Advanced Quantum Mechanics

Angelo Bassi

Academic Year 2020-21

Quantum Integral Transform

DEFINITION 6.1 (Discrete Integral Transform) Let $n \in \mathbb{N}$ and $S_n = \{0, 1, \ldots, 2^n - 1\}$ be a set of integers. Consider a map

$$K : S_n \times S_n \to \mathbb{C}. \tag{6.1}$$

For any function $f : S_n \to \mathbb{C}$, its **discrete integral transform** (DIT) $\tilde{f} : S_n \to \mathbb{C}$ with the **kernel** K is defined as:

$$\tilde{f}(y) = \sum_{x=0}^{2^n-1} K(y, x) f(x). \tag{6.2}$$

The transformation $f \to \tilde{f}$ is also called the discrete integral transform.

We define $N \equiv 2^n$ to simplify our notations. The kernel K is expressed as a matrix,

$$K = \begin{pmatrix} K(0,0) & \ldots & K(0, N-1) \\ K(1,0) & \ldots & K(1, N-1) \\ \ldots & \ldots & \ldots \\ K(N-1, 0) & \ldots & K(N-1, N-1), \end{pmatrix} \tag{6.3}$$

Quantum Integral Transform

PROPOSITION 6.1 Suppose the kernel K is unitary: $K^\dagger = K^{-1}$. Then the inverse transform $\tilde{f} \to f$ of a DIT exists and is given by

$$f(x) = \sum_{y=0}^{N-1} K^\dagger(x, y) \tilde{f}(y). \qquad (6.4)$$

Proof. By substituting Eq. (6.2) into Eq. (6.4), we prove

$$\sum_{y=0}^{N-1} K^\dagger(x, y) \tilde{f}(y) = \sum_{y=0}^{N-1} K^\dagger(x, y) \left[\sum_{z=0}^{N-1} K(y, z) f(z) \right]$$

$$= \sum_{z=0}^{N-1} \left[\sum_{y=0}^{N-1} K^\dagger(x, y) K(y, z) \right] f(z)$$

$$= \sum_{z=0}^{N-1} \delta_{xz} f(z) = f(x).$$

∎

Quantum Integral Transform

Now we make the connection with quantum computing

Let U be an $N \times N$ unitary matrix which acts on the n-qubit space $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$. Let $\{|x\rangle = |x_{n-1}, x_{n-2} \ldots, x_0\rangle\}$ $(x_k \in \{0, 1\})$ be the standard binary basis of $\mathcal{H}$, where $x = x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \ldots + x_0 2^0$. Then

$$U|x\rangle = \sum_{y=0}^{N-1} |y\rangle\langle y|U|x\rangle = \sum_{y=0}^{N-1} U(y, x)|y\rangle. \qquad (6.5)$$

The complex number $U(x, y) = \langle x|U|y\rangle$ is the (x, y)-component of U in this basis.

PROPOSITION 6.2 Let U be a unitary transformation, acting on $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$. Suppose U acts on a basis vector $|x\rangle$ as

$$U|x\rangle = \sum_{y=0}^{N-1} K(y,x)|y\rangle. \qquad (6.6)$$

Then U *computes** the DIT $\tilde{f}(y) = \sum_{x=0}^{N-1} K(y,x)f(x)$ for any $y \in S_n$, in the sense that

$$U\left[\sum_{x=0}^{N-1} f(x)|x\rangle\right] = \sum_{y=0}^{N-1} \tilde{f}(y)|y\rangle. \qquad (6.7)$$

Here $|x\rangle$ and $|y\rangle$ are basis vectors of $\mathcal{H}$.

Proof. In fact,

$$U\left[\sum_{x=0}^{N-1} f(x)|x\rangle\right] = \sum_{x=0}^{N-1} f(x)U|x\rangle$$

$$= \sum_{x=0}^{N-1} f(x)\left[\sum_{y=0}^{N-1} K(y,x)|y\rangle\right] = \sum_{y=0}^{N-1}\left[\sum_{x=0}^{N-1} K(y,x)f(x)\right]|y\rangle$$

$$= \sum_{y=0}^{N-1} \tilde{f}(y)|y\rangle. \qquad (6.8)$$

■

Quantum Integral Transform

The unitary matrix U implementing a discrete integral transform as in Eq. (6.7) is called the **quantum integral transform (QIT).**

Quantum Integral Transform

We will introduce three types of QIT:

1. Quantum Fourier Transform (QFT)
2. Walsh Hadamard Transform (which we already saw)
3. Selective Phase Rotation Transform

Quantum Fourier Transform

Fourier transform. Let ω_n be the Nth primitive root of 1;

$$\boxed{\omega_n = e^{2\pi i/N},}$$ (6.10)

where $N = 2^n$ as before. The complex number ω_n defines a kernel K by

$$\boxed{K(x,y) = \frac{1}{\sqrt{N}}\omega_n^{-xy}.}$$ (6.11)

The discrete integral transform with the kernel K,   The inverse DFT is given by

$$\tilde{f}(y) = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}\omega_n^{-xy}f(x), \qquad\longleftrightarrow\qquad (6.12) \qquad f(x) = \frac{1}{\sqrt{N}}\sum_{y=0}^{N-1}\omega_n^{xy}\tilde{f}(y).$$

is called the **discrete Fourier transform (DFT)**.
   The kernel K is unitary since

$$(KK^\dagger)(x,y) = \langle x|K\sum_z|z\rangle\langle z|K^\dagger|y\rangle = \sum_z K(x,z)K^\dagger(z,y)$$

$$= \frac{1}{N}\sum_z\omega_n^{-xz}\omega_n^{yz} = \frac{1}{N}\sum_z\omega^{-(x-y)z} = \delta_{xy}.$$

Quantum Fourier Transform

The quantum integral transform defined with this kernel is called the **quantum Fourier transform (QFT).**

$$U_{\mathrm{QFT}}|x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \omega_n^{-xy}|y\rangle \qquad \Longrightarrow \qquad U_{\mathrm{QFT}} = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \omega_n^{-xy}|y\rangle\langle x|$$

It is important to note that

$$U_{\mathrm{QFT}n}|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} |y\rangle, \qquad\qquad (6.16)$$

where $U_{\mathrm{QFT}n}$ is the n-qubit QFT gate. This equality shows that the QFT of $f(x) = \delta_{x0}$ is $\tilde{f}(y) = 1/\sqrt{2^n}$, which is similar to the FT of the Dirac delta function $\delta(x)$. Observe that a single application of $U_{\mathrm{QFT}n}$ on the state $|0\rangle$ has produced the superposition of all the basis vectors of $\mathcal{H}$.

Examples

The kernel for $n = 1$ is

$$K_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & e^{2\pi i/2} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad (6.13)$$

which is nothing but our familiar Hadamard gate. For $n = 2$, we have $\omega_2 = e^{2\pi i/4} = i$ and

$$K_2 = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega_2^{-1} & \omega_2^{-2} & \omega_2^{-3} \\ 1 & \omega_2^{-2} & \omega_2^{-4} & \omega_2^{-6} \\ 1 & \omega_2^{-3} & \omega_2^{-6} & \omega_2^{-9} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix}. \qquad (6.14)$$

Circuit implementation of QFT: n = 1

$\underline{n = 1}$

Eq. (6.13) shows that the kernel for $n = 1$ QFT is the Hadamard gate H, whose action on $|x\rangle$, $x \in \{0, 1\}$, is concisely written as

$$U_{\mathrm{H}}|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^x|1\rangle) = \frac{1}{\sqrt{2}}\sum_{y=0}^{1}(-1)^{xy}|y\rangle. \qquad (6.24)$$

In fact, this is the defining equation for $n = 1$ QFT as

$$U_{\mathrm{QFT1}}|x\rangle = \frac{1}{\sqrt{2}}\sum_{y=0}^{1}\omega_1^{-xy}|y\rangle = \frac{1}{\sqrt{2}}\sum_{y=0}^{1}(-1)^{xy}|y\rangle. \qquad (6.25)$$
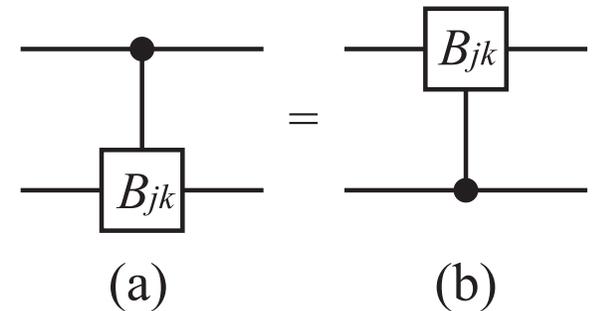
Circuit implementation of QFT: n = 2

<u>$n = 2$</u>

This case is considerably more complicated than the case $n = 1$. It also gives important insights into implementing QFT with $n \geq 3$. Let us introduce an important gate, the **controlled-B_{jk}** gate. The B_{jk} gate is defined by the matrix

$$B_{jk} = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\theta_{jk}} \end{pmatrix}, \quad \theta_{jk} = \frac{2\pi}{2^{k-j+1}}, \tag{6.26}$$

where $j, k \in \{0, 1, 2, \ldots\}$ and $k \geq j$.

LEMMA 6.1 The controlled-B_{jk} gate U_{jk} in Fig. 6.1 (a) acts on $|x\rangle|y\rangle$, $x, y \in \{0, 1\}$, as

$$U_{jk}|x, y\rangle = e^{-i\theta_{jk}xy}|x, y\rangle = \exp\left(-\frac{2\pi i}{2^{k-j+1}}xy\right)|x, y\rangle. \tag{6.27}$$

(a)   (b)

Circuit implementation of QFT: n = 2

Proof. The controlled-B_{jk} gate is written as

$$U_{jk} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes B_{jk}, \qquad (6.28)$$
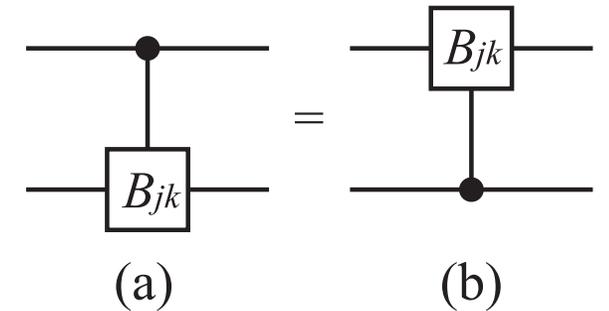
and its action on $|x, y\rangle$ is

$$
\begin{aligned}
U_{jk}|x, y\rangle &= |0\rangle\langle 0|x\rangle \otimes |y\rangle + |1\rangle\langle 1|x\rangle \otimes B_{jk}|y\rangle \\
&= \begin{cases} |x\rangle \otimes |y\rangle & x = 0 \\ |x\rangle \otimes B_{jk}|y\rangle & x = 1. \end{cases}
\end{aligned}
\qquad (6.29)
$$

Moreover, when $x = 1$ we have

$$
B_{jk}|y\rangle = \begin{cases} |y\rangle & y = 0 \\ e^{-i\theta_{jk}}|y\rangle & y = 1. \end{cases}
\qquad (6.30)
$$

Thus the action of U_{jk} on $|y\rangle$ is trivial if $xy = 0$ and nontrivial if and only if $x = y = 1$. These results may be summarized as Eq. (6.27). ∎

(a)   (b)

The action of the controlled-B_{jk} gate on a basis vector $|x\rangle|y\rangle$ is determined by the combination xy and not by x and y independently. Therefore the controlled-B_{jk} gate and the "inverted" controlled-B_{jk} gate are equivalent; see Fig. 6.1.

Circuit implementation of QFT: n = 2

Equation (6.6) in Proposition 6.2 states that our task is to find a unitary matrix U_{QFT2} such that

$$U_{\mathrm{QFT2}}|x\rangle = \frac{1}{2}\sum_{y=0}^{3}\omega_2^{-xy}|y\rangle. \qquad (6.32)$$

Let us write x and y in the binary form as $x = 2x_1 + x_0$ and $y = 2y_1 + y_0$, respectively. The action of U_{QFT2} on $|x\rangle$ is

$$U_{\mathrm{QFT2}}|x_1x_0\rangle = \frac{1}{2}\sum_{y=0}^{3}e^{-2\pi ixy/2^2}|y\rangle = \frac{1}{2}\sum_{y_0,y_1=0}^{1}e^{-2\pi ix(2y_1+y_0)/2^2}|y_1y_0\rangle$$

$$= \frac{1}{2}\sum_{y_1}e^{-2\pi ixy_1/2}|y_1\rangle \otimes \sum_{y_0}e^{-2\pi ixy_0/2^2}|y_0\rangle$$

$$= \frac{1}{2}\left(|0\rangle + e^{-2\pi ix/2}|1\rangle\right) \otimes \left(|0\rangle + e^{-2\pi ix/2^2}|1\rangle\right)$$

$$= \frac{1}{2}\left(|0\rangle + e^{-2\pi i(2x_1+x_0)/2}|1\rangle\right) \otimes \left(|0\rangle + e^{-2\pi i(2x_1+x_0)/2^2}|1\rangle\right)$$

$$= \frac{1}{2}\left(|0\rangle + e^{-\pi ix_0}|1\rangle\right) \otimes \left(|0\rangle + e^{-\pi ix_1}e^{-i(\pi/2)x_0}|1\rangle\right)$$

$$= \frac{1}{2}\left(|0\rangle + (-1)^{x_0}|1\rangle\right) \otimes B_{12}^{x_0}\left(|0\rangle + (-1)^{x_1}|1\rangle\right), \qquad (6.33)$$

$$B_{12} = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\theta_{12}} \end{pmatrix}, \quad \theta_{12} = \frac{2\pi}{2^{2-1+1}} = \frac{\pi}{2}$$

Then

$$B_{12} = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/2} \end{pmatrix}$$

Note that $B^{x_0}_{12}$ is the controlled-B gate with the control bit x_0 and the target bit x_1; $B^0_{12} = I$ while $B^1_{12} = B_{12}$. Note also that, in spite of its tensor product looking appearance, the last line of Eq. (6.33) is entangled due to this conditional operation.

