Phase in Quantum Computing

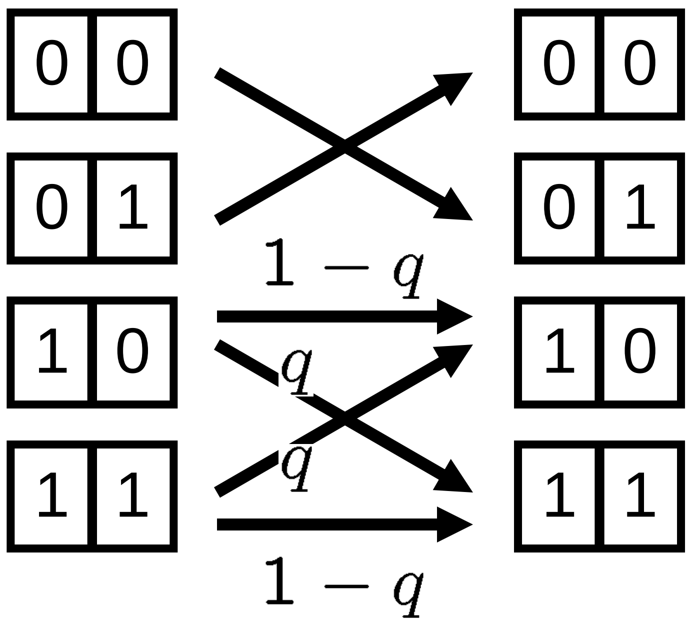
Main concepts of computing illustrated with simple examples

Quantum Theory Made Easy

Classical	Quantum
<u>probabilities</u>	<u>amplitudes</u>
0 — p_0	0 — a_0
1 — p_1	1 — a_1
p_i *is a real number*	a_i *is a complex number*
$p_0+p_1=1$	$\|a_0\|^2+\|a_1\|^2 =1$
$Prob(i)=p_i$	$Prob(i)=\|a_i\|^2$
bit	**qubit**

Quantum Theory Made Easy

Classical Evolution

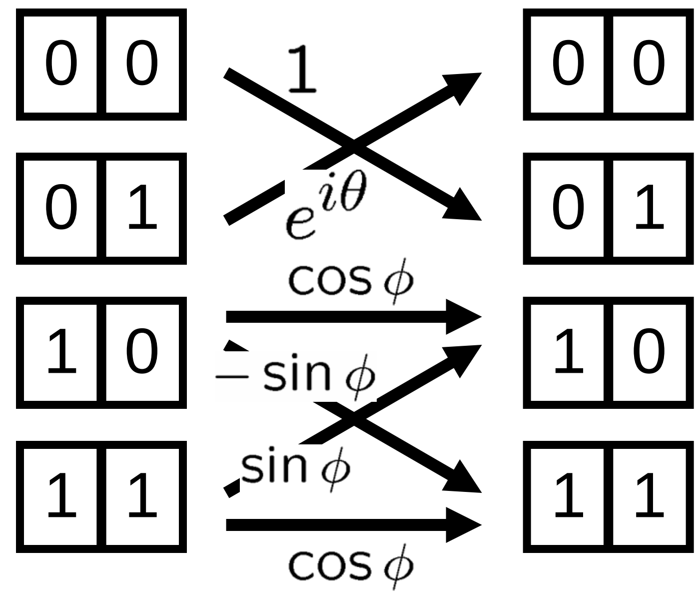


transition probabilities

$$\begin{bmatrix} p'_0 \\ p'_1 \\ p'_2 \\ p'_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1-p & q \\ 0 & 0 & p & 1-q \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

