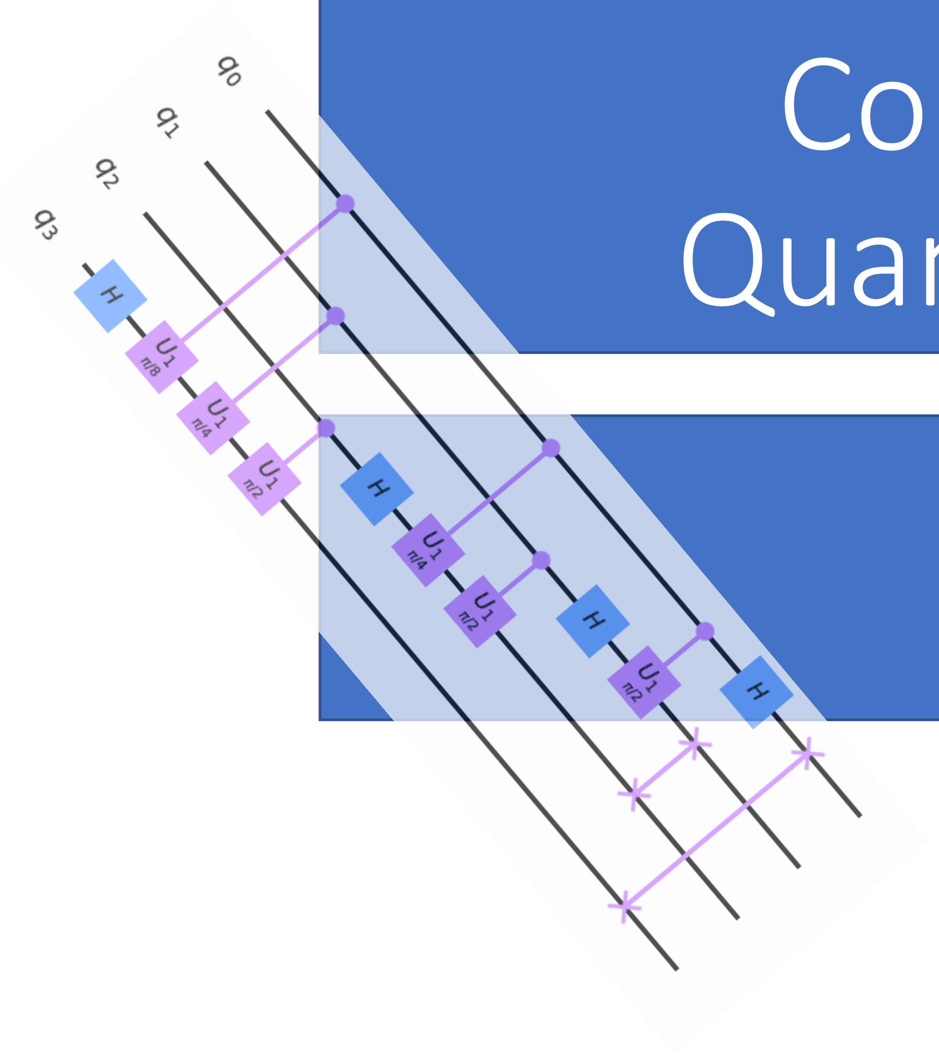
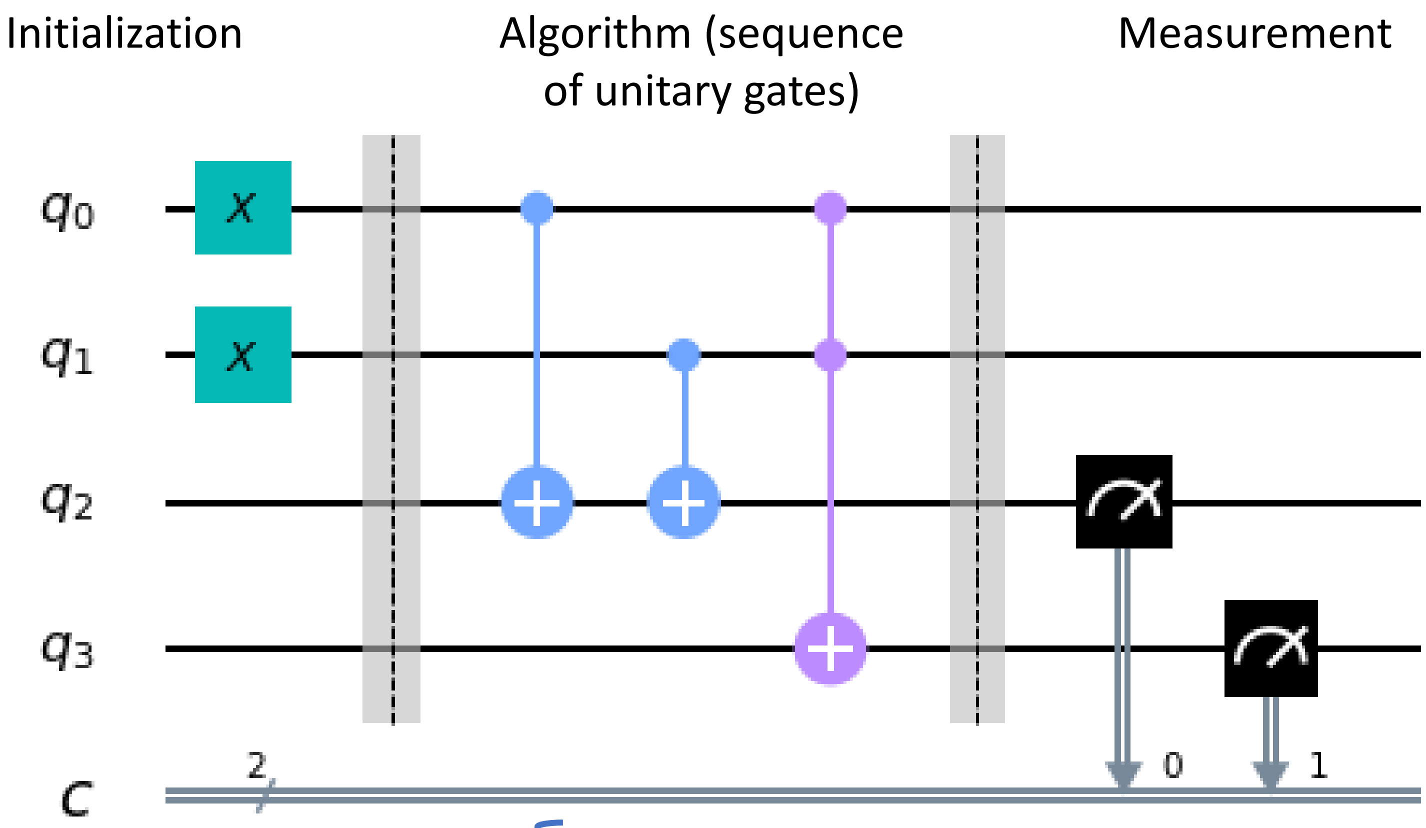
5. Principles of Quantum Computation and Quantum Algorithms

Quantum Computing



UNIVERSITÀ DI PARMA

Quantum circuits



Main differences from Classical Computer:

1. Inputs can be prepared in any superposition state
2. Quantum gates are unitary operators
3. Any measurement modifies the state of the qubits. You cannot simply stop, check and restart

Reversible calculation

Most logic gates are irreversible, because they correspond to a transformation 2 bits → 1 bit and the final state of a single bit does not allow to reconstruct the initial 2-bit state. E.g.:

XOR	
00	0
01	1
10	1
11	0

Equivalent reversible operation →

CNOT	
00	0**0**
01	0**1**
10	1**1**
11	1**0**

Any **irreversible** computation can be transformed into a **reversible** computation (usually by adding some extra lines to the circuit).

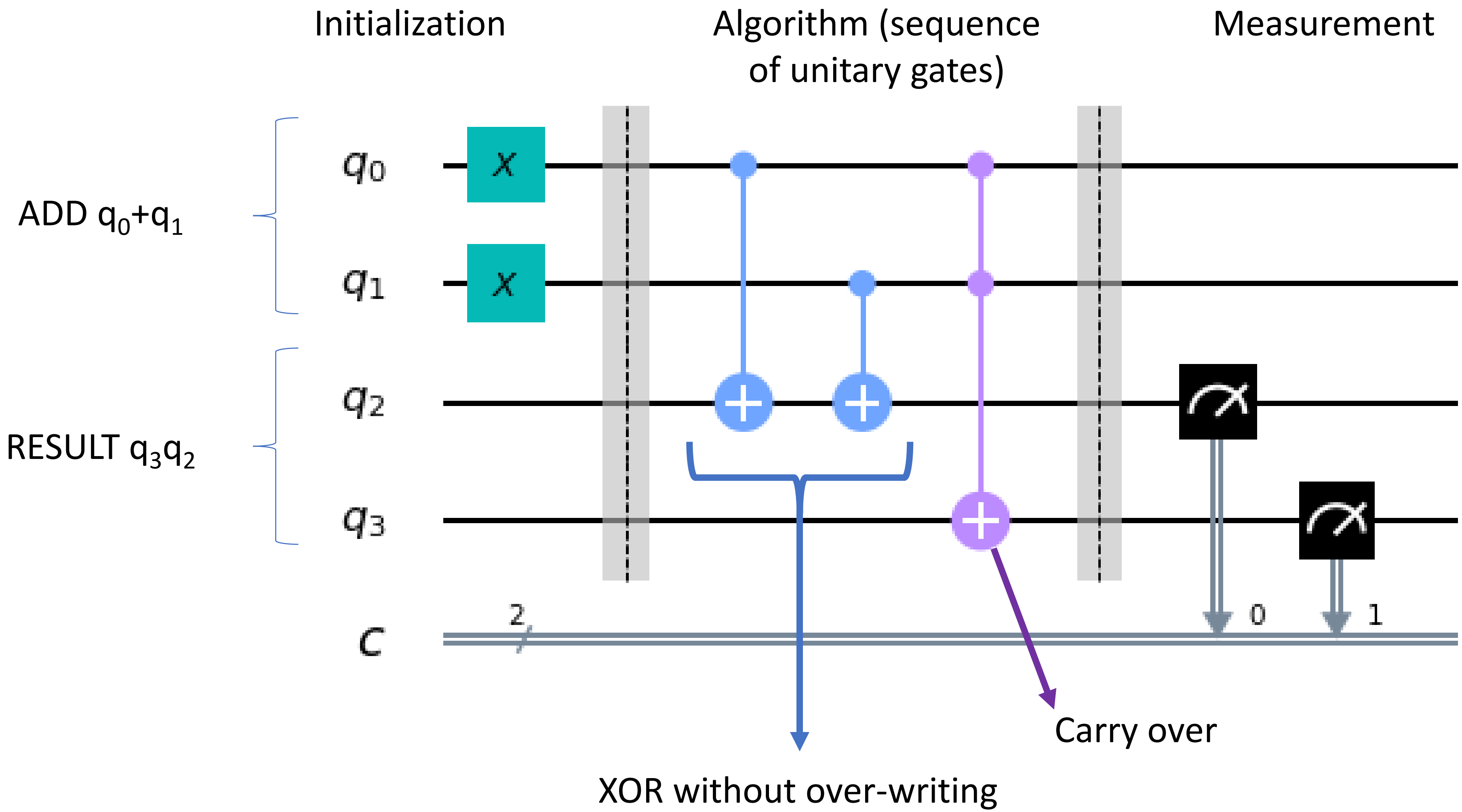
$$(x, y) \rightarrow (x, x \oplus y)$$

Using the CNOT and single-bit gates we can obtain linear Boolean functions.