Circuit implementation of QFT: n = 2

$$U_{\mathrm{QFT2}}|x_1 x_0\rangle = \frac{1}{\sqrt{2^2}} \left(|0\rangle + (-1)^{x_0}|1\rangle\right) \otimes B_{12}^{x_0} \left(|0\rangle + (-1)^{x_1}|1\rangle\right)$$

Equation (6.33) suggests that the $n = 2$ QFT are implemented with the Hadamard and the U_{12} gates. Before writing down the quantum circuit realizing Eq. (6.33), we should note that the first qubit has a power $(-1)^{x_0}$, while the second one has $(-1)^{x_1}$, when the input state is $|x_1 x_0\rangle$. If we naively applied the Hadamard gate to the second qubit, we would obtain

$$(I \otimes U_{\mathrm{H}})|x_1 x_0\rangle = |x_1\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{x_0}|1\rangle).$$

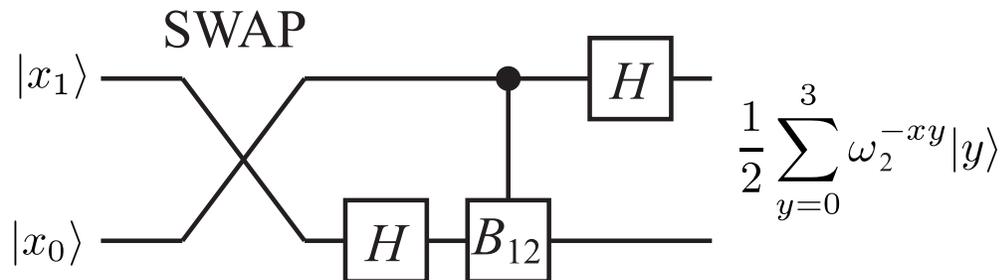These facts suggest that we need to swap the first and second qubits at the beginning of the implementation

Circuit implementation of QFT: n = 2

$$U_{\mathrm{QFT2}}|x_1 x_0\rangle = \frac{1}{\sqrt{2^2}} \left(|0\rangle + (-1)^{x_0}|1\rangle\right) \otimes B_{12}^{x_0} \left(|0\rangle + (-1)^{x_1}|1\rangle\right)$$

$$= (U_{\mathrm{H}} \otimes I)U_{12}(I \otimes U_{\mathrm{H}})|x_0, x_1\rangle$$

$$= (U_{\mathrm{H}} \otimes I)U_{12}(I \otimes U_{\mathrm{H}})U_{\mathrm{SWAP}}|x_1 x_0\rangle. \qquad (6.34)$$

PROPOSITION 6.3 The $n = 2$ QFT gate is implemented as

$$U_{\mathrm{QFT2}} = (U_{\mathrm{H}} \otimes I)U_{12}(I \otimes U_{\mathrm{H}})U_{\mathrm{SWAP}} \qquad (6.35)$$

(see Fig. 6.2).

FIGURE 6.2
Implementation of the $n = 2$ QFT, U_{QFT2}.

Circuit implementation of QFT: n = 3

$$U_{\mathrm{QFT3}}|x_2 x_1 x_0\rangle$$

$$= \frac{1}{\sqrt{2^3}}(|0\rangle + e^{-2\pi i x_0/2}|1\rangle) \otimes (|0\rangle + e^{-2\pi i(x_1/2 + x_0/2^2)}|1\rangle)$$

$$\otimes(|0\rangle + e^{-2\pi i(x_2/2 + x_1/2^2 + x_0/2^3)}|1\rangle)$$

$$= \frac{1}{\sqrt{2^3}}(|0\rangle + (-1)^{x_0}|1\rangle) \otimes B_{01}^{x_0}(|0\rangle + (-1)^{x_1}|1\rangle)$$

$$\otimes B_{02}^{x_0} B_{12}^{x_1}(|0\rangle + (-1)^{x_2}|1\rangle)$$

$$= (U_{\mathrm{H}} \otimes I \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I)U_{02}U_{12}(I \otimes I \otimes U_{\mathrm{H}})|x_0 x_1 x_2\rangle$$

$$= (U_{\mathrm{H}} \otimes I \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I)U_{02}U_{12}(I \otimes I \otimes U_{\mathrm{H}})P|x_2 x_1 x_0\rangle, \quad (6.36)$$

$$|x_1 x_0\rangle \rightarrow \frac{1}{\sqrt{2^2}} \sum_{y=0}^{2^2-1} e^{-2\pi i x y/2^2}|y\rangle$$

$$= \frac{1}{\sqrt{2^2}}(|0\rangle + e^{-2\pi i x_0/2}|1\rangle) \otimes (|0\rangle + e^{-2\pi i(x_1/2 + x_0/2^2)}|1\rangle).$$

For n= 2

where U_{jk} is the controlled-B_{jk} gate with the control qubit x_j, and the gate P reverses the order of the qubits as $P|x_2 x_1 x_0\rangle = |x_0 x_1 x_2\rangle$. For a three-qubit QFT, P is a SWAP gate between the first qubit (x_2) and the third qubit (x_0). Again note here that we should be careful in ordering the gates so that the control bit x_j acts in U_{jk} before it is acted by a Hadamard gate.

Circuit implementation of QFT: n = 3

$$U_{\mathrm{QFT3}} = (U_{\mathrm{H}} \otimes I \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I)U_{02}U_{12}(I \otimes I \otimes U_{\mathrm{H}})P. \qquad (6.38)$$

Equation (6.38) readily leads us to the quantum circuit in Fig. 6.3.
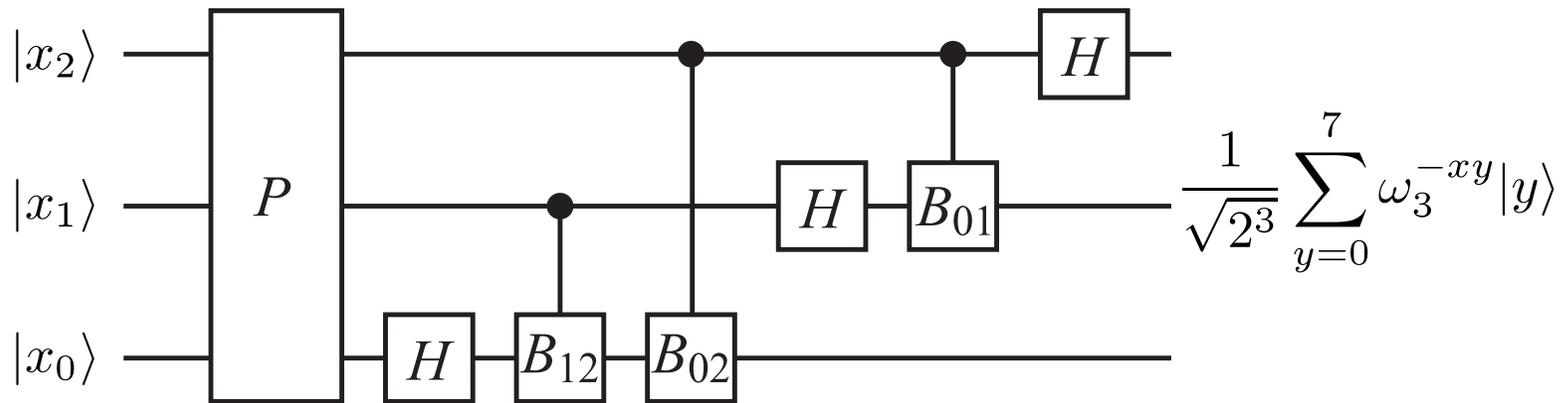
FIGURE 6.3

Implementation of the $n = 3$ QFT.

Exercise

EXERCISE 6.5 Let $x = 2^2 x_2 + 2x_1 + x_0$ and $y = 2^2 y_2 + 2y_1 + y_0$.
(1) Write down the RHS of

$$U_{\text{QFT3}}|x_2 x_1 x_0\rangle = \frac{1}{\sqrt{2^3}} \sum_{y=0}^{2^3 - 1} e^{-2\pi i x y / 2^3} |y\rangle \qquad (6.37)$$

explicitly in terms of x_i and y_i.
(2) Show that the RHS of Eq. (6.37) agrees with the first line of the RHS of Eq. (6.36).

Since Eq. (6.36) is true for any $|x_2 x_1 x_0\rangle$, we have found

$$U_{\text{QFT3}} = (U_{\text{H}} \otimes I \otimes I) U_{01} (I \otimes U_{\text{H}} \otimes I) U_{02} U_{12} (I \otimes I \otimes U_{\text{H}}) P. \qquad (6.38)$$

Circuit implementation of QFT: n general

Now the generalization of the present construction to $n \geq 4$ should be easy. The equation that generalizes Eq. (6.36) is

$$U_{\mathrm{QFT}n}|x_{n-1} \ldots x_1 x_0\rangle$$

$$= \frac{1}{\sqrt{N}}(|0\rangle + e^{-2\pi i x_0/2}|1\rangle) \otimes (|0\rangle + e^{-2\pi i(x_1/2+x_0/2^2)}|1\rangle)$$

$$\otimes(|0\rangle + e^{-2\pi i(x_2/2+x_1/2^2+x_0/2^3)}|1\rangle) \otimes \ldots$$

$$\ldots \otimes (|0\rangle + e^{-2\pi i(x_{n-1}/2+x_{n-2}/2^2+\ldots x_1/2^{n-1}+x_0/2^n)}|1\rangle)$$

$$= (U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{02}U_{12}$$

$$\times(I \otimes I \otimes U_{\mathrm{H}} \otimes \ldots \otimes I) \ldots$$

$$\times U_{0,n-1}U_{1,n-1} \ldots U_{n-2,n-1}(I \otimes \ldots \otimes I \otimes U_{\mathrm{H}})|x_0 x_1 \ldots x_{n-1}\rangle$$

$$= (U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{02}U_{12}$$

$$\times(I \otimes I \otimes U_{\mathrm{H}} \otimes \ldots \otimes I) \ldots U_{0,n-1}U_{1,n-1} \ldots U_{n-2,n-1}$$

$$\times(I \otimes \ldots \otimes I \otimes U_{\mathrm{H}})P|x_{n-1} \ldots x_1 x_0\rangle, \qquad (6.39)$$
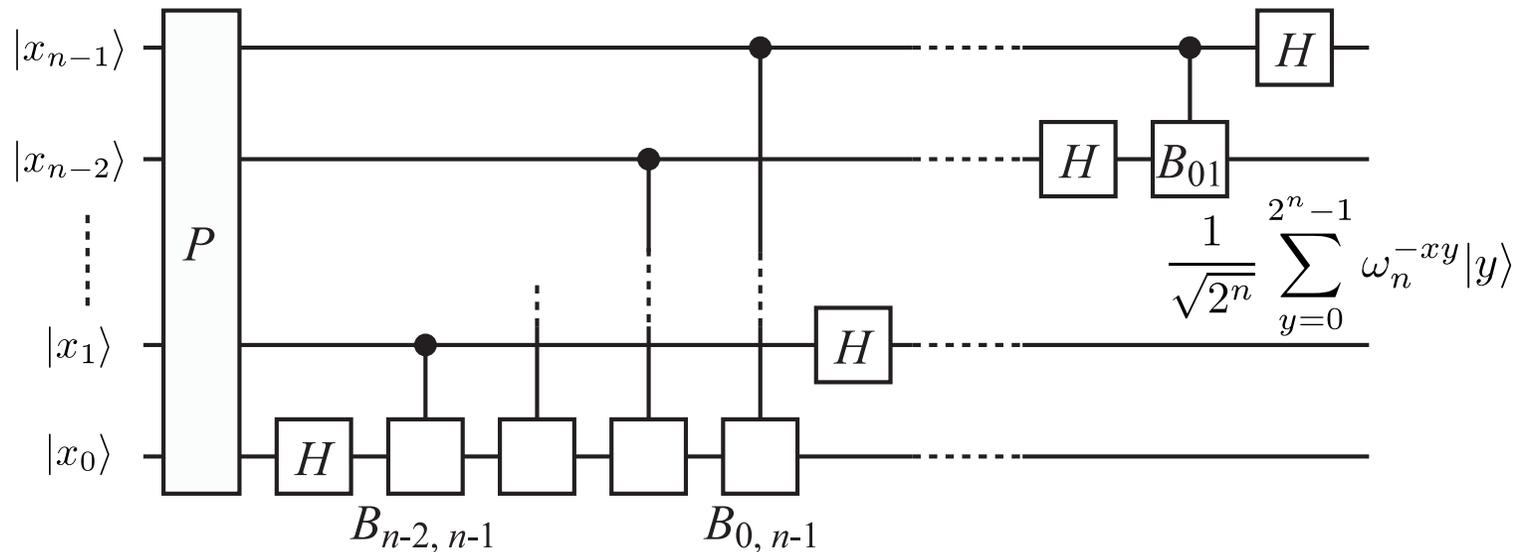
where P reverses the order of x_k as $P|x_{n-1} \ldots x_1 x_0\rangle = |x_0 x_1 \ldots x_{n-1}\rangle$.

Circuit implementation of QFT: n general

We finally find the following decompostion of $U_{\mathrm{QFT}n}$:

$$U_{\mathrm{QFT}n} = (U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{01}(I \otimes U_{\mathrm{H}} \otimes I \otimes \ldots \otimes I)U_{02}U_{12}$$
$$\times (I \otimes I \otimes U_{\mathrm{H}} \otimes \ldots \otimes I) \ldots$$
$$\times U_{0,n-1}U_{1,n-1}\ldots U_{n-2,n-1}(I \otimes \ldots \otimes I \otimes U_{\mathrm{H}})P. \qquad (6.40)$$

A quantum circuit which implements $U_{\mathrm{QFT}n}$ is found from Eq. (6.40) as in Fig. 6.4. It may be proved, by induction, for example, that the circuit in

$$\frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} \omega_n^{-xy}|y\rangle$$

Circuit implementation of QFT: n general

PROPOSITION 6.4 The n-qubit QFT may be constructed with $\Theta(n^2)$ elementary gates.

