Topics in quantum computing and quantum simulation - week 2

Michał Studziński, Sergii Strelchuk

Gdańsk, Poland, 17-21 July 2022

Abstract

This is a summary of lectures given during the second week of the summer school entitled "Next generation of quantum information scientists. Series of international schools for students in Gdansk". Presented notes are suitable for students who think about doing their master's in quantum computation and information from the perspective of theoretical computer science.

Contents

1 Quantum model of computation

1.1 Differences between classical and quantum computers

Before we go further and learn about certain quantum algorithms that can be used on potential quantum machines/computers. Let us stress here a few fundamental differences between classical and quantum computers. We elaborate on them more later on.

1. **Basic Unit of Information:**

   - **Classical Computing:** Classical computers use bits as the basic unit of information, where each bit can be in one of two states: 0 or 1.
   - **Quantum Computing:** Quantum computers use qubits (quantum bits) as the basic unit of information. Qubits can exist in multiple states simultaneously due to quantum superposition, representing both 0 and 1 at the same time.

2. **Processing Model:**

   - **Classical Computing:** Classical computers perform operations sequentially and deterministically. They process information one step at a time.
   - **Quantum Computing:** Quantum computers leverage quantum parallelism, enabling them to perform operations on multiple qubits simultaneously. They can explore multiple solutions to a problem at once, potentially offering exponential speedup for certain tasks.

3. **Gate Model vs. Quantum Circuit Model:**

   - Classical Computing: Classical computation is typically represented using logic gates (e.g., AND, OR, NOT gates) in a sequential circuit model.
   - Quantum Computing: Quantum computation is represented using quantum gates (e.g., Pauli$-X$, $Hadamard$, $CNOT$ gates) in a quantum circuit model. Quantum gates manipulate qubits through unitary transformations.

4. **Measurement:**

   - **Classical Computing:** Classical bits have definite values (0 or 1) when measured, and measurements are deterministic.
   - **Quantum Computing:** Qubits exist in a superposition of states until measured. Measurement collapses a qubit to one of its possible states with probabilities determined by its amplitudes.

5. **Quantum Entanglement:**

   - **Classical Computing:** Classical bits are independent and uncorrelated with each other.
   - **Quantum Computing:** Qubits can be entangled, meaning that the state of one qubit is intrinsically linked to the state of another qubit, even if they are physically separated.

6. **Error Correction:**

- **Classical Computing:** Classical computers use error-correcting codes to detect and correct errors that may occur during computation. Errors are typically deterministic.
- **Quantum Computing:** Quantum computers face unique challenges with quantum noise and decoherence. Quantum error correction codes are used to mitigate errors, but they require additional qubits and resources.

7. **Computational Complexity:**

   - **Classical Computing:** Classical algorithms have known computational complexity classes, such as P, NP, and exponential time algorithms (e.g., 2^N).
   - **Quantum Computing:** Quantum algorithms can have different complexity classes, with some problems demonstrating exponential speedup over classical algorithms (e.g., factoring with Shor's algorithm).

8. **Hardware Requirements:**

   - **Classical Computing:** Classical computers are built using classical electronic components, such as transistors and memory units.
   - **Quantum Computing:** Quantum computers require specialized hardware using quantum bits (qubits), which are typically implemented using superconducting circuits, trapped ions, or other quantum systems.

In summary, classical and quantum computing differ in their fundamental principles, processing models, error correction mechanisms, and potential speedup for specific problems. Quantum computing has the potential to revolutionize various fields but is still in the early stages of development and faces significant technical challenges. Classical computing remains essential for a wide range of everyday computing tasks.

