Contents

1 The computational model: quantum circuits

1.1 qubits

Definition - quantum computer A quantum computer is a quantum system composed of n qubits, whose dynamics can be completely *controlled* by an external observer.

Definition - qubit A qubit is a two-level quantum system. The two levels are usually labelled $|0\rangle$ and $|1\rangle$.

The Hilbert space associated to a qubit is therefore:

$$\mathcal{H}_1 = \{\alpha|0\rangle + \beta|1\rangle \quad st \quad \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1\}$$

And the Hilbert space associated to a quatum computer, i.e. the space of all possible *states* for a quantum computer is:

$$\begin{aligned}\mathcal{H}_n &= \otimes_{i=1}^{n} \mathcal{H}_0 \\ &= \left\{ \sum_{i=0}^{2^n-1} a_i|i\rangle \quad st \quad \forall i a_i \in \mathbb{C}, \quad \sum_{i=0}^{2^n-1} |a_i|^2 = 1 \right\}\end{aligned}$$

Above, $|i\rangle$, with i integer denotes the state $\otimes_{i=1}^{n}|b_i\rangle$ with b_i the i-th[1] bit in the binary decomposition of i. For small values of n, the states are typically written with bits directly.

dimension The dimension of the Hilbert space of a quantum computer with n qubits is therefore 2^n.

[1] or $(n-i)$-th bit, depending on the convention.

Initialization By convention, at the beginning of a quantum computation, the qubits are initialized at $|0\dots0\rangle$.

Measure As usual, $|a_i|^2$ denotes the probability of observing i when measuring the state of all qubits. I.e. the probability that qubit 1 is measured in state b_1, qubit 2 in state b_2, etc.

1.2 quantum gates

A quantum computer is a *closed* system, thereby following Hamiltonian/unitary dynamics, as per Schrödinger's equation.

When using a quantum computer, we modify its state through the application of unitary operations called *quantum gates*. These quantum gates are typically drawn from a fixed *gate set*, containing gates each involving only a small number of qubits at a time.

Definition - (universal gate set) A quantum computer comes with a *gate set*, a set of unitary operations involving a few qubits at a time. A universal gate set is able to generate any unitary operation.

Universal gate set are universal in light of the Solovay-Kitaev theorem.

Analogy with classical case Instruction sets for processors.

Usual gates CNOT, H, rotations, T, S (Phase)...

Bell pair creation H and then CNOT.

1.3 quantum circuit

A quantum circuit is a sequence/list of gates, drawn from a given gate set. It is the quantum equivalent of Boolean circuits, with "wires" representing qubits, and gates as boxes applied to them.

We typically refer to the number of gates in a quantum circuit as its *size*

1.4 Implementations for qubits and quantum computers

1.5 Computational model - quantum speedup

quantum algorithm The execution of a quantum circuit is the sequential application of all its gates on a quantum computer initialized at $|0...0\rangle$ followed by the measurement of all qubit. Intermediary measurements may also be applied, and condition the subsequent application of other gates.

classical complexity We count the number of elementary operations and upper-bound them aymptotically as a function of the input size[2].

quantum complexity We look at the size of quantum circuits involved in quantum algorithms solving the problem, and asymptotically upperbound it as a function of the input size.

poly-time quantum and poly-time classical Poly-time quantum is therefore defined in terms of poly-sized quantum circuits. Poly-time classical has been defined in vaguer terms here, but note that, up to theoretical subtleties beyond the scope of this lecture[3], classical poly-time complexity could be likewise defined in terms of Boolean circuit size complexity.

- Informally, the question of quantum algorithmics is: which computational problem have a quantum complexity that is asymptotically better than the complexity of any known/possible classical algorithm for the problem ?
- Are there even problems where there is a polynomial quantum algorithm and (as far as we know) only exponential/super-polynomial classical algorithms ?

2 Quantum algorithms

2.1 by hand example - Deutsch-Josza

From a classical circuit computing a Boolean function, it is fairly easy to get a reversible circuit computing that function. From a reversible circuit, it is easy to get a quantum version using Toffoli-s and CNOTs. This quantum version is usually an oracle, i.e. that does

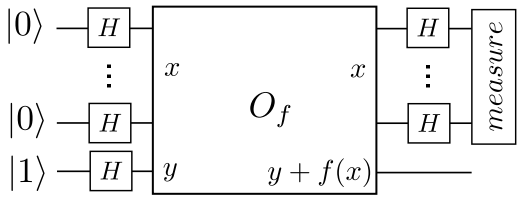
$$|x\rangle|y\rangle \underset{O_f}{\longrightarrow} |x\rangle|y \oplus f(x)\rangle$$

.

The Deutsch-Josza algorithm solves the following problem:
Input: a fonction $f : \{0,1\}^n \to \{0,1\}$, with the promise that it is either **balanced** ($|f^{-1}(0)| = |f^{-1}|(1)$) or **constant**.
Task: decide whether f is constant or balanced.

Classical algorithm Classically, one needs to compute f for at least half of the entries plus 1, i.e. $2^{n-1} + 1$ calls to f at least.

Quantum algorithm Quantumly, one call to the oracle is enough ! With the circuit below.



