee is allowed a small probability of error in both cases. No shared randomness or entanglement will be permitted between Alice and Bob. (If we do allow shared randomness or entanglement, the example to be considered no longer gives an interesting separation between quantum and classical.) The separation will be for the equality function. If we write R k (f) to denote the randomized communication complexity of f in the simultaneous message passing model, then we have In contrast, using a similar notation for the quantum case, we have The techniqu

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum fingerprinting. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Q121. What is Equality testing and why is it important in quantum computation?

Explanation. Let us begin with a simplified version of the theorem, which is sufficient to give the spirit of Holevo's Theorem, but not too much more. Holevo's Theorem (informal, simplified version). If Alice and Bob share no prior entanglement, then it is necessary for them to exchange at least n qubits (or bits) in order for Alice to communicate n classical bits of information to Bob. If they do share prior entanglement, then they must exchange

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Equality testing. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 21.

Lecture 21 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.

Lecture 22: Quantum computational complexity

Core mathematical relation

$$\text{BPP} \subseteq \text{BQP} \subseteq \text{PP} \subseteq \text{PSPACE}.$$

Q122. What does Promise Problems and Complexity Classes mean and why does it matter?

Explanation. Let us start with the notion of a promise problem. We have talked about promise problems in the black box setting, but for this lecture we will assume the input is given explicitly and there are no black boxes. A promise problem is a computational problem where two disjoint sets of inputs (yes inputs and no inputs) must be distinguished. For example, here is the Graph Isomorphism

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Promise Problems and Complexity Classes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q123. What does Quantum complexity classes mean and why does it matter?

Explanation. We can define quantum complexity classes in a similar way to the classes defined above, except based on quantum computations instead of classical (deterministic or probabilistic) computations. Although it is possible to define quantum Turing machines for the purpose of doing this, it turns out to be easier and equivalent to use the quantum circuit model. Let us briefly discuss some assumptions that we need to make concerning quantum circuits in order to have a good model for One technical concern that must be addressed when we discuss quantum circuits for complexity classes is the notion of uniformity. Suppose that we have a collection of quantum circuits one circuit Qx for each possible string x. Then we say that this collection is polynomial-time uniformly generated (or sometimes people say polynomial-time uniform or polynomial-time gen- erated) if there exists a deterministic polynomial-time algorithm that, on input x, outputs a classical Secondly, we need to fix a set of gates that are permitted. Although you could make different (but equivalent) choices, the following collection of gates works out nicely:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum complexity classes. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q124. What is Uniform quantum circuits and why is it important in quantum computation?

Explanation. We can define quantum complexity classes in a similar way to the classes defined above, except based on quantum computations instead of classical (deterministic or probabilistic) computations. Although it is possible to define quantum Turing machines for the purpose of doing this, it turns out to be easier and equivalent to use the quantum circuit model. Let us briefly discuss some assumptions that we need to make concerning quantum circuits in order to have a good model for One technical concern that must be addressed when we discuss quantum circuits for complexity classes is the notion of uniformity. Suppose that we have a collection of quantum circuits one circuit Qx for each possible string x. Then we say that this collection is polynomial-time uniformly generated (or sometimes people say polynomial-time uniform or polynomial-time gen- erated) if there exists a deterministic polynomial-time algorithm that, on input x, outputs a classical Secondly, we need to fix a set of gates that are permitted. Although you could make different (but equivalent) choices, the following collection of gates works out nicely:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Uniform quantum circuits. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q125. What is Quantum proofs: QMA and why is it important in quantum computation?

Explanation. Finally, I would like to just mention an interesting quantum complexity class that gives a quantum analogue of NP (or really MA because the verification is probabilistic). The class is called QMA. Formally, a promise problem A = (Ayes , Ano) is in QMA if there exists (i) a polynomial p, and (ii) a polynomial-time generated family {Qx : x {0, 1} } of circuits, where each Qx is of type (p(|x|), 1), such that these two properties hold:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Quantum proofs: QMA. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q126. What is BQP and why is it important in quantum computation?

Explanation. Now we are ready to define our first quantum complexity class: BQP. It is intended that this class abstracts the notion of a promise problem that could be solved efficiently with a quantum computer. Formally, a promise problem A = (Ayes , Ano) is in BQP if there exists a polynomial-time generated family {Qx : x {0, 1} } of type (0, 1) quantum circuits such that: The idea here is that the input string x is "hard coded" into the circuit Q x . When you run Qx , you start with no qubits, and end up with a 1 qubit mixed state (which we are simply denoting Q x , because it is produced by the circuit Qx). If you measure this qubit with respect to the standard basis, you get the outcome 1, which we interpret as "yes", with probability h1|Q x |1i. A decision problem version of factoring is is BQP because of Shor's algorithm:

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with BQP. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q127. What is QMA and why is it important in quantum computation?

Explanation. It can be shown that even this class is contained in PP. The proof is more difficult than the proof that BQP PP, so I won't have time to discuss it. An interesting question is whether quantum certificates can be more powerful than classical ones. An example of a problem that is known to be in QMA but not known to be in MA (or even QMA but restricting the proof to be classical) is Group Non-membership: given a group and an element of some common super-group, answer "yes" if the element is not contained in the group, and "no" otherwise. This is another example of how the structure of groups can be exploited by Figure 5 contains a diagram of the classes we have discussed in this lecture.

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with QMA. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Q128. What is Integer Factoring and why is it important in quantum computation?

Explanation. Input: Positive integers (N, k) with 2 k N . Yes: There exists a proper factor of N in the range {2, . . . , k}. No: There is no proper factor of N in the range {2, . . . , k}. It also happens to be the case that both Integer Factoring and its complement (where we switch the yes and no conditions) are in NP. We would typically state this as Integer Factoring NP co-NP. A natural question to ask is: how does BQP compare with the classical complexity classes defined above? It is fairly easy to prove that BPP BQP, based on the observation that quantum circuits can simulate coin-flips and perform deterministic computations efficiently. It is also easy to prove BQP EXP: if all of the gates of Qx are replaced by matrices and all of the matrix arithmetic is done explicitly, then at most exponential time (and space) is required to calculate h1|Q x |1i given a description of the circuit Qx . This method can be improved to show BQP PSPACE by performing the same computations in a very space-efficient manner. It may not be obvious how this can be done, but there are well-known methods that can be used to accomplish it. In essence, if you are willing to repeatedly calculate certain numbers over and over, once each time you need them, you don't need to actually store much information to perform matrix arithmetic. We can do somewhat

Study focus. Identify the assumptions, state representation, transformations and measurement outcomes associated with Integer Factoring. Explain which part is classical and which part is quantum, and trace the relevant example in Lecture 22.

Lecture 22 revision check

Explain the central formula above, define each symbol, and reconstruct at least one example or proof from the original notes.