stochastic matrix

Quantum Evolution

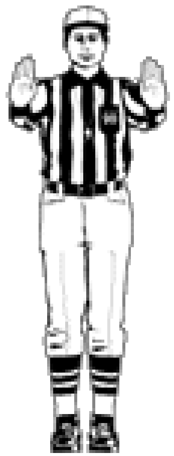


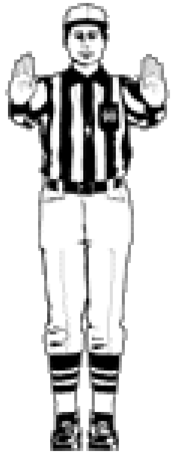
transition amplitudes

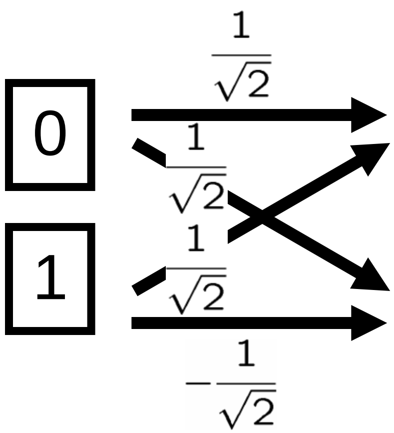
$$\begin{bmatrix} a'_0 \\ a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} = \begin{bmatrix} 0 & e^{i\theta} & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & \cos\phi & \sin\phi \\ 0 & 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

unitary matrix

Interference







$$:= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

qubit
input

$$\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \longrightarrow \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

measure

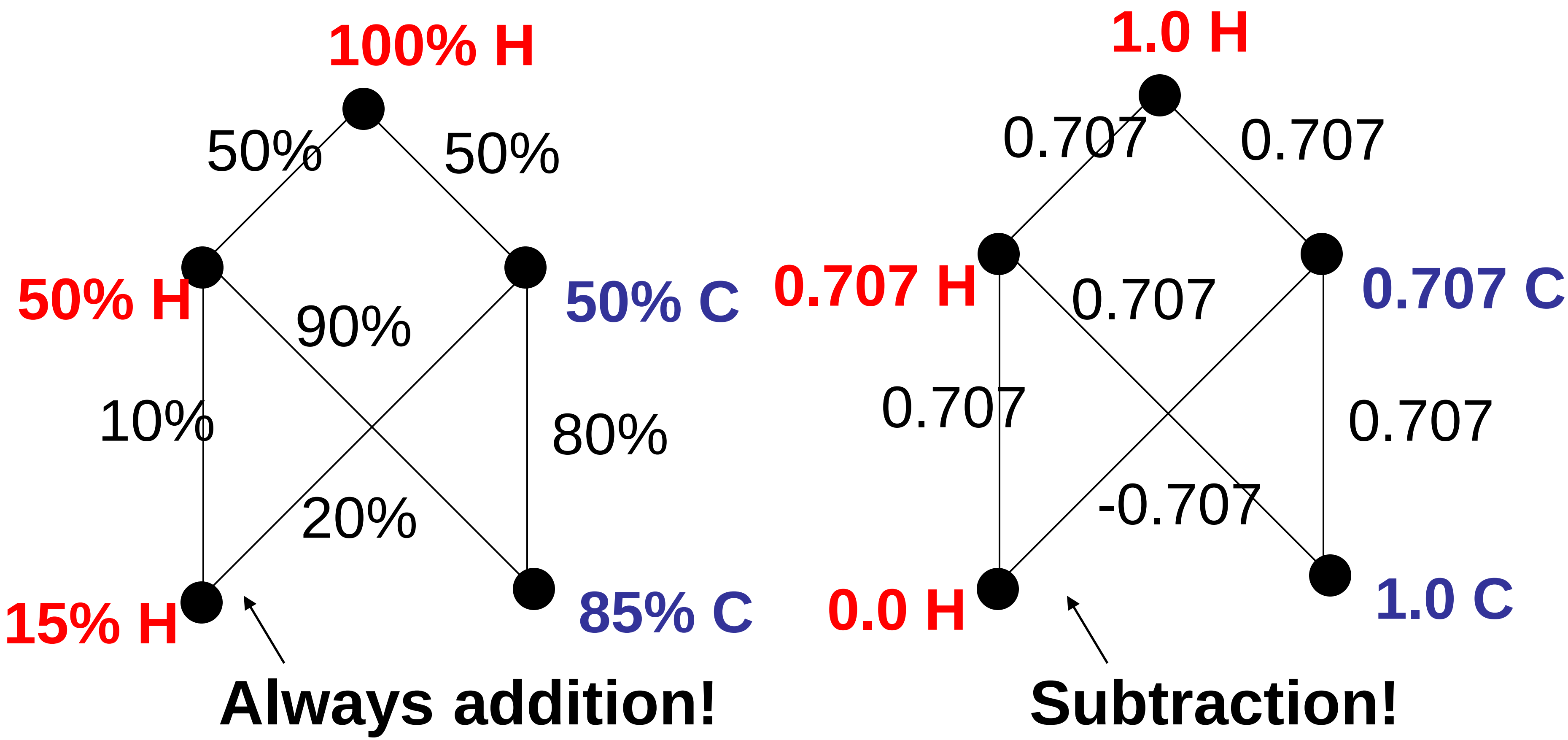
0	50%	100%
1	50%	0%

$$\begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \longrightarrow \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

