Quantum Gates

Miguel Sozinho Ramalho

November, 2018

Table of contents

1. Introduction
2. Conjugate Transpose
3. Hermitian Matrix
4. Unitary Operators
5. Quantum Gates
    - Hadamard
    - X Gate
    - Y Gate
    - Z Gate
    - Controlled gates
    - More Gates
6. Exercises
7. Where to learn more?

Introduction

This week is focused on recalling some linear algebra notions that are very useful for designing quantum circuits. We will briefly recall the concept of Hermitian and Unitary operators.

Afterwards, we will go through some important one-qubit gates and their QISKit representation, moving on to 2-qubit gates.

Lastly, we will have exercises to practice these topics and make them feel more natural, as they are the backbone of what's to come!

Conjugate Transpose $A^\dagger$

The Conjugate Transpose of a complex matrix A (denoted $A^\dagger$) is the transpose of A (A^T) after we take the complex conjugate of each element of A. Here's an example:

$$\begin{bmatrix} 1+i & -1 \\ 1 & -1-i \end{bmatrix}^\dagger = \begin{bmatrix} 1-i & 1 \\ -1 & -1+i \end{bmatrix}$$

Recall that the complex conjugate of $a+bi$ is simply $a-bi$ (geometrically, it is the reflection of the complex number over the real axis in the complex plane).

Hermitian Matrix $A = A^\dagger$

Hermitian refers to complex matrices ($H \in \mathbb{C}^2$) that equal their complex conjugate: $A = A^\dagger$.

These are matrices whose eigenvalues are always real numbers ($\lambda \in \mathbb{R}$).

Try to verify the following example:

$$A = \begin{bmatrix} 1 & 2+i & 3 \\ 2-i & 5 & i \\ 3 & -i & 1 \end{bmatrix}$$

$$A^\dagger = A \Rightarrow \text{ A is Hermitian}$$

Unitary Operators

A Linear Transformation (T) allows the conversion from a vector space (V) to another (W) $T : W \to V$ such that:

- $T(v1 + v2) = T(v1) + T(v2) : \ v1, v2 \in V$
- $T(\gamma \times v) = \gamma \times T(v)$ for any scalar γ

A Unitary Operator is a Linear Operator (an operator that performs a linear transformation), with the additional constraint that it is unitary: $AA^\dagger = A^\dagger A = I$.
These means we can apply a unitary operator to a vector space twice and get the original vector space. We care about them because this is exactly how quantum gates operate!

Quantum Gates

A quantum gate is an analogy of classic gates and is also a useful paradigm to think about qubit manipulation.
Also known as quantum logic gates, they have the following properties:

- Reversible (applied twice result in the original state) unlike some classical gates (for example the OR gate)
- Represented by Unitary matrices
- Each gate is only applied to the same number of qubits (one-qubit gates, two-qubit gates, ...)

We will see some of the more common gates and how to use them in QISKit, but more will come into play as your quantum knowledge increases.

Hadamard

The Hadamard gate is applied to a single qubit. If your qubit is in a state $|0\rangle$, after the Hadamard, it will be in a superposition of $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. If it is in $|1\rangle$ it will end up in $\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. It is essential to create a **superposition** from $|0\rangle$. Try to verify this using the vector version of the qubit and the following unitary operator:

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

Also, try to check that $HH^\dagger = H^\dagger H = I$.

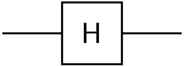
Hadamard in QISKit

After you have created your QuantumCircuit `circuit`

```
# Create a Quantum Register with 2 qubits
qr = QuantumRegister(2) # 1 would suffice

# Create a Quantum Circuit from the qr and cr registers
circuit = QuantumCircuit(qr, cr)

# H gate on qubit 0
circuit.h(qr[0])
```



You will have the chance to test it on this week's exercises.

