Lecture 7: Quantum Fourier Transform and Kitaev's Phase Estimation Algorithm

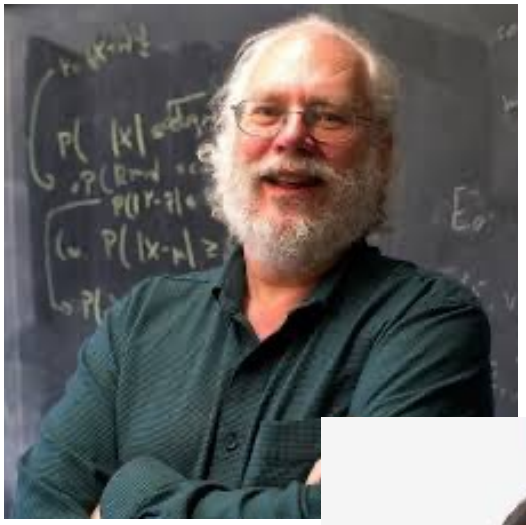
Andreas Wichert

Department of Computer Science and Engineering

Técnico Lisboa

Overview

- Discrete Fourier Transform
- Quantum Fourier Transform
- The QFT Period Algorithm
- Shor’s Algorithm for Factorization
- Kitaev’s Phase Estimation Algorithm
- Quantum Counting



Fourier Analysis

- It is always possible to analyze „complex“ periodic waveforms into a set of sinusoidal waveforms
- Any periodic waveform can be approximated by adding together a number of sinusoidal waveforms
- Fourier analysis tells us what particular set of sinusoids go together to make up a particular complex waveform

- The period is the duration of one cycle of an event and is the reciprocal of the frequency *f*
- For example, if we count 40 events in two seconds, the frequency is

$$\frac{40}{2\ s} = \frac{20}{1\ s} = 20\ \frac{1}{s} = 20\ hertz$$

- period is

$$T = p = \frac{1}{20} s.$$

- The frequency f is the inverse of the period

$$f = \frac{1}{T} = \frac{1}{p}$$

- If something changes rapidly, then we say that it has a high frequency.
- If it does not change rapidly, i.e., it changes smoothly, we say that it has a low frequency.

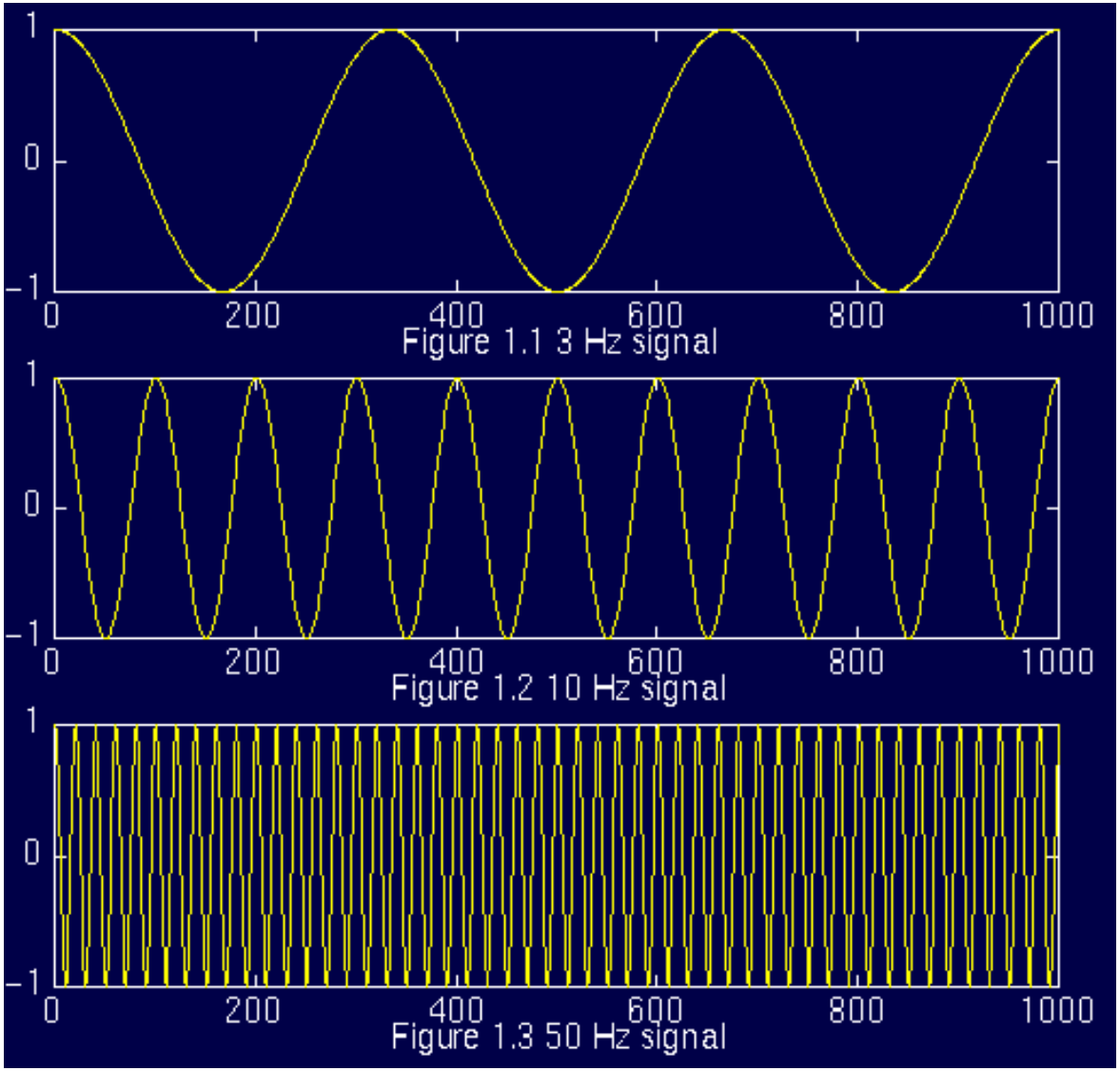


Figure 1.1 3 Hz signal

Figure 1.2 10 Hz signal

Figure 1.3 50 Hz signal

Discrete Fourier Transform

The discrete Fourier transforms discrete time-based maps space-based data into the frequency sequence-based data. Given a sequence α

$$\alpha_t : [1, 2, \cdots, n] \to C.$$

The discrete Fourier transform produces a sequence ω:

$$\omega_f : [1, 2, \cdots, n] \to C.$$

The discrete Fourier transform of $\alpha(t)$ is

$$\omega_f = \frac{1}{\sqrt{n}} \cdot \sum_{t=1}^{n} \alpha_t \cdot e^{-2 \cdot \pi \cdot i \cdot (t-1) \cdot \frac{(f-1)}{n}}$$

its wave frequency is $\frac{(f-1)}{n}$ events per sample. The inverse discrete Fourier transform of ω_f is

$$\alpha_t = \frac{1}{\sqrt{n}} \cdot \sum_{f=1}^{n} \omega_f \cdot e^{2 \cdot \pi \cdot i \cdot (t-1) \cdot \frac{(f-1)}{n}}.$$

Discrete Fourier transform (DFT) can be seen as a linear transform F talking the column vector α to a column

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_n \end{pmatrix} = \frac{1}{\sqrt{n}} \cdot \begin{pmatrix} e^{-2\cdot\pi\cdot i\cdot(0)\cdot\frac{(0)}{n}} & e^{-2\cdot\pi\cdot i\cdot(0)\cdot\frac{(1)}{n}} & \dots & e^{-2\cdot\pi\cdot i\cdot(0)\cdot\frac{(n-1)}{n}} \\ e^{-2\cdot\pi\cdot i\cdot(1)\cdot\frac{(0)}{n}} & e^{-2\cdot\pi\cdot i\cdot(1)\cdot\frac{(1)}{n}} & \dots & e^{-2\cdot\pi\cdot i\cdot(1)\cdot\frac{(n-1)}{n}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-2\cdot\pi\cdot i\cdot(n-1)\cdot\frac{(0)}{n}} & e^{-2\cdot\pi\cdot i\cdot(n-1)\cdot\frac{(1)}{n}} & \dots & e^{-2\cdot\pi\cdot i\cdot(n)\cdot\frac{(n-1)}{n}} \end{pmatrix} \cdot \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}$$

and the inverse discrete Fourier transform (IDFT) can be seen as a linear transform talking the column vector ω to a column vector α

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} = \frac{1}{\sqrt{n}} \cdot \begin{pmatrix} e^{2\cdot\pi\cdot i\cdot(0)\cdot\frac{(0)}{n}} & e^{2\cdot\pi\cdot i\cdot(0)\cdot\frac{(1)}{n}} & \dots & e^{2\cdot\pi\cdot i\cdot(0)\cdot\frac{(n-1)}{n}} \\ e^{2\cdot\pi\cdot i\cdot(1)\cdot\frac{(0)}{n}} & e^{2\cdot\pi\cdot i\cdot(1)\cdot\frac{(1)}{n}} & \dots & e^{2\cdot\pi\cdot i\cdot(1)\cdot\frac{(n-1)}{n}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{2\cdot\pi\cdot i\cdot(n-1)\cdot\frac{(0)}{n}} & e^{2\cdot\pi\cdot i\cdot(n-1)\cdot\frac{(1)}{n}} & \dots & e^{2\cdot\pi\cdot i\cdot(n)\cdot\frac{(n-1)}{n}} \end{pmatrix} \cdot \begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_n \end{pmatrix}.$$

An nth root of unity is a complex number ζ satisfying the equation

$$\zeta^n = 1 \qquad \left(e^{-2\cdot\pi\cdot i\cdot\frac{1}{n}}\right)^n = 1$$

with $n = 1, 2, 3, \cdots, n-1$ being a positive integer, for example

$$\zeta_n = e^{-2\cdot\pi\cdot i\cdot\frac{1}{n}} = \cos\left(2\cdot\pi\cdot\frac{1}{n}\right) - i\cdot\sin\left(2\cdot\pi\cdot\frac{1}{n}\right)$$

Using the nth root of unity the matrix can be represented as a Vandermonde matrix

$$F = \frac{1}{\sqrt{n}}\cdot\begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \zeta_n & \zeta_n^2 & \zeta_n^3 & \cdots & \zeta_n^{(n-1)} \\ 1 & \zeta_n^2 & \zeta_n^4 & \zeta_n^6 & \cdots & \zeta_n^{2\cdot(n-1)} \\ 1 & \zeta_n^3 & \zeta_n^6 & \zeta_n^9 & \cdots & \zeta_n^{3\cdot(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & \zeta_n^{(n-1)} & \zeta_n^{2\cdot(n-1)} & \zeta_n^{3\cdot(n-1)} & \cdots & \zeta_n^{(n-1)\cdot(n-1)} \end{pmatrix}.$$

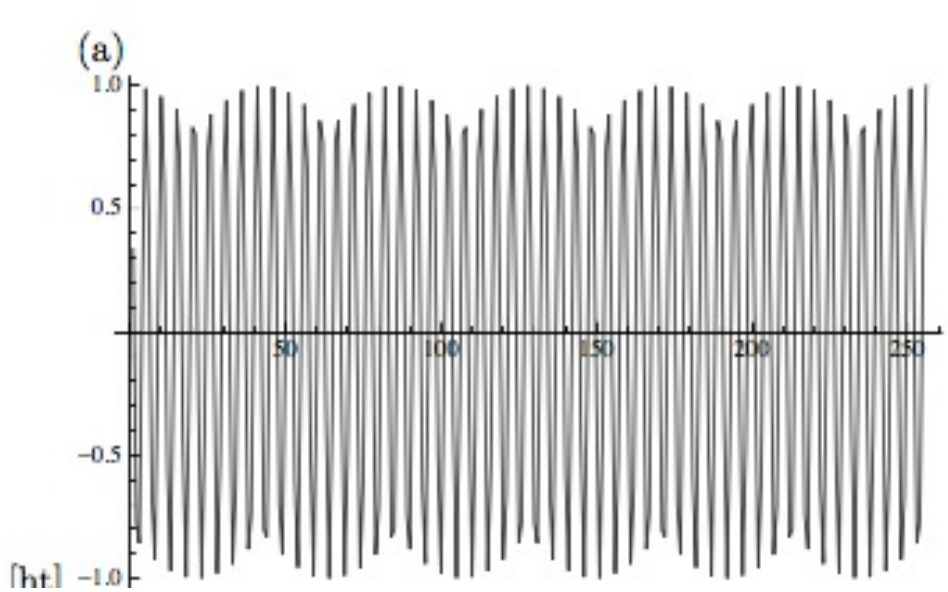
$$F = \frac{1}{\sqrt{n}} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \zeta_n & \zeta_n^2 & \zeta_n^3 & \cdots & \zeta_n^{(n-1)} \\ 1 & \zeta_n^2 & \zeta_n^4 & \zeta_n^6 & \cdots & \zeta_n^{2\cdot(n-1)} \\ 1 & \zeta_n^3 & \zeta_n^6 & \zeta_n^9 & \cdots & \zeta_n^{3\cdot(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & \zeta_n^{(n-1)} & \zeta_n^{2\cdot(n-1)} & \zeta_n^{3\cdot(n-1)} & \cdots & \zeta_n^{(n-1)\cdot(n-1)} \end{pmatrix}.$$