Let us walk through the steps of the algorithm for $n = 1$. In this case, either f is constant and $f(0) = f(1) = b$, or f is balanced and $f(0) = b$ while $f(1) = 1 - b$. For all binary numbers we write $\overline{x} = 1 - x$

To start with, Hadamard gates are applied:

$$\begin{aligned} |01\rangle &\xrightarrow{H \otimes H} \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) \\ &= \left(\frac{|00\rangle + |10\rangle - |01\rangle - |11\rangle}{2}\right) \end{aligned}$$

Then, the oracle, which yields:

$$\frac{|0f(0)\rangle + |1f(1)\rangle - |0\overline{f(0)}\rangle - |1\overline{f(1)}\rangle}{2}$$

Now let us distinguish the two cases:

- if f is constant, the application of $H \otimes I$ will yield:

$$\begin{aligned} \frac{|0f(0)\rangle + |1f(1)\rangle - |0\overline{f(0)}\rangle - |1\overline{f(1)}\rangle}{2} &= \frac{|0b\rangle + |1b\rangle - |0\bar{b}\rangle - |1\bar{b}\rangle}{2} \\ &= \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) \otimes \left(\frac{|b\rangle - |\bar{b}\rangle}{\sqrt{2}}\right) \\ &\xrightarrow[H \otimes I]{} |0\rangle \otimes \left(\frac{|b\rangle - |\bar{b}\rangle}{\sqrt{2}}\right) \end{aligned}$$

  and therefore 0 is measured.

[2] number of vertices/edges if the input is a graph, number of bits if the input is a number, number of elements if the input is a list...

[3] circuit “uniformity”

- if f is balanced, then $f(0) = b$ and $f(1) = \bar{b}$ and:

$$\begin{aligned}
\frac{|0f(0)\rangle + |1f(1)\rangle - |0\overline{f(0)}\rangle - |1\overline{f(1)}\rangle}{2} &= \frac{|0b\rangle + |1\bar{b}\rangle - |0\bar{b}\rangle - |1b\rangle}{2} \\
&\xrightarrow[H\otimes I]{} \frac{\cancel{|0b\rangle} + |1b\rangle + \cancel{|0\bar{b}\rangle} - |1\bar{b}\rangle - \cancel{|0\bar{b}\rangle} - |1\bar{b}\rangle - \cancel{|0b\rangle} + |1b\rangle}{\sqrt{8}} \\
&= |1\rangle \otimes \left(\frac{|b\rangle - |\bar{b}\rangle}{\sqrt{2}} \right)
\end{aligned}$$

2.2 Quantum Fourier Transform

Definition Looking a lot like a quantized version of the Fast Fourier Transform, the definition of the quantum Fourier transform is as such:

$$|j\rangle \xrightarrow{QFT} \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{\frac{2i\pi jk}{2^n}} |k\rangle$$

2.3 Phase Estimation

How can suh a thing be useful ? In fact building block of the blueprint for every quantum algorithms exhibiting a presumed exponential speed-up.

Imagine a unitary operator U with an eigenvector $|u\rangle$ and eigenvalue $e^{2i\pi\phi}$ with $\phi \in [0, 1[$. In the case of Shor's algorithm, algorithms for quantum chemistry, quantum linear algebra... *The hard part is computing ϕ for a specific U.*

The QFT (or rather its inverse) allows to extract ϕ from the following state:

$$\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2i\pi\phi k} |k\rangle$$

Indeed we can write $\phi = 0.\phi_0...\phi_{n-1} = \sum_k \frac{\phi_k}{2^k}$. Which we also write in reverse $\phi = 0.\tilde{\phi}_{n-1}...\tilde{\phi}_0 = \sum_l \frac{\tilde{\phi}_l 2^l}{2^n}$.

$$\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2i\pi\phi k} |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{\frac{2i\pi k \left(\sum_l \tilde{\phi}_l 2^l\right)}{2^n}} |k\rangle$$

Which becomes, when applying an inverse QFT, we get the state $|\tilde{\phi}_0...\tilde{\phi}_{n-1}\rangle$

But how do we get this state: $\frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1} e^{2i\pi\phi k}$? Let us write:

$$\begin{aligned}
\frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1} e^{2i\pi\phi k}|k\rangle &= \frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1} e^{2i\pi\phi\left(\sum_l k_l 2^l\right)}|k\rangle \\
&= \frac{1}{\sqrt{2^n}}\sum_{k_1=0,1}\cdots\sum_{k_n=0,1}\prod_l e^{2i\pi\phi k_l 2^l}|k_1 ... k_n\rangle \\
&= \otimes_{l=0}^{n-1}\left(\frac{|0\rangle + e^{2i\pi\phi 2^l}|1\rangle}{\sqrt{2}}\right)
\end{aligned}$$

A state that is almost equal to what we want is $(H\otimes ...\otimes H)\,|0...0\rangle = \otimes_{l=0}^{n-1}\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)$. From it, we need to apply a phase on qubit l *when it is equal to* 1 *but not when it is equal to* 0. We can do that with a controlled operation.

We just need to apply U^{2^j} on $|u\rangle$, controlled by qubit j.

2.4 Overview: other algorithms and their speedup

3 What you will do in the TP: QPE and VQE for quantum chemistry