Properties	**Classical computers**	**Quantum computers**
Basic unit	bits (0 or 1)	qubits (0,1 or both)
Logical Operations	Boolean Logic (AND, OR, NOT)	Quantum Gates (Hadamard, CNOT, etc.)
Superposition	not possible	qubits exist in superposition
Entanglement	not possible	qubits can be entangled
Processing	perform tasks one by one	able to perform multiple calculations simultaneously
Speed	Limited by Moore's Law and physical limitations	Faster and more efficient for certain problems
Error sensitivity	Not as sensitive	Extremely sensitive to noise and errors
Algorithms	Classical algorithms	Quantum algorithms
Applications	Various industries and fields	Cryptography, optimization, chemistry, etc. (discussed later)

Table 1: Comparison between classical and quantum computers

2 Introduction to quantum computing

2.1 Fundamentals of famous quantum algorithms with applications

2.1.1 Shor's Algorithm

Shor's algorithm, developed by mathematician Peter Shor in 1994, is a quantum algorithm that efficiently factors large composite numbers into their prime factors. Factoring large numbers is a computationally challenging problem for classical computers, and Shor's algorithm is one of the most famous quantum algorithms because it demonstrates a significant advantage over classical factoring methods.

Background:

1. Factoring is the process of decomposing a composite number N into its prime factors. For example, factoring the number $N = 15$ would yield the prime factors 3 and 5.

2. Factoring large numbers is a fundamental problem in number theory and is used in cryptography, where the security of many encryption methods relies on the difficulty of factoring large semiprime numbers (the product of two prime numbers).

Steps of Shor's Algorithm:

1. **Classical Preprocessing:** In the classical preprocessing step, the algorithm selects a random number a that is less than N and coprime to N (i.e., the greatest common divisor of a and N is 1). This step requires classical computation.

2. **Quantum Period Finding:** Shor's algorithm leverages quantum computation to efficiently find the period of the function $f(x) = a^x \bmod N$, where x is a positive integer. Quantum circuits are used to perform quantum modular exponentiation, which computes the values of $a^x \bmod N$ for various values of x simultaneously using superposition and quantum parallelism. The quantum part of Shor's algorithm includes the Quantum Fourier Transform (QFT), which plays a crucial role in determining the period of the function. The quantum period-finding step is the core of Shor's algorithm and provides a significant speedup compared to classical methods.

3. **Classical Postprocessing:** After finding the period r, which is a positive integer, classical algorithms (e.g., continued fractions) are used to analyze the period and obtain potential factors of N. If r is even and $a^(r/2)$ is not congruent to -1 modulo N, then the greatest common divisor of $a^(r/2)-1$ and N, and the greatest common divisor of $a^(r/2)+1$ and N are likely to be non-trivial factors of N.

Key Points:

1. Shor's algorithm is probabilistic, meaning that it may require multiple runs to find the correct factors of N with high probability.

2. The time complexity of Shor's algorithm is polynomial in the number of digits in N, making it significantly faster than the best-known classical factoring algorithms, which have exponential complexity.

3. The potential applications of Shor's algorithm are both constructive and potentially disruptive. It has the potential to break widely used cryptographic systems, such as RSA, which rely on the difficulty of factoring large semiprime numbers. As a result, the development of post-quantum cryptography is an active area of research to ensure security in a post-quantum world.

In summary, Shor's algorithm is a groundbreaking quantum algorithm that efficiently factors large composite numbers, posing a significant challenge to classical cryptography and raising the need for quantum-resistant cryptographic techniques.

2.1.2 Grover's Algorithm

Steps of Grover's Algorithm:

1. **Initialization:**

2. Start with a quantum register of n qubits, where n is chosen based on the size of the search space ($N = 2^n$).

3. Initialize the qubits in a superposition of all possible states: $|\psi\rangle = (1/\sqrt{N}) \sum |x\rangle$, where $|x\rangle$ represents possible solutions.

4. **Oracle Query:**

   - Introduce an oracle operator U_f that marks the solution(s) we are looking for. The oracle operator flips the sign of the amplitude of the target state(s).
   - The oracle operator U_f is applied to the quantum state $|\psi\rangle : U_f|\psi\rangle$.