The matrix F, also called DFT matrix is unitary

$$F^{-1} = F^* = IF.$$

Because F is unitary it implies that the length of a vector is preserved as stated in Parseval's theorem

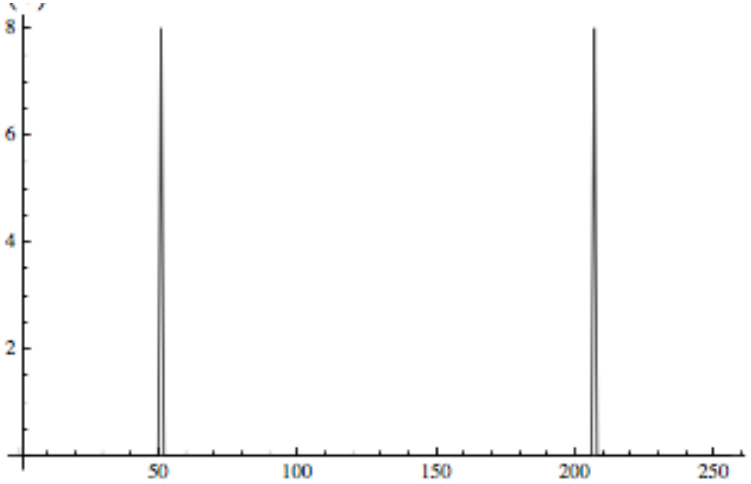
$$\|\omega\| = \|F \cdot \alpha\| = \|\alpha\|.$$



We generates a list with 256 = 2^8 elements containing a periodic signal

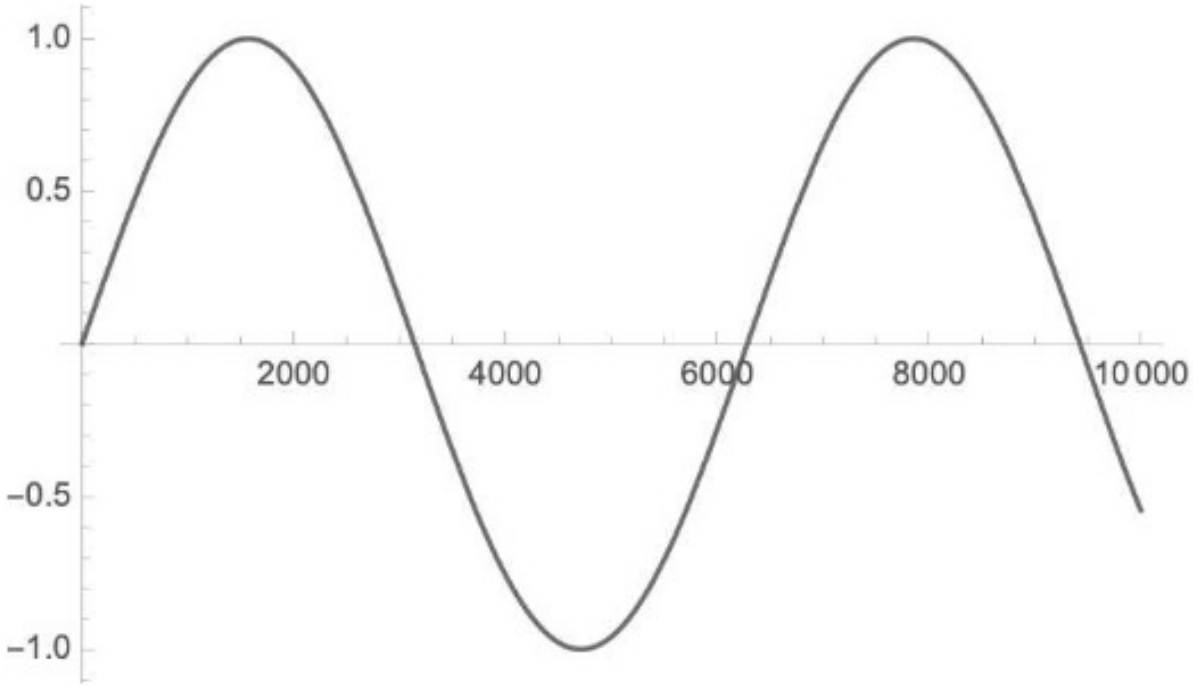
$$\alpha_t = sin\left(\frac{50 \cdot t \cdot 2 \cdot \cdot \pi}{256}\right).$$

The frequency spectrum of a **real** valued signal is always symmetric



The discrete Fourier transform ω_f of the real valued signal αt is symmetric. It shows a strong peak at *50 + 1* and a symmetric peak at *256 − 50 + 1* representing the frequency component of the signal

Quantum Fourier Transform



Quantum Fourier transform can be used to determine the period of a periodic function in polynomial time

Quantum Fourier Transform

- The QFT on a state $|x\rangle$ of m qubits in a n-dimensional Hilbert space $H_n = H_2^m$ can be represented as

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle.$$

- It is just the discrete inverse Fourier transform of α(t) in the bra-ket notation

$$\omega_f = \frac{1}{\sqrt{n}} \cdot \sum_{t=1}^{n} \alpha_t \cdot e^{2 \cdot \pi \cdot i \cdot \frac{(f-1)}{n} \cdot (t-1)}.$$

Conventions for the *sign of the phase factor exponent vary*; here the quantum Fourier transform has the same effect as the *inverse discrete Fourier transform* and follows *the qiskit notation*.

convention choice

- QFT ≈ inverse DFT
- inverse QFT ≈ DFT

For one qubit $m = 1$, $n = 2$

$$\zeta_2 = e^{2 \cdot \pi \cdot i \cdot \frac{1}{2}} = e^{-\pi \cdot i} = e^{\pi \cdot i} = -1$$

and the QFT F_1 is

$$F_1 = \frac{1}{\sqrt{2}} \cdot \begin{pmatrix} 1 & 1 \\ 1 & \zeta_2 \end{pmatrix} = \frac{1}{\sqrt{2}} \cdot \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = H_1.$$

For two qubits $m = 2$, $n = 4$

$$\zeta_4 = e^{2 \cdot \pi \cdot i \cdot \frac{1}{4}} = e^{\cdot \pi \cdot i \cdot \frac{1}{2}} = i$$

and the QFT F_2 is

$$F_2 = \frac{1}{\sqrt{4}} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \zeta_4 & \zeta_4^2 & \zeta_4^3 \\ 1 & \zeta_4^2 & \zeta_4^4 & \zeta_4^6 \\ 1 & \zeta_4^3 & \zeta_4^6 & \zeta_4^9 \end{pmatrix} = \frac{1}{2} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

QFT F_2 is

$$F_2 = \frac{1}{\sqrt{4}} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \zeta_4 & \zeta_4^2 & \zeta_4^3 \\ 1 & \zeta_4^2 & \zeta_4^4 & \zeta_4^6 \\ 1 & \zeta_4^3 & \zeta_4^6 & \zeta_4^9 \end{pmatrix} = \frac{1}{2} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

inverse QFT is

$$IF_2 = F_2^* = \frac{1}{2} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix}.$$

For three qubits $m = 3$, $n = 8$

$$\zeta_8 = e^{2\cdot\pi\cdot i\cdot\frac{1}{8}} = e^{\pi\cdot i\cdot\frac{1}{4}} = \frac{1+i}{\sqrt{2}}$$

$$F_3 = \frac{1}{\sqrt{8}} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & e^{\frac{i\pi}{4}} & i & e^{\frac{3i\pi}{4}} & -1 & e^{-\frac{1}{4}(3i\pi)} & -i & e^{-\frac{1}{4}(i\pi)} \\ 1 & i & -1 & -i & 1 & i & -1 & -i \\ 1 & e^{\frac{3i\pi}{4}} & -i & e^{\frac{i\pi}{4}} & -1 & e^{-\frac{1}{4}(i\pi)} & i & e^{-\frac{1}{4}(3i\pi)} \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & e^{-\frac{1}{4}(3i\pi)} & i & e^{-\frac{1}{4}(i\pi)} & -1 & e^{\frac{i\pi}{4}} & -i & e^{\frac{3i\pi}{4}} \\ 1 & -i & -1 & i & 1 & -i & -1 & i \\ 1 & e^{-\frac{1}{4}(i\pi)} & -i & e^{-\frac{1}{4}(3i\pi)} & -1 & e^{\frac{3i\pi}{4}} & i & e^{\frac{i\pi}{4}} \end{pmatrix}$$

The first row of F_3 is the DC average of the amplitude of the input state when measured, the following rows represent the AC (difference) of the input state amplitudes.

QFT Decomposition

- QFT can be factored into the tensor product of *m* single-qubit operations using the binary fractions are represented as

$$0.x_m x_{m\ 1} x_{m-2} \cdots x_2 x_1 = \frac{x_m}{2^1} + \frac{x_{m-1}}{2^2} + \cdots + \frac{x_1}{2^m}.$$

- The representation involves the input in the tensor decomposition
- The product of *m* single-qubit operations of the QFT allows us to define a quantum circuit

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$\frac{1}{\sqrt{n}} \cdot \left(\sum_{y_m \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_m \cdot 0.x_1} |y_1\rangle \right) \left(\sum_{y_{m-1} \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_{m-1} \cdot 0.x_2 x_1} |y_2\rangle \right) \cdot$$

$$\cdots \left(\sum_{y_1 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_m \cdots x_2 x_1} |y_m\rangle \right)$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle \right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle \right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m \cdots x_2 x_1} \cdot |1\rangle \right)$$

Why?

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{-2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle.$$

It is just the discrete Fourier transform of $\alpha(t)$ in the bra-ket notation

$$\omega_f = \frac{1}{\sqrt{n}} \cdot \sum_{t=1}^{n} \alpha_t \cdot e^{-2 \cdot \pi \cdot i \cdot \frac{(f-1)}{n} \cdot (t-1)}.$$

The binary representation of x of m bits is given by

$$x = x_m \cdot 2^{m-1} + x_{m-1} \cdot 2^{m-2} + \cdots + x_2 \cdot 2^1 + x_1 \cdot 2^0$$

and of y by

$$y = y_m \cdot 2^{m-1} + y_{m-1} \cdot 2^{m-2} + \cdots + y_2 \cdot 2^1 + y_1 \cdot 2^0.$$

this is little-endian notation as used in qiskit

We can represent the multiplication of

$$e^{\frac{-2\cdot\pi\cdot i\cdot y\cdot x}{n}} = e^{\frac{-2\cdot\pi\cdot i\cdot y\cdot x}{2^m}}$$

binary representation

$$= e^{\frac{-2\cdot\pi\cdot i\cdot(y_m\cdot 2^{m-1}+\cdots 1+y_1\cdot 2^0)\cdot\cdots(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0)}{2^m}} =$$

$$= e^{\frac{-2\cdot\pi\cdot i\cdot(y_m\cdot 2^{m-1}\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0)+\cdots+y_1\cdot 2^0\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0))}{2^m}}.$$

