Quantum Protocols and Quantum Algorithms - Notes

Grant Talbert

December 7th, 2023

Contents

1 Quantum Circuits

Prerequisite knowledge for this section:
The Tensor power for some matrix M is denoted as follows.

$$M^{\otimes n} = \underbrace{M \otimes M \otimes \cdots \otimes M}_{\text{n times}}$$

The Exponential function for some value x is denoted as follows.

$$\exp(x) = e^x \forall x \in \mathbb{C}$$

1.1 Introduction

So far, we've seen various single- and multi-qubit gates. We've also explored how to use these gates in concert with other components to build quantum circuits.
Before implementing quantum algorithms on real quantum computers, it's important to highlight the definition of a quantum circuit concretely, as we will be building quantum circuits to implement these algorithms.

1.2 What is a Quantum Circuit?

A quantum circuit is a computation routine consisting of *coherent quantum operations on quantum data, such as qubits, and concurrent real-time classical computation.* It's an ordered sequence of *quantum gates, measurements and resets*, all of which may be conditioned on and use real-time data from the real-time classical computation.
A set of quantum gates is said to be universal if any unitary transformation of theq uantum data can be efficiently approximated arbitrarily well as a sequence of gates in the set. Any quantum program can be represented by a sequence of quantum circuits and non-concurrent classical computation.

1.3 Example: Quantum Teleportation

Consider the following circuit. In later lessons, it will be shown that it implements the quantum teleportation algorithm.

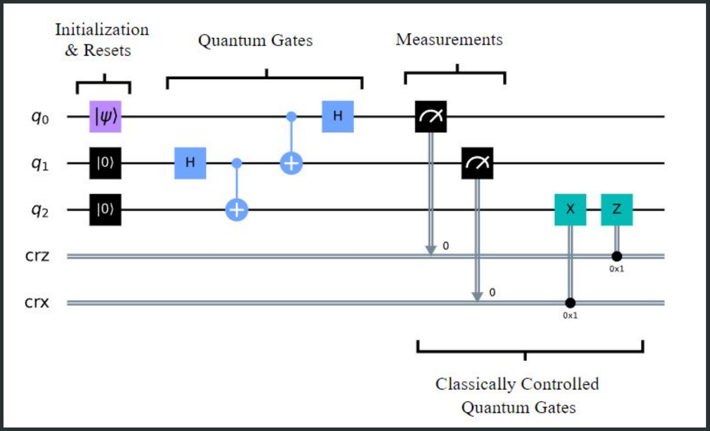

Figure 1: Quantum Teleportation Algorithm

The quantum circuit uses three qubits and two classical bits. There are four main components in this circuit.

1.3.1 Initialization and Reset

We need to initialize our computation with a well-defined quantum state. This is achieved using the initialization and reset operations. The resets can be performed by a combination of single-qubit gates and concurrent real-time classical computation that monitors whether we have successfully created the desired state through measurements. The initialization of q_0 into a desired state $|\psi\rangle$ can then follow by applying single-qubit gates.

1.3.2 Quantum Gates

Next, we apply a sequence of quantum gates that manipulate the three qubits as required by the teleportation algorithm. In this case, we only need to apply single-qubit Hadamard (H) and two-qubit Controlled-X ($\oplus$) gates.

1.3.3 Measurements

Next, we measure two of the three qubits. A classical computer interprets the measurements of each qubit as classical outcomes (0 and 1) and stores them in the two classical bits.

1.3.4 Classically Conditioned Quantum Gates

Fourth, we apply single-qubit Z and X quantum gates on the third qubit. These gates are conditioned on the results of the measurements that are stored in the two classical bits. In this case, we are using the results of the classical computation concurrently in real-time within the same quantum circuit. These are basically CX and CZ gates, but controlled by classical bits rather than qubits.

1.4 Example: Variational Quantum Eigensolvers

Consider the following quantum algorithm, which will later be shown to implement the variational quantum eigensolver. A classical computer herein works *non-concurrently* in concert with a quantum computer.