measure

0	50%	0%
1	50%	100%

Interfering Pathways



Classical **Quantum**

Superposition

Qubits

	amplitudes
0	a_0
1	a_1

$$|\psi\rangle = a_0|0\rangle + a_1|1\rangle$$

a_i is a complex number

Schrödinger's Cat

$\frac{1}{\sqrt{2}}\Big(\Big|$  $\Big\rangle + \Big|$  $\Big\rangle\Big)$

Classical versus quantum computers

Some differences between classical and quantum computers

The state of a classical reversible computer is confined to being one of the computational basis states at any time (queries to the black box for f can only be made one at a time)

Quantum computers can branch out over exponentially many computational basis states, like

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

Using the black box for f *only once*, one can then evaluate f(x) for exponentially many x in superposition:

superposition

$$\sum_{x=0}^{2^n-1} \frac{1}{\sqrt{2^n}} |x\rangle |f(x)\rangle$$

Such states can be further processed (quantumly) to extract hidden properties of f

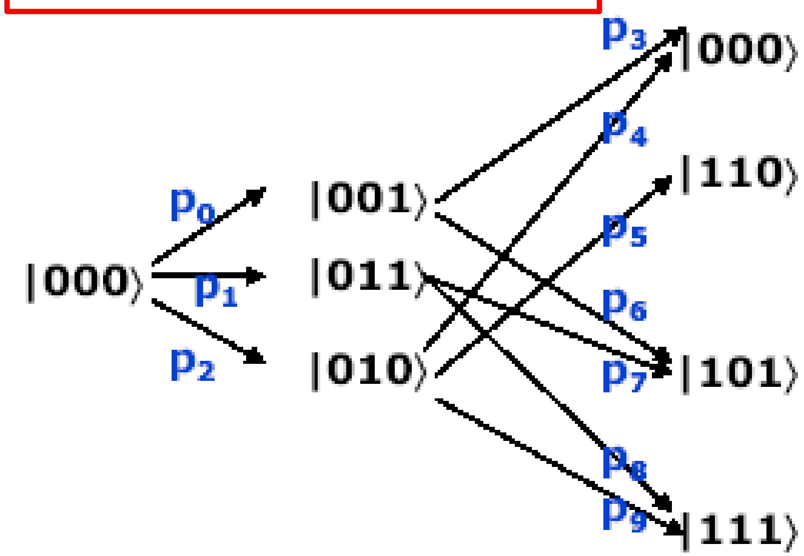
Hidden properties of oracles

Randomised Classical Computation versus Quantum Computation

Deterministic Turing machine

Recall that a deterministic computation can be regarded as a path through "configuration space" of all configurations of a Turing machine (each configuration corresponds to an element of the computational basis)

A randomised computation can be regarded as a tree

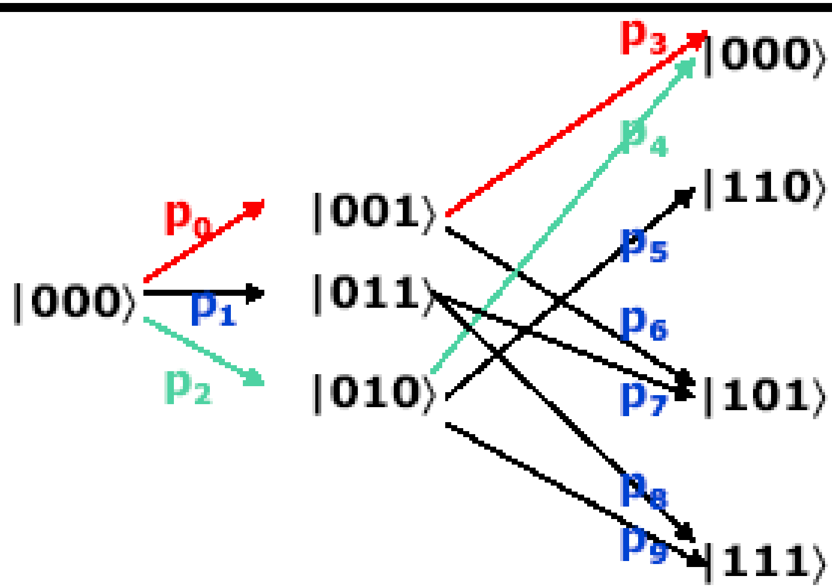


Probabilistic Turing machine

where each branch has a probability p_i associated with it

Probabilities of reaching states

Randomised Classical v. Quantum Computers (2)