Because

$$e^{-2\cdot\pi\cdot i\cdot(a+b+c)} = e^{(-2\cdot\pi\cdot i\cdot a)+(-2\cdot\pi\cdot i\cdot b)+(-2\cdot\pi\cdot i\cdot c)} =$$

$$= e^{(-2\cdot\pi\cdot i\cdot a)}\cdot e^{(-2\cdot\pi\cdot i\cdot b)}\cdot e^{(-2\cdot\pi\cdot i\cdot c)}$$

root of unity $e^{-2\cdot\pi\cdot i\cdot n} = 1, \quad n \in N_0 = \{0, 1, 2, 3, \cdots\}.$

we can ignore in

$$e^{\frac{-2\cdot\pi\cdot i\cdot(y_m\cdot 2^{m-1}\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0)+\cdots+y_1\cdot 2^0\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0))}{2^m}}$$

the terms divisible by $n = 2^m$. For example

$$e^{-2\cdot\pi\cdot i\cdot(1+\frac{1}{2}+2)} = e^{(-2\cdot\pi\cdot i\cdot 1)}\cdot e^{(-2\cdot\pi\cdot i\cdot\frac{1}{2})}\cdot e^{(-2\cdot\pi\cdot i\cdot 3)} = 1\cdot e^{(-2\cdot\pi\cdot i\cdot\frac{1}{2})}\cdot 1 = -1.$$

$$e^{-2\cdot\pi\cdot i\cdot(a+b+c)} = e^{(-2\cdot\pi\cdot i\cdot a)}\cdot e^{(-2\cdot\pi\cdot i\cdot b)}\cdot e^{(-2\cdot\pi\cdot i\cdot c)}$$

root of unity $e^{-2\cdot\pi\cdot i\cdot n} = 1, \quad n \in N_0 = \{0, 1, 2, 3, \cdots\}.$

we can ignore in

$$e^{\frac{-2\cdot\pi\cdot i\cdot(y_m\cdot 2^{m-1}\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0)+\cdots+y_1\cdot 2^0\cdot(x_m\cdot 2^{m-1}\cdots+x_2\cdot 2^1+x_1\cdot 2^0))}{2^m}}$$

the terms divisible by $n = 2^m$. For example

$$e^{-2\cdot\pi\cdot i\cdot(1+\frac{1}{2}+2)} = e^{(-2\cdot\pi\cdot i\cdot 1)}\cdot e^{(-2\cdot\pi\cdot i\cdot\frac{1}{2})}\cdot e^{(-2\cdot\pi\cdot i\cdot 3)} = 1\cdot e^{(-2\cdot\pi\cdot i\cdot\frac{1}{2})}\cdot 1 = -1.$$

It follows that

$$e^{\frac{-2\cdot\pi\cdot i\cdot y\cdot x}{2^m}} =$$

$$= e^{-2\cdot\pi\cdot i\cdot\left(y_m\cdot\frac{x_1}{2^1}+y_{m-1}\cdot\left(\frac{x_2}{2^1}+\frac{x_1}{2^2}\right)+y_{m-2}\cdot\left(\frac{x_3}{2^1}+\frac{x_2}{2^2}+\frac{x_1}{2^3}\right)+\cdots+y_1\cdot\left(\frac{x_m}{2^1}+\frac{x_{m-1}}{2^2}+\cdots+\frac{x_1}{2^m}\right)\right)}$$

It follows that

$$e^{\frac{-2\cdot\pi\cdot i\cdot y\cdot x}{2^m}} =$$

$$= e^{-2\cdot\pi\cdot i\cdot\left(y_m\cdot\frac{x_1}{2^1}+y_{m-1}\cdot\left(\frac{x_2}{2^1}+\frac{x_1}{2^2}\right)+y_{m-2}\cdot\left(\frac{x_3}{2^1}+\frac{x_2}{2^2}+\frac{x_1}{2^3}\right)+\cdots+y_1\cdot\left(\frac{x_m}{2^1}+\frac{x_{m-1}}{2^2}+\cdots+\frac{x_1}{2^m}\right)\right)}$$

using the binary fraction notation for binary numbers

$$e^{\frac{-2\cdot\pi\cdot i\cdot y\cdot x}{2^m}} =$$

$$= e^{-2\cdot\pi\cdot i\cdot(y_m\cdot 0.x_1+y_{m-1}\cdot 0.x_2x_1+y_{m-2}\cdot 0.x_3x_2x_1+\cdots+y_1\cdot 0.x_mx_{m\ 1}x_{m-2}\cdots x_2x_1)}$$

binary fractions are represented as

$$0.x_mx_{m\ 1}x_{m-2}\cdots x_2x_1 = \frac{x_m}{2^1}+\frac{x_{m-1}}{2^2}+\cdots+\frac{x_1}{2^m}.$$

- Since

$$e^{-2\cdot\pi\cdot i\cdot(a+b+c)} = e^{(-2\cdot\pi\cdot i\cdot a)+(-2\cdot\pi\cdot i\cdot b)+(-2\cdot\pi\cdot i\cdot c)} =$$

$$= e^{(-2\cdot\pi\cdot i\cdot a)} \cdot e^{(-2\cdot\pi\cdot i\cdot b)} \cdot e^{(-2\cdot\pi\cdot i\cdot c)}$$

Binary

- and

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y\in B^m} e^{-2\cdot\pi\cdot i\cdot(y_m\cdot 0.x_1 + y_{m-1}\cdot 0.x_2x_1 + y_{m-2}\cdot 0.x_3x_2x_1 + \cdots + y_1\cdot 0.x_m x_{m\ 1}x_{m-2}\cdots x_2x_1)} |y\rangle$$

$$y = y_m \cdot 2^{m-1} + y_{m-1} \cdot 2^{m-2} + \cdots + y_2 \cdot 2^1 + y_1 \cdot 2^0.$$

- The QFT can be factored into the tensor product of m single-qubit operations,

$$y = y_m \cdot 2^{m-1} + y_{m-1} \cdot 2^{m-2} + \cdots + y_2 \cdot 2^1 + y_1 \cdot 2^0.$$

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{-2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$\frac{1}{\sqrt{n}} \cdot \left(\sum_{y_m \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_m \cdot 0.x_1} |y_m\rangle \right) \left(\sum_{y_{m-1} \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_{m-1} \cdot 0.x_2 x_1} |y_{m-1}\rangle \right)$$

$$\cdots \left(\sum_{y_1 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_m \cdots x_2 x_1} |y_1\rangle \right)$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle \right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle \right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m \cdots x_2 x_1} \cdot |1\rangle \right)$$

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$\frac{1}{\sqrt{n}} \cdot \left(\sum_{y_m \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_m \cdot 0.x_1} |y_m\rangle \right) \left(\sum_{y_{m-1} \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_{m-1} \cdot 0.x_2 x_1} |y_{m-1}\rangle \right)$$

$$\cdots \left(\sum_{y_1 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_m \cdots x_2 x_1} |y_1\rangle \right)$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle \right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle \right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m \cdots x_2 x_1} \cdot |1\rangle \right)$$

- Qiskit (IBM): little-endian $x_m x_{m-1} \ldots x_1$

binary fractions

$$0.x_m x_{m\ 1} x_{m-2} \cdots x_2 x_1 = \frac{x_m}{2^1} + \frac{x_{m-1}}{2^2} + \cdots + \frac{x_1}{2^m}.$$

- Cirq (Google): big-endian
- Q# (Microsoft): big-endian
- PennyLane (Xanadu): big-endian

$x_1 x_2 \ldots x_m$

- **Quantum Physics** and some **Quantum Computing** textbooks

binary fractions

$$0.x_1 x_2 x_3 \cdots x_2 x_m = \frac{x_1}{2^1} + \frac{x_2}{2^2} + \cdots + \frac{x_m}{2^m}.$$

Big-endian convention is used in **Quantum Physics** and **Quantum Computing** textbooks *(reordered)*

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$\frac{1}{\sqrt{n}} \cdot \left(\sum_{y_1 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_m} |y_1\rangle \right) \cdot \left(\sum_{y_2 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_2 \cdot 0.x_{m-1} x_m} |y_2\rangle \right) \cdots$$

$$\cdots \left(\sum_{y_m \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_1 \cdots x_{m-1} x_m} |y_m\rangle \right)$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m} \cdot |1\rangle \right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_{m-1} x_m} \cdot |1\rangle \right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1 \cdots x_{m-1} x_m} \cdot |1\rangle \right)$$

Little-endian will be used in the following

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$\frac{1}{\sqrt{n}} \cdot \left(\sum_{y_m \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_m \cdot 0.x_1} |y_m\rangle \right) \left(\sum_{y_{m-1} \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_{m-1} \cdot 0.x_2 x_1} |y_{m-1}\rangle \right)$$

$$\cdots \left(\sum_{y_1 \in \{0,1\}} e^{2 \cdot \pi \cdot i \cdot y_1 \cdot 0.x_m \cdots x_2 x_1} |y_1\rangle \right)$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle \right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle \right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m \cdots x_2 x_1} \cdot |1\rangle \right)$$

- The circuit will use a controlled phase gate CP_k that performs following mapping on two qubits

$$CP_k|00\rangle = |00\rangle, \quad CP_k|01\rangle = |01\rangle,$$
$$CP_k|10\rangle = |10\rangle, \quad CP_k|11\rangle = e^{2\cdot\pi\cdot i\cdot/2^k}\cdot|11\rangle.$$

$$CP_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-2\cdot\pi\cdot i\cdot/2^k} \end{pmatrix}.$$

$$CP(|00\rangle \longrightarrow |00\rangle \qquad CP|01\rangle \longrightarrow |01\rangle$$
$$CP|10\rangle \longrightarrow |10\rangle \qquad CP|11\rangle \longrightarrow e^{i\cdot\lambda}\cdot|11\rangle$$

$$CP(\lambda) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\cdot\lambda} \end{pmatrix}$$

QFT for two qubits

We demonstrate the definition of the quantum circuit on F_2

$$F_2|x_2x_1\rangle = \frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_2x_1} \cdot |1\rangle\right)$$

on the input $|x_2x_1\rangle$. The "first" operation can be represented as

$$\frac{1}{\sqrt{2}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_2x_1} \cdot |1\rangle\right) = \frac{1}{\sqrt{2}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot \frac{x_2}{2^1}} \cdot e^{2\cdot\pi\cdot i\cdot \frac{x_1}{2^2}} \cdot |1\rangle\right)$$

and can be represented as

$$CP_1 \cdot (H_1 \otimes I_1) \cdot |x_2x_1\rangle.$$

$$F_2|x_2x_1\rangle = \frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_2x_1} \cdot |1\rangle\right)$$

$$CP_1 \cdot (H_1 \otimes I_1) \cdot |x_2x_1\rangle.$$

The H_1 gate puts the qubit into superposition and applies a phase of $e^{2\cdot\pi\cdot ix_2/2}$ with resulting state

$$\frac{1}{\sqrt{2}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot\frac{x_2}{2^1}} \cdot |1\rangle\right)$$

We applying a Controlled-Phase gate with a rotation of $\pi/2$ using $|x_1\rangle$ as control wire with resulting state

$$\frac{1}{\sqrt{2}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot\frac{x_2}{2^1}} \cdot e^{2\cdot\pi\cdot i\cdot\frac{x_1}{2^2}} \cdot |1\rangle\right) \otimes |x_1\rangle$$

Finally, we apply an $I_1 \otimes H_1$ gate to $|x_1\rangle$

$$\frac{1}{\sqrt{2}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_1} \cdot |1\rangle\right)$$

Together we get

$$(I_1 \otimes H_1) \cdot CP_1 \cdot (H_1 \otimes I_1) \cdot |x_2 x_1\rangle =$$

$$= \frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle\right)$$

$$F_2|x_2 x_1\rangle = \frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle\right)$$

The arrangement of the bits is not correct.
This is because the last qubit in the result uses the first input qubit and so on...