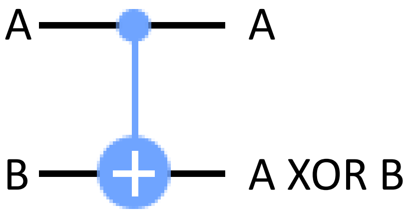
The Toffoli gate (non-linear) allows us to reproduce reversibly all classical Boolean functions.

$$(x, y, z) \rightarrow (x, y, z \oplus xy)$$

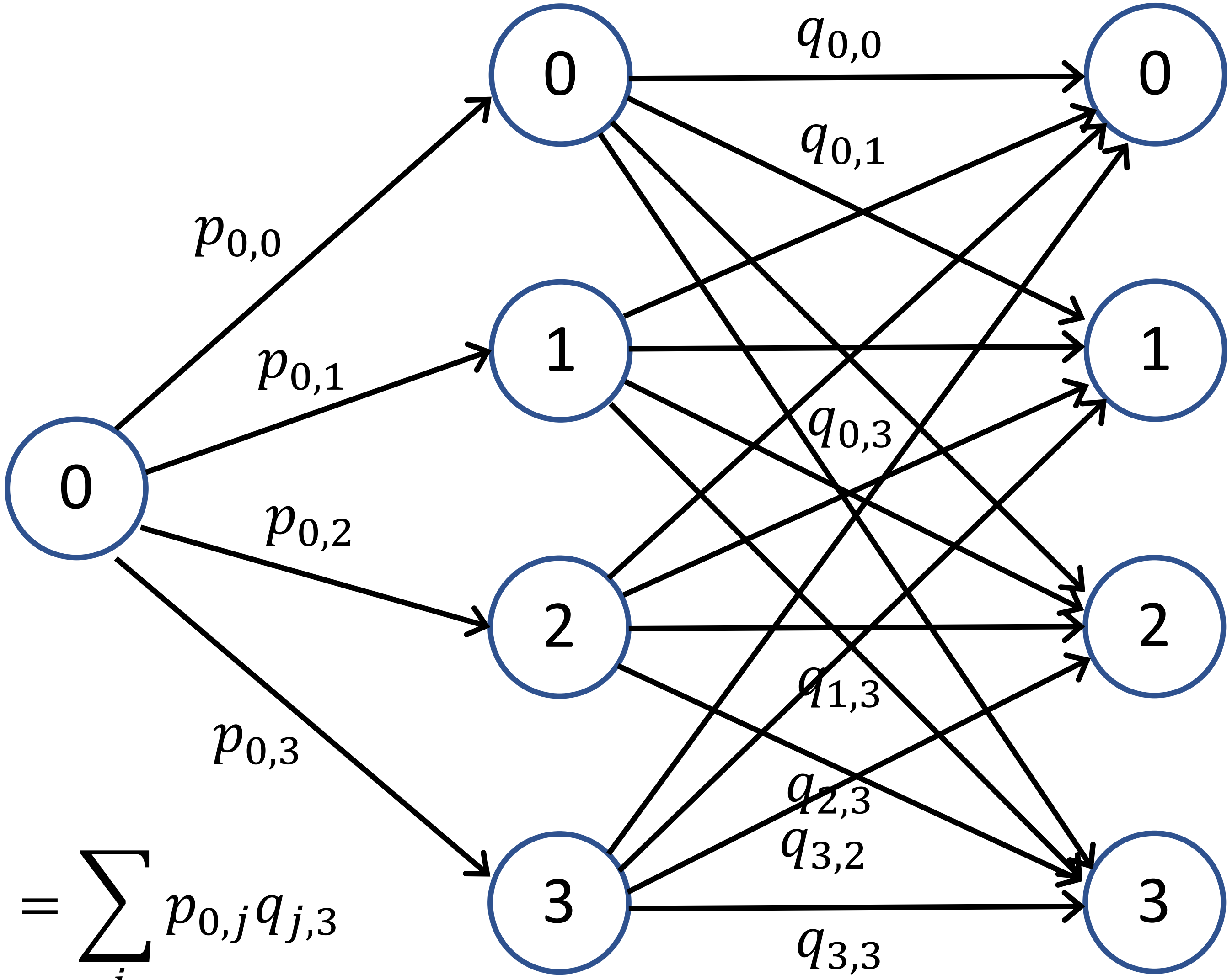
Adder circuit on Qiskit



q_0	q_1	q_0+q_1
0	0	00
0	1	01
1	0	01
1	1	10

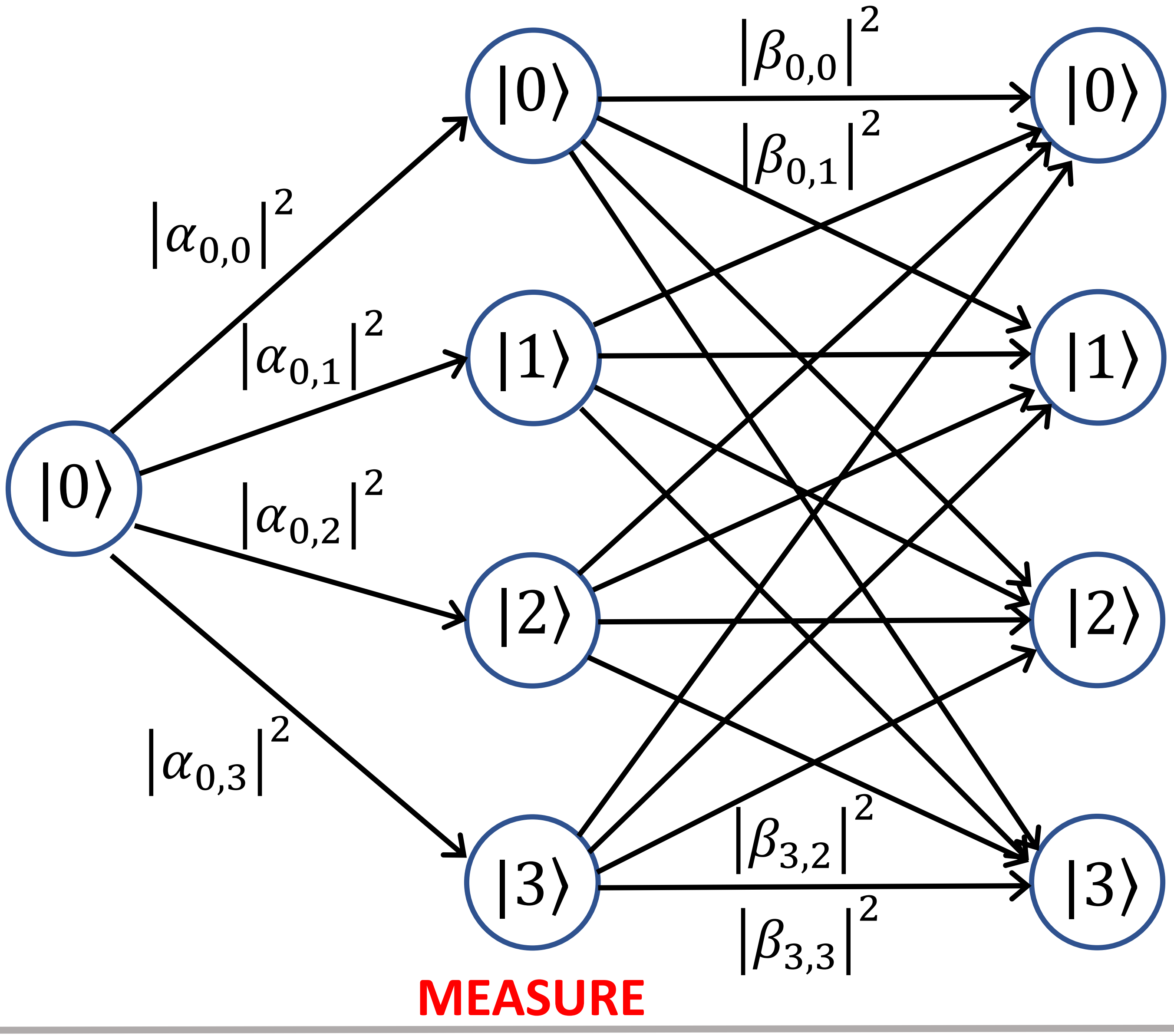


Probabilistic vs. Quantum Algorithms



$$P(\text{fin} = 3) = \sum_j p_{0,j} q_{j,3}$$

Probabilistic vs. Quantum Algorithms



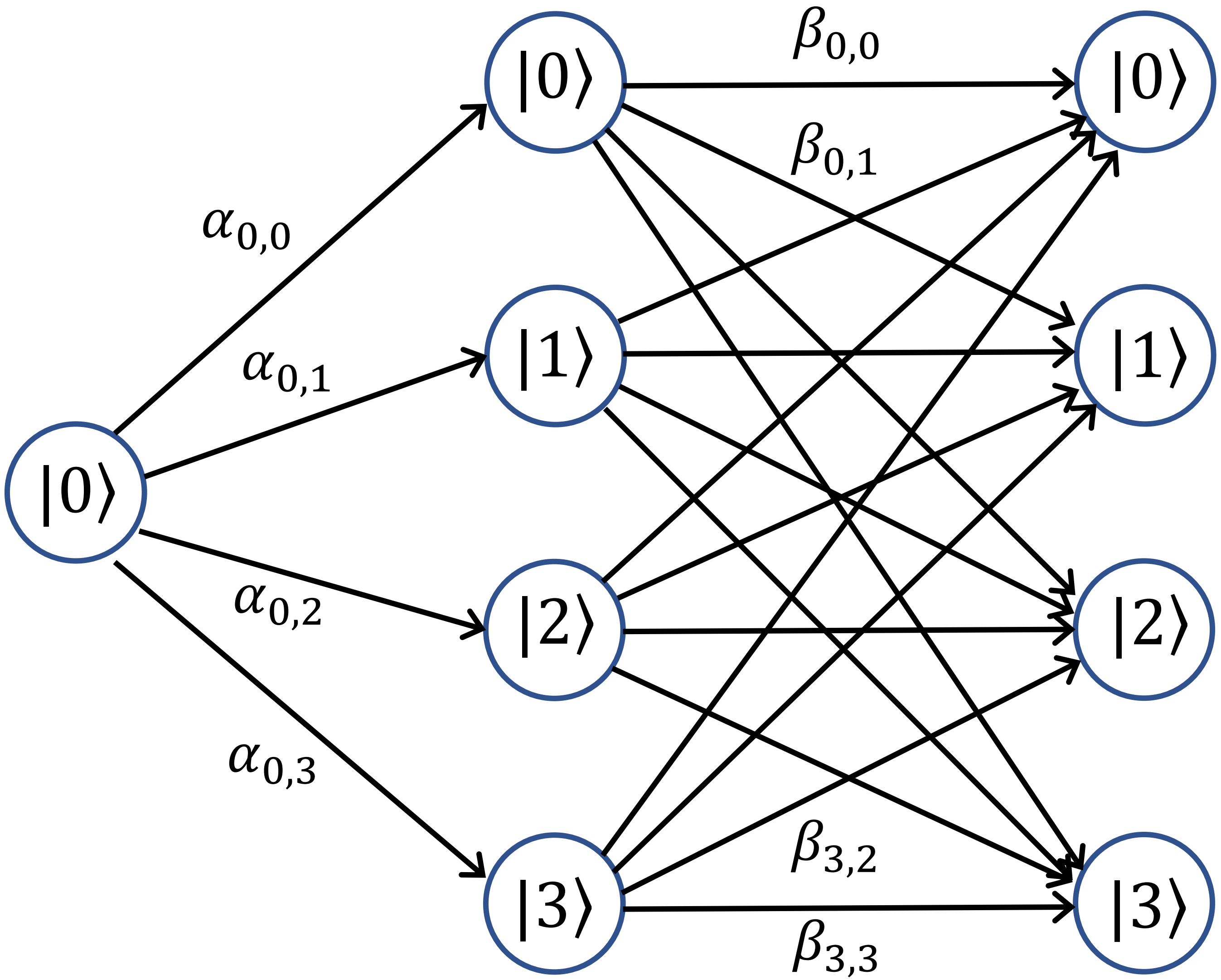
$$p_{0,j} = |\alpha_{0,j}|^2$$

$$q_{j,k} = |\beta_{j,k}|^2$$

$$P(\text{fin} = 3) = \sum_j |\alpha_{0,j}|^2 |\beta_{j,3}|^2$$

$$= \sum_j |\alpha_{0,j}\beta_{j,3}|^2$$

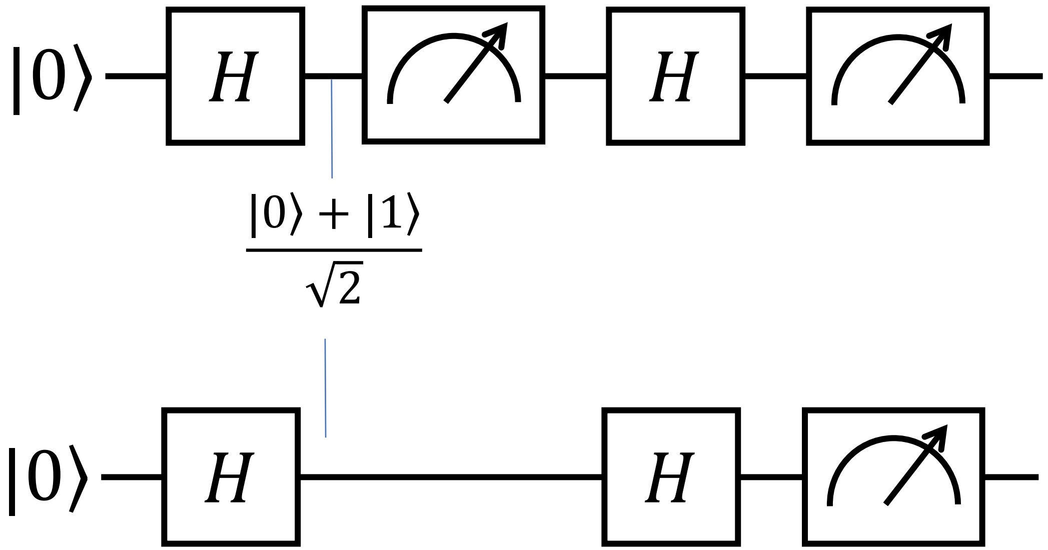
Fully quantum computation



$$P(\text{fin} = 3) = \left|\sum_j \alpha_{0,j}\beta_{j,k}\right|^2$$
$$\neq \sum_j \left|\alpha_{0,j}\beta_{j,k}\right|^2$$

INTERFERENCE

A circuit with quantum interference



No quantum interference: we finally get either $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ or $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ with 0.5 probability

Quantum interference: we finally get $|0\rangle$ with probability 1.

- Classical probabilistic algorithms can always be easily simulated by quantum algorithms.
- Classical probabilistic algorithms can also efficiently simulate quantum algorithms with small amount of entanglement (Gottesmann-Knill th.)

P. Kaye, R. Laflamme, M. Mosca, *An introduction to Quantum Computing, Oxford University Press*

Principles of Quantum Computation

A quantum processor would produce the transformation

$$|x\rangle \rightarrow U|x\rangle = |f(x)\rangle$$

desired binary number $\leq 2^n - 1$
(n is the number of qubits)

any function of x, $0 \leq f(x) \leq 2^n - 1$

However, this is not true for all functions. Indeed, unitary transformations preserve the overlap between any pair of states. Hence, given two input states $|x_1\rangle \neq |x_2\rangle$ such that $|f(x_1)\rangle = |f(x_2)\rangle$

$$|\langle f(x_1)|f(x_2)\rangle| = 1 \qquad\qquad 0 = \langle x_1|x_2\rangle = \langle x_1 U^\dagger | U x_2\rangle$$

$$\Longrightarrow U|x\rangle \neq |f(x)\rangle \qquad \text{at least for some } x$$

To **reversibly** compute **any** function, we introduce a second bit string (initialized in $|y\rangle$), so that the processor performs the transformation

$$|x\rangle\otimes|y\rangle \longrightarrow U|x\rangle\otimes|y\rangle = |x\rangle\otimes|y\oplus f(x)\rangle$$

Now $|x_1\rangle\otimes|y\oplus f(x_1)\rangle$ and $|x_2\rangle\otimes|y\oplus f(x_2)\rangle$ are orthogonal even if $f(x_1) = f(x_2)$.

String of bits in which each bit is determined by modulo 2 addition of the bit strings y and $f(x)$