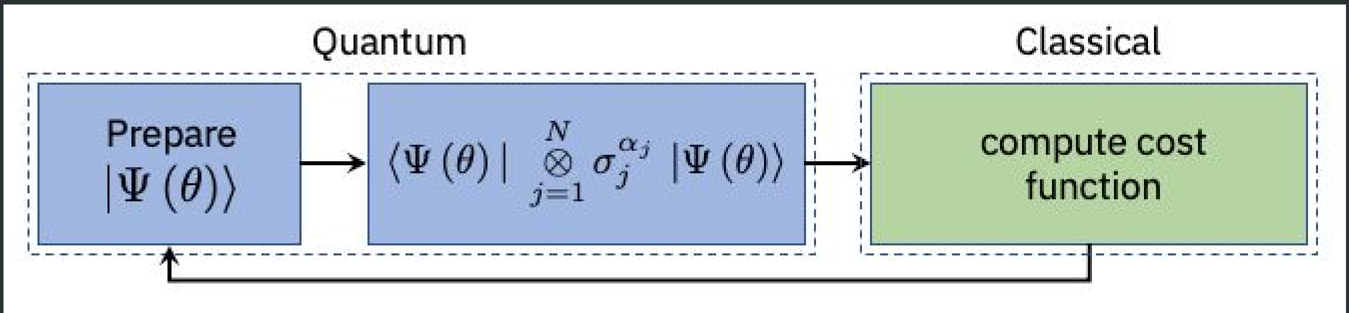
Quantum
Classical
Prepare $|\Psi(\theta)\rangle$
$\langle\Psi(\theta)|\bigotimes_{j=1}^{N}\sigma_j^{\alpha_j}|\Psi(\theta)\rangle$
compute cost function

Figure 2: Variational Quantum Eigensolver Algorithm

1.4.1 The Quantum Block

As with the quantum teleportation example above, a quantum state $|\Psi(\theta)\rangle$ is prepared by a combination of resets with single- and multi-qubit quantum gates. Here, the parameters of the state are parameterized with the quantity θ. Once prepared, the quantum state is then manipulated using quantum gates and measured. All of the operations within the quantum block consist of quantum circuits.

1.4.2 The Classical Block

Once a quantum state has been measured, a classical computer interprets those measurement outcomes and computes their cost using a cost function that has been chosen for the intended application. Based on this cost, the classical computer determines another value for the parameter θ.

1.4.3 Combined Operation

Once the classical computer determines the next parameter for θ, a sequence of resets, single- and multi-qubit quantum gates are used in a quantum circuit to prepare $|\Psi(\theta)\rangle$, and this process continues until the cost of the measured quantum state stabilizes, or until another pre-determined outcome is met.

1.5 Why the Classical Parts?

While a universal quantum computer can do anything a classical computer can, we often add classical parts to our quantum circuits because quantum states are fragile.

When we measure the qubit, we collapse its state and destroy a lot of the information. Since all measurement does is destroy information, we can in theory always measure last and lose no computational advantage. In reality, measuring early offers many practical advantages.

For example, in the teleportation circuit, we measure the qubits so we can send the information over classical channels instead of quantum channels. The advantage is that classical channels are very stable, while we don't really have a way of sending quantum information to other people since the channels are so difficult to create.

In the variational quantum eigensolver example, splitting the computation up into smaller quantum computations actually loses us some computational advantage, but makes up for this on noisy hardware by reducing the time our qubits are in superposition. This means there is less chance interference will introduce inaccuracies in our results.

Finally, to use the results of our quantum computation in our classical, everyday world, we need to measure and interpret these states at the end of our computation.

2 Deutsch-Jozsa Algorithm

2.1 Introduction

In this section, we consider the Deutsch-Jozsa problem, and both the classical and quantum solutions. We then implement the quantum solution with Qiskit.

2.1.1 Deutsch-Jozsa Problem

We are given a hidden *boolean* function f, which takes as input a string of bits, and returns either 0 or 1, that is:

$$f(\{x_0, x_1, x_2, \ldots\}) \to 0 \vee 1, \text{ where } x_n \in \{0, 1\}$$

The property of the given boolean function is that it's guaranteed to be either balanced or constant - that is, it will either return 0's for exactly half of all inputs and 1's for the others (balanced), or it will return either all 0s or all 1s (constant). The task is to determine whether the function is balanced or constant.

2.1.2 The Classical Solution

In the best case, two queries to the oracle can determine if the hidden Boolean function $f(x)$ is balanced. If we get both $f(0, 0, 0, \ldots) \to 0$ and $f(1, 0, 0 \ldots) \to 1$, then we know the function is balanced as we have obtained the two different outputs.

In the worst case, we may continue to see the same output for each input, meaning we will have to check exactly half of all possible inputs plus one in order to be certain $f(x)$ is constant, equal to $2^{n-1} + 1$ trials in the worst case. Thus, for a 4-bit string (which has 16 different possible states over all bits), we would need to check 9 different combinations in the worst case, as it's possible even if we check 8, that the other 8 all output the opposite value. This is, however, highly unlikely, and if we get the same result continually in succession, we give the probability that the function is constant as a function of k inputs as:

$$P_{\text{constant}}(k) = 1 - \frac{1}{2^{k-1}} \quad \text{for } 1 < k \leq 2^{n-1}$$

We can, thus, truncate the classical algorithm early if we need only be certain to an arbitrary accuracy. However for 100% confidence, we need to check a maximum of $2^{n-1} + 1$ inputs.