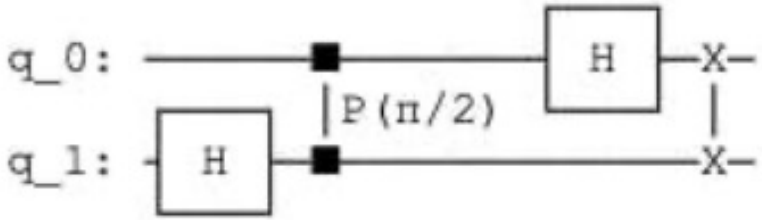
$$F_2 \cdot |x_2 x_1\rangle = SWAP(I_1 \otimes H_1) \cdot CP_1 \cdot (H_1 \otimes I_1) \cdot |x_2 x_1\rangle =$$

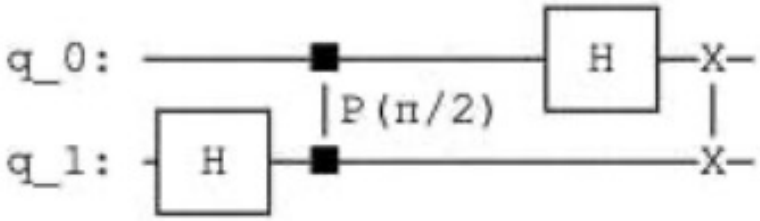
$$\frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle\right)$$

QFT for two qubits (qiskit)

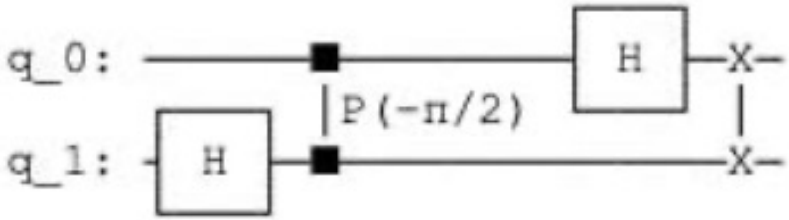
$$F_2 \cdot |x_2 x_1\rangle = SWAP(I_1 \otimes H_1) \cdot CP_1 \cdot (H_1 \otimes I_1) \cdot |x_2 x_1\rangle =$$

$$\frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle\right)$$





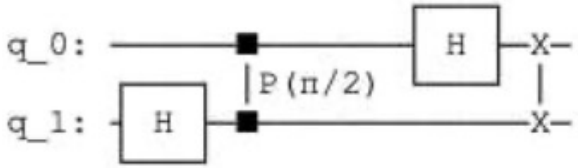
QFT two qubits



Inverse QFT two qubits

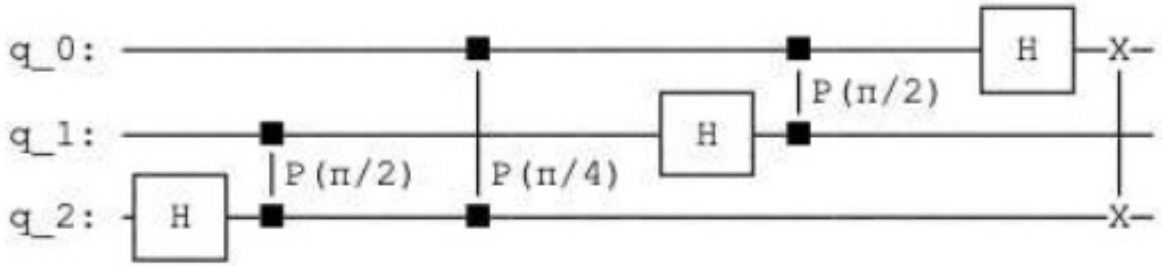
QFT for three qubits

$$F_2|x_2x_1\rangle = \frac{1}{\sqrt{4}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_2x_1} \cdot |1\rangle\right)$$

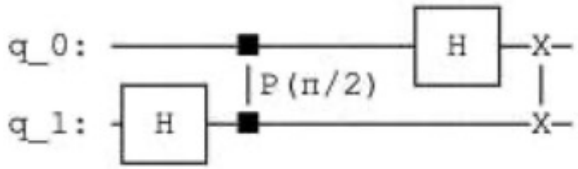


$$F_3|x_3x_2x_1\rangle = \frac{1}{\sqrt{8}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_2x_1} \cdot |1\rangle\right) \otimes$$

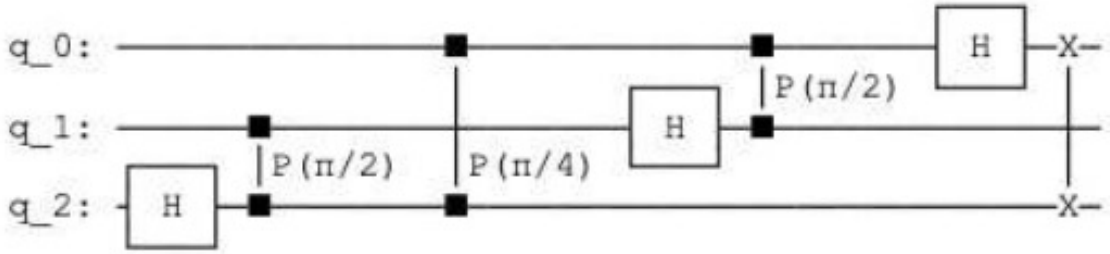
$$\otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot 0.x_3x_2x_1} \cdot |1\rangle\right)$$



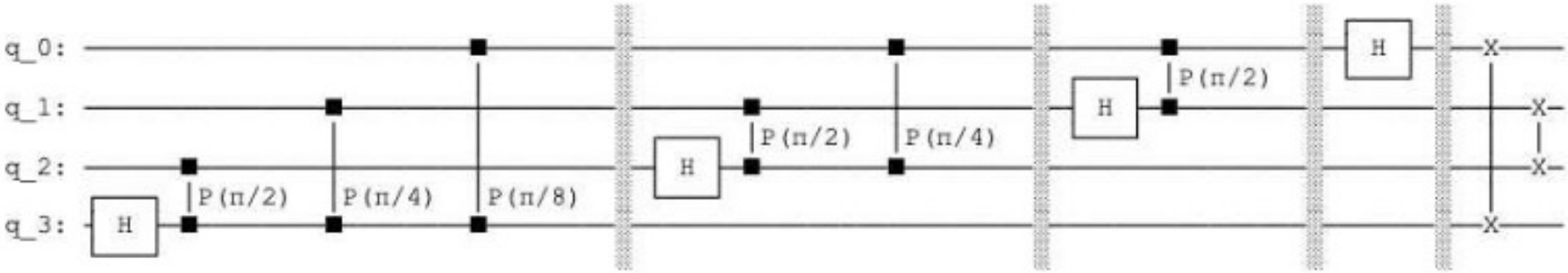
$$\begin{bmatrix}
\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} \\
\frac{\sqrt{2}}{4} & \frac{1}{4}+\frac{i}{4} & \frac{\sqrt{2}i}{4} & -\frac{1}{4}+\frac{i}{4} & -\frac{\sqrt{2}}{4} & -\frac{1}{4}-\frac{i}{4} & -\frac{\sqrt{2}i}{4} & \frac{1}{4}-\frac{i}{4} \\
\frac{\sqrt{2}}{4} & \frac{\sqrt{2}i}{4} & -\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}i}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}i}{4} & -\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}i}{4} \\
\frac{\sqrt{2}}{4} & -\frac{1}{4}+\frac{i}{4} & -\frac{\sqrt{2}i}{4} & \frac{1}{4}+\frac{i}{4} & -\frac{\sqrt{2}}{4} & \frac{1}{4}-\frac{i}{4} & \frac{\sqrt{2}i}{4} & -\frac{1}{4}-\frac{i}{4} \\
\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} \\
\frac{\sqrt{2}}{4} & -\frac{1}{4}-\frac{i}{4} & \frac{\sqrt{2}i}{4} & \frac{1}{4}-\frac{i}{4} & -\frac{\sqrt{2}}{4} & \frac{1}{4}+\frac{i}{4} & -\frac{\sqrt{2}i}{4} & -\frac{1}{4}+\frac{i}{4} \\
\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}i}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}i}{4} & \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}i}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}i}{4} \\
\frac{\sqrt{2}}{4} & \frac{1}{4}-\frac{i}{4} & -\frac{\sqrt{2}i}{4} & -\frac{1}{4}-\frac{i}{4} & -\frac{\sqrt{2}}{4} & -\frac{1}{4}+\frac{i}{4} & \frac{\sqrt{2}i}{4} & \frac{1}{4}+\frac{i}{4}
\end{bmatrix}$$



QFT two qubits

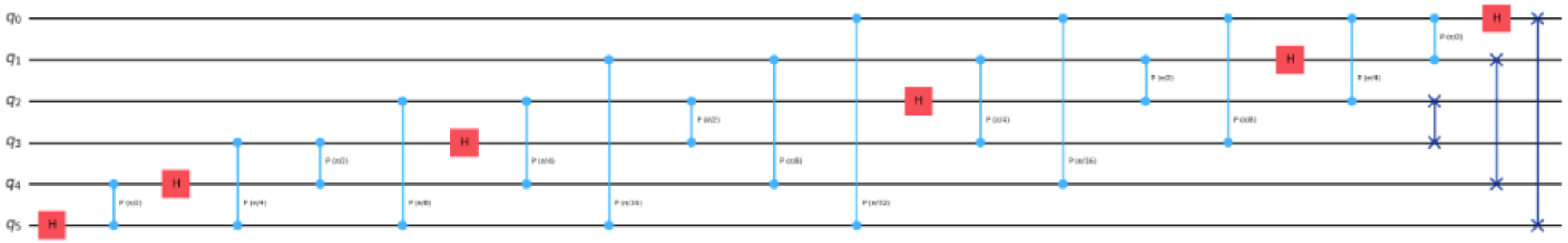


QFT three qubits



QFT four qubits

QFT six qubits



QFT costs

- The first term requires one Hadamard gate, the second one requires a Hadamard gate and a controlled phase gate
- Each following term requires an additional controlled phase gate
  - Summing up

$$1 + 2 + 3 + \cdots (m-1) + m = \frac{m \cdot (m-1)}{2} = O(m^2).$$

- The cost of a QFT are O(m^2) compared to the cost of FFT with O($2^m \cdot m$) on a conventional computer, so the costs of QFT are **exponentially less**

The QFT Period Algorithm

- In the algorithm based on QFT the function *f(x)* **must be periodic**
  - The determined property is the period of the function f(x)
- **We cannot** use QFT to determine if a function is periodic or not
- We represent the function f(x) by a quantum Boolean circuit represented by a unitary operator U_f that acts on two registers of *m* qubits,

$$U_f \cdot |x\rangle|0^{\otimes m}\rangle = |x\rangle|f(x)\rangle$$

after the application of U_f the two registers are **entangled**

- We build a superposition of *m* qubits

$$H_m \cdot |0^{\otimes m}\rangle|0^{\otimes m}\rangle = \frac{1}{\sqrt{2^m}} \sum_{x \in B^m} |x\rangle|0^{\otimes m}\rangle.$$

- In the second step we apply the U_f operator

$$U_f \left(\frac{1}{\sqrt{2^m}} \sum_{x \in B^m} |x\rangle|0^{\otimes m}\rangle \right) = \frac{1}{\sqrt{2^m}} \sum_{x \in B^m} U_f \cdot |x\rangle|0^{\otimes m}\rangle = \frac{1}{\sqrt{2^m}} \sum_{x \in B^m} |x\rangle|f(x)\rangle.$$

- In the third step we measure the “second” register of the compound system
- The state of the system is projected to the subspace that corresponds to the **observed state** and the vector representing the state is renormalized to the unit length
- Because the function *f(x)* is periodic, the new amplitude distribution is normalized and has the same period as *f(x)*.

- The ***measured*** value ***γ*** corresponds to all ***k*** x_i values for which the periodic function is $\gamma = f(x_i)$
- The function $\alpha(x)$ after the measurement is defined as

$$\alpha(x) = \begin{cases} \frac{1}{\sqrt{k}} & if \quad \gamma = f(x) \\ 0 & else \end{cases} .$$

- After the measurement the state is represented as

$$\sum_{x \in B^m} \alpha(x) \cdot |x\rangle |\gamma\rangle .$$

"second" register

- We apply *QFT* that computes the discrete Fourier transform
- The discrete Fourier transform of *α(x)* is *ω(x)*
- F_m is a linear transform talking the column vector *α* to a column vector *ω*

$$F_m \cdot \sum_{x \in B^m} \alpha(x) \cdot |x\rangle|\gamma\rangle = \sum_{x \in B^m} \omega(x) \cdot |x\rangle|\gamma\rangle.$$

- We measure the first register
  - The measurement gives us a value *v* that is close to a multiple value of

$$v \approx \frac{n}{period} = \frac{2^m}{period}.$$

Three Cases

Period r happens to be power of 2, the discrete Fourier transform gives exact multiplies

$$v = t \cdot \frac{n}{r} = t \cdot \frac{2^m}{r}.$$

In this case we can estimate r by several experiments if necessary

$$\frac{v}{2^m} = \frac{t}{r}$$

where the lowest term of $\frac{v}{2^m}$ will yield a fraction $\frac{t}{r}$ whose denominator is the period r.

