Quantum Algorithms 2021/2022: Exercices 2

Benoît Vermersch (benoit.vermersch@lpmmc.cnrs.fr) -October 16, 2022

1 Implementation of the quantum Fourier transform

Ref: Nielsen and Chuang. The quantum Fourier transform realizes the transformation

$$U|j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi ijk/N} |k\rangle, \tag{1}$$

with $j, k = 0, \ldots, N-1$. Our goal is to implement this transformation for $N = 2^n$, using 2 coupled circuits of n qubits.

1. Binary representation $j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0$

2. We use the notation $0.j_l \ldots j_n = j_l/2 + \ldots j_n/2^{n-l+1}$.

$$\begin{aligned}
U|j\rangle &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi ijk/N} |k\rangle \\
&= \frac{1}{\sqrt{N}} \sum_{k_1,\ldots,k_n=0}^{1} e^{2\pi ij(k_1 2^{n-1}+\cdots+k_n 2^0)/2^n} |k_1,\ldots,k_n\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k_1,\ldots,k_n} \bigotimes_l \left(e^{2\pi ijk_l 2^{-l}} |k_l\rangle\right) \\
&= \frac{1}{2^{n/2}} \bigotimes_l \left[\sum_{k_l} e^{2\pi ijk_l 2^{-l}} |k_l\rangle\right] \\
&= \frac{1}{2^{n/2}} \bigotimes_l \left[|0\rangle + e^{2\pi ij2^{-l}} |1\rangle\right] \\
&= \frac{1}{2^{n/2}} (|0\rangle + e^{2i\pi j/2} |1\rangle)(|0\rangle + e^{2i\pi(j/4)} |1\rangle) \ldots (|0\rangle + e^{2i\pi(j/2^n)} |1\rangle) \\
&= \frac{1}{2^{n/2}} (|0\rangle + e^{2i\pi j_n/2} |1\rangle)(|0\rangle + e^{2i\pi(j_{n-1}/2+j_n/4)} |1\rangle) \ldots (|0\rangle + e^{2i\pi(j_1/2+\cdots+j_n/2^n} |1\rangle) \\
&= \frac{1}{2^{n/2}} (|0\rangle + e^{2i\pi 0.j_n} |1\rangle)(|0\rangle + e^{2i\pi 0.j_{n-1}j_n} |1\rangle) \ldots (|0\rangle + e^{2i\pi 0.j_1\cdots j_n} |1\rangle)
\end{aligned} \tag{2}$$

3. We have

$$R_2 = \begin{bmatrix} 1 & 0 \\ 0 & e^{2i\pi/2^2} \end{bmatrix}. \tag{3}$$

$$\begin{aligned}
C[R_2]H_1|j\rangle &= \frac{1}{\sqrt{2}} C[R_2] \left(|0\rangle + e^{2i\pi 0.j_1} |1\rangle\right) |j_2 \ldots j_n\rangle \\
&= \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2i\pi 0.j_1} e^{2i\pi j_2/2^2} |1\rangle\right) |j_2 \ldots j_n\rangle \\
&= \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2i\pi 0.j_1 j_2} |1\rangle\right) |j_2 \ldots j_n\rangle
\end{aligned} \tag{4}$$

After the first R_n rotations

$$C[R_n] \ldots C[R_2]H_1|j\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2i\pi 0.j_1 j_2 \cdots j_n} |1\rangle\right) |j_2 \ldots j_n\rangle \tag{5}$$

4. After the Hadamard on the second qubit, we obtain

$$\frac{1}{2} \left(|0\rangle + e^{2i\pi 0.j_1 j_2 \cdots j_n} |1\rangle\right) \left(|0\rangle + e^{2i\pi 0.j_2} |1\rangle\right) |j_3 \ldots j_n\rangle \tag{6}$$

After the controlled R_k rotations on the second qubit, we obtain

$$\frac{1}{2} \left(|0\rangle + e^{2i\pi 0.j_1 j_2 \cdots j_n} |1\rangle\right) \left(|0\rangle + e^{2i\pi 0.j_2 \cdots j_n} |1\rangle\right) |j_3 \ldots j_n\rangle \tag{7}$$

At the end of the circuit

$$\frac{1}{2^{n/2}}\left(|0\rangle + e^{2i\pi 0.j_1 j_2 \ldots j_n}|1\rangle\right)\left(|0\rangle + e^{2i\pi 0.j_2 \ldots j_n}|1\rangle\right)\ldots\left(|0\rangle + e^{2i\pi 0.j_n}|1\rangle\right) \tag{8}$$

Up to a swap transformation, this is the desired transformation.