Proof. The n-qubit QFT is made of a P gate, n Hadamard gates and $(n-1) + (n-2) + \ldots + 2 + 1 = n(n-1)/2$ controlled-B_{jk} gates (see Fig. 6.4). It has been shown in §4.2.3 that it requires three CNOT gates to construct a SWAP gate. Furthermore, a P gate for n qubits requires $\lfloor n/2 \rfloor$ SWAP gates,[†] assuming that there exists a SWAP gate for any pair of qubits. Thus a P gate requires $3 \times \lfloor n/2 \rfloor = \Theta(n)$ elementary gates. Proposition 4.1 states that a controlled-B_{ij} gate is constructed with at most six elementary gates. Thus it has been proved that the n-qubit QFT is made of $\Theta(n^2)$ elementary gates.
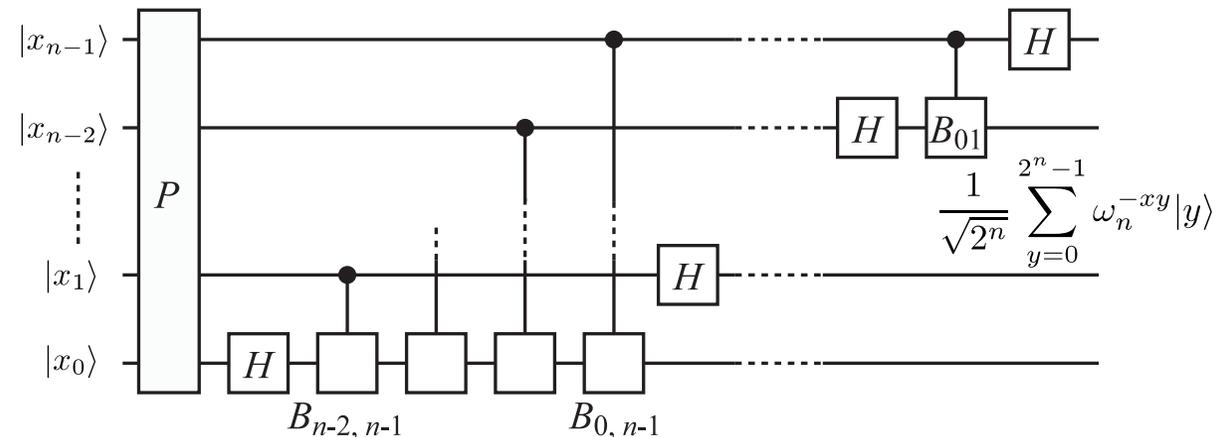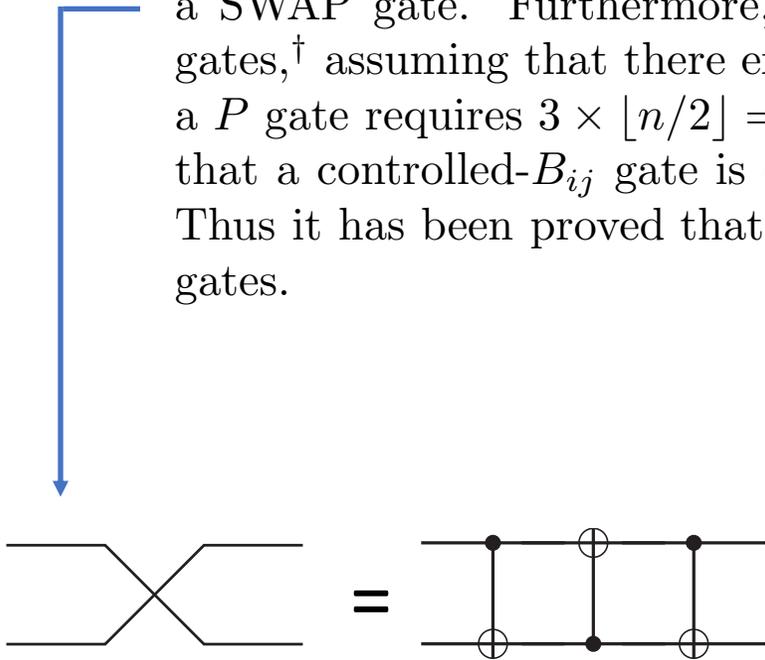
Walsh Hadamard Transform

We have already encountered the Walsh-Hadamard transform in §4.2.2 and §5.2. Let $x, y \in S_n = \{0, 1, \dots, N-1\}$ with binary expressions $x_{n-1}x_{n-2}\dots x_0$ and $y_{n-1}y_{n-2}\dots y_0$, where $N = 2^n$. The Walsh-Hadamard transform, written in the form of Eq. (5.7), shows that it is a quantum integral transform with a kernel $W_n : S_n \times S_n \to \mathbb{C}$ defined by

$$W_n(x, y) = \frac{1}{\sqrt{N}}(-1)^{x \cdot y} \quad (x, y \in S_n), \tag{6.41}$$

where $x \cdot y = x_{n-1}y_{n-1} \oplus x_{n-2}y_{n-2} \oplus \dots \oplus x_0 y_0$. This kernel defines a discrete integral transform

$$\tilde{f}(y) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} (-1)^{x \cdot y} f(x). \tag{6.42}$$
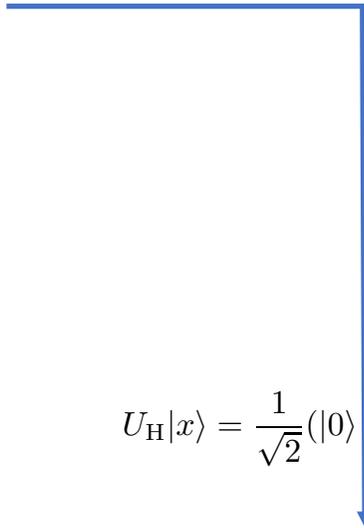
$$U_{\mathrm{H}}|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^x |1\rangle) = \frac{1}{\sqrt{2}} \sum_{y \in \{0,1\}} (-1)^{xy}|y\rangle,$$

$$\begin{aligned}
W_n |x\rangle &= (U_{\mathrm{H}}|x_{n-1}\rangle)(U_{\mathrm{H}}|x_{n-2}\rangle) \dots (U_{\mathrm{H}}|x_0\rangle) \\
&= \frac{1}{\sqrt{2^n}} \sum_{y_{n-1}, y_{n-2}, \dots, y_0 \in \{0,1\}} (-1)^{x_{n-1}y_{n-1} + x_{n-2}y_{n-2} + \dots + x_0 y_0} \\
&\quad \times |y_{n-1}y_{n-2}\dots y_0\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n - 1} (-1)^{x \cdot y}|y\rangle, \tag{5.7}
\end{aligned}$$

Selective Phase Rotation Transform

DEFINITION 6.2 (Selective Phase Rotation Transform) Let us define a kernel

$$K_n(x, y) = e^{i\theta_x}\delta_{xy}, \quad \forall x, y \in S_n, \tag{6.43}$$

where $\theta_x \in \mathbb{R}$. The discrete integral transform

$$\tilde{f}(y) = \sum_{x=0}^{N-1} K(x, y)f(x) = \sum_{x=0}^{N-1} e^{i\theta_x}\delta_{xy}f(x) = e^{i\theta_y}f(y) \tag{6.44}$$

with the kernel K_n is called the **selective phase rotation transform**.

EXERCISE 6.7 Show that K_n defined above is unitary. Write down the inverse transformation K_n^{-1}.

Selective Phase Rotation Transform

The matrix representations for K_1 and K_2 are

$$K_1 = \begin{pmatrix} e^{i\theta_0} & 0 \\ 0 & e^{i\theta_1} \end{pmatrix}, \quad K_2 = \begin{pmatrix} e^{i\theta_0} & 0 & 0 & 0 \\ 0 & e^{i\theta_1} & 0 & 0 \\ 0 & 0 & e^{i\theta_2} & 0 \\ 0 & 0 & 0 & e^{i\theta_3} \end{pmatrix}.$$

Selective Phase Rotation Transform

The implementation of K_n is achieved with the universal set of gates as follows. Take $n = 2$, for example. The kernel K_2 has been given above. This is decomposed as a product of two two-level unitary matrices as

$$K_2 = A_0 A_1, \qquad (6.45)$$

where

$$A_0 = \begin{pmatrix} e^{i\theta_0} & 0 & 0 & 0 \\ 0 & e^{i\theta_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, A_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{i\theta_2} & 0 \\ 0 & 0 & 0 & e^{i\theta_3} \end{pmatrix}. \qquad (6.46)$$

Note that

$$A_0 = |0\rangle\langle 0| \otimes U_0 + |1\rangle\langle 1| \otimes I, \quad U_0 = \begin{pmatrix} e^{i\theta_0} & 0 \\ 0 & e^{i\theta_1} \end{pmatrix},$$

$$A_1 = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U_1, \quad U_1 = \begin{pmatrix} e^{i\theta_2} & 0 \\ 0 & e^{i\theta_3} \end{pmatrix}.$$

Selective Phase Rotation Transform

Thus A_1 is realized as an ordinary controlled-U_1 gate while the control bit is negated in A_0. Then what we have to do for A_0 is to negate the control bit first and then to apply ordinary controlled-U_0 gate and finally to negate the control bit back to its input state. In summary, A_0 is implemented as in Fig. 6.5. In fact, it can be readily verified that the gate in Fig. 6.5 is written as

$$A_0 = |0\rangle\langle 0| \otimes U_0 + |1\rangle\langle 1| \otimes I,$$

$$A_1 = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U_1,$$

$$(X \otimes I)(|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes U_0)(X \otimes I)$$
$$= X|0\rangle\langle 0|X \otimes I + X|1\rangle\langle 1|X \otimes U_0 = |1\rangle\langle 1| \otimes I + |0\rangle\langle 0| \otimes U_0 = A_0.$$

Thus these gates are implemented with the set of universal gates. In fact, the order of A_i does not matter since $[A_0, A_1] = 0$.
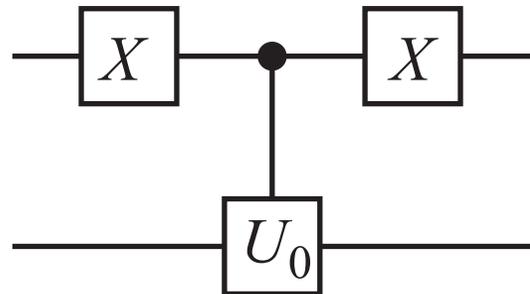
Back on Grover's search algorithm

We need to prove that the D gate used to perform the quantum search can be implemented efficiently. We now show that

$$D = W_n R_0 W_n, \tag{7.6}$$

where W_n is the Walsh-Hadamard transform,

$$W_n(x, y) = \frac{1}{\sqrt{N}}(-1)^{x \cdot y}, \quad (x, y \in S_n) \tag{7.7}$$

and R_0 is the selective phase rotation transform defined by

$$R_0(x, y) = e^{i\pi(1-\delta_{x0})}\delta_{xy} = (-1)^{1-\delta_{x0}}\delta_{xy}. \tag{7.8}$$

Back on Grover's search algorithm

Proof

$$\langle x|D|y\rangle = \langle x|\left[-I + 2|\varphi_o\rangle\langle\varphi_0|\right]|y\rangle = -\delta_{xy} + \frac{2}{N}$$

$$|\varphi_0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle$$

$$\langle x|W_n R_0 W_n|y\rangle = \sum_{u,v}\langle x|W_n|u\rangle\langle u|R_0|v\rangle\langle v|W_n|y\rangle = \frac{1}{N}\sum_{u,v}(-1)^{x\cdot u}(-1)^{1-\delta_{u0}}\delta_{uv}(-1)^{v\cdot y}.$$

$$= \frac{1}{N}\sum_{u}(-1)^{x\cdot u}(-1)^{y\cdot u}(-1)^{1-\delta_{u0}}$$

$$= \frac{1}{N}\left[1 - \sum_{u\neq 0}(-1)^{x\cdot u}(-1)^{y\cdot u}\right]$$

Back on Grover's search algorithm

$$\frac{1}{N}\left[1 - \sum_{u \neq 0}(-1)^{x \cdot u}(-1)^{y \cdot u}\right] = A$$

x = y:
$$A = \frac{1}{N}\left[1 - \sum_{u \neq 0}\right] = \frac{1}{N}\left[1 - (N-1)\right] = -1 + \frac{2}{N}$$

x ≠ y. As discussed in relation to the Deutsch-Jozsa algorithm

$$\sum_{u=0}^{N-1}(-1)^{x \cdot u} = 0 \;\rightarrow\; \sum_{u \neq 0}^{N-1}(-1)^{x \cdot u} = -1$$

