# Quantum Computation 101 for Physicists

Class exercise 4

## 1 Question 1 (Deutsch-Jozsa problem)

Quick reminder of the Deutch algorithm we saw in class (2.2 in Mermin's book): we have a function $f : \{0,1\} \to \{0,1\}$ and we want to determine whether $f(0) = f(1)$. In class we saw that if we have a unitary $U_f|x,y\rangle = |x, y \oplus f(x)\rangle$, we can solve the problem using one application of $U_f$:

$$U_f(H_0 \otimes H_1)(X_0 \otimes X_1)|00\rangle = U_f(|0\rangle - |1\rangle)(|0\rangle - |1\rangle) = |0\rangle|f(0)\rangle - |0\rangle|\tilde{f}(0)\rangle - |1\rangle|f(1)\rangle + |1\rangle|\tilde{f}(1)\rangle =$$
$$\begin{cases} (|0\rangle - |1\rangle)(|f(0)\rangle - |\tilde{f(0)}\rangle) & f(0) = f(1), \\ (|0\rangle + |1\rangle)(|f(0)\rangle - |\tilde{f(0)}\rangle) & f(0) = \tilde{f}(1). \end{cases}$$

We can now measure the first qubit in the basis $(|+\rangle, |-\rangle)$ and get the answer.

The Deutsch-Jozsa algorithm is an extension to the Deutch algorithm. The problem goes like this:

For an integer $n$ we are given a function $f : \{0,1\}^n \to \{0,1\}$ ($\{0,1\}^n$ is a common notation for a bit string of length $n$) such that either all $f(i)$ have the same value ("constant"), or $2^n/2$ of the values are 0 and $2^n/2$ are 1 ("balanced"). The goal is to find out whether $f$ is constant or balanced.

1. What is the (worst case) number of times we will have to query the function classicaly in order to determined whether $f$ is constant or balanced?

2. Use the method of Deutch to solve the problem quantumly using $O(n)$ number of qubits and $O(n)$ number of quantum gates.

### 1.1 Solution

1. Classically, we can only query the function until we get two different values, in which case we know it is balanced, or until we have queried more than half the possibile arguments of the function ($2^n/2$) and got the same result, in this case we know $f$ is constant. So we see that if $f$ is constant, we need an exponential number of queries to $f$, specifically $2^n/2 + 1$.

2. Assume we have a unitary $U_f$ such that $U_f|x\rangle_n|y\rangle_1 = |x\rangle_n|y \oplus f(x)\rangle$. We now start from $|0\rangle^{n+1}$, and apply $X$ to the last qubit to get $|0\rangle^n|1\rangle$, and apply $H^{\otimes(n+1)}$ to our state to get $\frac{1}{2^{n/2}}\sum_{i=0}^{2^n-1}|i\rangle_n|-\rangle$.

   We now apply the unitary $U_f$ to the circuit, which results in $\frac{1}{2^{n/2}}\sum_{i=0}^{2^n-1}|i\rangle(f(i) - \tilde{f}(i))$. We notice that if the function is constant, then for all $i$, the last qubit is in the state $(|f\rangle - |\tilde{f}\rangle)$ where $f$ is the constnt value of the function. If it is balanced, then for half of the possibilities $i$, the last qubit will be in the state $(|0\rangle - |1\rangle)$ and for the other half the last qubit will be in the state $(|1\rangle - |0\rangle)$. In both cases, like in Deutch's algorithm, we see that the first $n$ qubits and the last qubit are disentangled after the application of $U_f$:

   $$U_f \frac{1}{2^{n/2}} \sum_{i=0}^{2^n-1} |i\rangle(|0\rangle - |1\rangle) = \frac{1}{2^{n/2}} \sum_{i=0}^{2^n-1} |i\rangle(|f(i)\rangle - |\tilde{f}(i)\rangle) = \begin{cases} \frac{1}{2^{n/2}}\sum_{i=1} 2^n|i\rangle \otimes (|f\rangle - |\tilde{f}\rangle) & f \text{ is constant}, \\ \frac{1}{2^{n/2}}\sum_{i=1} 2^n(-1)^{f(i)}|i\rangle \otimes |-\rangle & f \text{ is balanced}. \end{cases}$$