Principles of Quantum Computation

If $y = 0$ a measurement of the final state of the second string of qubits directly returns $f(x)$

$$|x\rangle = \mathrm{H}^{\otimes n}|0\rangle^{\otimes n} = 2^{-n/2}(|0\rangle + |1\rangle)\otimes(|0\rangle + |1\rangle)\otimes\cdots\otimes(|0\rangle + |1\rangle) = 2^{-n/2}\sum_{\nu=0}^{2^n-1}|\nu\rangle$$

$$U|x\rangle\otimes|0\rangle = 2^{-n/2}\sum_{\nu=0}^{2^n-1}|\nu\rangle\otimes|f(\nu)\rangle$$

Highly entangled output

The **single** quantum processor **computes simultaneously** the values of $f(\nu)$ for **all** ν, in the sense that states corresponding to all of these values are present in the transformed state

Origin of the **quantum speed-up**: performing U with an array of quantum gates requires a time that is polynomial in n. The prepared state, however, contains a superposition of 2^n values, so our processor has performed an exponential (in n) number of calculations in a polynomial time. We can expect, at least for some problems, an exponential speed up using a quantum computer.

Phase kick-back

$$\text{CNOT}: |0\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} \longrightarrow |0\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

$$\text{CNOT}: |1\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} \longrightarrow -|1\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

$\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ is an eigenstate of X gate with eigenvalue -1. The resulting phase can be moved in front of the control qubit (Note that in the Hadamard gate the role of control and target are swapped).

$$\text{CNOT}: |x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} \longrightarrow (-1)^x|x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} \qquad x\in\{0,1\}$$

$$\text{CNOT}: (\alpha|0\rangle+\beta|1\rangle)\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} \longrightarrow (\alpha|0\rangle-\beta|1\rangle)\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

Z gate on the control qubit

$$U|x\rangle\otimes|y\rangle = |x\rangle\otimes|y\oplus f(x)\rangle$$

$$\boxed{U|x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}} = |x\rangle\otimes\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}} = \begin{cases} |x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} & f(x)=0 \\ -|x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}} & f(x)=1 \end{cases} \boxed{=(-1)^{f(x)}\,|x\rangle\otimes\frac{|0\rangle-|1\rangle}{\sqrt{2}}}$$

Phase kick-back

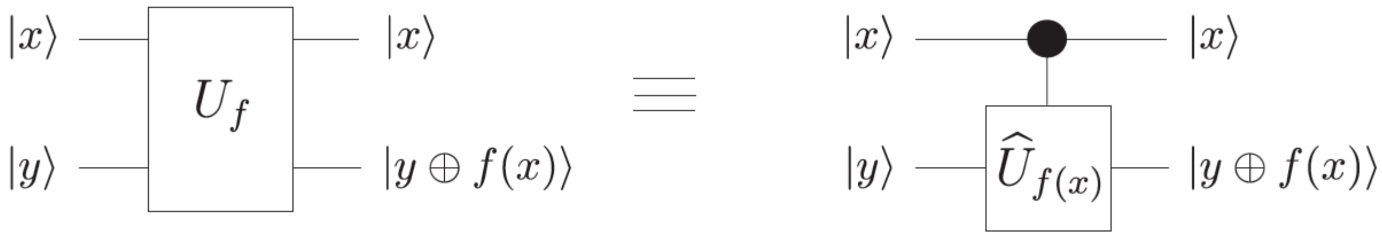


Fig. 6.6 The 2-qubit gate $U_f : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$ can be thought of as a 1-qubit gate $\widehat{U}_{f(x)}$ acting on the second qubit, controlled by the first qubit.