Therefore:
$$A = \frac{1}{N}\left[1 - (-1)\right] = \frac{2}{N}$$

Back on Grover's search algorithm

Therefore the D gate can be implemented efficiently. The overall circuit is
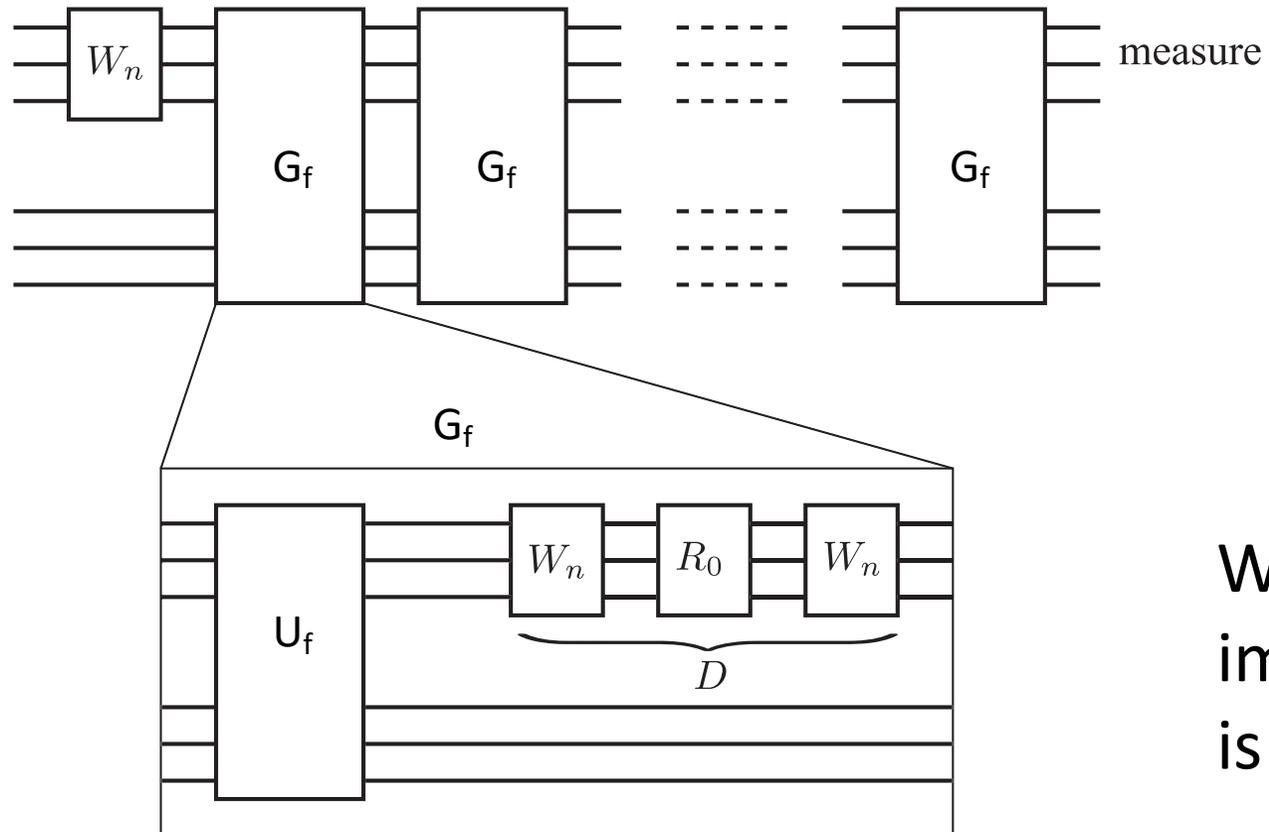
We are not interested on how to implement the oracle U_f since this is supposed to be given

Shor's factorization algorithm

Shor's algorithm is a polynomial-time quantum computer algorithm for integer factorization. It solves the following problem: Given an integer N, find its prime factors. It was invented in 1994 by Peter Shor.

Shor's algorithm consists of two parts:
1. A reduction, which can be done on a classical computer, of the factoring problem to the problem of **order-finding**.
2. A quantum algorithm to solve the order-finding problem.

The first part can be done easily. We will see the second part.

Order finding – the problem

Number to factorize

Define $f_N : \mathbb{N} \to \mathbb{N}$ by $a \mapsto m^a \bmod N$. Find the smallest $P \in \mathbb{N}$, such that $m^P \equiv 1 \bmod N$. The number P is called the **order** or **period**. It is known that this takes exponentially large steps in any classical algorithm, but it takes only polynomial steps in Shor's algorithm. A quantum computer is required only in this step, and the rest may be executed in polynomial steps even with a classical computer.

Order finding – the quantum solution

Our quantum computer has two n-qubit registers which we call $|\mathrm{REG1}\rangle$ and $|\mathrm{REG2}\rangle$:

$$|\mathrm{REG1}\rangle|\mathrm{REG2}\rangle = |a\rangle|b\rangle = |a_{n-1}\ldots a_1 a_0\rangle|b_{n-1}\ldots b_1 b_0\rangle, \qquad (8.7)$$

where decimal numbers $a, b \in S_n$ are expressed in binary numbers in the RHS;

$$a = \sum_{j=0}^{n-1} a_j 2^j, \;\; b = \sum_{j=0}^{n-1} b_j 2^j.$$

Step 0. Set the registers to the initial state

$$|\psi_0\rangle = |\mathrm{REG1}\rangle|\mathrm{REG2}\rangle = |\underbrace{00\ldots0}_{n \text{ qubits}}\rangle|\underbrace{00\ldots0}_{n \text{ qubits}}\rangle. \qquad (8.9)$$

Order finding – the quantum solution

Step 1. The QFT $\mathcal{F}$ is applied on the first register;

$$|\psi_0\rangle = |0\rangle|0\rangle \overset{\mathcal{F}\otimes I}{\mapsto} |\psi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|0\rangle. \qquad (8.10)$$

The first register is in a superposition of all the states $|x\rangle$ $(0 \leq x \leq Q-1)$,

with $Q = 2^n$. Remember that QFT on all |0>'s gives the equal weighted superposition of all computational basis states

Order finding – the quantum solution

Step 2. Let us define a function f :

$$f(x) = m^x \bmod N, \quad x \in S_n = \{0, 1, \ldots, Q-1\} \quad (8.11)$$

Suppose that the unitary gate U_f realizes the action of f on x in such a way that $U_f |x\rangle |0\rangle = |x\rangle |f(x)\rangle$. Apply U_f on the state prepared in step 2.1 to yield

$$U_f |\psi_1\rangle = |\psi_2\rangle \equiv \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |f(x)\rangle. \quad (8.12)$$

Order finding – the quantum solution

Step 3.   Apply QFT on $|\mathrm{REG1}\rangle$ again to yield

$$|\psi_3\rangle = (\mathcal{F} \otimes I)|\psi_2\rangle = \frac{1}{Q}\sum_{x=0}^{Q-1}\sum_{y=0}^{Q-1}\omega_n^{-xy}|y\rangle|f(x)\rangle$$

$$= \frac{1}{Q}\sum_{y=0}^{Q-1}|y\rangle|\Upsilon(y)\rangle = \frac{1}{Q}\sum_{y=0}^{Q-1}\big\|\,|\Upsilon(y)\rangle\,\big\| \cdot |y\rangle\frac{|\Upsilon(y)\rangle}{\big\|\,|\Upsilon(y)\rangle\,\big\|}, \qquad (8.13)$$

where

$$|\Upsilon(y)\rangle = \sum_{x=0}^{Q-1}\omega_n^{-xy}|f(x)\rangle. \qquad (8.14)$$

Order finding – the quantum solution

Step 4. $|\text{REG1}\rangle$ is measured. The result $y \in S_n$ is obtained with the probability

$$\text{Prob}(y) = \frac{\||\Upsilon(y)\rangle\|^2}{Q^2}, \tag{8.15}$$

and, at the same time, the state collapses to

$$|y\rangle \frac{|\Upsilon(y)\rangle}{\||\Upsilon(y)\rangle\|}.$$

The measurement process generates a random variable following a classical probability distribution $\mathcal{S}$ over S_n, in which "symbols" $y \in S_n$ are generated with the probability (8.15).

Step 5. Extract the order P from the measurement outcome.

Order finding – the quantum solution

P is what we want to find

PROPOSITION 8.1 Let $Q = 2^n = Pq + r$, $(0 \leq r < P)$, where q and r are uniquely determined non-negative integers. Let $Q_0 = Pq$. Then

$$\text{Prob}(y) = \begin{cases} \dfrac{r\sin^2\left(\frac{\pi Py}{Q}\left(\frac{Q_0}{P}+1\right)\right) + (P-r)\sin^2\left(\frac{\pi Py}{Q}\cdot\frac{Q_0}{P}\right)}{Q^2\sin^2\left(\frac{\pi Py}{Q}\right)} & (Py \not\equiv 0 \bmod Q) \\[6mm] \dfrac{r(Q_0+P)^2 + (P-r)Q_0^2}{Q^2P^2} & (Py \equiv 0 \bmod Q). \end{cases}$$

Proof. It is found from the definition that[¶]

$$|\Upsilon(y)\rangle = \sum_{x=0}^{Q-1} \omega^{-xy}|f(x)\rangle = \sum_{x=0}^{Q_0-1} \omega^{-xy}|f(x)\rangle + \sum_{x=Q_0}^{Q-1} \omega^{-xy}|f(x)\rangle$$

Definition + splitting the sum

$$= \sum_{x_0=0}^{P-1} \sum_{x_1=0}^{Q_0/P-1} \omega^{-(Px_1+x_0)y}|f(Px_1+x_0)\rangle$$

$x = Px_1 + x_0$

$$+ \sum_{x_0=0}^{r-1} \omega^{-[P(Q_0/P)+x_0]y}|f(P(Q_0/P)+x_0)\rangle$$

$$= \sum_{x_0=0}^{P-1} \omega^{-x_0 y} \left(\sum_{x_1=0}^{Q_0/P-1} \omega^{-Px_1 y} \right) |f(x_0)\rangle + \sum_{x_0=0}^{r-1} \omega^{-x_0 y} \omega^{-Py(Q_0/P)}|f(x_0)\rangle$$

$$= \sum_{x_0=0}^{r-1} \omega^{-x_0 y} \sum_{x_1=0}^{Q_0/P-1} \omega^{-Pyx_1}|f(x_0)\rangle$$

$$+ \sum_{x_0=r}^{P-1} \omega^{-x_0 y} \sum_{x_1=0}^{Q_0/P-1} \omega^{-Pyx_1}|f(x_0)\rangle + \sum_{x_0=0}^{r-1} \omega^{-x_0 y} \omega^{-Py(Q_0/P)}|f(x_0)\rangle$$

Splitting the sum

$$= \sum_{x_0=0}^{r-1} \omega^{-x_0 y} \left(\sum_{x_1=0}^{Q_0/P} \omega^{-Pyx_1} \right) |f(x_0)\rangle$$
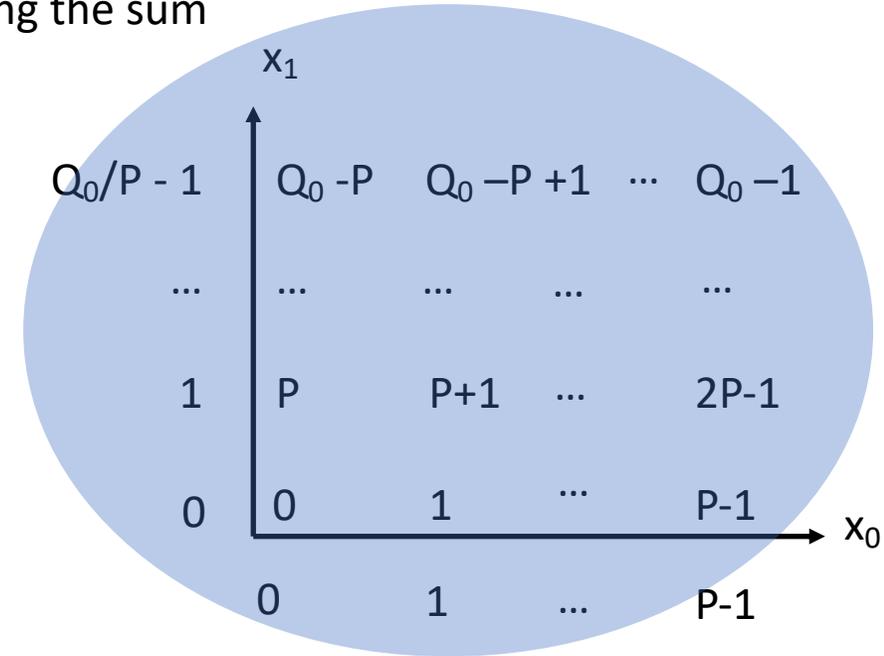
$$+ \sum_{x_0=r}^{P-1} \omega^{-x_0 y} \left(\sum_{x_1=0}^{Q_0/P-1} \omega^{-Pyx_1} \right) |f(x_0)\rangle.$$

Merges the two sums

So far we have

$$|\Upsilon(y)\rangle = \sum_{x_0=0}^{r-1} \omega^{-x_0 y} \left(\sum_{x_1=0}^{Q_0/P} \omega^{-Pyx_1} \right) |f(x_0)\rangle + \sum_{x_0=r}^{P-1} \omega^{-x_0 y} \left(\sum_{x_1=0}^{Q_0/P-1} \omega^{-Pyx_1} \right) |f(x_0)\rangle.$$

Note that the map $f : a \mapsto m^a \bmod N$ is $1 : 1$ on $\{0, 1, 2, \ldots, P-1\}$

This implies that $|f(0)\rangle, |f(1)\rangle, \ldots, |f(P-1)\rangle$ are mutually orthogonal. Accordingly

$$\langle \Upsilon(y) | \Upsilon(y) \rangle = r \left| \sum_{x_1=0}^{Q_0/P} \omega^{-Pyx_1} \right|^2 + (P-r) \left| \sum_{x_1=0}^{Q_0/P-1} \omega^{-Pyx_1} \right|^2.$$

In case $Py \equiv 0 \bmod Q$, we put $Py = aQ,\ a \in \mathbb{N}$ and obtain

$$\omega^{-Pyx_1} = e^{-2\pi i (Py/Q)x_1} = e^{-2\pi i a x_1} = 1.$$

Therefore

$$\langle \Upsilon(y) | \Upsilon(y) \rangle = r \cdot \left(\frac{Q_0}{P} + 1 \right)^2 + (P - r) \left(\frac{Q_0}{P} \right)^2,$$

which leads to the result independent of y,

$$\mathrm{Prob}(y) = \frac{r(Q_0 + P)^2 + (P - r)Q_0^2}{P^2 Q^2} = \frac{r(q + 1)^2 + (P - r)q^2}{Q^2}. \qquad (8.16)$$

If $Py \not\equiv 0 \bmod Q$, on the other hand, we obtain

$$\langle \Upsilon(y)|\Upsilon(y)\rangle = r \left| \frac{\omega^{-Py(Q_0/P+1)} - 1}{\omega^{-Py} - 1} \right|^2 + (P-r) \left| \frac{\omega^{-Py(Q_0/P)} - 1}{\omega^{-Py} - 1} \right|^2$$

$$= r \left| \frac{e^{-(2\pi i/Q)Py(Q_0/P+1)} - 1}{e^{-(2\pi i/Q)Py} - 1} \right|^2 + (P-r) \left| \frac{e^{-(2\pi i/Q)Py(Q_0/P)} - 1}{e^{-(2\pi i/Q)Py} - 1} \right|^2$$

Here we find from

$$|e^{i\theta} - 1|^2 = 2(1 - \cos\theta) = 4 \sin^2 \frac{\theta}{2}$$

that

$$\langle \Upsilon(y)|\Upsilon(y)\rangle = r \frac{\sin^2 \frac{\pi}{Q} Py \left(\frac{Q_0}{P} + 1 \right)}{\sin^2 \frac{\pi}{Q} Py} + (P-r) \frac{\sin^2 \frac{\pi}{Q} Py \frac{Q_0}{P}}{\sin^2 \frac{\pi}{Q} Py}.$$

Therefore, the probability distribution is given by

$$\mathrm{Prob}(y) = \frac{\||\Upsilon(y)\rangle\|^2}{Q^2} = \frac{r \sin^2 \left[\frac{\pi}{Q} Py \left(\frac{Q_0}{P} + 1 \right) \right] + (P-r) \sin^2 \left[\frac{\pi}{Q} Py \frac{Q_0}{P} \right]}{Q^2 \sin^2 \frac{\pi}{Q} Py},$$

$$(8.17)$$

which proves the proposition. ∎

COROLLARY 8.1 Suppose $Q/P \in \mathbb{Z}$ (namely $Q_0 = Q$). Then the probability of obtaining a measurement outcome y is

$$\text{Prob}(y) = \begin{cases} 0 & (Py \not\equiv 0 \bmod Q) \\ \dfrac{1}{P} & (Py \equiv 0 \bmod Q) \end{cases}$$

$\longrightarrow$ **r = 0**

$\longrightarrow$ Peaks are repeated at distance q, because we are in the first situation until y = q

Proof. When $Py \not\equiv 0 \bmod Q$, $r = 0$ implies $Q = Pq$. Therefore

$$\text{Prob}(y) = \frac{P \sin^2 \pi y}{Q^2 \sin^2 \frac{\pi y}{q}} = 0.$$

In case $Py \equiv 0 \bmod Q$, we obtain

$$\text{Prob}(y) = \frac{PQ^2}{Q^2 P^2} = \frac{1}{P}.$$

Factoring 15 (Credits: Dr. G. Crognaletti)

N = 15.

m = 7.