Three Cases

Period r is not power of 2, the discrete Fourier transform gives approximate multiples

$$v \approx t \cdot \frac{n}{r} = t \cdot \frac{2^m}{r}.$$

In this case we can estimate r by continued finite fraction expansion of $\frac{v}{2^m}$ resulting in a unique fraction $\frac{p}{q}$ with $r \approx q$. For unique fraction $\frac{p}{q}$ of $\frac{v}{2^m}$ with $q < M$

$$\left| \frac{v}{2^m} - \frac{p}{q} \right| < \frac{1}{M^2}.$$

The fraction can be obtained by the following algorithm:

$$a_0 = \left\lfloor \frac{v}{2^m} \right\rfloor, \quad \epsilon_0 = \frac{v}{2^m} - a_0, \quad p_0 = a_0, \quad q_0 = 1$$

Floor always rounds towards negative infinity, *floor(2.4)=2*

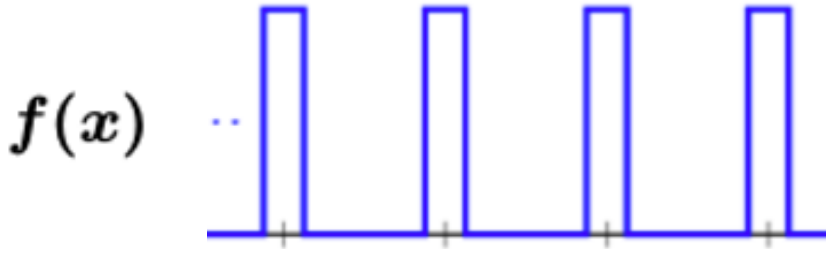
$$a_1 = \left\lfloor \frac{1}{\epsilon_0} \right\rfloor, \quad \epsilon_1 = \frac{1}{\epsilon_0} - a_0, \quad p_1 = a_1 \cdot a_0 + 1, \quad q_1 = a_1$$

$$a_i = \left\lfloor \frac{1}{\epsilon_{i-1}} \right\rfloor, \quad \epsilon_i = \frac{1}{\epsilon_{i-1}} - a_i, \quad p_i = a_i \cdot p_{i-1} + p_{i-2}, \quad q_i = a_i \cdot q_{i-1} + q_{i-2}.$$

We stop the algorithm with the output $\frac{p_i}{q_i}$ with $r \approx q_i$ if

$$q_i < M \leq q_{i+1}.$$

Three Cases



$f(x)$ is a periodic block function, the measured value γ in a block function corresponds to all k x_i values for which the periodic function is $\gamma = f(x_i)$. The amplitude function $\alpha(x)$ after the measurement has less or equal number of zeros with $n - k \leq k$.

$$\alpha(x) = \begin{cases} \frac{1}{\sqrt{k}} & if \quad \gamma = f(x) \\ 0 & else \end{cases} .$$

The $\frac{1}{\sqrt{k}}$ dominates the amplitude distribution. After the DFT the DC average of the amplitude dominates the distribution. The measured value will be with high probability $v = 1$, we cannot estimate the period.

Factorization

Peter Williston Shor developed the Shor's algorithm, a quantum algorithm for factoring **exponentially faster** than the best currently-known algorithm running on a classical computer

- The widely used RSA-public key cryptography scheme is secure (?)
- Its security corresponds to the difficulty in factoring large numbers on conventional computers
- Shor's algorithm indicated how to do it on a quantum computer in **polynomial time**

- Given an integer number *M* to be factored, a function $f(x)_M$ is defined

$$f_M(x) = a^x \ mod \ M,$$

- ***a*** is a randomly chosen coprime to *M*, means the greatest common divisor of a and *M* is *1*
- $f(x)_M$ is periodic
  - For a value ***a*** the period of a modulo *M* is *r*

$$a^r = 1 \ mod \ M,$$

- if r is an even number
  - (*r* is dependent on ***a***, if *r* is not even, chose different ***a***)

$$\left(a^{\frac{r}{2}}\right)^2 = 1 \; mod \; M$$

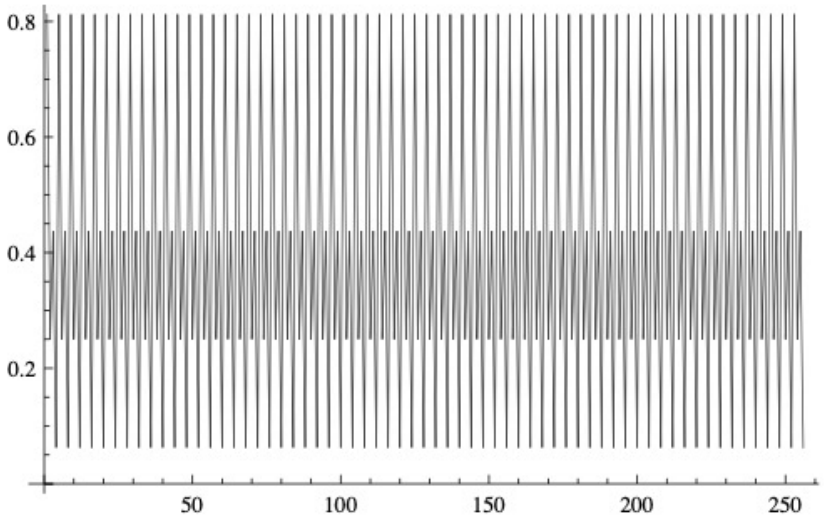
$$\left(a^{\frac{r}{2}}\right)^2 - 1 = 0 \; mod \; M \Rightarrow \left(a^{\frac{r}{2}}\right)^2 - 1^2 = 0 \; mod \; M$$

$$\left(a^{\frac{r}{2}} - 1\right) \cdot \left(a^{\frac{r}{2}} + 1\right) = 0 \; mod \; M$$

The product $\left(a^{\frac{r}{2}} - 1\right) \cdot \left(a^{\frac{r}{2}} + 1\right)$ is some integer multiple of M, because if we divide it by M the reminder is zero. A common factor between them can be efficiently determined by the greatest common divisor (gcd) Euclidean algorithm.

$$gcd\left(\left(a^{\frac{r}{2}} - 1\right), M\right), \quad gcd\left(\left(a^{\frac{r}{2}} + 1\right), M\right).$$

Example

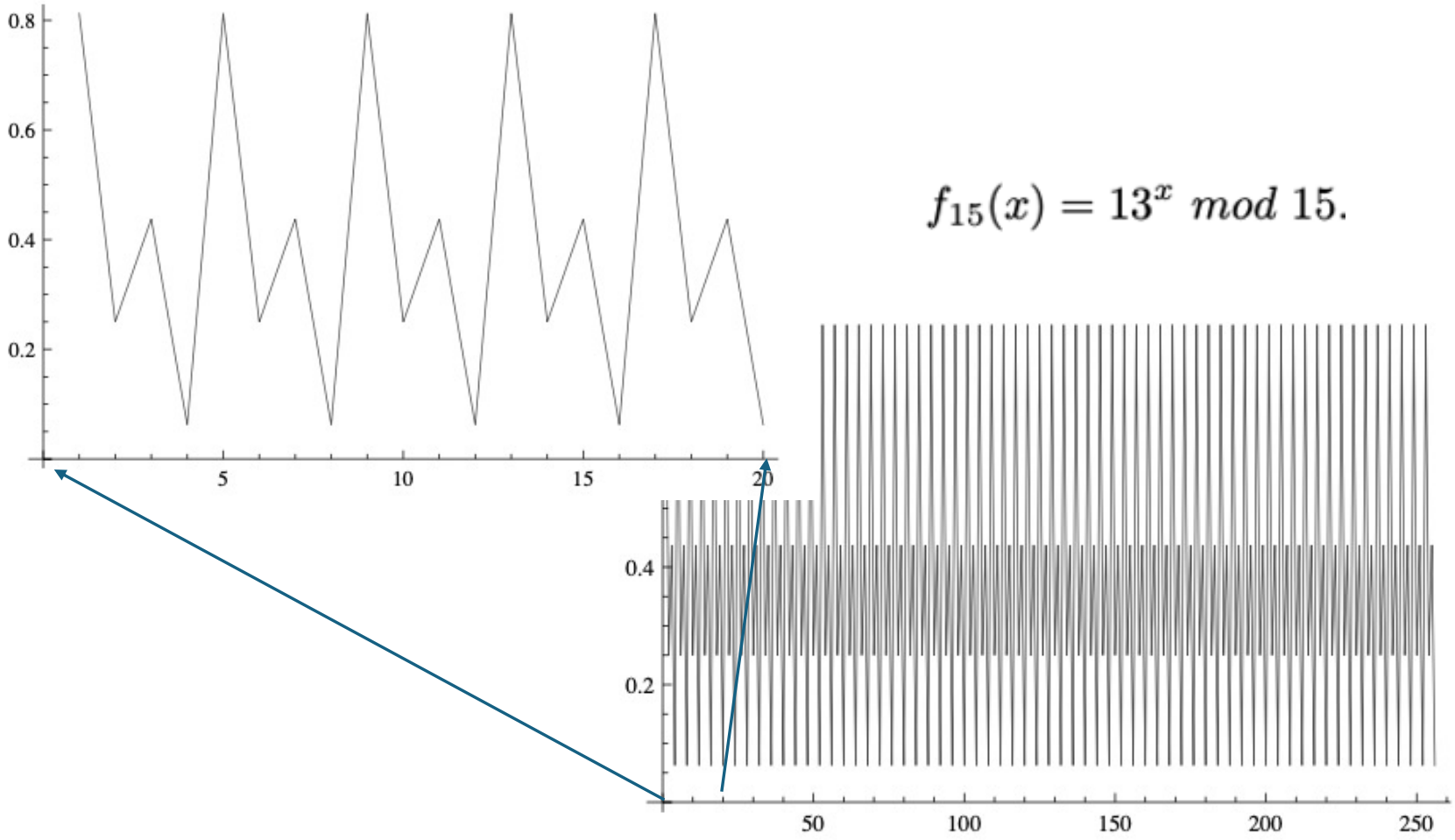


In this example we will factor the number $M = 15$. We chose $a = 13$.

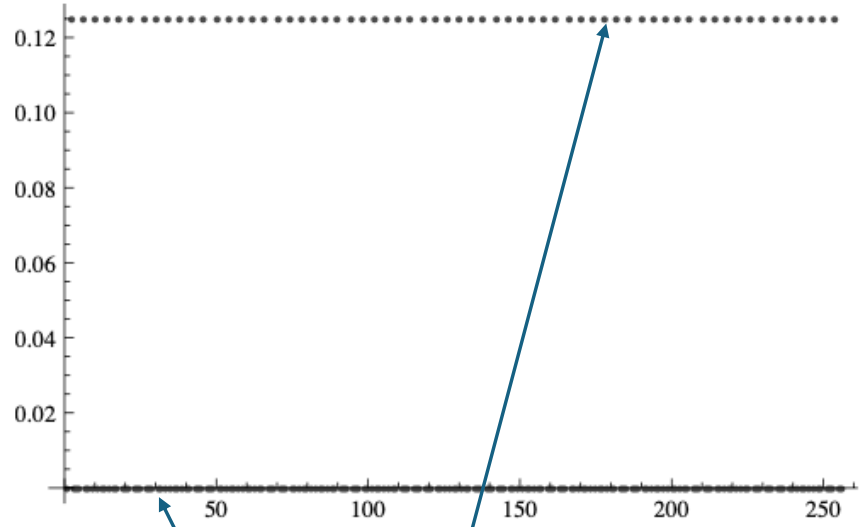
$$f_{15}(x) = 13^x \ mod\ 15.$$

We apply the $U_{f_{15}}$ operator

$$U_{f_{15}}\left(\frac{1}{\sqrt{2^8}}\sum_{x\in B^8}|x\rangle|0^{\otimes 8}\rangle\right) = \frac{1}{16}\sum_{x\in B^8}|x\rangle|f(x)\rangle.$$