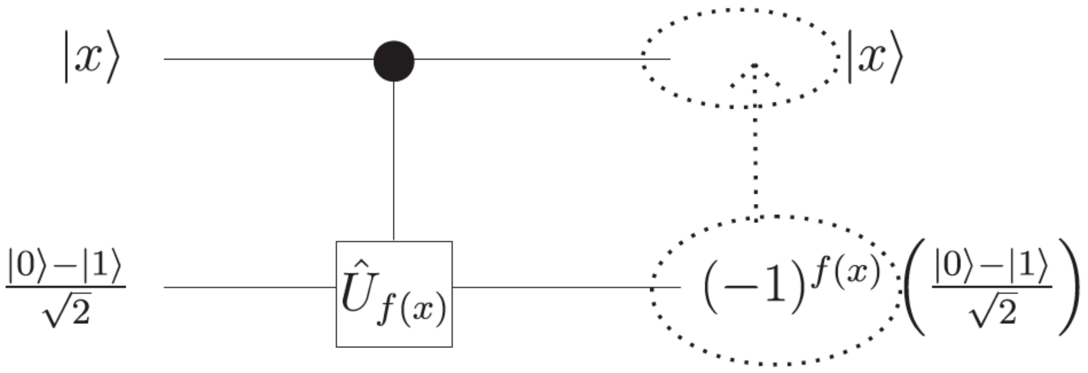
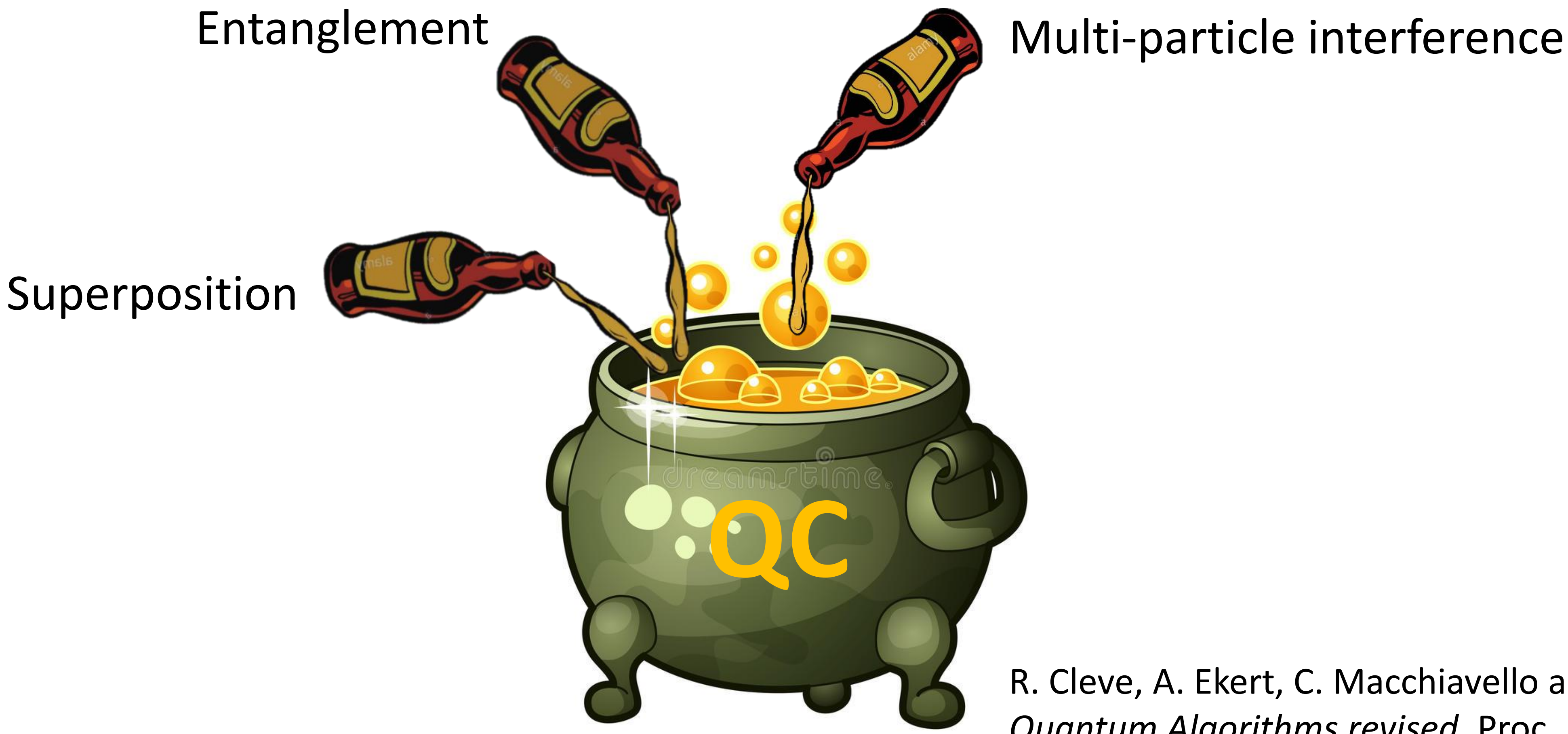


Fig. 6.7 The state $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ of the target register is an eigenstate of $\hat{U}_{f(x)}$. The eigenvalue $(-1)^{f(x)}$ can be 'kicked back' in front of the target register.

P. Kaye, R. Laflamme, M. Mosca, *An introduction to Quantum Computing, Oxford University Press*

Basic Ingredients of Quantum Computation

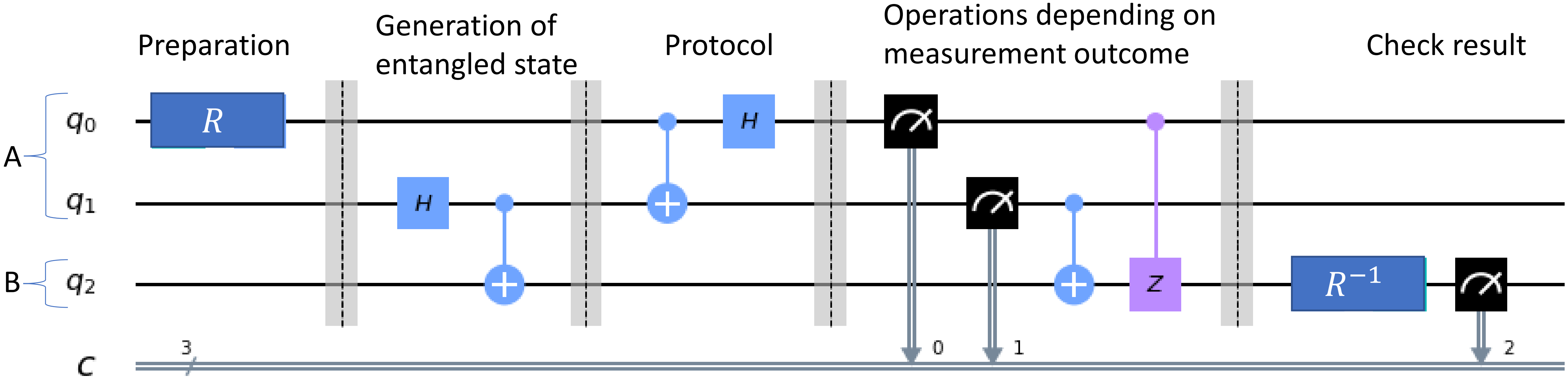


R. Cleve, A. Ekert, C. Macchiavello and M. Mosca, *Quantum Algorithms revised*, Proc. R. Soc. Lond. A (1998) 454, 339-354 (1998)

Quantum teleportation

The state of q_0 is transmitted from one location to another, with the help of classical communication and a Bell pair.

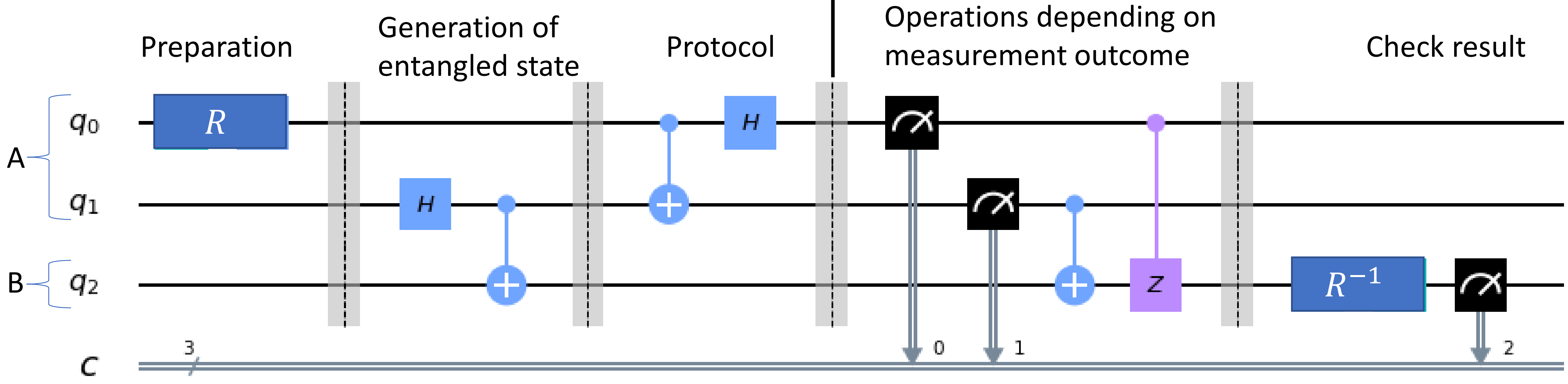
The protocol destroys the quantum state of a qubit in one location and recreates it on a qubit at a distant location, with the help of shared entanglement.