5. **Amplitude Amplification:**

   - Apply a series of quantum operations to amplify the amplitudes of the marked state(s) while reducing the amplitudes of other states. This step involves two main components: **Grover Diffusion Operator (D):** This operator reflects the state vector about the average amplitude. **Amplitude Amplification Operator (A):** This operator amplifies the amplitudes of marked states.

- These operations are applied iteratively, typically around $\sqrt{N}$ times or a specific number of iterations determined by Grover's algorithm.

6. **Measurement:**
   - Measure the quantum state, which collapses it to one of the possible solutions.
   - Repeating the algorithm several times increases the probability of measuring one of the marked states.

Key points:

1. Grover's algorithm is probabilistic, and the number of iterations required to find the solution with high probability depends on the search space size.
2. The algorithm achieves a quadratic speedup over classical search algorithms, making it valuable for applications like database searching and optimization problems.
3. Grover's algorithm has broader implications beyond search problems and is used in quantum algorithms for solving various other problems, such as Boolean satisfiability and cryptographic protocols like Grover's quantum search-based attack on symmetric-key encryption.

In summary, Grover's algorithm is a quantum algorithm that efficiently searches unstructured databases or solves unstructured search problems with a quadratic speedup over classical algorithms. It showcases the potential of quantum computing for solving practical problems faster and has important implications for quantum computing applications.

2.2 Quantum error correction, elements of fault tolerant quantum computing

2.3 Quantum complexity classes

Quantum complexity classes are sets of decision problems or computational tasks that can be efficiently solved by quantum computers. They are analogous to classical complexity classes but take into account the capabilities of quantum computation. Quantum complexity theory helps us understand the computational power of quantum computers and their relationship to classical computers. Here are some important quantum complexity classes:

1. **BQP (Quantum Polynomial Time),** or "bounded-error quantum polynomial time," is the most well-known quantum complexity class. It consists of problems that can be efficiently solved by a quantum computer in polynomial time with a bounded probability of error. BQP includes problems like factoring large integers, simulating quantum systems, and searching unsorted databases faster than classical computers (Shor's algorithm and Grover's algorithm).

2. **BPP (Bounded-Error Probabilistic Polynomial Time):** is the classical counterpart of BQP. BQP is a subset of BPP because quantum computers can simulate classical probabilistic algorithms with at most polynomial overhead. Problems in BPP can be solved by classical probabilistic algorithms in polynomial time with bounded error.

3. **NPQ (Nondeterministic Polynomial Time with Quantum Verification):** is an extension of the classical complexity class NP. It includes decision problems for which a quantum witness can be efficiently checked by a quantum computer in polynomial time. NPQ is considered more powerful than NP because quantum computers can verify solutions more efficiently than classical computers.

4. **QMA (Quantum Merlin-Arthur):** is a quantum version of the classical complexity class MA. It consists of problems for which a quantum computer can verify a quantum solution provided by a prover with bounded error. The verifier can use quantum operations to check the validity of the proof.

5. **QCMA (Quantum Classical Merlin-Arthur):** extends the QMA class by allowing the verifier to be a classical computer. In QCMA, a quantum prover provides a quantum proof to a classical verifier.

6. **PostBQP** also known as **PP (Quantum Polynomial Time)**, includes problems that can be efficiently solved by a quantum computer with access to a quantum version of the complexity class PP. PP is a class of problems related to counting solutions to decision problems, and PostBQP is its quantum counterpart.

These quantum complexity classes provide a framework for understanding the computational power of quantum computers and their relationships to classical complexity classes. They help categorize problems based on their quantum computational complexity, shedding light on the potential advantages and limitations of quantum computation for various tasks.

3 Is everything so great?

Advantages of Quantum Computers/Computing:

Speed: Quantum computers can perform certain calculations much faster than classical computers, making them well-suited for tasks that involve a large amount of data or complex mathematical calculations.

Parallelism: Quantum computing allows for the parallel processing of information, which means that multiple computations can be performed simultaneously. This can significantly speed up certain tasks, such as searching large

databases.

Cryptography: Quantum computing has the potential to break many of the encryption methods currently used to secure data. However, it also has the potential to develop new and more secure encryption methods, which could be more resistant to attacks by hackers.