The outcome |000⟩ in this computation can be reached by two paths (red and green)

Probability of reaching |000⟩ by the red path is $|a_{00}|^2 = p_0 p_3$

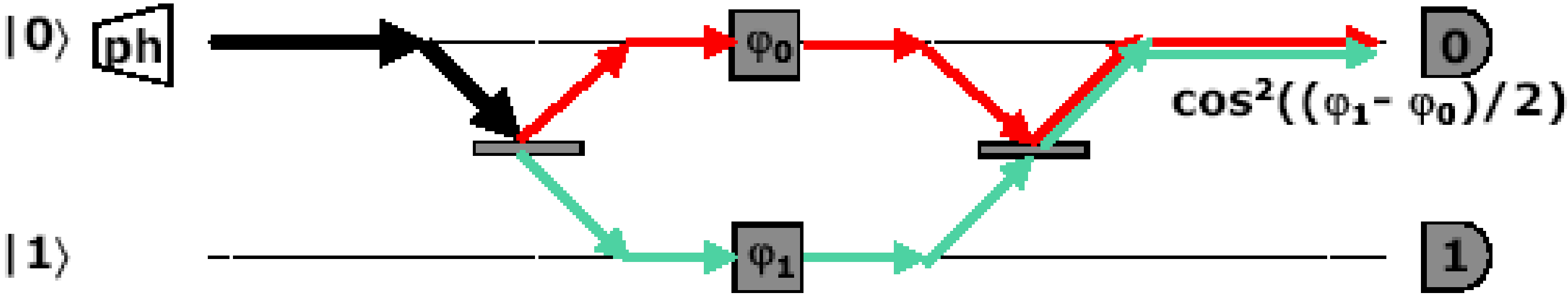
Probability of reaching |000⟩ by the green path is $|a_{10}|^2 = p_2 p_4$

The *total* probability of reaching |000⟩ is thus $|a_{00}|^2 + |a_{10}|^2$

Formulas for reaching states

Randomised Classical v. Quantum Computers (3)

In our interferometry experiment, recall that there are two "computational paths" that lead to the outcome 0 (red path and green path):



The probability *amplitude* of reaching 0 by the red path is

$$a_{00} = \exp(i\varphi_0)/2$$

The probability *amplitude* of reaching 0 by the green path is

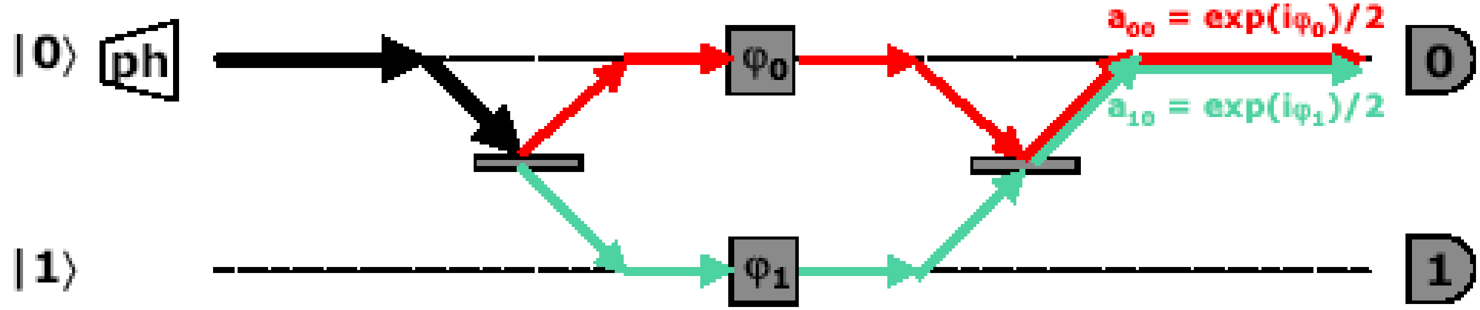
$$a_{10} = \exp(i\varphi_1)/2$$

The *total* probability of reaching 0 is

$$|a_{00} + a_{10}|^2 = \cos^2((\varphi_1 - \varphi_0)/2)$$

Relative phase, destructive and constructive inferences

Randomised Classical v. Quantum Computers (4)