Quantum teleportation

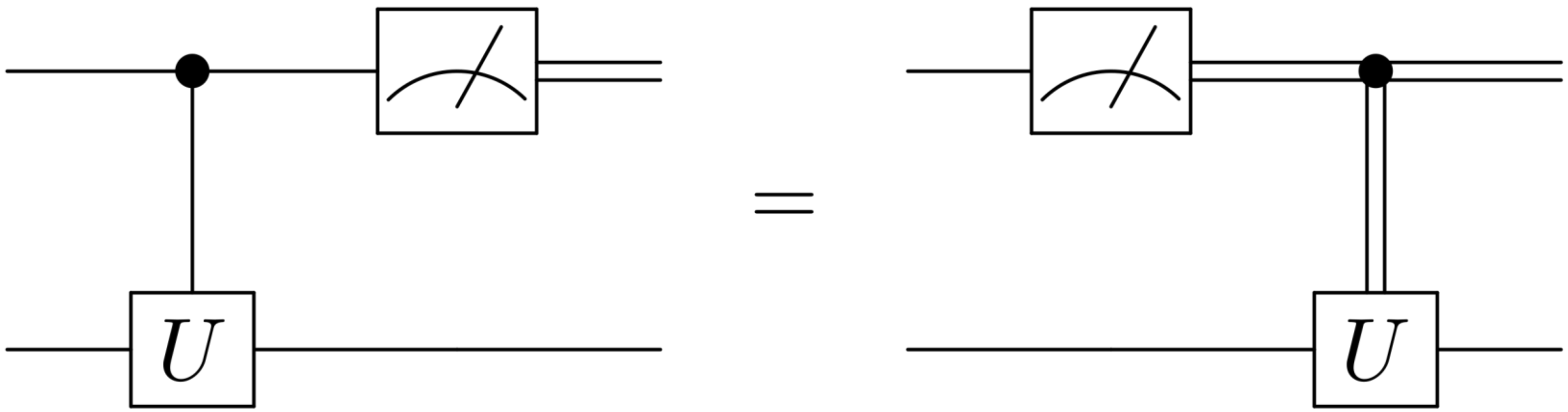
$$
\begin{aligned}
\frac{1}{2}[&|00\rangle(\alpha|0\rangle+\beta|1\rangle) \xrightarrow{\mathbb{I}} |00\rangle(\alpha|0\rangle+\beta|1\rangle)\\
+&|01\rangle(\alpha|1\rangle+\beta|0\rangle) \xrightarrow{X} |01\rangle(\alpha|0\rangle+\beta|1\rangle)\\
+&|10\rangle(\alpha|0\rangle-\beta|1\rangle) \xrightarrow{Z} |10\rangle(\alpha|0\rangle+\beta|1\rangle)\\
+&|11\rangle(\alpha|1\rangle-\beta|0\rangle)] \xrightarrow{ZX} |11\rangle(\alpha|0\rangle+\beta|1\rangle)
\end{aligned}
$$

The protocol does not violate:
- No cloning theorem
- Special relativity



Quantum Computing

Deferred measurement principle



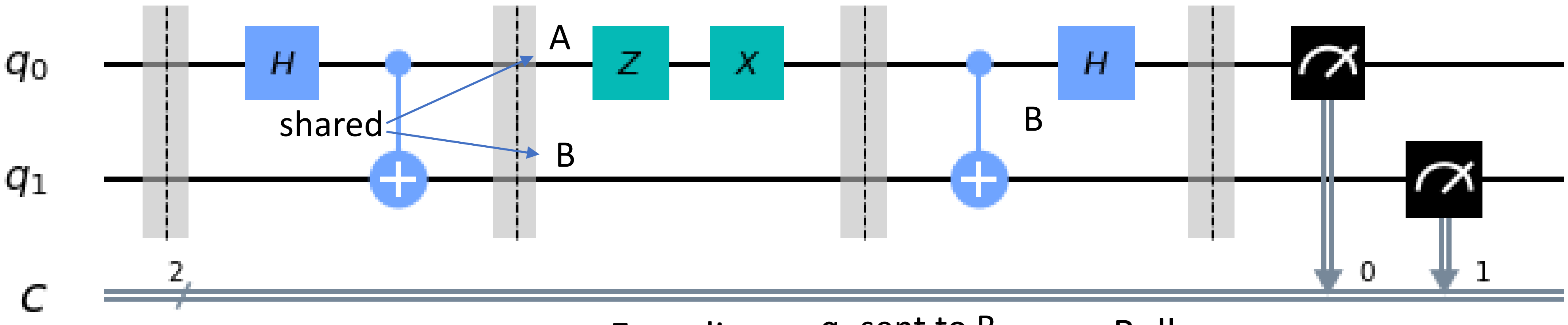
On the real hardware we cannot perform operations depending on a previous measurement outcome. But we can get the same result if we first perform a conditional gate and then we measure

Some drawbacks:

- By measuring early, we could reuse qubits or **reduce the time these qubits are in fragile superposition**.
- In quantum teleportation, the early measurement would have allowed us to transmit a qubit state **without a direct quantum communication channel** (much less stable than a classical one).
- Hence, in NISQ devices measuring earlier yields more reliable results (see e.g. VQE algorithm).

Superdense coding

Procedure that allows one to **send two classical bits** to another party **using just a single qubit** of communication.



Prepare $\frac{|00\rangle+|11\rangle}{\sqrt{2}}$

Encoding protocol on q_0

q_0 sent to B

Bell measurements

Teleportation	Superdense coding
Transmit 1 qubit using two c-bits	Transmit 2 c-bits using 1 qubit

Message	Gate	Output
00	I	$\|00\rangle + \|11\rangle$
01	X	$\|10\rangle + \|01\rangle$
10	Z	$\|00\rangle - \|11\rangle$
11	ZX	$\|10\rangle - \|01\rangle$

CNOT
$\|00\rangle + \|10\rangle$
$\|11\rangle + \|01\rangle$
$\|00\rangle - \|10\rangle$
$\|11\rangle - \|01\rangle$

H
$\|00\rangle$
$\|01\rangle$
$\|10\rangle$
$\|11\rangle$

Deutsch-Josza algorithm

First example of **quantum exponential speed-up.** Problem: given a Boolean function

f returns the same result for all inputs

$$f: \ \{0,1\}^n \rightarrow \{0,1\}$$

f returns 0 for half of the 2^n possible inputs, 1 for the others

Establish whether f is *constant* or *balanced*. On a classical computer you need to evaluate f an exponential ($2^{n-1}+1$) number of times to get a certain result

On a quantum computer a **single evaluation** is sufficient. ⇒ **Exponential speed-up!**

In the worst case we need to evaluate f for half +1 of the possible inputs

We need **two registers**:

A. An n −qubit register initialized in $|+\rangle_A^{\otimes n} = H^{\otimes n}|0\rangle_A$
B. A single-qubit register initialized in $|-\rangle_B = H|1\rangle_B = HX|0\rangle_B$

Oracle: black-box performing the transformation U_f: $|x\rangle_A|y\rangle_B \rightarrow |x\rangle_A|y\oplus f(x)\rangle_B$

f-controlled-NOT

X-basis measurement (i.e. Hadamard followed by Z-measurement) of the first register

Deutsch's algorithm: how it works

Let's start from $n = 1$: $|x\rangle_A \dfrac{|0\rangle_B - |1\rangle_B}{\sqrt{2}}$

$$\Big\downarrow U_f$$

$$|x\rangle_A \frac{|f(x)\rangle_B - |1 \oplus f(x)\rangle_B}{\sqrt{2}} = (-1)^{f(x)} |x\rangle_A \frac{|0\rangle_B - |1\rangle_B}{\sqrt{2}}$$

$$\longrightarrow \frac{1}{2}\left[(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right]_A \otimes (|0\rangle - |1\rangle)_B$$

constant $f(0) = f(1)$

$$\frac{1}{2}(|0\rangle + |1\rangle)_A \otimes (|0\rangle - |1\rangle)_B$$

$f(0) \neq f(1)$ balanced

$$\frac{1}{2}(|0\rangle - |1\rangle)_A \otimes (|0\rangle - |1\rangle)_B$$

$$\Big\downarrow H_A$$

$$|0\rangle_A \frac{|0\rangle_B - |1\rangle_B}{\sqrt{2}} \qquad |1\rangle_A \frac{|0\rangle_B - |1\rangle_B}{\sqrt{2}}$$

Measuring A gives the answer

Deutsch-Josza algorithm

$$H^{\otimes n}|0\rangle_A = 2^{-n/2}\sum_{x=0}^{2^n-1}|x\rangle_A \Rightarrow$$

Initial state

$$2^{-(n+1)/2}\sum_{x=0}^{2^n-1}|x\rangle_A\otimes(|0\rangle-|1\rangle)_B$$

$$\xrightarrow{U_f} \quad 2^{-(n+1)/2}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle_A\otimes(|0\rangle-|1\rangle)_B$$

f constant

$$2^{-(n+1)/2}\sum_{x=0}^{2^n-1}|x\rangle_A\otimes(|0\rangle-|1\rangle)_B$$

Up to a global phase

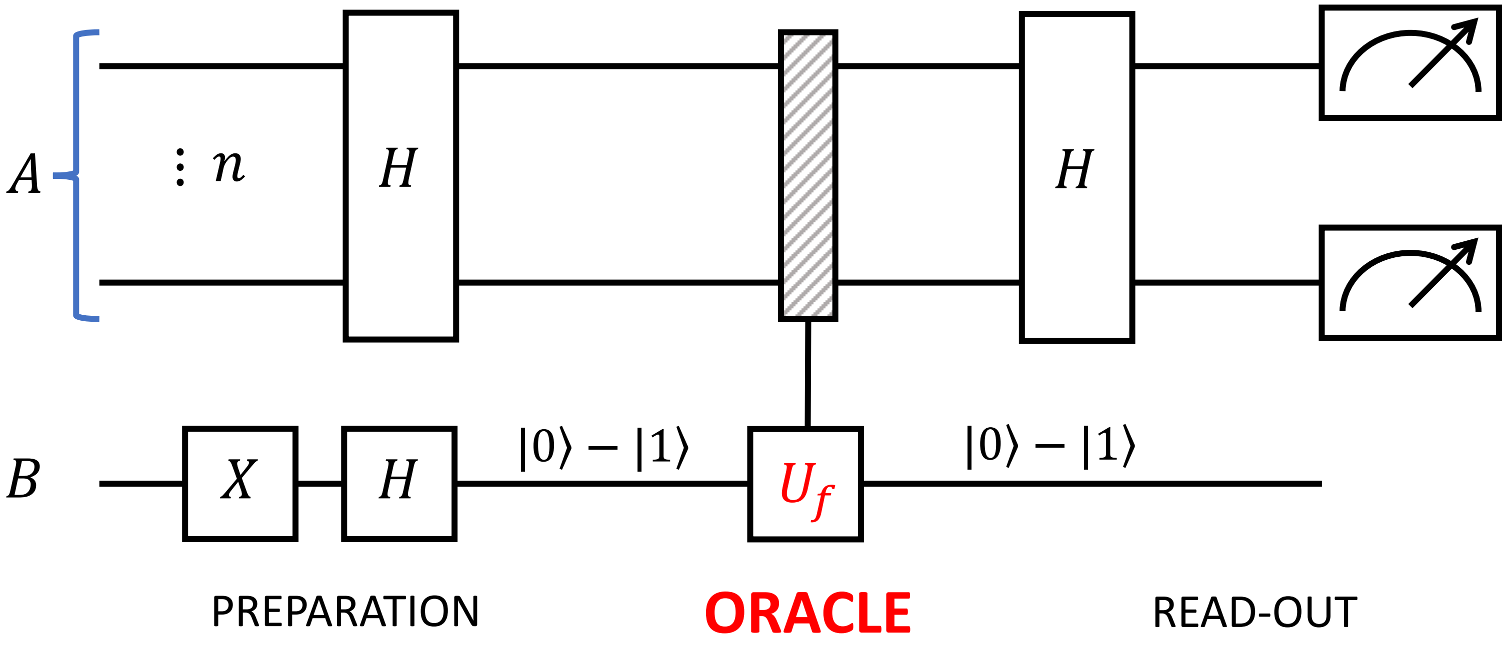
$$\Big\downarrow H_A^{\otimes n}$$

$$|0\cdots 0\rangle_A\frac{|0\rangle_B-|1\rangle_B}{\sqrt{2}}$$

f balanced

Orthogonal state (negative phase added to half of the states)

Deutsch-Josza algorithm: general structure



n −qubit Hadamard: $|x\rangle \xrightarrow{H} \sum_{y\in\{0,1\}^n} (-1)^{x\cdot y}\, |y\rangle$

$x \cdot y = (x_1 \wedge y_1) \oplus (x_2 \wedge y_2) \oplus \cdots \oplus (x_n \wedge y_n)$ Scalar product modulo 2

At the end of the algorithm $\sum_{x,y=0}^{2^n-1} (-1)^{f(x)} (-1)^{x\cdot y} |y\rangle\, (|0\rangle - |1\rangle)$ $\quad P_{|0\rangle^{\otimes n}} = \left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \right|^2 = \begin{cases} 1 & \text{constant} \\ 0 & \text{balanced} \end{cases}$

Bernstein-Vazirani algorithm

PROBLEM:

Given a black-box function $f_s(x) = x \cdot s \pmod 2$ we aim to determine the string s

Classically, this requires querying the oracle n times.