Chemistry: Quantum computing can simulate the behavior of molecules at a level of detail that is not possible with classical computing. This could lead to new discoveries in drug design, materials science, and other areas.

Disadvantages of Quantum Computers/Computing:

Quantum computers have the potential to revolutionize the field of computing, but they also come with a number of disadvantages. Some of the main challenges and limitations of quantum computing include:

Noise and decoherence: One of the biggest challenges in building a quantum computer is the problem of noise and decoherence. Quantum systems are extremely sensitive to their environment, and any noise or disturbance can cause errors in the computation. This makes it difficult to maintain the delicate quantum state of the qubits and to perform accurate and reliable computations.

Scalability: Another major challenge is scalability. Building a large-scale quantum computer with a large number of qubits is extremely difficult, as it requires the precise control of a large number of quantum systems. Currently, the number of qubits that can be controlled and manipulated in a laboratory setting is still quite small, which limits the potential of quantum computing.

Error correction: Error correction is another major challenge in quantum computing. In classical computing, errors can be corrected using error-correcting codes, but in quantum computing, the errors are much more difficult to detect and correct, due to the nature of quantum systems. Lack of robust quantum algorithms: Even though some quantum algorithms have been developed, their number is still limited, and many problems that can be solved using classical computers have no known quantum algorithm.

High cost: Building and maintaining a quantum computer is extremely expensive, due to the need for specialized equipment and highly trained personnel. The cost of building a large-scale quantum computer is also likely to be quite high, which could limit the availability of quantum computing to certain groups or organizations.

Power consumption: Quantum computers are extremely power-hungry,

due to the need to maintain the delicate quantum state of the qubits. This makes it difficult to scale up quantum computing to larger systems, as the power requirements become prohibitively high.

4 Existing quantum machines

There are several multinational companies that have built and are currently working on building quantum computers. Some examples include:

- **IBM:** IBM has been working on quantum computing for several decades, and has built several generations of quantum computers. The company has made significant progress in the field, and its IBM Q quantum Experience platform allows anyone with an internet connection to access and runs experiments on its quantum computers. IBM's most recent quantum computer, the IBM Q System One, is a 20-qubit machine that is designed for commercial use.
- **Google:** Google has been working on quantum computing for several years and has built several generations of quantum computers, including the 72-qubit Bristlecone quantum computer. The company claims that its quantum computer has reached "quantum supremacy," meaning it can perform certain calculations faster than any classical computer.
- **Alibaba:** Alibaba has been investing heavily in quantum computing, and in 2017 it announced that it had built a quantum computer with 11 qubits. The company has also been developing its own quantum chips and is planning to release a cloud-based quantum computing service in the near future.
- **Rigetti Computing:** Rigetti Computing is a startup company that is building and developing superconducting qubits-based quantum computers. They offer a cloud-based quantum computing platform for researchers and developers to access their quantum computers.
- **Intel:** Intel has been developing its own quantum computing technology and has been building quantum processors and cryogenic control chips, which are used to control the quantum bits. In 2019, they announced the development of a 49-qubit quantum processor, one of the largest processors of its kind developed so far.
- **D-Wave Systems:** D-Wave Systems is a Canadian quantum computing company, founded in 1999, which is known for its development of the D-Wave One, the first commercially available quantum computer. D-Wave's quantum computers are based on a technology called quantum annealing, which is a type of quantum optimization algorithm. They claim to have built the first commercially available quantum computer, but their system is not a fully general-purpose computer and it's mainly used for optimization problems.

- **Xanadu:** Xanadu is a Canadian startup company that is building a new type of quantum computer based on a technology called photonic quantum computing. Photonic quantum computing is based on the manipulation of light particles (photons) to perform quantum computations. Xanadu's approach is different from other companies that are building quantum computers, as it uses light instead of superconducting qubits. They are focusing on developing a general-purpose quantum computer that can run multiple algorithms.