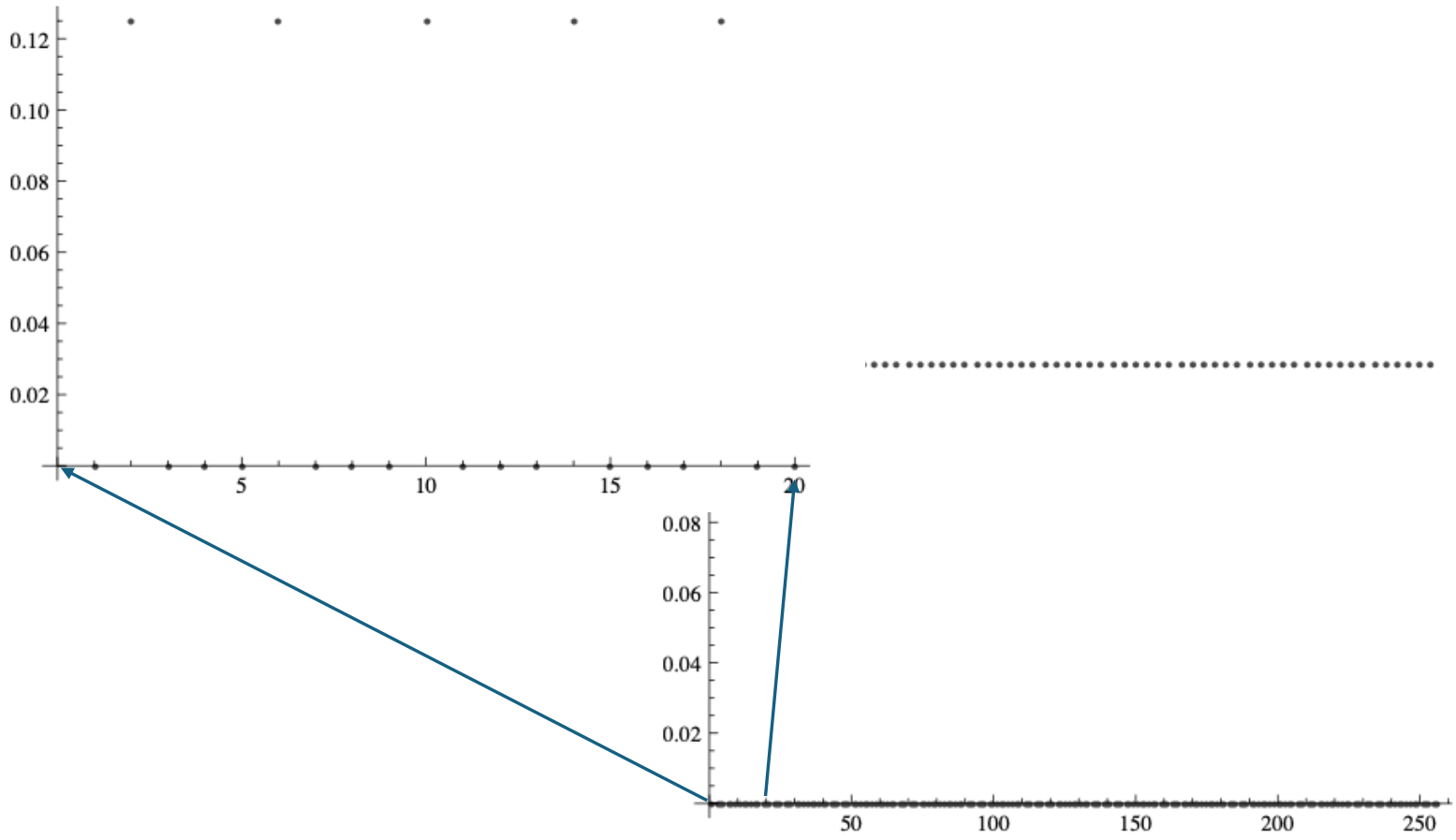


$$f_{15}(x) = 13^x \ mod\ 15.$$



- We measure the “second” register of the compound system
- In our experiment the function *α(x)* of the first register after the measurement is defined as

$$\alpha(x) = \begin{cases} \frac{1}{\sqrt{64}} & if \quad 0.25 = f(x) \\ 0 & else \end{cases}$$


0.12
0.10
0.08
0.06
0.04
0.02
5
10
15
20
0.08
0.06
0.04
0.02
50
100
150
200
250

- We apply QFT that computes the discrete Fourier transform in the first register

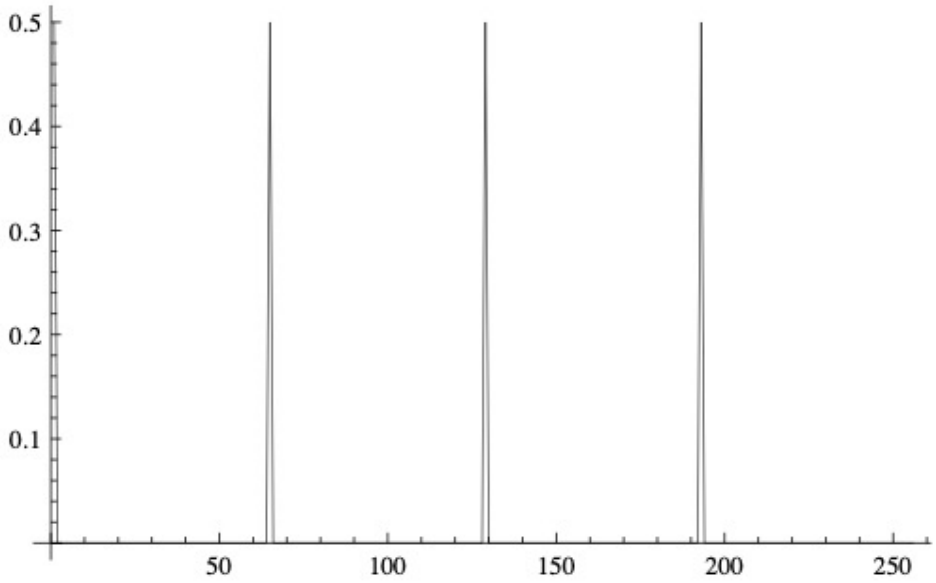


Fig. 9.8 DFT transform of the first register. It shows a strong peak at 1 and $64 + 1$, $2 \cdot 64 + 1 = 129$ and $3 \cdot 64 + 1 = 193$.

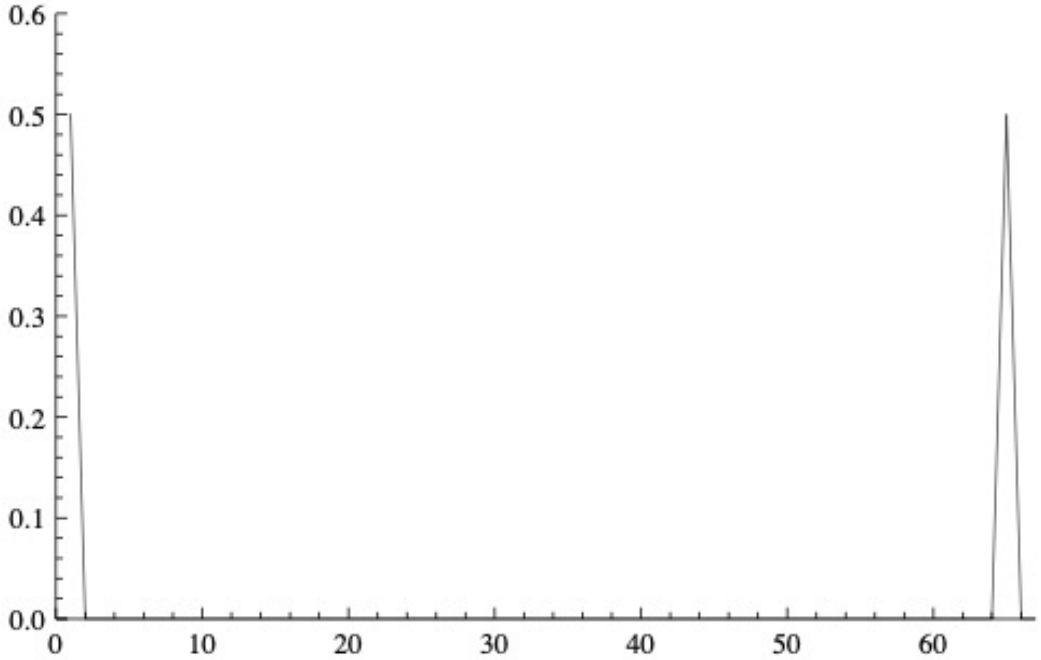


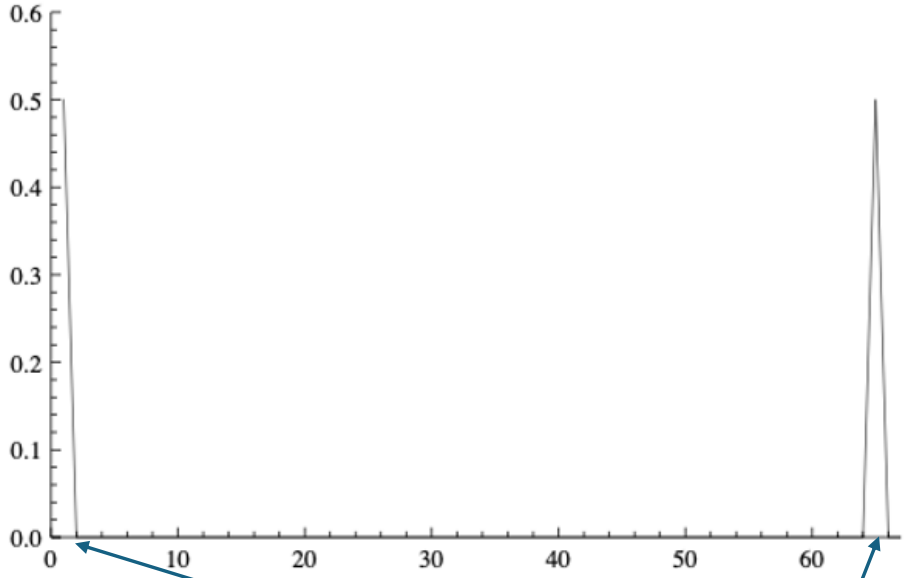
Fig. 9.9 DFT transform of the first register, higher resolution. The zero frequency term represents the DC average and appears at position 1 instead at the position 0.

- The measurement gives us a value ***v*** that is close to a multiple value of

$$\frac{n}{period}$$

- We measure the first period register
- The measurement gives us a value *64+1* that is close to a multiple value of

$$\frac{256}{period}$$

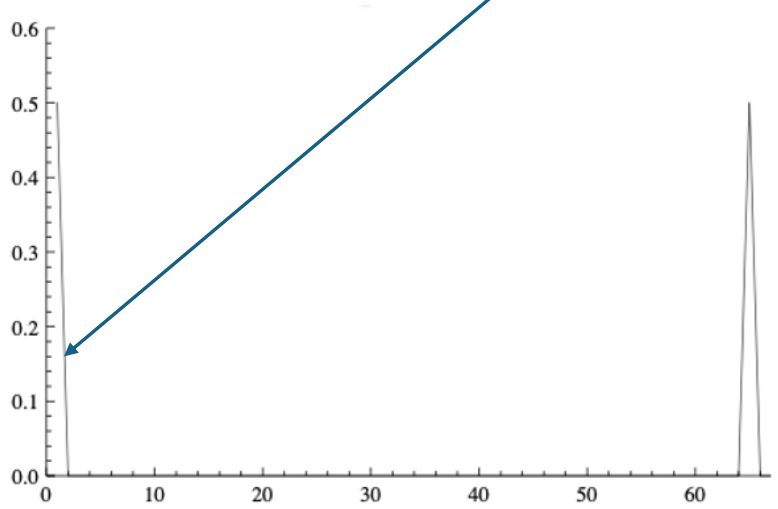


- In our experiment the period *r* happens to be power of *2*
- The zero frequency term represents the DC average and appears at position *1* instead at the position *0*, so ***v = 64***
- It follows for the period $r = \frac{256}{v} = \frac{256}{64} = 4$

A common factor between them can be efficiently determined by the greatest common divisor (gcd) Euclidean algorithm,

$$gcd\left(\left(13^{\frac{4}{2}}-1\right),15\right)=3,\quad gcd\left(\left(13^{\frac{4}{2}}+1\right),15\right)=5.$$

The factors of 15 are 3 and 5. The algorithm is probabilistic and it can fail. Suppose the measurement gives us the value 1, in this case we have to repeat the whole procedure.