X Gate

The X gate (also known as Pauli-X) is applied to a single qubit. It can be seen as the equivalent of the NOT gate in classical circuitry, with respect to the zero-one basis. It maps $|0\rangle$ to $|1\rangle$ and $|1\rangle$ to $|0\rangle$. Try to verify this using the vector version of the qubit and the following unitary operator:

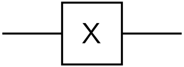
$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Also, try to check that $XX^\dagger = X^\dagger X = I$.

X Gate in QISKit

After you have created your QuantumCircuit `circuit`

```
# X gate on qubit 1
circuit.x(qr[1])
```



You will have the chance to test it on this week's exercises.

Y Gate

The Y gate (also known as Pauli-Y) is applied to a single qubit. It is usually called the phase-flip because it maps $|0\rangle$ to $i\,|1\rangle$ and $|1\rangle$ to $-i\,|0\rangle$. Try to verify this using the vector version of the qubit and the following unitary operator:

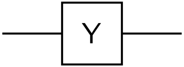
$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

Also, try to check that $YY^\dagger = Y^\dagger Y = I$.

Y Gate in QISKit

After you have created your QuantumCircuit `circuit`

```
# Y gate on qubit 1
circuit.y(qr[1])
```



You will have the chance to test it on this week's exercises.

Z Gate

The Z gate (also known as Pauli-Z or phase-flip) is applied to a single qubit. It is usually called the phase-flip because it leaves the state $|0\rangle$ unchanged and maps $|1\rangle$ to $-|1\rangle$. Try to verify this using the vector version of the qubit and the following unitary operator:

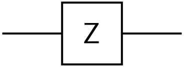
$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Also, try to check that $ZZ^\dagger = Z^\dagger Z = I$.

Z Gate in QISKit

After you have created your QuantumCircuit `circuit`

```
# Z gate on qubit 1
circuit.z(qr[1])
```



You will have the chance to test it on this week's exercises.

Curiosity*: The squared matrices of Pauli gates (X, Y, Z) is the identity itself: $X^2 = Y^2 = Z^2 = I$

Controlled gates

Controlled gates are an extension of lower order gates. Usually, they perform the same operation, but use one or more extra qubits that are used to control/ascertain whether the original operation should be applied.
For instance, controlled gates for 2 qubits add a **control** qubit and, if it is $|1\rangle$, the original gate is applied to the other qubit; if the control qubit is $|0\rangle$ the other qubit remains the same (typically).

$$CNOT = \begin{cases} \text{control qubit is } |0\rangle \Rightarrow \text{leave source qubit} \\ \text{control qubit is } |1\rangle \Rightarrow \text{apply gate to source qubit} \end{cases}$$

Given two qubits, it flips the second iff the first is $|1\rangle$.

Controlled Not (CNOT)

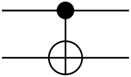
One example of this is the CNOT gate, one of the most important 2-qubit gates. It is a controlled gate derived from the X gate, so it is also know as cx gate.Its unitary operator is:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

CNOT Gate in QISKit

After you have created your QuantumCircuit `circuit` with at least 2 qubits.

```
# CNOT gate on qubits 0 and 1
# qubit 0 is the control qubit
# qubit 1 is the target qubit
circuit.cx(qr[0], qr[1])
```



You will have the chance to test it on this week's exercises.

More Gates

As you dive deeper into the Quantum World, more gates will present themselves, you may even create your own (just remember their properties and do check if no one has come up with it yet).
To spike your interest, here are a few more names you will eventually come across for quantum gates:

- SWAP
- Phase shift family
- CCNOT (AKA Toffoli)
- CSWAP (**C** usually stands for ‘Controlled’)
- XX (not the band, no)
- Deutsch (not hardware implementation yet)

Exercises

Hands-on

This week you will start writing your first quantum gates, run simple circuits on a simulator, plot the readings. Do all of that in a real quantum device (and learn about real world concerns you should take).
When you feel ready, move on to the exercises and, if you feel stuck, you may look at the solutions.

Where to learn more?

- Khan Academy on Linear Algebra
- Medium article on common Quantum Gates
- Quantiki entry on Quantum Gates