2.1.3 Quantum Solution

Using quantum computation, we can solve this problem to 100% accuracy after only one call to $f(x)$, provided we implement f as a quantum oracle mapping the state $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$, where $\oplus$ is addition modulo 2. Below is the generic circuit to implement the Deutsch-Josza algorithm. Let's go through the steps to understand this properly.

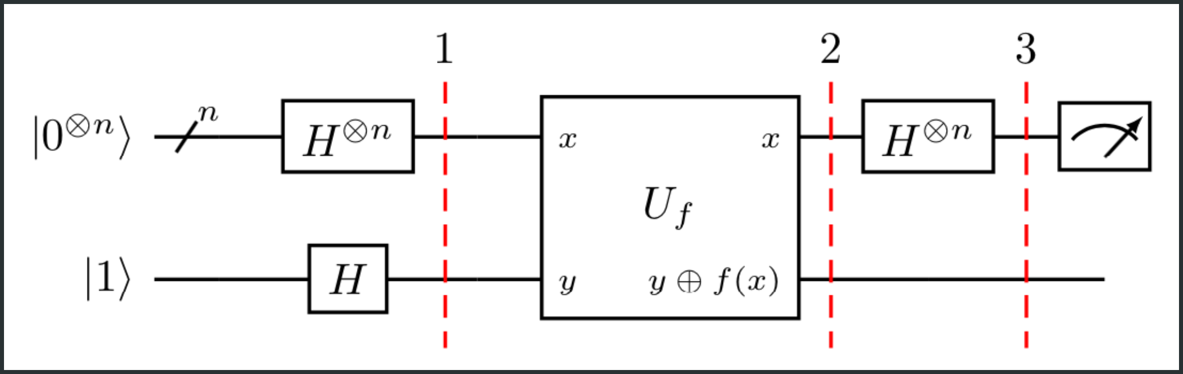

Figure 3: Deutsch-Josza Algorithm

1. Prepare two quantum registers. We initialize an n-qubit register to $|0\rangle$, and another single-qubit register to $|1\rangle$.
$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle$$

2. Apply a Hadamard to each qubit.
$$|\psi_1\rangle = \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle - |1\rangle)$$
We apply a Hadamard to n gates in state $|0\rangle$, giving $|+\rangle^{\otimes n}$, which gives us our $\frac{1}{\sqrt{2^n}}$. We also have a singular $H|1\rangle = |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, multiplying one more scalar term in. This also constructs a column vector of size 2^{n+1}, so a linear combination of 2^n column vectors with 2 entries is sufficient to construct this vector. We let x start at 0, so we give $2^n - 1$ rather than 2^n. The $|x\rangle$ term takes the values of all the basis states (for a column vector with 4 entries, $|x\rangle \in \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$). The tensor product is then taken with the $|0\rangle - |1\rangle$ term to produce a column vector with 2 distinct entries.

3. Apply the quantum oracle $|x\rangle \otimes |y\rangle$ to $|x\rangle \otimes |y \oplus f(x)\rangle$.
$$\begin{aligned}|\psi_2\rangle =& \frac{1}{\sqrt{2^{2n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle)\\ =& \frac{1}{\sqrt{2^{2n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|f(x)\rangle - |1 \oplus f(x)\rangle)\\ =& \frac{1}{\sqrt{2^{2n+1}}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)\end{aligned}$$
since for each x, $f(x) \in \{0, 1\}$. Thus, if $x \mapsto 1$, we would have $|1\rangle - |1 \oplus 1\rangle = |1\rangle - |0\rangle$, which is equal to $-(|0\rangle - |1\rangle) = (-1)^{f(x)}(|0\rangle - |1\rangle)$.

4. The second single qubit register can now be ignored. We apply a Hadamard to each qubit in the first register.
$$\begin{aligned}|\psi_3\rangle =& \frac{1}{2^n} = \sum_{x=0}^{2^n-1}(-1)^{f(x)}\left[\sum_{y=0}^{2^n-1}(-1)^{x\cdot y}|y\rangle\right]\\ =& \frac{1}{2^n}\sum_{y=0}^{2^n-1}\left[\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{x\cdot y}\right]|y\rangle\end{aligned}$$
where $x \cdot y = x_0y_0 \oplus x_1y_1 \oplus \ldots \oplus x_{n-1}y_{n-1}$ is the sum of the bitwise product.

5. Measure the first register. The probability of measuring $|0\rangle^{\otimes n}$ is
$$|0\rangle^{\otimes n} = \left|\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\right|^2$$
For a balanced $f(x)$, you have the sum of an equal amount of 1s and -1s, giving 0. For a constant $f(x)$, you have either $|\frac{2^n}{2^n}|^2$ or $|-\frac{2^n}{2^n}|^2$, and both equal to 1.