Kitaev's Phase Estimation Algorithm

- Given a unitary operator U on m qubits with an eigenvector $|u\rangle$ with an unknown eigenvalue $e^{2\cdot\pi\cdot i\cdot\theta}$ we want to determine the **phase** θ

$$U \cdot |u\rangle = e^{2\cdot\pi\cdot i\cdot\theta} \cdot |u\rangle$$

eigenvector eigenvalue eigenvector

Alexei Yurievich Kitaev (Russian: Алексей Юрьевич Китаев; born August 26, 1963) is a Russian–American professor of physics at the California Institute of Technology

Idea

$$U \cdot |u\rangle = e^{2 \cdot \pi \cdot i \cdot \theta} \cdot |u\rangle$$

if we apply U to $|u\rangle$ w times we get

$$U^w \cdot |u\rangle = U^{w-1} \cdot \left(e^{2 \cdot \pi \cdot i \cdot \theta} \cdot |u\rangle\right) = \left(e^{2 \cdot \pi \cdot i \cdot \theta}\right)^w \cdot |u\rangle = e^{2 \cdot \pi \cdot i \cdot \theta \cdot w} \cdot |u\rangle.$$

- We will not gain any information
- $|u\rangle$ and $e^{2 \cdot \pi \cdot i \cdot \theta \cdot w} \cdot |u\rangle$ are equivalent states, they represent the same state when a measurement is preformed.

Controlled U^w operator CU^w

- Instead of the unitary operator U^w we use the controlled U^w operator CU^w
- If the control qubit is set then U^w is applied to the target qubits, otherwise not.
  - The operator CU^w is unitary and defines an injective mapping on two qubits that is reversible

$$CU^w \cdot |0\rangle|u\rangle = |0\rangle|u\rangle, \quad CU^w \cdot |1\rangle|u\rangle = |1\rangle \left(e^{2\cdot\pi\cdot i\cdot\theta\cdot w} \cdot |u\rangle\right) = e^{2\cdot\pi\cdot i\cdot\theta\cdot w} \cdot |1\rangle|u\rangle.$$

So with $w = 2^j$

$$CU^{2^j} \cdot \left(\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) \cdot |u\rangle\right) = \left(\frac{|0\rangle + e^{2\cdot\pi\cdot i\cdot\theta\cdot 2^j}|1\rangle}{\sqrt{2}}\right) \cdot |u\rangle.$$

QFT and $C_{j+1}U^{2j}$

- The QFT can be factored into the tensor product of *m* single-qubit operations

$$F_m \cdot |x\rangle = \frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle =$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_1} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_2 x_1} \cdot |1\rangle\right) \otimes \cdots \otimes$$

$$\otimes \left(|0\rangle + e^{2 \cdot \pi \cdot i \cdot 0.x_m \cdots x_2 x_1} \cdot |1\rangle\right).$$

- For *m* control qubits we define $C_{j+1}U^{2j}$ in the following way:
- For $j \in \{0,1,2,\cdots,m-1\}$ the control qubit ***j + 1*** of the *m* qubits is set then $C_{j+1}U^{2j}$ is applied to the target $|u\rangle$, otherwise not.

The Algorithm

- The initial state of the algorithm is

$$|0^{\otimes m}\rangle|u\rangle$$

u being the eigenvector of *U*

- In the first step we build a superposition of *m* control qubits

$$H_m \cdot |0^{\otimes m}\rangle|u\rangle = \frac{1}{\sqrt{2^m}} \sum_{x \in B^m} |x\rangle|u\rangle =$$

$$= \frac{1}{\sqrt{n}} \cdot (|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)\,|u\rangle.$$

- In the second step we apply *m* $C^{j+1}U^{2j}$ operators to the target $|u\rangle$

$$\prod_{j=0}^{m-1} C_{j+1}U^{2^j} \cdot \left(\frac{1}{\sqrt{n}} \cdot (|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)\,|u\rangle\right) =$$

$$= \frac{1}{\sqrt{n}} \cdot \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot\left(\theta\cdot 2^{m-1}\right)} \cdot |1\rangle\right) \otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot\left(\theta\cdot 2^{m-2}\right)} \cdot |1\rangle\right) \otimes \cdots$$

$$\otimes \left(|0\rangle + e^{2\cdot\pi\cdot i\cdot\left(\theta\cdot 2^{0}\right)} \cdot |1\rangle\right) \cdot |u\rangle$$

- In the third step we apply inverse QFT to the m control qubit

$$IF_m \cdot \left(\frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot y \cdot \theta} \cdot |y\rangle \right) \cdot |u\rangle =$$

$$= IF_m \cdot \left(\frac{1}{\sqrt{n}} \sum_{y \in B^m} e^{2 \cdot \pi \cdot i \cdot \frac{y}{n} \cdot x} \cdot |y\rangle \right) \cdot |u\rangle = |x\rangle |u\rangle.$$

- In the fourth step we measure the first register composed of *m* control qubits and estimate θ

$$\theta = 0.x_m \cdots x_2 x_1 = \frac{x}{n} = \frac{x}{2^m}.$$

Example with T Gate

The T gate corresponds to the unitary matrix

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}.$$

With the eigenvector $|1\rangle$ and the eigenvalue $e^{2\cdot\pi\cdot i\cdot\theta}$

$$U \cdot |u\rangle = T \cdot |1\rangle = e^{2\cdot\pi\cdot i\cdot\theta} \cdot |u\rangle = e^{2\cdot\pi\cdot i\cdot\theta} \cdot |1\rangle$$

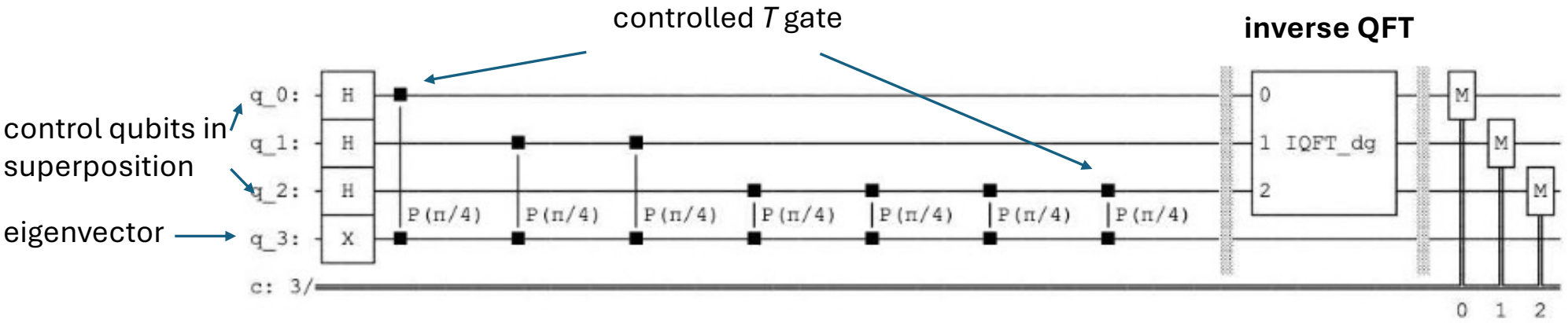
with the phase is $\theta = \frac{1}{8}$ since

$$T|1\rangle = e^{2\cdot\pi\cdot i\cdot\theta} = e^{i\frac{\pi}{4}} \cdot |1\rangle = e^{2\cdot i\frac{\pi}{8}} \cdot |1\rangle.$$

- The phase estimation algorithm will write the phase of *T* to the *m* qubits in the control register
- The value of *m* determines the precision of the result
- In our simple case *m* = *3*

$$\theta = 0.x_3x_2x_1 = \frac{x}{8} = \frac{x}{2^3}.$$

- The phase θ estimate should be *1/8*



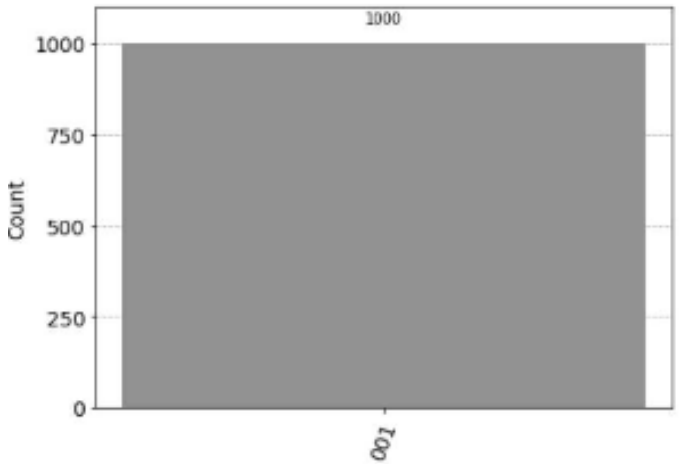
- The controlled *T* gate is represented by the controlled phase gate *CP(λ)* with $\lambda = \pi/4$
  - Qubits *0, 1, 2* represent the *3* qubits in the control register.
- The qubit *3* represents the **eigenvector** $|1\rangle$.
- The control register controls the unitary operations *T* applied to the target eigenvector $|1\rangle$ resulting in the Fourier basis representation of the three control qubits.
  - To estimate the phase *θ* we perform the **inverse QFT** and measure the three qubits and estimate *θ*

- To estimate the phase θ we perform the **inverse QFT** and measure the three qubits and estimate θ

$$\theta = 0.x_m \cdots x_2 x_1 = \frac{x}{n} = \frac{x}{2^m}.$$

$$x = |q_2 q_1 q_0\rangle$$

$$\theta = \frac{x}{2^3}.$$



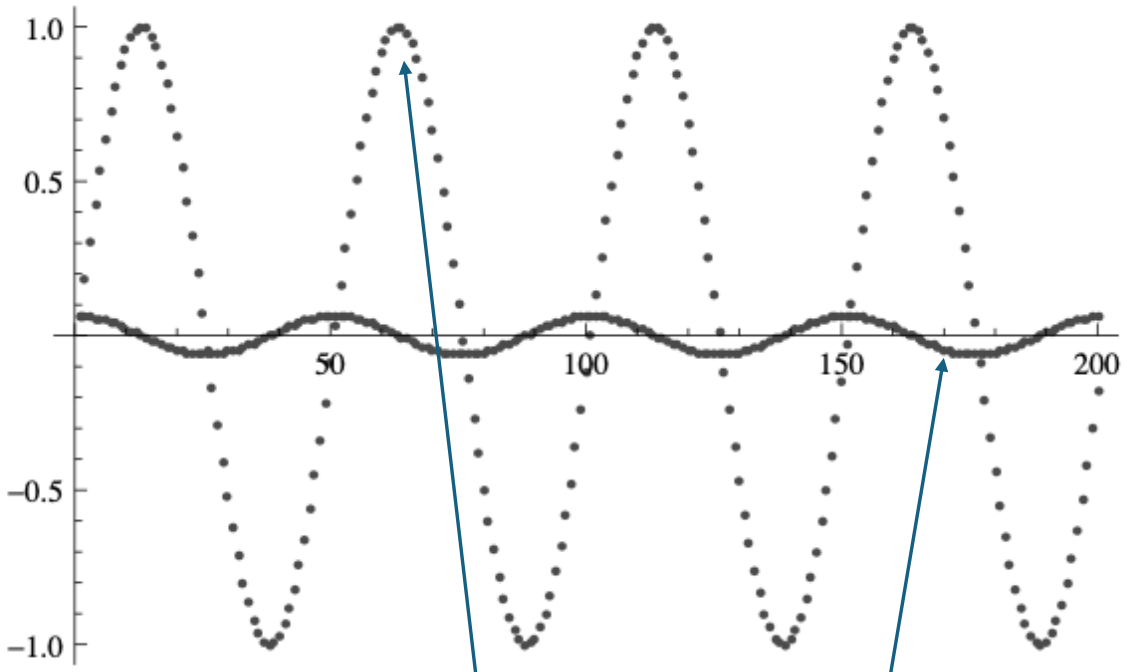
- The measured value corresponds to the binary value *001* equal to one indicating phase $\theta = 0.125 = 1/2^3$

Quantum Counting

- In Grover's amplification the number of iterations r is the largest integer not greater than t^*

$$r = \left\lfloor \frac{\pi}{4} \cdot \sqrt{\frac{n}{k}} \right\rfloor$$

- The value of r depends on the relation of *n* versus *k*, with *k* being the number of solutions
- We can estimate *k* by quantum counting
  - we use the quantum phase estimation algorithm to find an **eigenvalue** of a Grover search iteration



The values are n = 256, k = 1 and 1 ≤ t ≤ 200

One can represent the state $|\tau\rangle$ after t Grover's amplification by two subspaces $|\tau_{solution}\rangle$ and $|\tau_{non}\rangle$ representing the states representing the solutions and non solutions with

$$|\tau\rangle = \alpha_t \cdot |\tau_{solution}\rangle + \beta_t \cdot |\tau_{non}\rangle$$

- After amplification we get

$$\sqrt{\frac{k}{n}} \cdot |\tau_{solution}\rangle + \sqrt{\frac{n-k}{n}} \cdot |\tau_{non}\rangle$$

$$\sin^2\theta = \frac{k}{n}$$

$$\cos^2\theta = \frac{n-k}{n} = 1 - \frac{k}{n}.$$

We can represent the Grover's amplification matrix in the two dimensional basis $|\tau_{solution}\rangle$ and $|\tau_{non}\rangle$

$$G = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$G = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

The matrix G has two eigenvectors:

$$\mathbf{u}_1 = \begin{pmatrix} \frac{i}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} \frac{-i}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

with two eigenvalues $\lambda_1 = e^{2 \cdot i\theta}$ and $\lambda_2 = e^{-2 \cdot i\theta}$.