   Now, we continue in a similar manner to the one in Deutch: We apply Hadamard gates on the first $n$ qubits and measure all of them. If the function is constant, the first $n$ qubits are in the state $\frac{1}{2^{n/2}}\sum_{i=1} 2^n|i\rangle$ and as we have already seen several times, applying $H^{\otimes n}$ on

this state will result in $|0\rangle^{\otimes n}$. Let's see what happens if we apply $H^{\otimes n}$ on the first $n$ qubits when $f$ is balanced: We have seen in class that $H^{\otimes n}|i\rangle_n = \frac{1}{2^{n/2}}\sum_{y=0}^{2^n-1}(-1)^{i\cdot y}|y\rangle$, where $i \cdot y = \sum_{\alpha=1}^{n} i_\alpha \wedge y_\alpha$ (i.e. we get a factor of $(-1)$ for each bit that equals 1 both in $i$ and $y$). Use it to get: $H^{\otimes n}\frac{1}{2^{n/2}}\sum_{i=1} 2^n(-1)^{f(i)}|i\rangle = \frac{1}{2^n}\sum_{i,y=0}^{2^n-1}(-1)^{f(i)+i\cdot y}|y\rangle$. Let's check what is the probability to get $|0\rangle^{\otimes n}$ in this case:

$$\langle 0|^{\otimes n}\frac{1}{2^n}\sum_{i,y=0}^{2^n-1}(-1)^{f(i)+i\cdot y}|y\rangle = \frac{1}{2^n}\sum_{i=0}^{2^n-1}(-1)^{f(i)+i\cdot 0} = \frac{1}{2^n}\sum_{i=0}^{2^n-1}(-1)^{f(i)} = 0.$$

We see that if we get $|0\rangle^{\otimes n}$ in our measurement we are guaranteed that the function is constant, and balanced otherwise, using $2n+1$ gates and one application of $U_f$.

# 2 Question 3 - Constructing Toffoli gates

(2.6 in Mermin's book) In theory, we can construct any $n \times n$ unitary gate and apply it to our system. However, experiment-wise (or engineering-wise), it is extremely hard to construct gates that involve more than two qubits. Above we assumed we have some way of creeating unitaries that act on a general number of $n$ qubits, which is really something no one expects to be feasible anywhere in the near future.

Luckily, we can construct any $n-$qubit gate from 1- and 2-qubit gates, i.e. we can build a *universal* set of gates out of 1- and 2-qubit gates alone (recall what universal means from week 1!). An exmample for a universal set (though we will not prove its universaity here) is $\{CNOT, H, T\}$, where $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$. Here we will see a small example of this property by constructing the Toffoli gate out of 2-qubit gates. Here we will see a way to use 8 CNOT gates. Mermin also shows a way to construct Toffoli gates out of 6 CNOT gates (this is the best way known of so far), which you can read if you are interested (section 2.6).

Recall that the Toffoli gate:

$$U_T|x, y, z\rangle = |x, y, z \oplus xy\rangle.$$

Now that we are more familiar with the notation, we can see that infact the Toffoli gate implements logical And between $x$ and $y$.

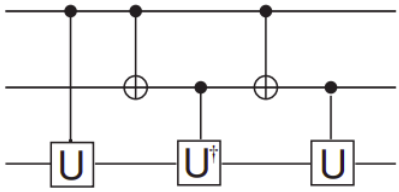
1. Show that if we have a *controlled-U* gate, $cU|x, y\rangle = |x, U^x \oplus y\rangle$, we can construct a *controlled-controlled-$U^2$*out of two $cU$s, one $cU^\dagger$ and two $cNOT$s. Notice the notation $U^x$, which indicates $U$ acts if and only if $x = 1$.

2. define the 1-qubit gate $\sqrt{X}$ and the 2-qubit gate $c\sqrt{X}$. Substitute $U = \sqrt{x}$ in the above result to get the Toffoli gate.

   At home you will show that any controlled-$U$ can be constructed from two $cNOT$ gates and 1-qubit gates, so in total this means we used 8 $cNOT$ gates.

## 2.1 Solution

1. We use the hint we got, which is the exact numbers and types of two qubit gates we should use. Note that if we apply $UU$ gates on the target qubit we get $U^2$, but a pair $UU^\dagger = \mathbb{I}$. So what we could do is:

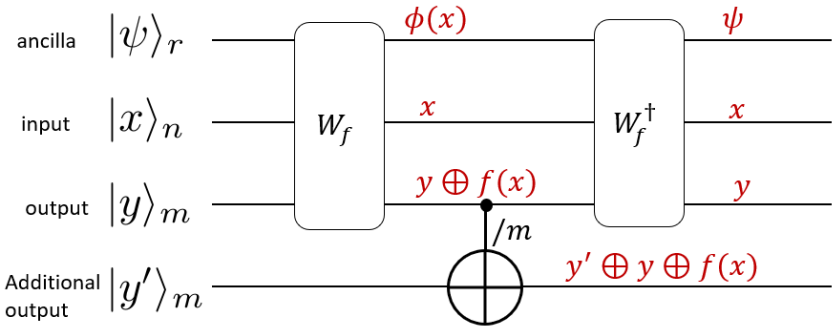



   You can verify for any of the 4 possibilities of the last two qubits - if both are 1, we apply $U^2$. Otherwise, we either apply $UU^\dagger, U^\dagger U$ or nothing.

