Introduction to Quantum Computing

Week 6: Magic of Quantum Computing

Taha Selim

t.i.m.m.selim2@hva.nl

Quantum Talent and Learning Center
Amsterdam University of Applied Sciences

Week 6
April 26th, 2024

Agenda for today:

- Recap of previous sessions.
- Bloch Sphere representation.
- How to do computations with quantum circuits?
- Quantum computing programming with Qiskit.

Every 30 min or so we will have a feed from other QTLC groups.
Feel free to ask questions at any time!
It is an interactive workshop, we all learn from each other!

Workshop facilities

- Join Discord and display your name instead of the nickname or the username.
- Check the invite email for the Discord link.

JupyterHub facilities

- JupyterHub is up and running; please check your accounts.
- If you have any issues, please let us know.
- We should test it between 1:30 until 14:30.

Server's link:

`https://jupyter.snellius.surf.nl/jhshr001/`

Session recording

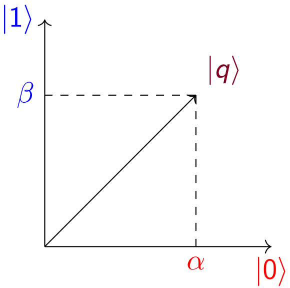
- The session is being recorded.

Representing a qubit state

A qubit is a quantum bit. It is similar to a classical bit, but it can be in a superposition of states.

In analogy to vectors, we can write the wavefunction as a ket $|q\rangle$:

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle$$



In a vector notation:

$$|q\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix},$$ where α and β are complex coefficients,

and

$$\alpha|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \alpha|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$ normalization: $|\alpha|^2 + |\beta|^2 = 1$.

Probability of finding an outcome during measurements

Amplitudes give the probability of finding the system in a given state when performing a measurement.

The probability of finding the system in state $|0\rangle$ is $|\alpha|^2$, and the probability of finding the system in state $|1\rangle$ is $|\beta|^2$.

The sum of the probabilities of finding the system in the two states must be equal to 1.

Hence,

$$|\alpha|^2 + |\beta|^2 = 1$$

The particle exists by itself in a superposition of states.

Quantum computing vs. classical computing

In classical computing, we use bits to represent information.
In quantum computing, we use qubits to represent information.

Classical Computing	Quantum Computing
Bits	Qubits
0 or 1	0, 1, or superposition of 0 and 1
AND, OR, NOT gates	Quantum gates
classical circuits	quantum circuits

Similar to manipulating bits in classical computing using classical gates like *AND* and *NOR* gates, we manipulate the qubit states using quantum gates.

Quantum computing vs. classical computing

Let's learn about the action of classical gates on data bits. Data in classical digital computers are in a form of bits, each bit can be in a state of 0 or 1.

Basically anything that can be represented as a sequence of bits.

0	1	1	0	1	0	1	0	1	0	1	0	1

Example: ascii text to binary converter (to be checked)

Word	Binary
Hello	01001000 01100101 01101100 01101100 01101111
World	01010111 01101111 01110010 01101100 01100100

Classical gates operate on those bits.

Quantum computing vs. classical computing

Examples of typical classical gates:

One-bit NOT gate

It flips the bit value.

bit	Out
0	1
1	0

$x \to \bar{x}$

Two-bit AND gate

It operates on two bits.

bit 1	bit 2	Out
0	0	0
0	1	0
1	0	0
1	1	1

$x \wedge y$

$\sim$ bit 1 * bit 2

Two-bit OR gate

It operates on two bits.

bit 1	bit 2	Out
0	0	0
0	1	1
1	0	1
1	1	1

$x \vee y$

Two-bit XOR gate

It operates on two bits.

bit 1	bit 2	Out
0	0	0
0	1	1
1	0	1
1	1	0

$x \oplus y$

$\sim (bit\ 1 + bit\ 2)(mod\ 2)$

All together, any logical operation can be formed out of those gates.

Quantum computing vs. classical computing

Why can't we use classical gates in quantum computing?

One-bit `NOT` **gate**, **two-bit** `AND` **gate**, and **two-bit** `OR` **gate** form the basis of classical computing. They form a universal set of gates; any logical operation could be composed out of these gates.