Quindi: $f(x) = 7^x \bmod 15$

Il numero di qubit n è stabilito da $2 \log_2(N) < n < 2 \log_2(N)+1$, in questo caso $7.8 < n < 8.8 \Rightarrow n = 8$, necessito di $2^8 = 256$ ampiezze di probabilità.

$$|R_0\rangle = |0\rangle^{\otimes q} \quad\boxed{H^{\otimes q}} \quad \begin{array}{c} a \\ \\ U_{\mathrm{mod\ exp}} \\ \\ t \end{array} \quad \begin{array}{c} a \\ \\ \\ \\ t \oplus x^a \bmod N \end{array} \quad \boxed{QFT} \quad \boxed{\nearrow} \quad |c\rangle$$

$$|R_1\rangle = |0\rangle^{\otimes q} \qquad\qquad\qquad\qquad\qquad\qquad \boxed{\nearrow} \quad |k\rangle$$
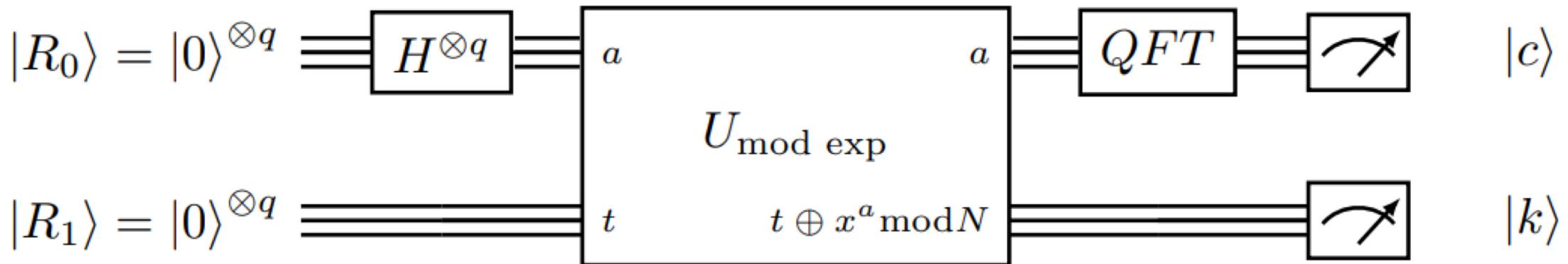
Factoring 15

Ad ogni stato della macchina è associato
un istogramma di questo tipo:

- Il primo asse rappresenta la base
  computaizionale del primo registro, i cui
  valori verranno indicati con c.
- Il secondo rappresenta la base
  computazionale del secondo, limitato ai
  valori ottenuti nell pratica (in questo
  caso 13), i cui valori verranno indicati
  con k.
- L'asse verticale rappresenta la
  probabilità di misura P(c,k) associata ad
  ogni elemento della base della coppia di
  registri. Es: lo stato iniziale
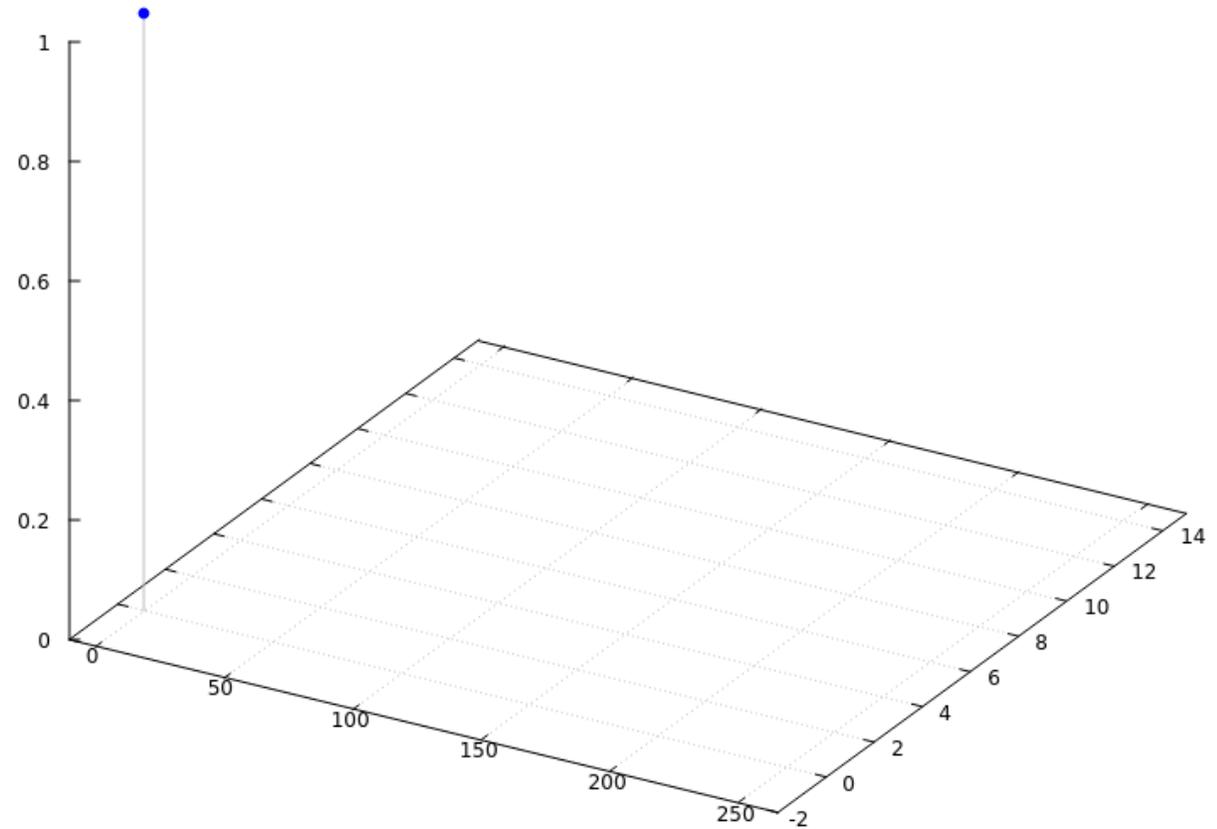
$$|R_0\rangle |R_1\rangle = |\mathbf{00...0}\rangle_8 |\mathbf{00...0}\rangle_8 = |0\rangle |0\rangle$$

Factoring 15

1. **Trasformata di Hadamard**

Creo lo stato sovrapposto di tutta la base computazionale. Ciò richiede in totale n operazioni (applicazione di H ad ognuno dei Qubit)
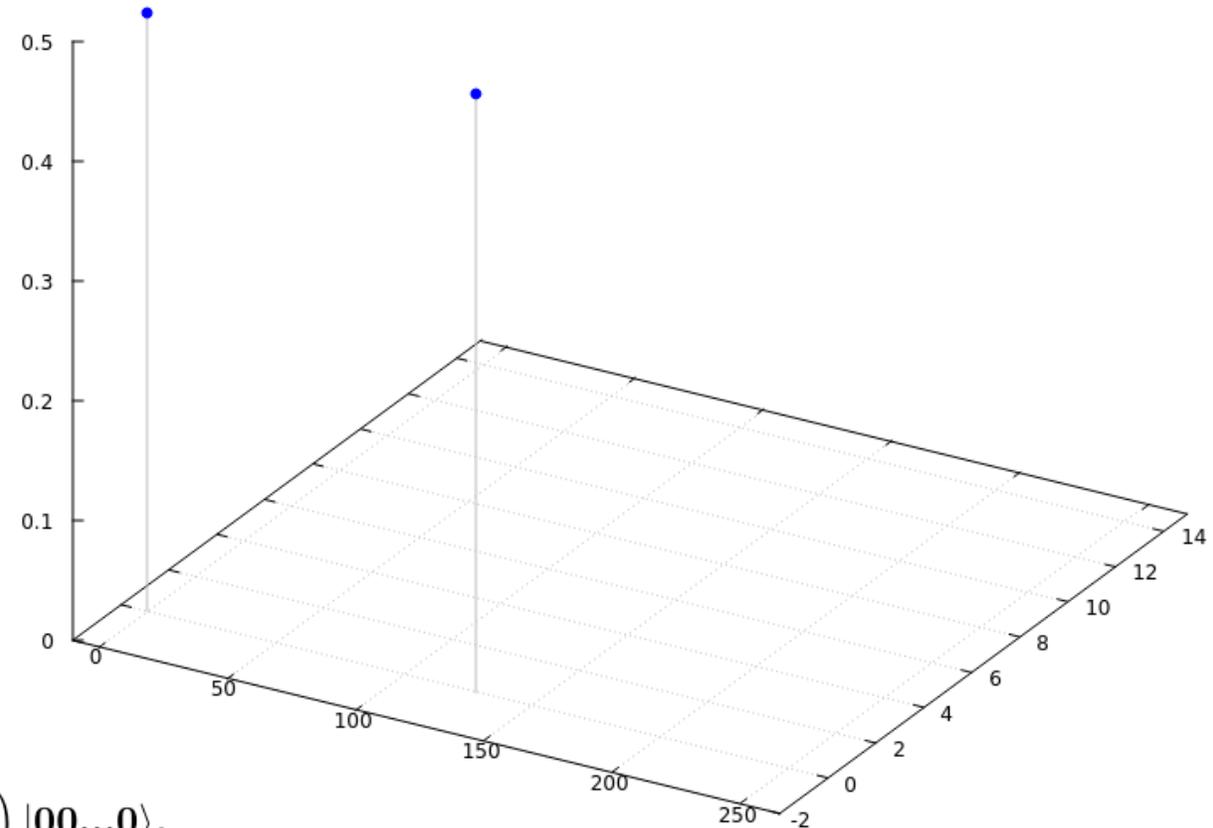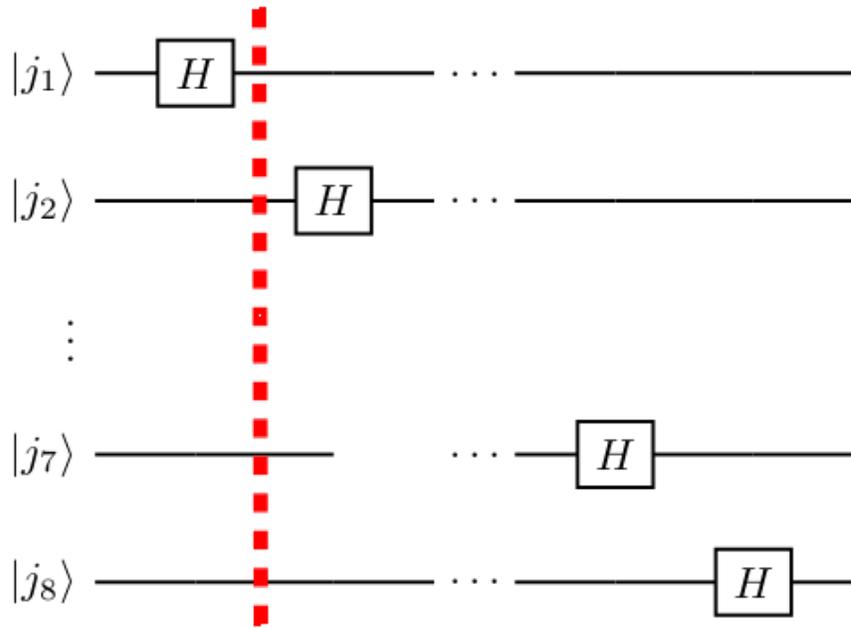
$$|R_0\rangle\,|R_1\rangle = \left[\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)|00...0\rangle_7\right]|00...0\rangle_8 = \frac{1}{\sqrt{2}}\left(|00..0\rangle_8 + |10...0\rangle_8\right)|00...0\rangle_8$$