2. We need to find the matrix that diagonalizes $X$, take the square root of the eigenvalues and apply the diagonalizing matrix from the other side. Luckily, we already found this matrix, when we proved $Z = HXH$. So $\sqrt{X} = H\sqrt{Z}H$. $c\sqrt{X}$ is constructed in a similar way to the one we used to constract $cX$: $\begin{pmatrix} \mathbb{I}_2 & 0 \\ 0 & \sqrt{X} \end{pmatrix}$.

# 3 Uncompute trick - example

Reminder from class: When we have a computation that receives $n$ qubits as input and returns $m$ qubits as output, sometimes we need additional $r$ ancilla qubits for implementing the computation. Therefore, for some function $f : \{0,1\}^n \to \{0,1\}^m$ we cannot implement the unitary $U_f|x\rangle_n|y\rangle_m = |x\rangle_n|y \oplus f(x)\rangle_m$, but rather $W_f|x\rangle_n|y\rangle_m|\psi\rangle_r = |x\rangle_n|y \oplus f(x)\rangle_m|\phi\rangle_r$. Note that $W_f$ is also a unitary (otherwise, we wouldn't have been able to implement it). If $|\phi\rangle_r$ depends on $x$, which it does since we just used the ancillary qubits for the computation, this is a bit of a problem - the ancillary qubits are now entangled to the "main" qubits, and the "main" qubits are not in the state we expected. In fact, we cannot assign to them a state by themselves, since they are entangled to the ancillary qubits. However, in class, Moshe showed that we can always disentangle the ancillary qubits from the "main" qubits:




Let us see that this works even if the input is in a superposition, so that at the end the input and the additional output will be entangled, as required, but the ancilla and output would be decoupled. For simplicity, we take $n = m = r = 1$. We will only consider the initial state $|y\rangle = |\psi\rangle = |0\rangle$, since this is usually the case, but you are welcome to check that the circuit above will work for any initial state of the output and ancilla.

Assume our function $f$ is $f(0) = 0, f(1) = 1$ and therefore

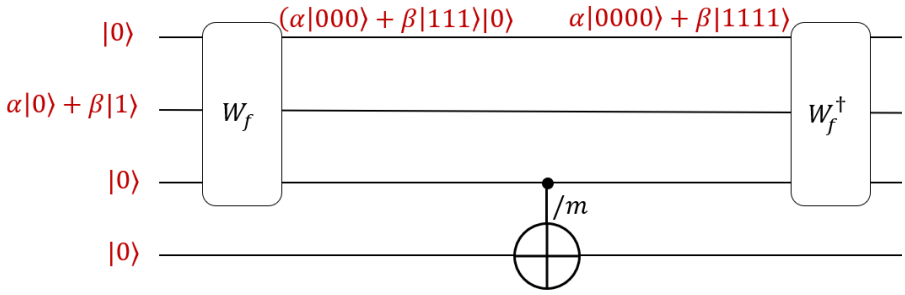
$$U_f|00\rangle = |00\rangle, U_f|10\rangle = |11\rangle.$$

Unfortunately, we cannot implement $U_f$, but only $W_f$, for which

$$W_f|000\rangle = |000\rangle, W_f|010\rangle = |111\rangle,$$

where the leftmost qubit is the ancillary qubit, the middle qubit is the input and the rightmost bit is the output.

Let's see what happens when we apply the circuit above when the input is in a super position:




Now, the last gate is $W_f^\dagger \otimes \mathbb{I}$.

$$W_f^\dagger \otimes \mathbb{I}|0000\rangle = W_f^\dagger|000\rangle \otimes |0\rangle = |0000\rangle, W_f^\dagger \otimes \mathbb{I}|1111\rangle = W_f^\dagger|111\rangle \otimes |1\rangle = |0101\rangle.$$

So the final state is $\alpha|0000\rangle + \beta|0101\rangle$. Now the input qubits and the additional output qubits are disentangled from the other qubits, and are in the state $U_f(\alpha|0\rangle + \beta|1\rangle)|0\rangle = \alpha|00\rangle + \beta|11\rangle$ as we intended.