QUANTUM SOLUTION:

The DJ circuit (register A) can be used to determine the bit string s of the hidden function:

$$f_s(x) = x \cdot s \pmod 2 = (x_1 \wedge s_1) \oplus (x_2 \wedge s_2) \oplus \cdots \oplus (x_n \wedge s_n)$$

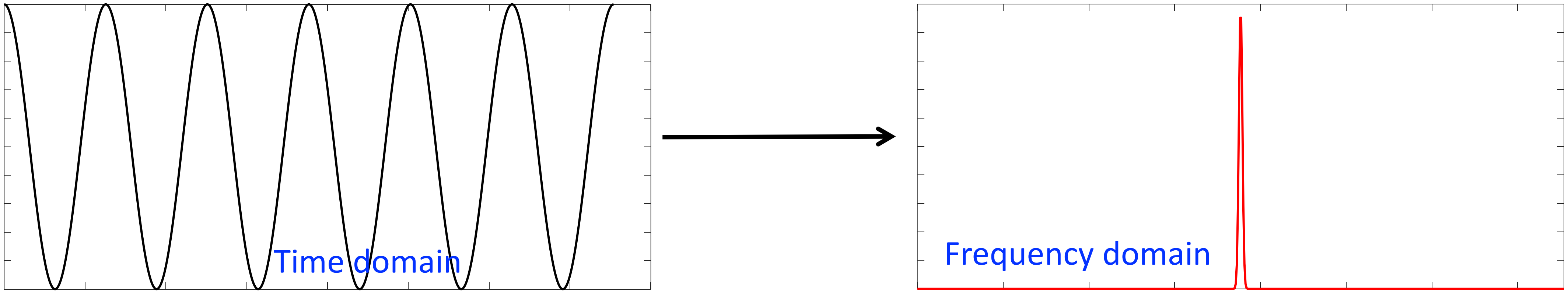
$$|0\rangle \xrightarrow{H^{\otimes n}} 2^{-n/2} \sum_{x \in \{0,1\}^n} |x\rangle \xrightarrow{f_s(x)} 2^{-n/2} \sum_{x \in \{0,1\}^n} (-1)^{x \cdot s} |x\rangle \xrightarrow{H^{\otimes n}} |s\rangle$$

Quantum Fourier Transform

Physicists often solve problems by ***transforming*** it into another problem for which a solution is known. A few such transformations appear so often and in so many different contexts that these transformations are studied for their own sake.

Some of these transformations can be computed **much faster on a quantum computer** than on a classical computer and fast algorithms were constructed to achieve this goal.

One such transformation is the *discrete* ***Fourier*** *transform.*



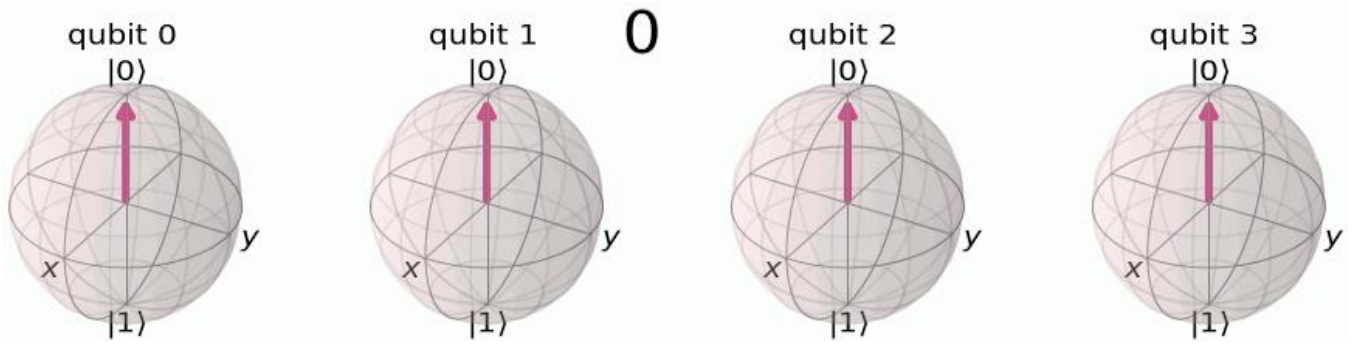
Quantum Fourier Transform

$$U_N^{QFT} = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} e^{2\pi i x y/N} |y\rangle\langle x| \qquad N = 2^n$$

$$|x\rangle = \sum_{i=0}^{N-1} x_i |i\rangle$$

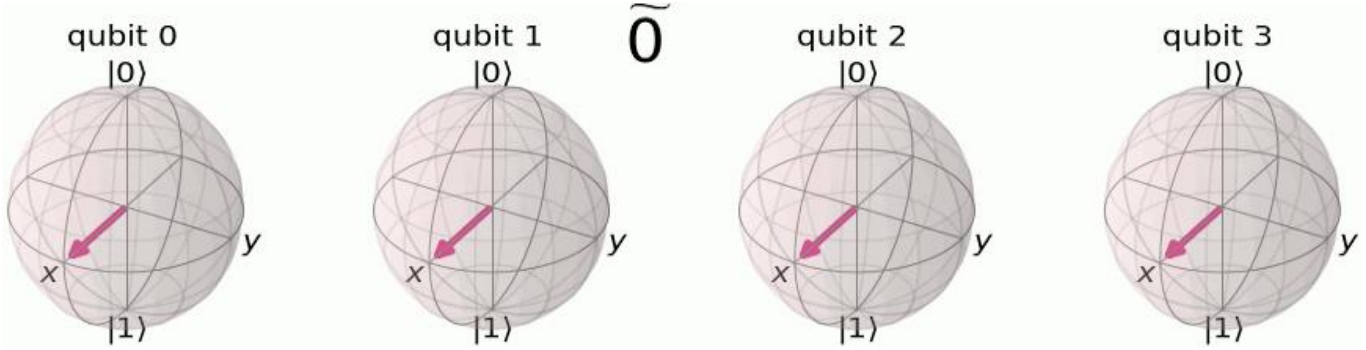
$$\Big\downarrow \; \mathcal{F}_N$$

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



Highest frequency

Z basis



Fourier basis

Highest frequency

$\tilde{x}$ dictates the angle at which each qubit is rotated around the Z-axis.

UNIVERSITÀ DI PARMA

Quantum Computing

Quantum Fourier Transform

$$y=\sum_{k=0}^{n-1} y_k 2^k = 2^n \sum_{k=0}^{n-1} y_k 2^{k-n} = 2^n \sum_{j=1}^{n} y_j 2^{-j} \Rightarrow \frac{y}{2^n}=\sum_{j=1}^{n}\frac{y_j}{2^j}$$

$$|\tilde{x}\rangle = \frac{1}{\sqrt{N}}\sum_{y=0}^{N-1} e^{2\pi i x y/N}|y\rangle = \frac{1}{\sqrt{N}}\sum_{y=0}^{N-1} e^{2\pi i x \sum_{j=1}^{n} y_j/2^j}|y_1\dots y_n\rangle = \frac{1}{\sqrt{N}}\sum_{y=0}^{N-1}\prod_{j=1}^{n} e^{2\pi i x y_j/2^j}|y_1\dots y_n\rangle$$

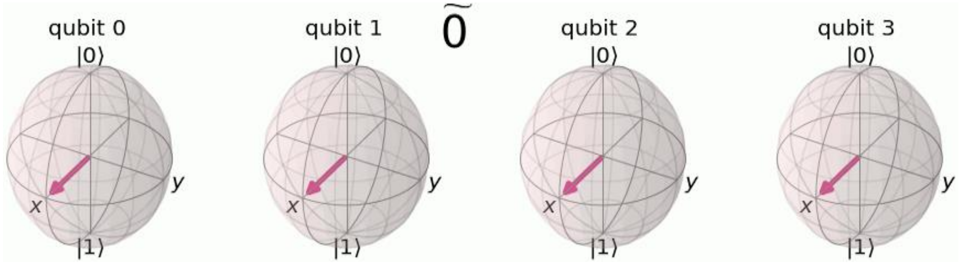
$$\sum_{y=0}^{N-1}|y\rangle = \sum_{y_1=0}^{1}\sum_{y_2=0}^{1}\cdots\sum_{y_n=0}^{1}|y_1\dots y_n\rangle$$

$|\tilde{x}\rangle$ is **unentangled**. However, the phases depend on the state encoded in the whole y

$$\Rightarrow |\tilde{x}\rangle = \frac{1}{\sqrt{N}}\left(|0\rangle + e^{\frac{2\pi}{2}ix}|1\rangle\right)\otimes\left(|0\rangle + e^{\frac{2\pi}{2^2}ix}|1\rangle\right)\otimes\dots\otimes\left(|0\rangle + e^{\frac{2\pi}{2^n}ix}|1\rangle\right)$$

The circuit **requires also controlled gates**

On each qubit, the exponent contains the rotation frequency



1-qubit QFT

$$N = 2$$

$$|x\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

$$|\tilde{x}\rangle = \begin{pmatrix} \tilde{\alpha} \\ \tilde{\beta} \end{pmatrix}$$

$$\tilde{\alpha} = \frac{1}{\sqrt{2}}\left(\alpha\, e^{2\pi i 0\times 0/2} + \beta e^{2\pi i 1\times 0/2}\right) = \frac{\alpha + \beta}{\sqrt{2}}$$

$$\tilde{\beta} = \frac{1}{\sqrt{2}}\left(\alpha\, e^{2\pi i 0\times 1/2} + \beta e^{2\pi i 1\times 1/2}\right) = \frac{\alpha - \beta}{\sqrt{2}}$$

$$U_2^{QFT}|x\rangle = \tilde{\alpha}|0\rangle + \tilde{\beta}|1\rangle = \frac{\alpha + \beta}{\sqrt{2}}|0\rangle + \frac{\alpha - \beta}{\sqrt{2}}|1\rangle$$

$$U_2^{QFT} = H$$

Circuit for the QFT

We use 2 gates:
- Single-qubit $\quad H|x_k\rangle = |0\rangle + \exp\frac{2\pi i x_k}{2}|1\rangle$
- Two-qubit $C\varphi_k$ $\quad C\varphi_{k\to j}|1x_j\rangle = \exp\frac{2\pi i}{2^k}x_j\,|1x_j\rangle \qquad C\varphi_k|0x_j\rangle = |0x_j\rangle$