=0   =128

Factoring 15

1. Trasformata di Hadamard

$$|R_0\rangle = |0\rangle^{\otimes q} \equiv \boxed{H^{\otimes q}}$$

$$|R_1\rangle = |0\rangle^{\otimes q} \equiv$$

$U_{\text{mod exp}}$

a ... a

t ... $t \oplus x^a \bmod N$
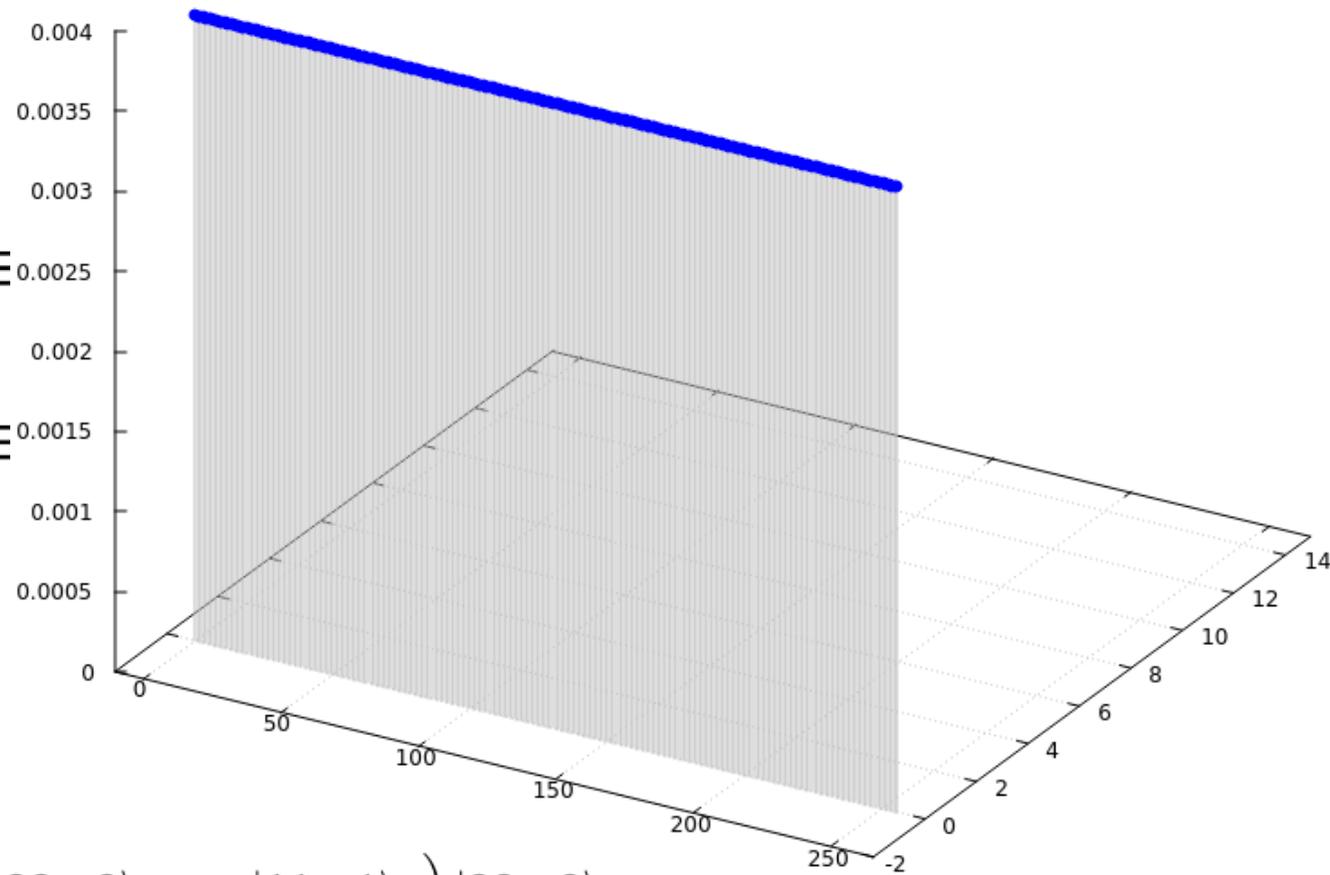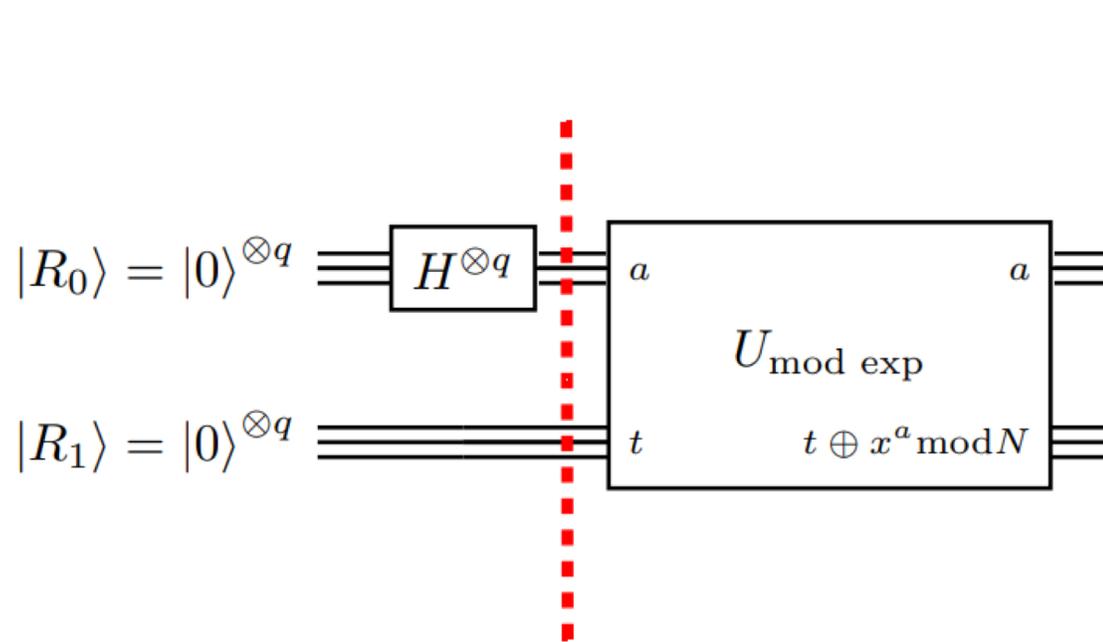
$$|R_0\rangle\,|R_1\rangle = \left[\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)\cdots\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)\right] |00...0\rangle_8 = \frac{1}{\sqrt{256}}\left(|00...0\rangle_8 + ...\, |11...1\rangle_8\right) |00...0\rangle_8$$

$=0 \qquad =255$

Factoring 15

2. Applico l'operatore esponenziale modulare U_f

$$|R_0\rangle |R_1\rangle = \frac{1}{\sqrt{256}} \Big(|0\rangle |1\rangle + |1\rangle |7\rangle + |2\rangle |4\rangle + |3\rangle |13\rangle + |4\rangle |1\rangle + |5\rangle |7\rangle \dots |255\rangle |13\rangle \Big)$$

È uno stato non separabile, descrivibile come sovrapposizione con uguale ampiezza di probabilità di 4 stati separabili

$$|R_0\rangle |R_1\rangle = \frac{1}{\sqrt{4}} \left(\frac{|0\rangle + |4\rangle + \dots + |252\rangle}{\sqrt{64}} \right) |1\rangle + \frac{1}{\sqrt{4}} \left(\frac{|1\rangle + |5\rangle + \dots + |253\rangle}{\sqrt{64}} \right) |7\rangle$$

$$+ \frac{1}{\sqrt{4}} \left(\frac{|2\rangle + |6\rangle + \dots + |254\rangle}{\sqrt{64}} \right) |4\rangle + \frac{1}{\sqrt{4}} \left(\frac{|3\rangle + |7\rangle + \dots + |255\rangle}{\sqrt{64}} \right) |13\rangle$$
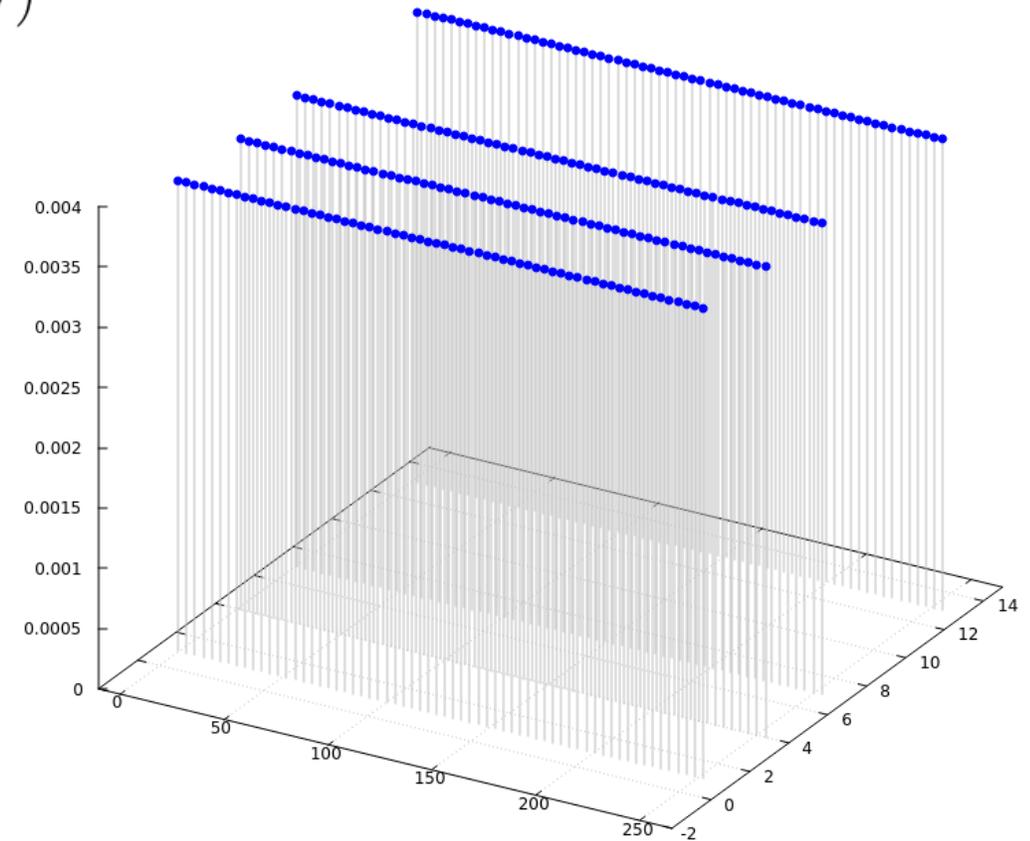
Factoring 15