2.1.4 Why Does This Work?

2.2 Worked Example

This is extremely confusing, so let's consider a worked example. **Note for the future: I got confused after the quantum oracle was introduced, work so far is on the board so you should probs finish that today**
Consider an example for a two bit balanced function. That is, $f(x_0, x_1) = x_0 \oplus x_1$. Thus,
$f(0,0) = 0$
$f(0,1) = 1$
$f(1,0) = 1$
$f(1,1) = 0$
There is a corresponding phase oracle to this two bit oracle, $U_f|x_1, x_0\rangle := (-1)^{f(x_1,x_0)}|x\rangle$. Let's consider this oracle on the state

$$|\psi_0\rangle = |00\rangle_{01} \otimes |1\rangle_2$$

The subscripts here denote the qubit that each state corresponds to. We have the first register of two qubits initialized to $|00\rangle$ and the second to $|1\rangle$.
We apply the Hadamard to all qubits.

$$H^{\otimes 3}|\psi_0\rangle := |\psi_1\rangle = \frac{1}{2}|{+}{+}\rangle_{01} \otimes \frac{1}{\sqrt{2}}|-\rangle_2$$

$$= \frac{1}{2\sqrt{2}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)_{01} \otimes (|0\rangle - |1\rangle)_2$$

The oracle function can be implemented as $Q_f = CX_{02}CX_{12}$.
We distribute over the tensor product before applying the oracle to make this easier to understand.:

$$|\psi_1\rangle = \frac{1}{2\sqrt{2}}((|00\rangle_{01} \otimes (|0\rangle - |1\rangle)_2) + (|01\rangle_{01} \otimes (|0\rangle - |1\rangle)_2) + (|10\rangle_{01} \otimes (|0\rangle - |1\rangle)_2) + (|11\rangle_{01} \otimes (|0\rangle - |1\rangle)_2))$$

We then apply the CNOT operation, analogous to an XOR operation, analogous to addition modulo 2. We take our 0th qubit as our first control, and the 1st qubit as the second control, so we map each term $|q_0, q_1\rangle \otimes |q_2\rangle \mapsto |q_0, q_1\rangle \otimes |q_2 \oplus q_0 \oplus q_1\rangle$.

$$\begin{aligned} Q_f|\psi_1\rangle := |\psi_2\rangle = &\frac{1}{2\sqrt{2}}[|00\rangle_{01} \otimes (|0 \oplus 0 \oplus 0\rangle - |1 \oplus 0 \oplus 0\rangle)_2 + |01\rangle_{01} \otimes (|0 \oplus 0 \oplus 1\rangle - |1 \oplus 0 \oplus 1\rangle)_2 \\ &+ |10\rangle_{01} \otimes (|0 \oplus 1 \oplus 0\rangle - |1 \oplus 1 \oplus 0\rangle)_2 + |11\rangle_{01} \otimes (|0 \oplus 1 \oplus 1\rangle - |1 \oplus 1 \oplus 1\rangle)_2] \end{aligned}$$

We simplify and obtain the following:

$$\begin{aligned} |\psi_2\rangle = &\frac{1}{2\sqrt{2}}[|00\rangle_{01} \otimes (|0\rangle - |1\rangle)_2 + |01\rangle_{01} \otimes (|1\rangle - |0\rangle)_2 + |10\rangle_{01} \otimes (|1\rangle - |0\rangle)_2 + |11\rangle_{01} \otimes (|0\rangle - |1\rangle)_2] \\ = &\frac{1}{2\sqrt{2}}[|00\rangle_{01} \otimes (|0\rangle - |1\rangle)_2 - |01\rangle_{01} \otimes (|0\rangle - |1\rangle)_2 - |10\rangle_{01} \otimes (|0\rangle - |1\rangle)_2 + |11\rangle_{01} \otimes (|0\rangle - |1\rangle)_2] \\ = &\frac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle)_{01} \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)_2 \\ = &\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)_0 \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)_1 \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)_2 \\ = &|{-}{-}{-}\rangle_{012} \end{aligned}$$

Now we reapply Hadamards on the first register. In the above definition we drop the second register, however it will be kept for this example, hadamards will just not be applied to it.

$$|\psi_3\rangle := H|-\rangle \otimes H|-\rangle \otimes |-\rangle$$

$$= |1\rangle \otimes |1\rangle \otimes |-\rangle$$

Measuring the first two qubits gives the measurement 11, and as we saw earlier the probability for measuring the 0 state in a balanced function was theorized to be 0%. Thus, we have shown this algorithm applies to our state.

3 Bernstein-Vazirani Algorithm

4 Simon's Algorithm

5 Quantum Fourier Transform

6 Quantum Phase Estimation

7 Shor's Algorithm

8 Grover's Algorithm

9 Quantum Counting

10 Quantum Walk Search Algorithm

11 Quantum Teleportation

12 Superdense Coding

13 Hidden Shift Problem

14 Quantum Key Distribution