1. Hadamard on the first qubit H_1: $\quad H_1|x_1x_2\dots x_n\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \exp\frac{2\pi i x_1}{2}|1\rangle\right)\otimes|x_2\dots x_n\rangle$
2. $C\varphi_{2\to1}:\longrightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + \exp\left(\frac{2\pi i}{2}x_1 + \frac{2\pi i}{2^2}x_2\right)|1\rangle\right)\otimes|x_2\dots x_n\rangle$
3. $C\varphi_{n\to1}:\longrightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + \exp\left(\frac{2\pi i}{2}x_1 + \frac{2\pi i}{2^2}x_2 + \cdots + \frac{2\pi i}{2^n}x_n\right)|1\rangle\right)\otimes|x_2\dots x_n\rangle$
$$= \frac{1}{\sqrt{2}}\left(|0\rangle + \exp\frac{2\pi i x}{2^n}|1\rangle\right)\otimes|x_2\dots x_n\rangle$$
4. Repeat by starting with H_2 and then $C\varphi_{3\to2}$ ... $C\varphi_{n\to2}$

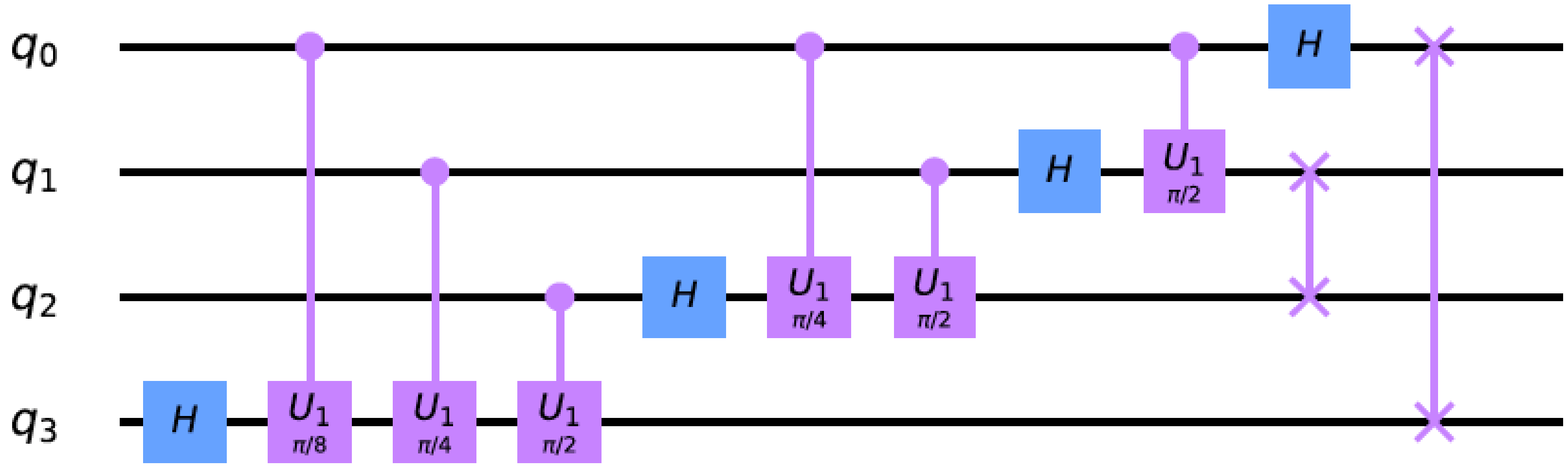
$$\longrightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + \exp\frac{2\pi i x}{2^n}|1\rangle\right)\otimes\frac{1}{\sqrt{2}}\left(|0\rangle + \exp\frac{2\pi i x}{2^{n-1}}|1\rangle\right)\otimes\dots\otimes\frac{1}{\sqrt{2}}\left(|0\rangle + \exp\frac{2\pi i x}{2^1}|1\rangle\right)$$

QFT: Scaling of resources

Total number of gates: $n(n+2)/2$

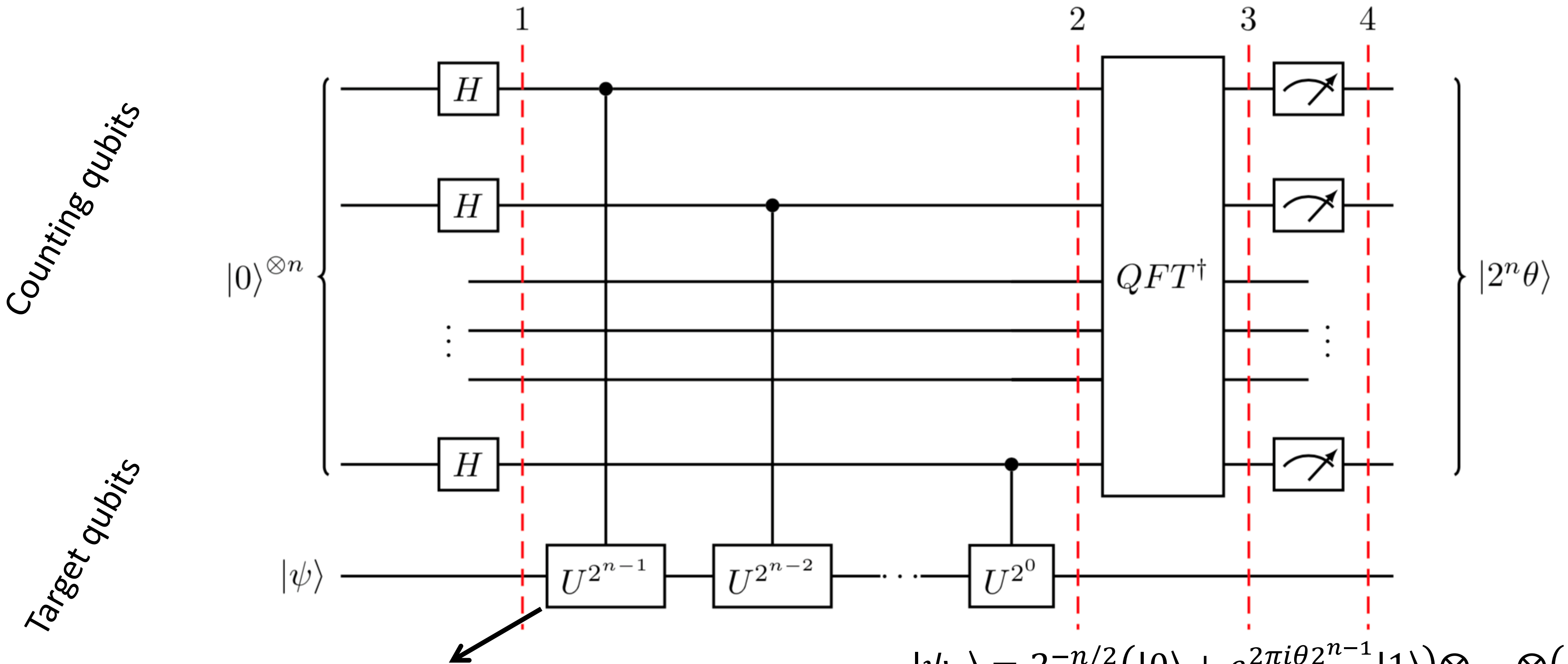
The best **classical** algorithm (Fast Fourier Transform) requires an **exponential number of gates**.

qubit	H	$C\varphi$	$SWAPs$
1	1	$n-1$	
2	1	$n-2$	
3	1	$n-3$	
…			
n	1	0	
TOTAL	n	$n(n-1)/2$	$n/2$



Quantum Phase Estimation

PROBLEM: Given a unitary operator U, estimate θ in $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$



$U^{2^k}|\psi\rangle = e^{2\pi i\theta 2^k}|\psi\rangle$

$$|\psi_2\rangle = 2^{-n/2}\left(|0\rangle + e^{2\pi i\theta 2^{n-1}}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i\theta 2^1}|1\rangle\right)$$

$$\otimes\left(|0\rangle + e^{2\pi i\theta 2^0}|1\rangle\right)\otimes|\psi\rangle = 2^{-n/2}\sum_{m=0}^{2^n-1} e^{2\pi i\theta m}\,|m\rangle\otimes|\psi\rangle$$

… remember the QFT

$$|\psi_2\rangle = 2^{-n/2}\left(|0\rangle + e^{2\pi i\theta 2^{n-1}}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i\theta 2^{1}}|1\rangle\right)\otimes\left(|0\rangle + e^{2\pi i\theta 2^{0}}|1\rangle\right)\otimes|\psi\rangle =$$

$$= 2^{-n/2}\left(|0\rangle + e^{2\pi i x/2}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i x/2^{n-1}}|1\rangle\right)\otimes\left(|0\rangle + e^{\frac{2\pi i x}{2^n}}|1\rangle\right)\otimes|\psi\rangle = U_{QFT}|x\rangle\otimes|\psi\rangle$$

$x = 2^n\theta$

$$|\psi_2\rangle = 2^{-n/2}\sum_{m=0}^{2^n-1} e^{2\pi i\theta m}\,|m\rangle\otimes|\psi\rangle \xrightarrow{U_{QFT}^{\dagger}} 2^{-n}\sum_{x=0}^{2^n-1}\sum_{m=0}^{2^n-1} e^{-\frac{2\pi i m}{2^n}(x-2^n\theta)}\,|x\rangle\otimes|\psi\rangle$$

This expression peaks close to $x = 2^n\theta$.

For integer $2^n\theta$, measuring the first register (counting qubits) exactly gives θ. Otherwise (see notebook) we can obtain a good approximation.

QPE is a **fundamental subroutine** in many Quantum Algorithms.

If we prepare the target register in a state $|\xi\rangle = \sum_n c_n|\psi_n\rangle$ (with $U|\psi_n\rangle = e^{i2\pi\theta_n}|\psi_n\rangle$), by measuring the counting register we get a good estimate of θ_n with probability $|c_n|^2$.