2 Factorizing 21 with Shor's algorithm

We take $N = 21$.

1. **Classical part** Assume we randonly pick $a = 2$. Show that the function $f(x) = a^x \bmod(N)$ is 6 periodic. $2^6 = 64 = 1 + 21 \times 3 = 1 \bmod(N)$. $f(x+6) = a^{x+6} \bmod(N) = a^x(1 + N \times 3) \bmod(N) = a^x \bmod(N) = f(x)$.

2. Find two non-trivial divisors of N. We have: N divides $a^6 - 1$. Therefore, with $b = a^3 = 8$, N divides $b^2 - 1$. According to the result presented in Lecture 3, $\gcd(N, b \pm 1) = 7, 3$ are non-trivial divisors of N.

3. **Quantum subroutine** The quantum subroutine of Shor's algorithm consists in finding the period $r = 6$ of $f(x)$. How many qubits do we need to implement this algorithms? $N^2 = 441$, we thus need 2 registers of $q = 9$ qubits.

4. Write the state of the system after modular exponentiation.

$$|\psi\rangle = \frac{1}{\sqrt{Q}} \sum_x |x\rangle \otimes |f(x)\rangle\,, \tag{9}$$

with $Q = 2^q = 512$.

5. Write the state after inverse quantum Fourier transform and the probability $P(y)$ to observe the bitstring y after measuring the first q qubits.

$$|\psi\rangle = \frac{1}{Q} \sum_x (\sum_y e^{-2i\pi xy/Q} |y\rangle) \otimes |f(x)\rangle \tag{10}$$

$$|\psi\rangle = \frac{1}{Q} \sum_y \left(|y\rangle \otimes \sum_x e^{-2i\pi xy/Q} |f(x)\rangle \right) \tag{11}$$

$$\begin{aligned} P(y) &= \sum_{y'} |\langle y, y'|\, |\psi\rangle|^2 = \frac{1}{Q^2} \sum_{x_1, x_2} e^{-2i\pi(x_2 - x_1)y/Q} \langle f(x_1)| \sum_{y'} |y'\rangle \langle y'|\, |f(x_2)\rangle \\ &= \frac{1}{Q^2} \sum_{x_1, x_2} e^{-2i\pi(x_2 - x_1)y/Q} \langle f(x_1)|\, |f(x_2)\rangle \end{aligned}$$

We obtain the dominant contributions for $x_2 - x_1 = sr$.

$$P(y) \approx \frac{1}{Q^2} \sum_{x_1, s} e^{-2i\pi sry/Q}. \tag{12}$$

6. Plot the function $P(y)$ and give the table of the three most likely measured bitstrings.

7. We expect $P(y)$ to be maximal for $ry/Q \approx p$, thus $y/Q \approx p/r$. The continued fraction algorithm gives us the closest fraction p/r to the measured y/Q rational, with a maximum r_{max} tunable value for r. For Python, this is implemented as fractions.Fraction(float).limit_denominator(rmax). Give the attributed value for each most likely bitstring r. Comment. For $y = 85$, we search for a fraction such that $85/512 = 0.166 \approx 1/6$. We end up with $b = \sqrt{2^6} = 8$. For $y = 171$, $r = 3 \to$ fail, or $r = 6$, $\to$ success. For $y = 512$, we attibute $r = 2$ from $1/2$ (instead of 6 from $3/6$, which results in a fail.

8. Repeat the same exercise with $a = 13$. The function is $r = 2$ periodic. We find $b = 13$, and $7, 3$ as non-trivial divisors. In this case, the success probability is $1/2$, obtained when measuring $y = Q/2$.