- The eigenvectors $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are represented in the $|\tau_{solution}\rangle$ and $|\tau_{non}\rangle$ basis as

$$|\tau_1\rangle = \frac{i}{\sqrt{2}} \cdot |\tau_{solution}\rangle + \frac{1}{\sqrt{2}} \cdot |\tau_{non}\rangle$$

$$|\tau_2\rangle = \frac{i}{\sqrt{2}} \cdot |\tau_{solution}\rangle - \frac{1}{\sqrt{2}} \cdot |\tau_{non}\rangle$$

$$|\tau\rangle = e^{i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_1\rangle + e^{-i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_2\rangle \longleftrightarrow |\tau\rangle = \alpha_t \cdot |\tau_{solution}\rangle + \beta_t \cdot |\tau_{non}\rangle$$

Superposition of Eigenvectors

In the original quantum phase estimation algorithm the required eigenvector

$$U \cdot |u\rangle = e^{2 \cdot \pi \cdot i \cdot \theta} \cdot |u\rangle$$

However we do not need to prepare our register in either of these eigenvectors, the register is actually in a superposition of the eigenvectors of the Grover operator

$$G \cdot |\tau\rangle = G \cdot \left(e^{i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_1\rangle + e^{-i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_2\rangle \right)$$

$$G \cdot |\tau\rangle = \lambda_1 \cdot \left(e^{i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_1\rangle \right) + \lambda_2 \cdot \left(e^{-i\theta} \cdot \frac{1}{\sqrt{2}} \cdot |\tau_2\rangle \right)$$

$$U \cdot |u\rangle = e^{2 \cdot \pi \cdot i \cdot \theta} \cdot |u\rangle$$

Since our eigenvalues are $\lambda_1 = e^{2 \cdot i\theta}$ and $\lambda_2 = e^{-2 \cdot i\theta}$ and we have m control qubits

$$\theta = 0.x_m \cdots x_2 x_1 = \frac{x}{n} = \frac{x}{2^m}. \qquad \theta = \frac{x \cdot \pi}{2^m} \tag{20.20}$$

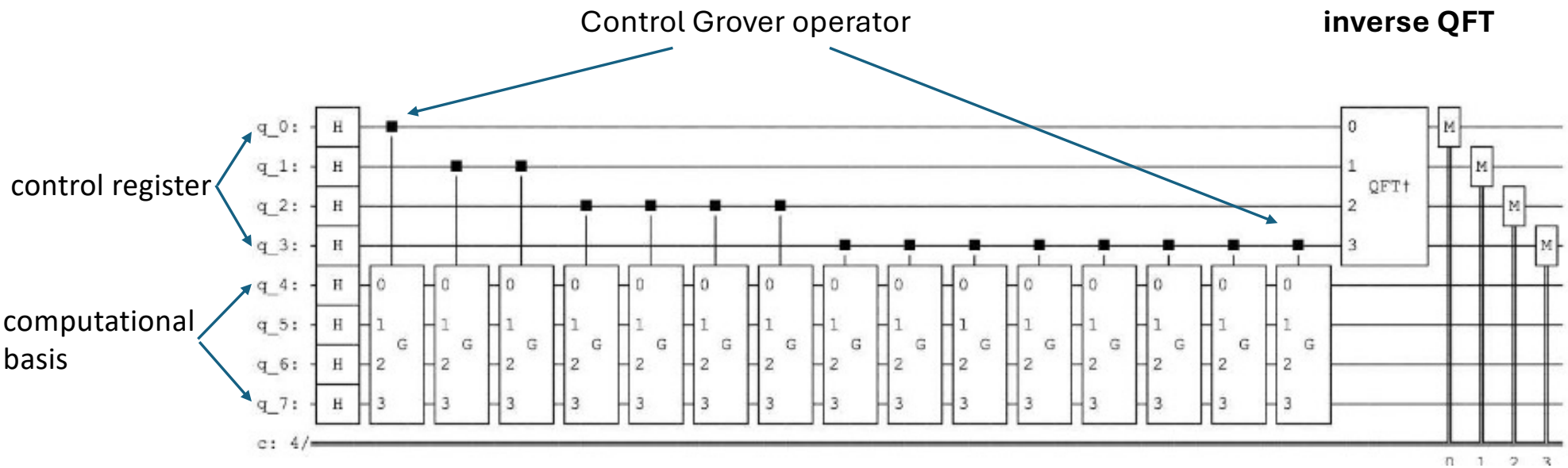
and with n being the number of state of Grover's amplification

$$\sin^2 \theta = \frac{k}{n} \tag{20.21}$$

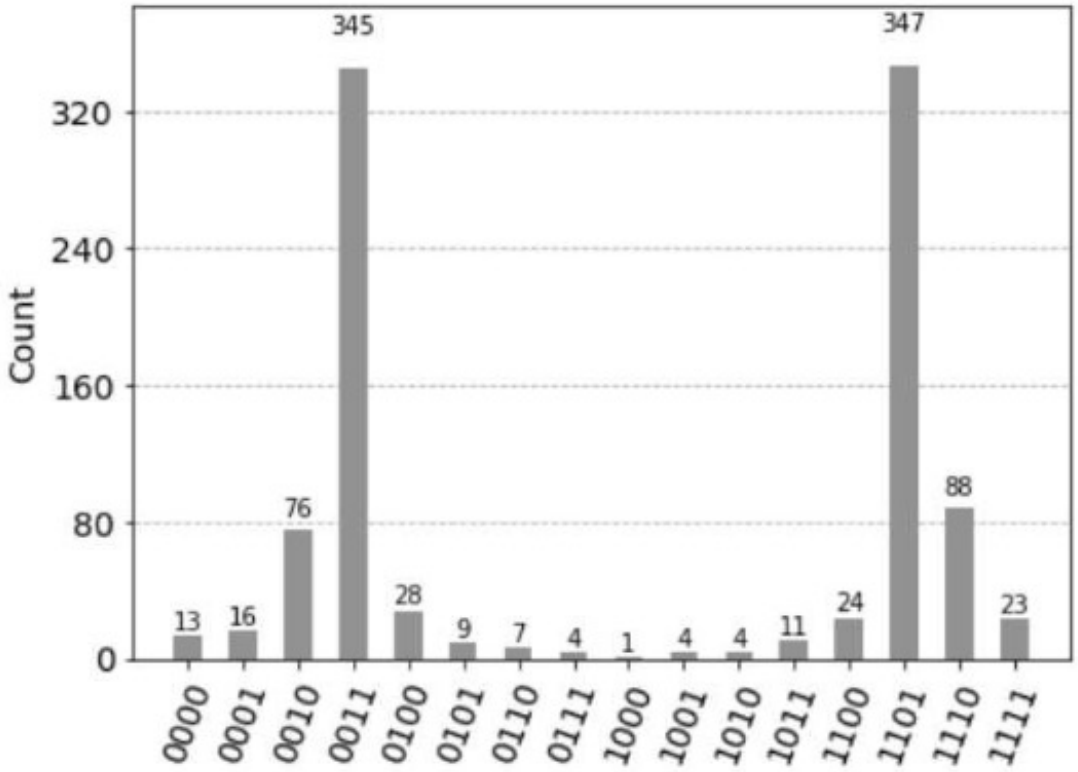
with

$$k = \sin^2 \theta \cdot n = \sin^2 \left(\frac{x \cdot \pi}{2^m} \right) \cdot n.$$

We use the *.control()* method to create a controlled gate from from the Grover operator



The controlled Grover operator is implemented using use the circuit library with an oracle that marks four solutions (***k = 4***) out of *16* state



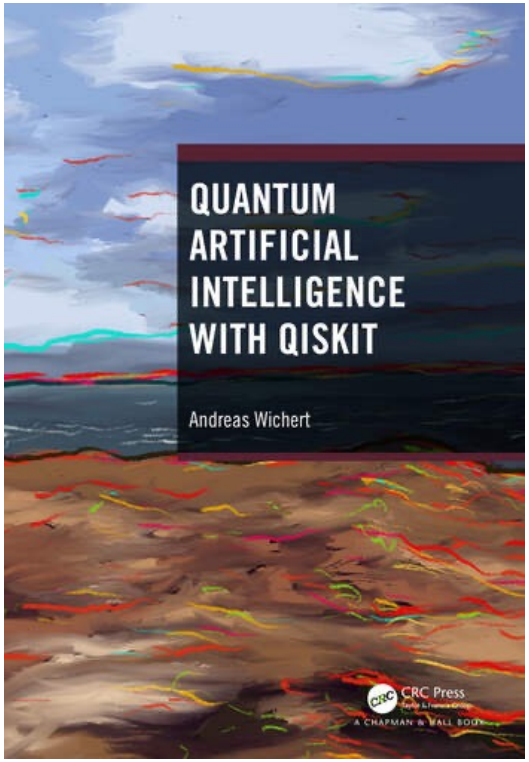
The maximal measured value corresponds to the binary value 0011 (3 decimal) or 1101 (13 decimal) indicating phase $\theta_1 = \pi \cdot 3/16$ or $\theta_2 = \pi \cdot 13/16$.

$$k = 4.9 \approx \sin^2\left(\frac{\pi \cdot 3}{16}\right) \cdot 16 = sin^2\left(\frac{\pi \cdot 13}{16}\right) \cdot 16.$$

In our example four control qubits lead only to **an approximate estimation** of the phase θ.

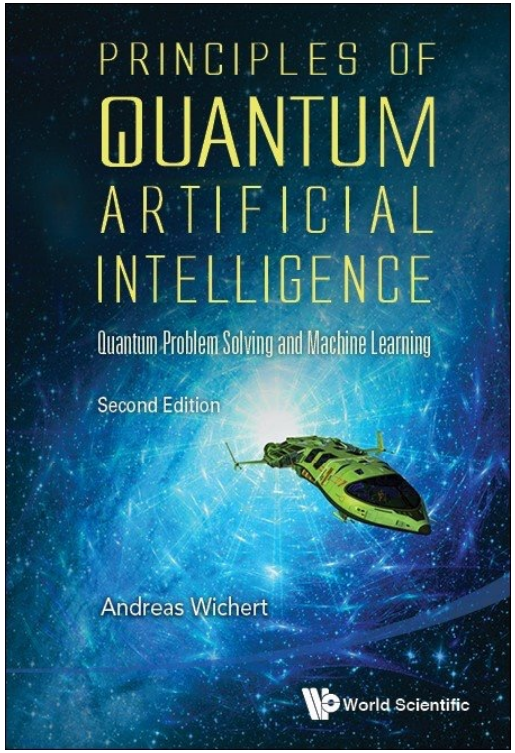
- The more control qubits (m) we use, the higher precision we get
- In our example four control qubits lead only to an **approximate estimation** of the phase θ

- More control qubits would be required for less approximate result



- Chapter 19
- Chapter 20

Quantum Artificial Intelligence with Qiskit, A. Wichert, Chapman and Hall/CRC, 2024



- Chapter 9

Principles of Quantum Artificial Intelligence: Quantum Problem Solving and Machine Learning, 2nd Edition, A. Wichert, World Scientific, 2020