Example on qiskit: estimating π by QPE

$$QPE(U, |0\rangle_n, |\psi\rangle_m) = |\tilde{\theta}\rangle_n |\psi\rangle_m$$

$$U|\psi\rangle_m = \mathrm{e}^{i2\pi\theta}\,|\psi\rangle_m$$

Binary approximation to $2^n\theta$

$$U = u_1(\varphi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix} \qquad |\psi\rangle_1 = |1\rangle \qquad u_1(\varphi)|1\rangle = e^{i\varphi}|1\rangle$$

From QPE we measure an estimate of $x = 2^n\theta$. Hence, $\theta = \frac{x}{2^n}$

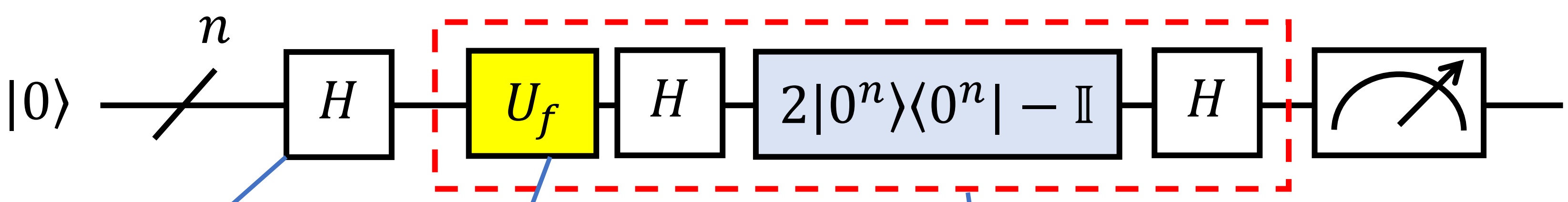
Here we have chosen $\varphi = 2\pi\theta = 1 \Longrightarrow \pi = \frac{\varphi}{2\theta} = \frac{2^n}{2x} = \frac{2^{n-1}}{x}$

Grover's algorithm

PROBLEM: **search** in an unstructured data-base.

BASIC **TRICK**: **Amplitude amplification** (used in many algorithms)

Quadratic advantage compared to classical counterpart. (Classically you would need on average $N/2$ trials)



Repeat $O(\sqrt{N})$ times

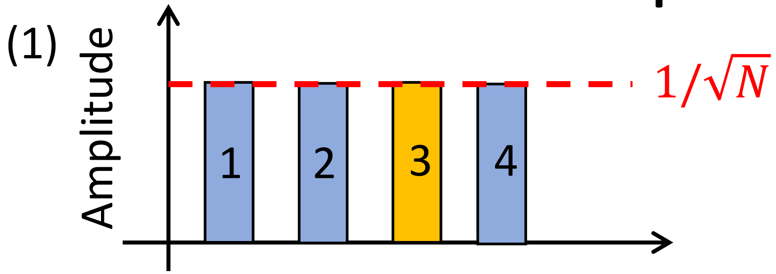
Prepare in uniform superposition
$|s\rangle = H^{\otimes n}|0\rangle$

Oracle function:
$U_f|x\rangle = (-1)^{f(x)}|x\rangle$

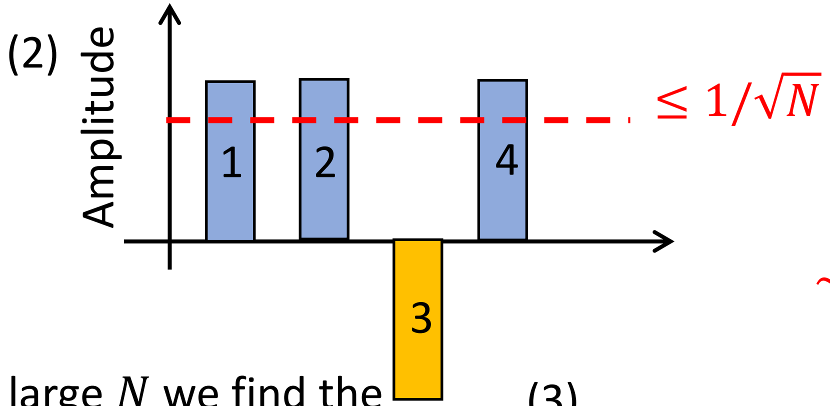
$$f(x) = \begin{cases} 0 & x \neq a \\ 1 & x = a \end{cases}$$

Amplitude amplification:
$U_s = 2|s\rangle\langle s| - \mathbb{I} = H^{\otimes n}(2|0^n\rangle\langle 0^n| - \mathbb{I})H^{\otimes n}$

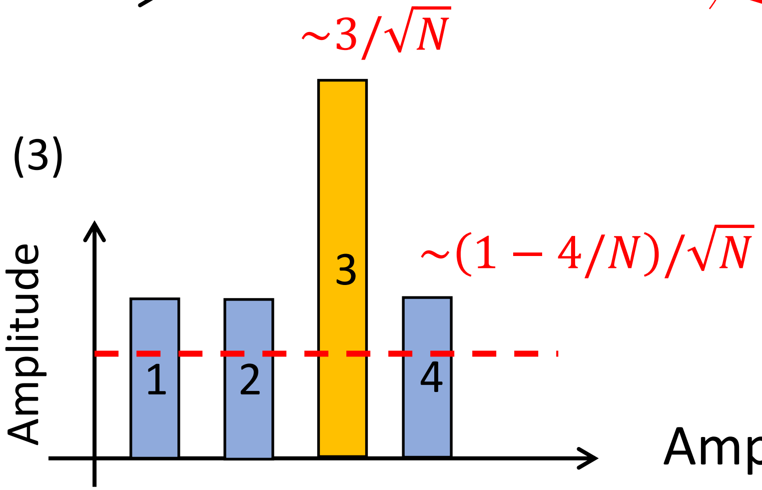
Amplitude amplification



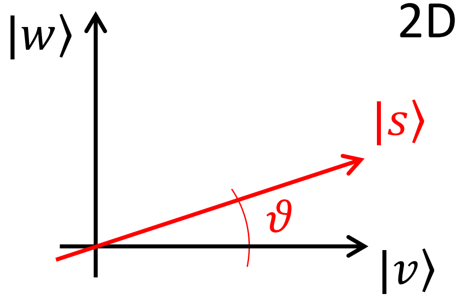
We start from a uniform superposition $|s\rangle = H^{\otimes n}|0\rangle$



For large N we find the required element in the database with high probability using $\approx \frac{\pi}{4}\sqrt{N}$ queries of the oracle (Barnett)



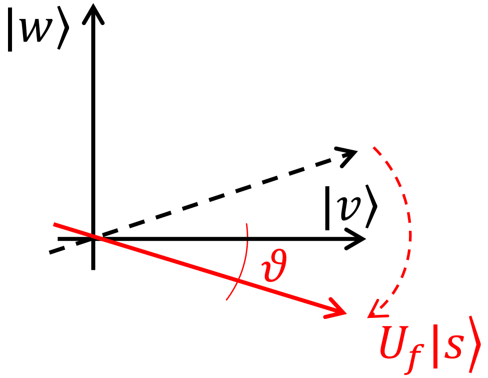
2D space spanned by vectors $|w\rangle$ and $|v\rangle$



$$|s\rangle = \sin\vartheta\,|w\rangle + \cos\vartheta\,|v\rangle,$$

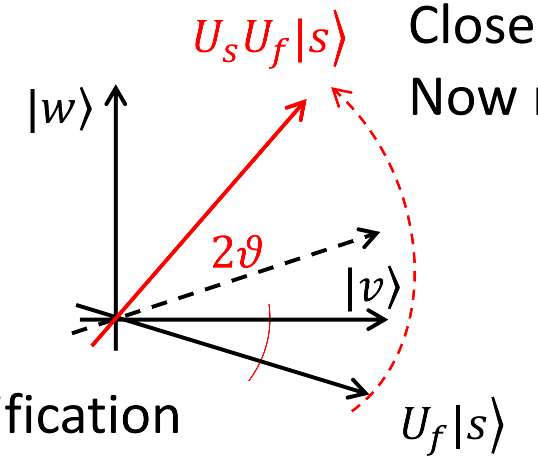
$$\sin\vartheta = \langle s|w\rangle = 1/\sqrt{N}$$

Oracle function



Closer to $|w\rangle$!

Now repeat!



Amplitude amplification

Quantum Computing

UNIVERSITÀ DI PARMA

Quantum Computing

Example: $N = 4$

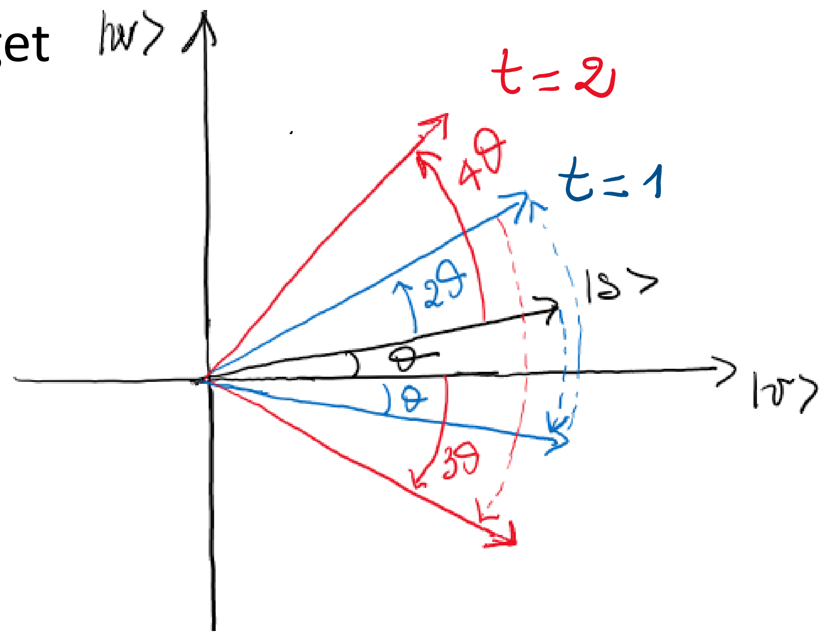
After n applications of the Grover's circuit (oracle+diffuser) we get

$$|\psi\rangle = (U_s U_f)^t \, |s\rangle = \sin\theta_t \, |w\rangle + \cos\theta_t |v\,\rangle$$

$$\theta_t = (2t+1)\theta$$

For $N = 4$, $\theta = \arcsin\frac{1}{2} = \frac{\pi}{6}$

To obtain $|w\rangle$, $\theta_t = \frac{\pi}{2}$ and hence after $t = 1$ we'll find the searched element. In general we need $\sim\sqrt{N}$ rotations.



e.g. $|w\rangle = |11\rangle$

$$Z_1 Z_2 U_{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$U_f = U_{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} = 2|00\rangle\langle 00| - \mathbb{I}$$