The *total* probability of reaching 0 is $|a_{00} + a_{10}|^2 = \cos^2((\varphi_1 - \varphi_0)/2)$

The *relative phase* between the probability amplitudes of the two paths matters (no such concept in the classical case), and can result in *constructive* or *destructive* interference

e.g. destructive interference occurs when $a_{00} = -a_{10}$,

e.g. constructive interference occurs when $a_{00} = a_{10}$

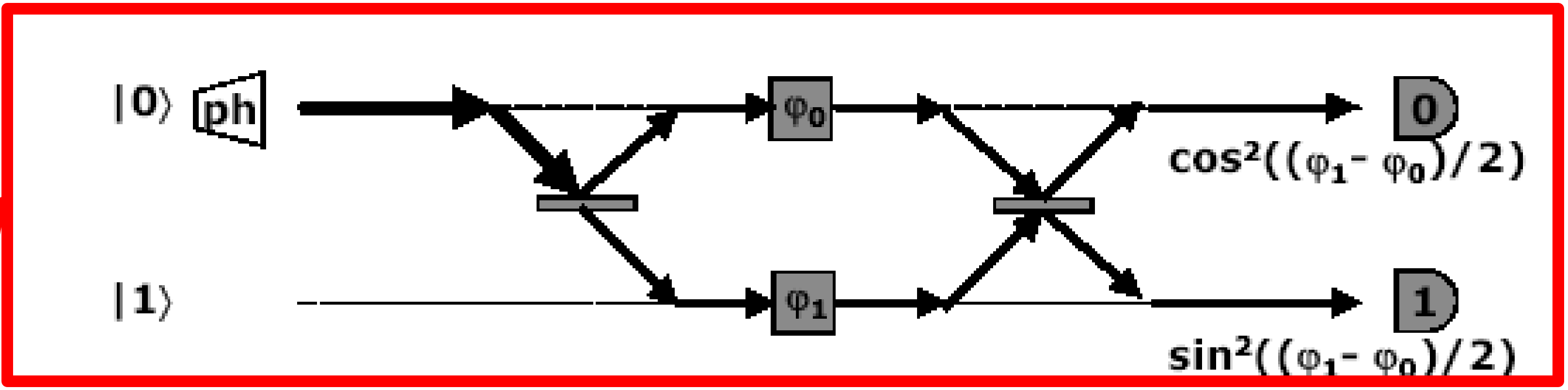
Destructive interference

Constructive interference

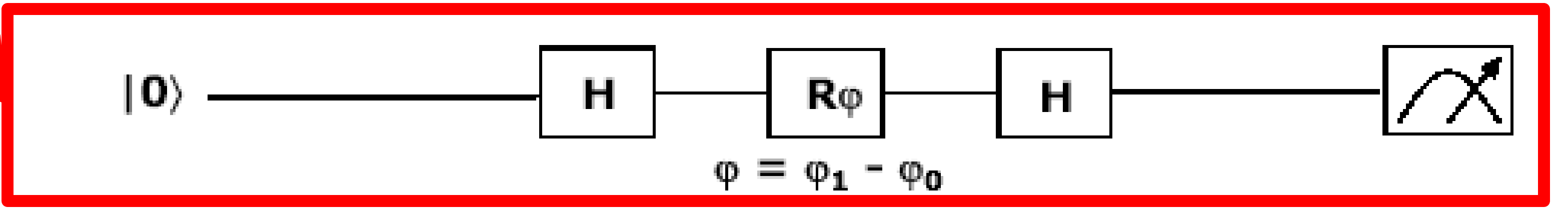
One goal of quantum algorithms is to induce constructive interference on *good* outcomes and destructive interference on *bad* outcomes