3. Applico la Trasformata di Fourier Quantistica

L'algoritmo utilizzato in questo caso è quello relativo alla QFT esatta, schematizzato dal circuito:
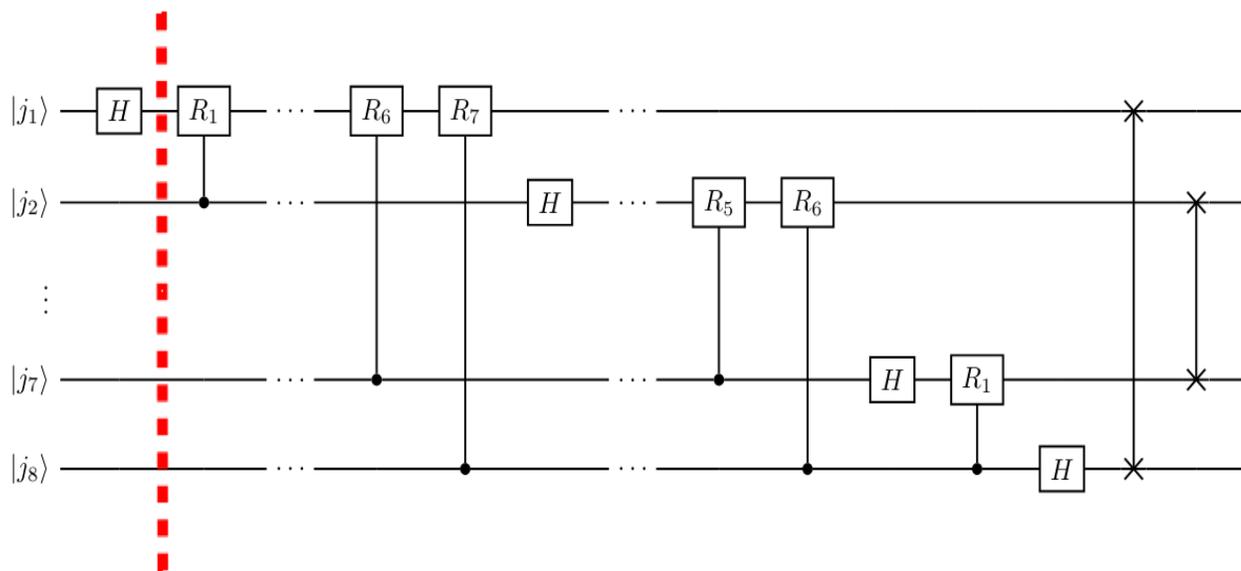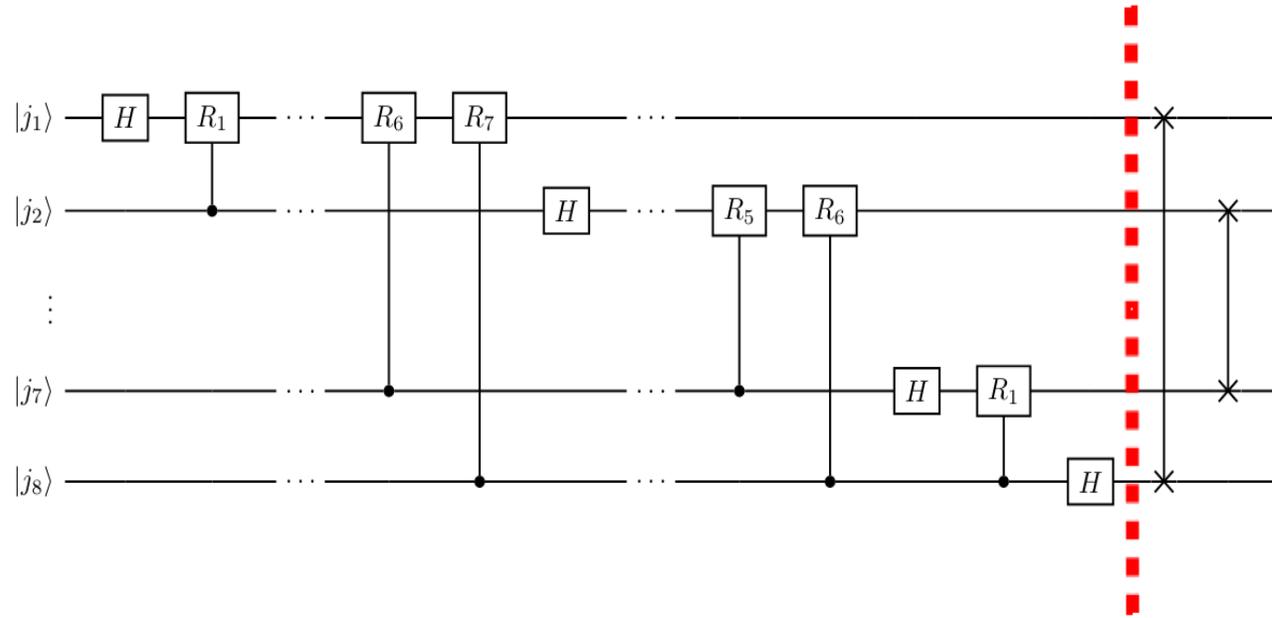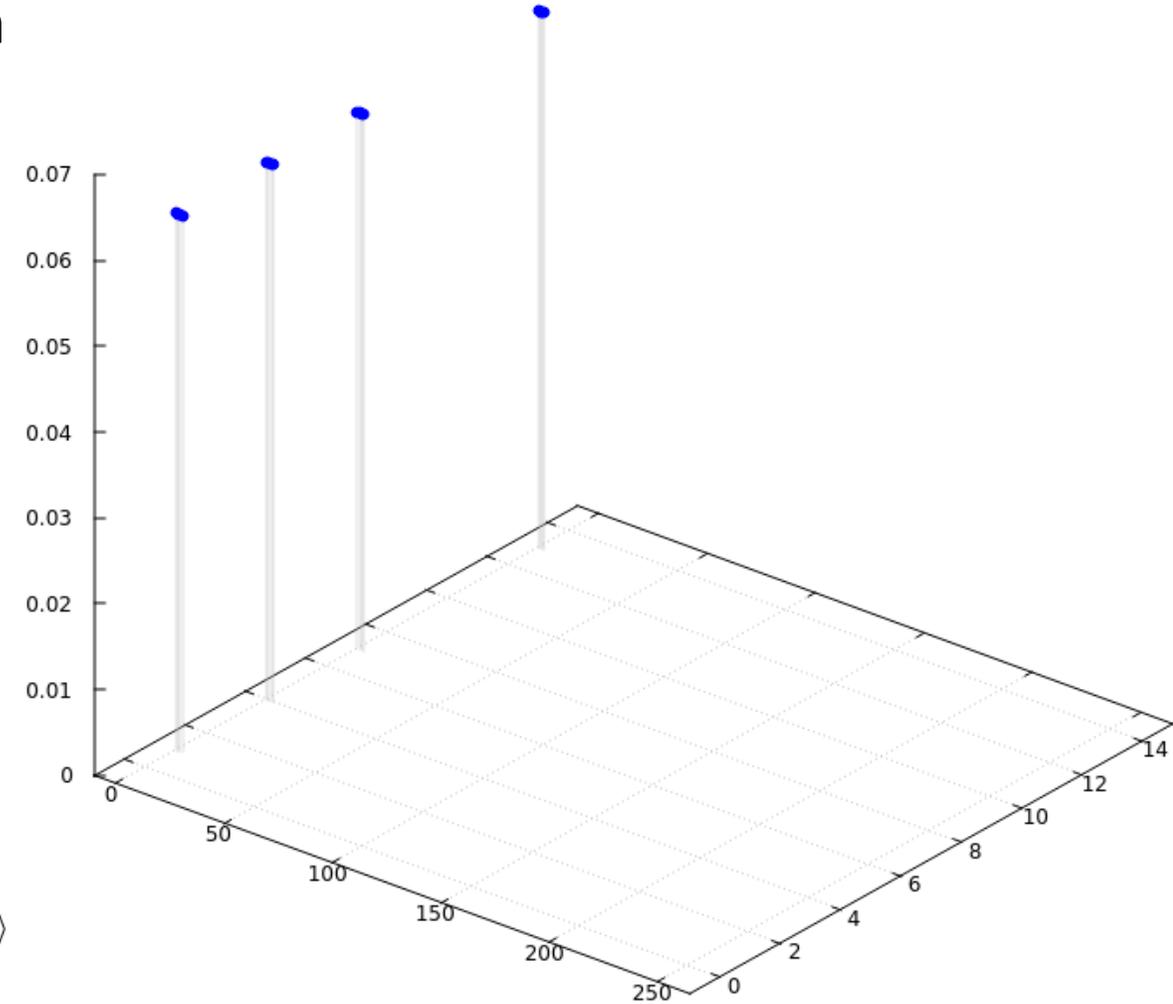

$$|R_0\rangle |R_1\rangle = \frac{1}{\sqrt{4}} \left(\frac{|0\rangle + |4\rangle + ... + |124\rangle}{\sqrt{32}} \right) |1\rangle + \frac{1}{\sqrt{4}} \left(\frac{|1\rangle + |5\rangle + ... + |125\rangle}{\sqrt{32}} \right) |7\rangle$$

$$+ \frac{1}{\sqrt{4}} \left(\frac{|2\rangle + |6\rangle + ... + |126\rangle}{\sqrt{32}} \right) |4\rangle + \frac{1}{\sqrt{4}} \left(\frac{|3\rangle + |7\rangle + ... + |127\rangle}{\sqrt{32}} \right) |13\rangle$$
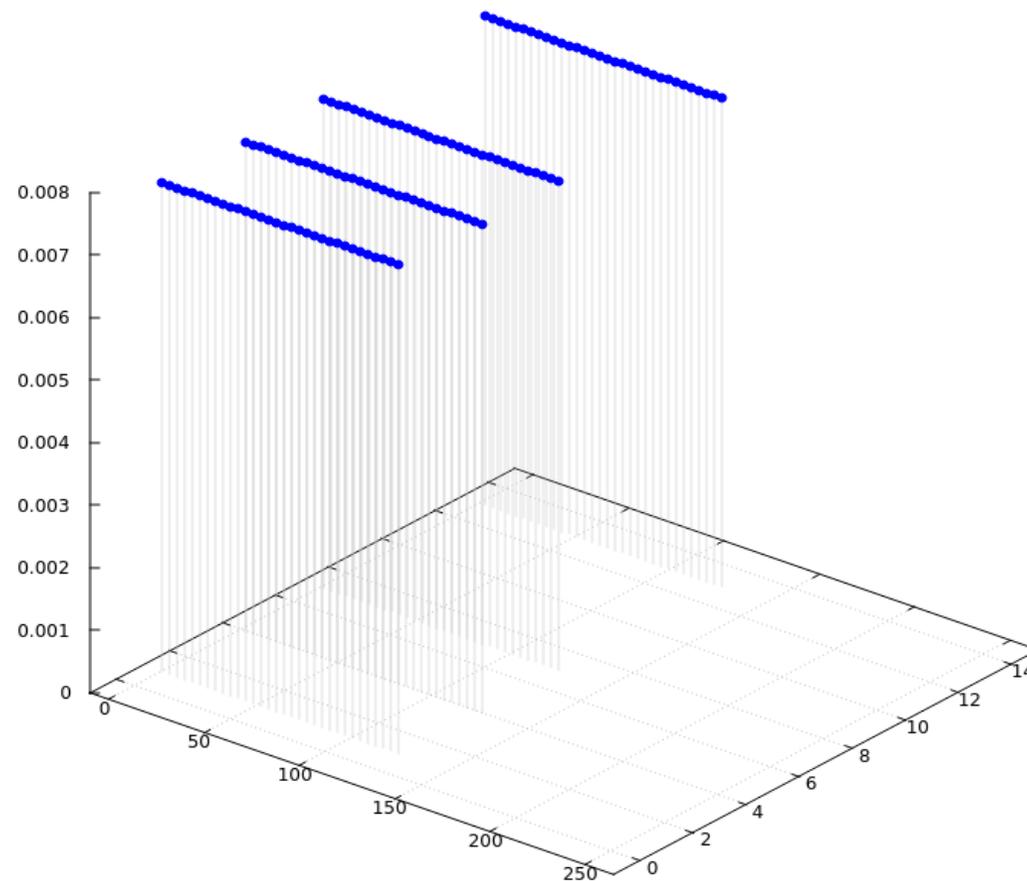
Factoring 15

3. **Applico la Trasformata di Fourier Quantistica**

$$|R_0\rangle |R_1\rangle = \frac{1}{\sqrt{4}} \left(\frac{|0\rangle + |1\rangle + |2\rangle + |3\rangle}{\sqrt{4}} \right) |1\rangle + \frac{1}{\sqrt{4}} \left(\frac{|0\rangle - |1\rangle + i|2\rangle - i|3\rangle}{\sqrt{4}} \right) |7\rangle$$

$$+ \frac{1}{\sqrt{4}} \left(\frac{|0\rangle + |1\rangle - |2\rangle - |3\rangle}{\sqrt{4}} \right) |4\rangle + \frac{1}{\sqrt{4}} \left(\frac{|0\rangle - |1\rangle - i|2\rangle + i|3\rangle}{\sqrt{4}} \right) |13\rangle$$

Factoring 15
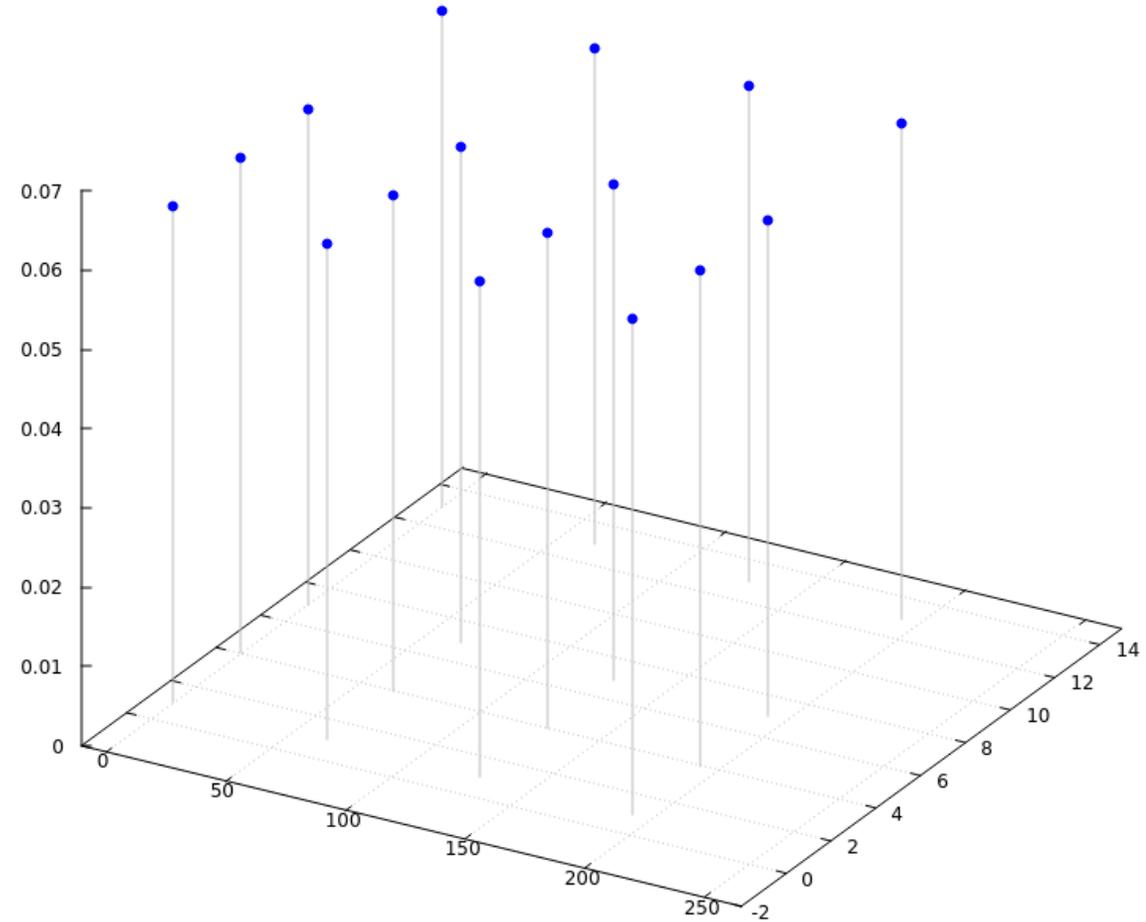
3. Applico la Trasformata di Fourier Quantistica

$$|R_0\rangle |R_1\rangle = \frac{1}{\sqrt{4}} |0\rangle \left(\frac{|1\rangle + |7\rangle + |4\rangle + |13\rangle}{\sqrt{4}} \right) + \frac{1}{\sqrt{4}} |64\rangle \left(\frac{|1\rangle + i|7\rangle - |4\rangle - i|13\rangle}{\sqrt{4}} \right)$$