Quantum computing is fundamentally different from classical computing: A quantum computer uses quantum gates to manipulate qubits.

Quantum gates	Classical gates
$\sim$ Reversible operations on qubits.	$\sim$ Irreversible operations on bits
Condition: unitary operation.	Condition: Different inputs and outputs.
Unitary operation: $U^\dagger U = I$	`NOT` gate is reversable.
Inverse operation: $U^{-1} = U^\dagger$	`NOT^2` $= 1$
More one-qubit quantum gates are available:	`AND`, `OR`, `XOR` gates are not reversable.
`X, Y, Z, H, S, T, ...`	Only One-bit `NOT` gate.

Quantum gates can create superposition states, and entangled states, which are not possible with classical gates.

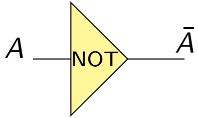
Example: Classical vs. quantum gates

Classical gates are used to manipulate bits.
Quantum gates are used to manipulate qubits.

Classical NOT gate versus quantum NOT (Pauli-X) gate

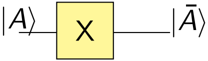
Classical NOT gate	Quantum NOT gate
$0 \rightarrow 1$	$\lvert 0\rangle \rightarrow \lvert 1\rangle$
$1 \rightarrow 0$	$\lvert 1\rangle \rightarrow \lvert 0\rangle$

Classical NOT Gate



A	$\bar{A}$
0	1
1	0

Quantum NOT Gate



$\lvert A\rangle$	$\lvert \bar{A}\rangle$
0	1
1	0

What is a quantum gate?

- Quantum gates are the building blocks of quantum circuits.
- They are used to manipulate qubits.
- Quantum gates are reversible operations.
- They are represented by unitary matrices.
- The action of a quantum gate on a qubit is represented by matrix-vector multiplication.
- The matrix representation of a quantum gate is called the unitary matrix.

Quantum circuits

A quantum circuit is typically composed of:

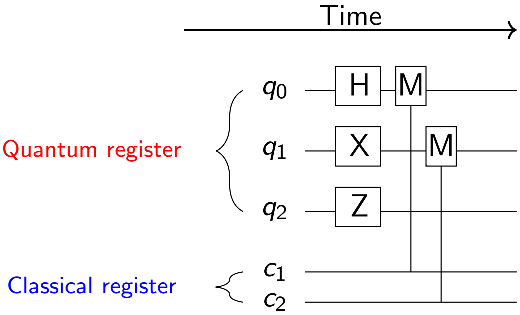
1. quantum register: ollection/stack of qubits.
2. Classical register: collection of classical bits,
3. Quantum gates: operations that act on qubits.
4. Quantum gates are arranged in a particular order/sequence to perform a specific task, a certain computation.

Quantum register: $q_0, q_1, q_2, \cdots$

Classical register: $c_0, c_1, c_2, \cdots$

Quantum gates: $H, X, Z, \cdots$

Measurement gates: M



Examples of quantum gates: `Hadamard` gate

A `Hadamard` gate is a one-qubit gate that puts a qubit in a **superposition** state.

The `Hadamard` gate is represented by the matrix:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

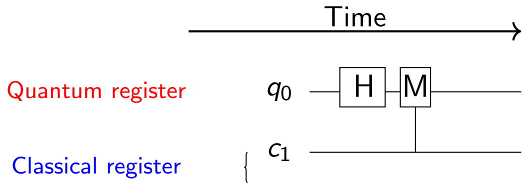
The qubit is initially in the state $|0\rangle$:

$$|\psi\rangle = |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

The action of the Hadamard gate on the qubit:

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \end{aligned}$$

Measurements: $|1/\sqrt(2)|^2 = 0.5$.



References

- Hughes, R. J., Nordholt, J. E., Mink, A., & Lanzagorta, M. (2021). Quantum Computing for the Quantum Curious. Springer.
- Nielsen, M. A., & Chuang, I. L. (2010). Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press.
- IBM Quantum Experience: `https://quantum-computing.ibm.com/`

Thank you!