$$+ \frac{1}{\sqrt{4}} |128\rangle \left(\frac{|1\rangle - |7\rangle + |4\rangle - |13\rangle}{\sqrt{4}} \right) + \frac{1}{\sqrt{4}} |192\rangle \left(\frac{|1\rangle - i|7\rangle - |4\rangle + i|13\rangle}{\sqrt{4}} \right)$$
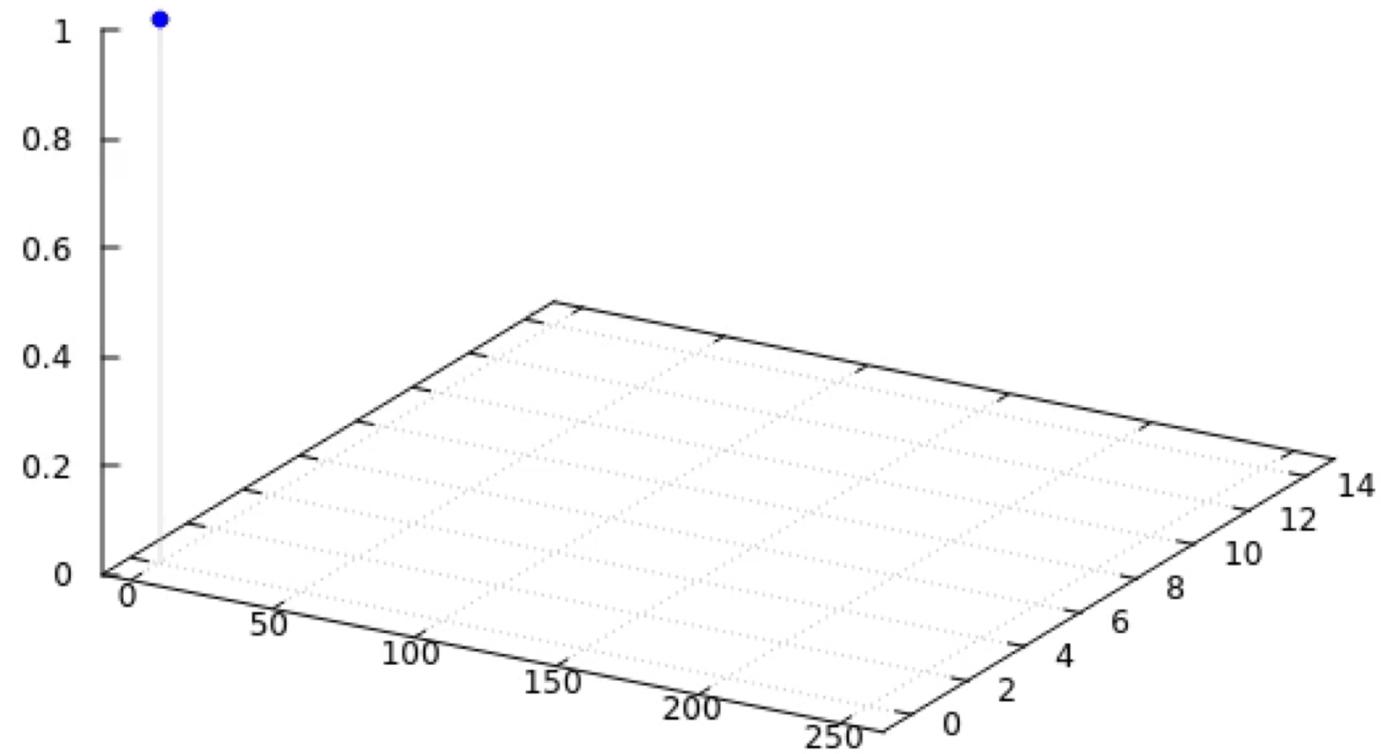
dove si ricordano le espressioni in binario

$$64 \rightarrow 01000000$$

$$128 \rightarrow 10000000$$
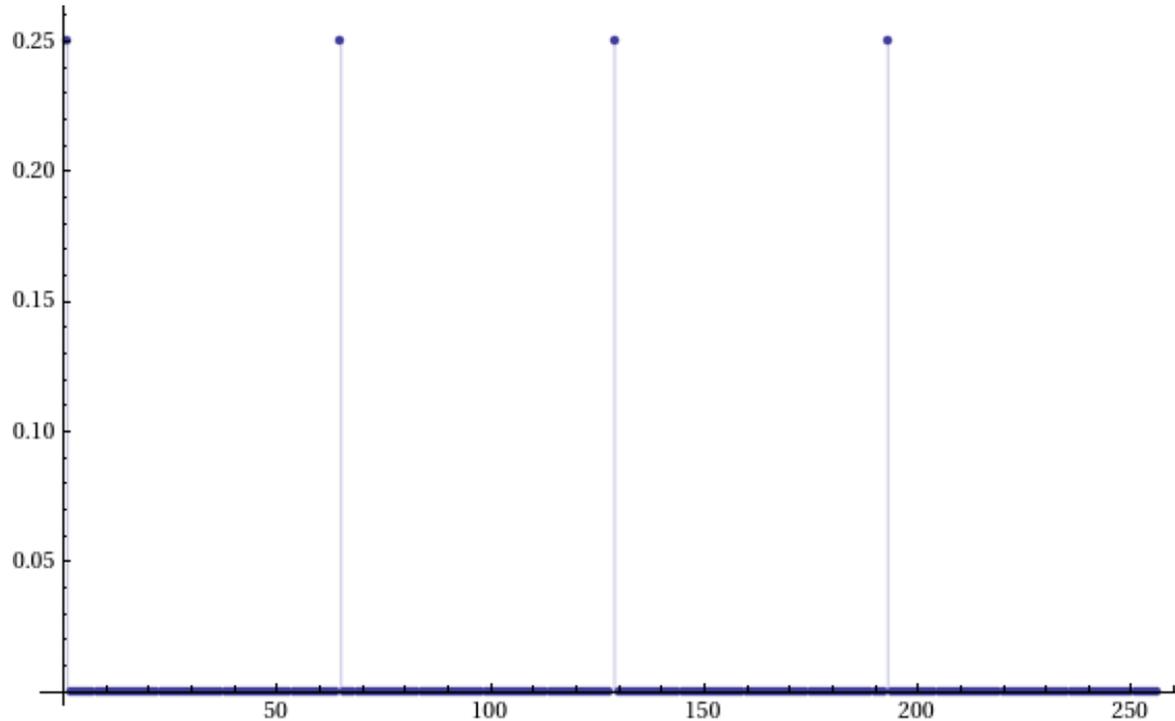
$$192 \rightarrow 11000000$$

Factoring 15

Factoring 15

A questo punto l'algoritmo prevede la misura del primo registro: La distribuzione di probabilità marginale ottenuta è riassunta in figura:

$$P(c) = \sum_k P(c, k)$$

Quindi q = 64

e

Q/P = 256/64 = 4

Che è l'ordine cercato

Example: Factorize 799. Take m = 7.

We have to find the order P of the function f(a) = 7ª mod 799.
(The answer is P = 368). We take n = 20

STEP **0**: The initial state is

$$|\psi_0\rangle = |0\rangle|0\rangle. \tag{8.18}$$

STEP **1**: The QFT on the first register results in

$$|\psi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|0\rangle, \tag{8.19}$$

Example: Factorize 799. Take m = 7.

STEP 2: Application of U_f on $|\psi_1\rangle$ produces

$$|\psi_2\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |7^x \bmod 799\rangle$$

$$= \frac{1}{\sqrt{Q}} \Big[|0\rangle|1\rangle + |1\rangle|7\rangle + |2\rangle|49\rangle + |3\rangle|343\rangle + |4\rangle|4\rangle + |5\rangle|28\rangle$$

$$+ \ldots + |368\rangle|1\rangle + |369\rangle|7\rangle + |370\rangle|49\rangle + \ldots$$

$$+ |Q-2\rangle|756\rangle + |Q-1\rangle|498\rangle \Big]. \tag{8.20}$$

Note that there are only $P = 368$ different states in the second register.

Example: Factorize 799. Take m = 7.

STEP 3: The QFT with $\omega = e^{2\pi i/Q}$, $Q = 2^n$, is applied to the first register. This results in

$$|\psi_3\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} \frac{1}{\sqrt{Q}} \sum_{y=0}^{Q-1} \omega^{-xy} |y\rangle |7^x \bmod 799\rangle \equiv \frac{1}{Q} \sum_{y=0}^{Q-1} |y\rangle |\Upsilon(y)\rangle,$$

where

$$|\Upsilon(y)\rangle = \sum_{x=0}^{Q-1} \omega^{-xy} |7^x \bmod 799\rangle = \sum_{x=0}^{Q-1} e^{-2\pi ixy/Q} |7^x \bmod 799\rangle$$

$$|\Upsilon(y)\rangle = \sum_{x=0}^{Q-1} e^{-2\pi i x y / Q} |7^x \bmod 799\rangle$$

$$= |1\rangle + \omega^{-y}|7\rangle + \omega^{-2y}|49\rangle + \omega^{-3y}|343\rangle + \dots$$
$$+ \omega^{-368y}|1\rangle + \omega^{-369y}|7\rangle + \omega^{-370y}|49\rangle + \omega^{-371y}|343\rangle + \dots$$
$$+ \dots +$$
$$+ \omega^{-736y}|1\rangle + \omega^{-737y}|7\rangle + \omega^{-738y}|49\rangle + \omega^{-739y}|343\rangle + \dots$$
$$+ \dots +$$
$$+ \omega^{-1048432y}|1\rangle + \omega^{-1048433y}|7\rangle + \omega^{-1048434y}|49\rangle + \omega^{-1048435y}|343\rangle$$
$$\dots + \omega^{-1048575y}|498\rangle$$
$$= (1 + \omega^{-368y} + \omega^{-736y} + \dots + \omega^{-1048432y})|1\rangle$$
$$+ (\omega^{-y} + \omega^{-369y} + \omega^{-737y} + \dots + \omega^{-1048433y})|7\rangle$$
$$+ (\omega^{-2y} + \omega^{-370y} + \omega^{-738y} + \dots + \omega^{-1048434y})|49\rangle$$
$$+ (\omega^{-3y} + \omega^{-371y} + \omega^{-739y} + \dots + \omega^{-1048435y})|343\rangle$$
$$+ \dots$$
$$+ (\omega^{-87y} + \omega^{-455y} + \omega^{-823y} + \dots)|794\rangle. \tag{8.22}$$

There are P = 368 ket vectors in the above expansion.
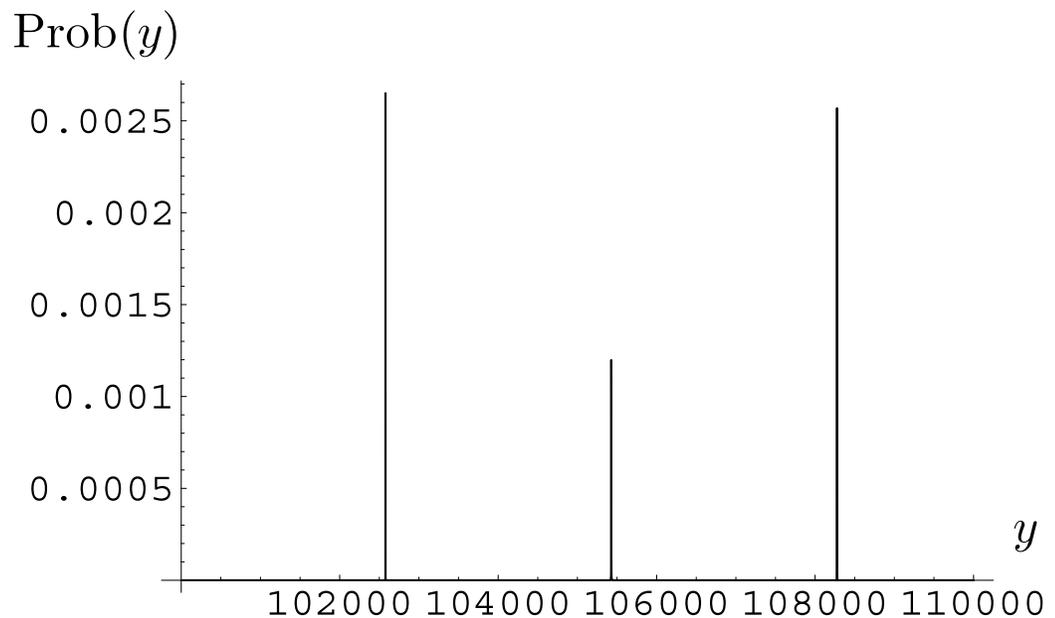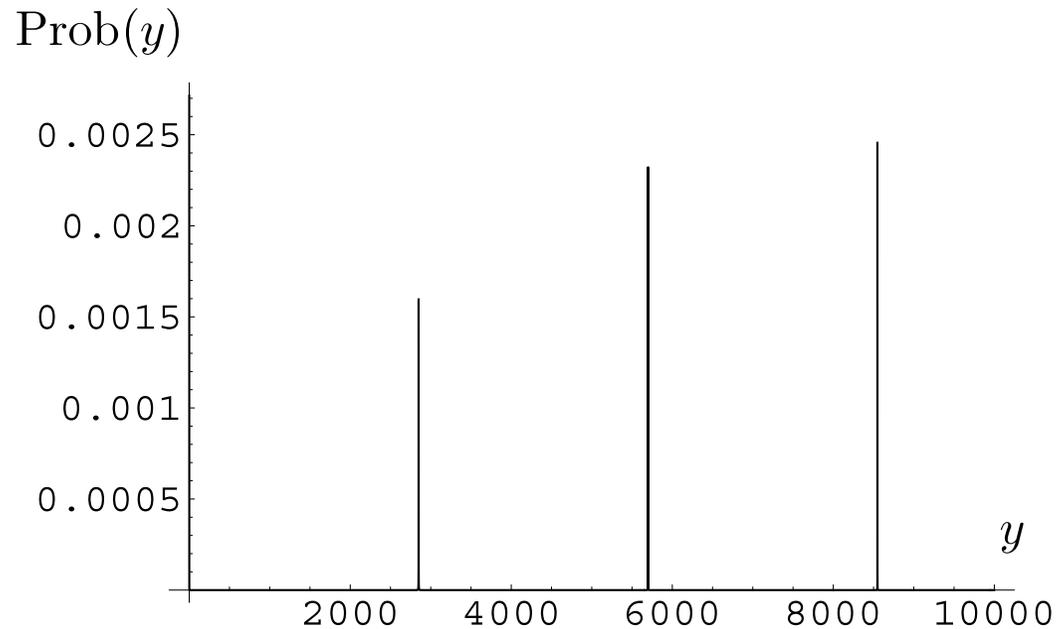
The coefficient of each vector becomes sizeable when and only when y is approximately a multiple of 2849. That means that q ∼ 2849 (in general r ≠ 0) and therefore P ∽ Q/2849 ∽ 368.0505. The order thus obtained is probabilistic, and its plausibility must be checked. This strategy is not practical when N is considerably large. There is a powerful method of continued fraction expansion by which we find the order P with a single measurement of the first register.