Most quantum algorithms can be viewed as big interferometry experiments

Equivalent circuits



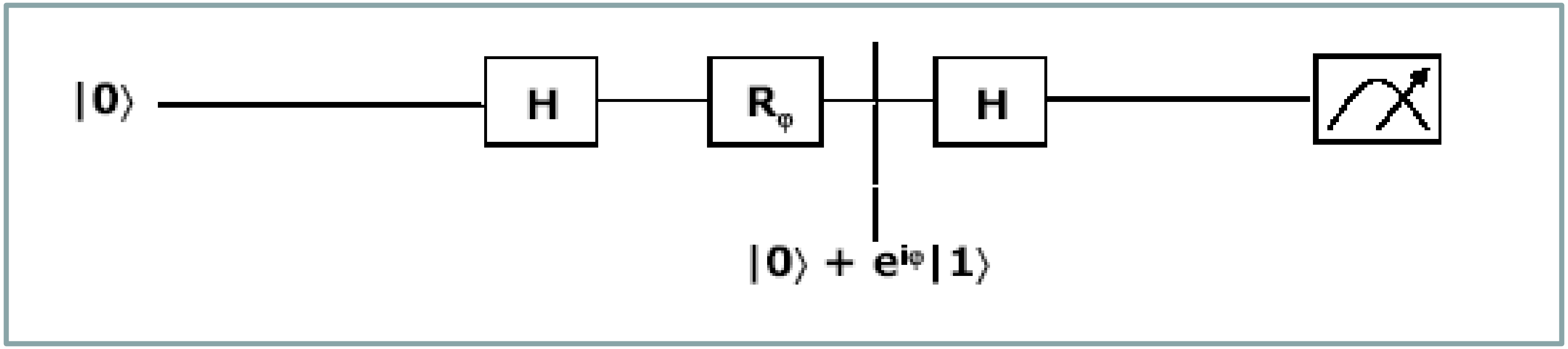
≡



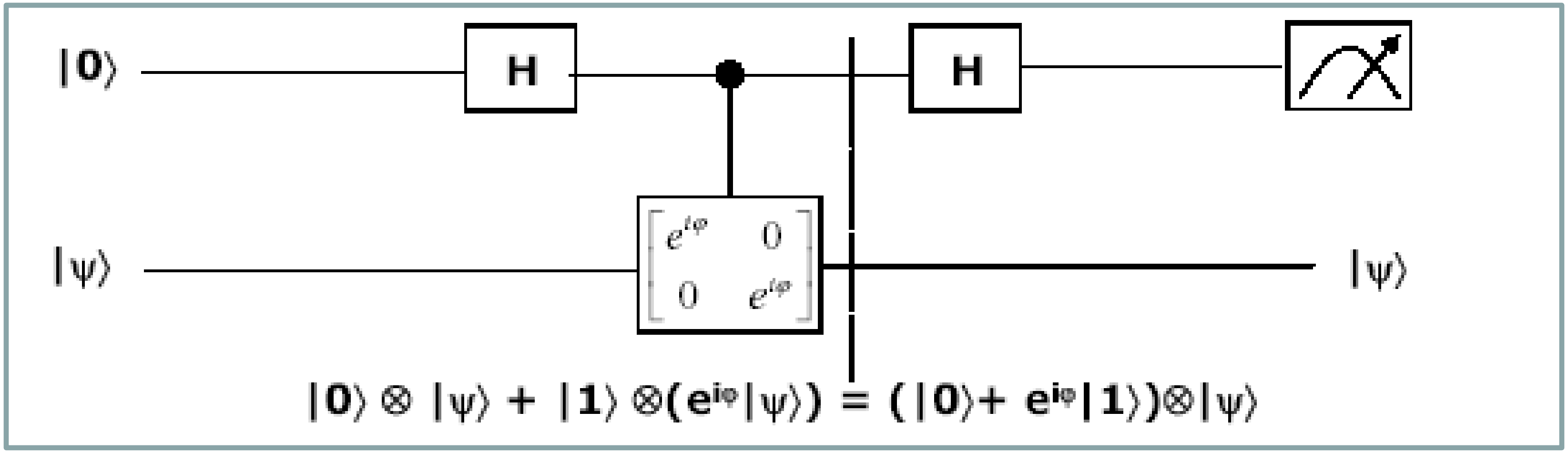
Basic idea: the measurement can distinguish the two cases $\varphi=0$ and $\varphi=\pi$

The “eigenvalue kick-back” concept

There are also some other ways to introduce a relative phase



is equivalent to

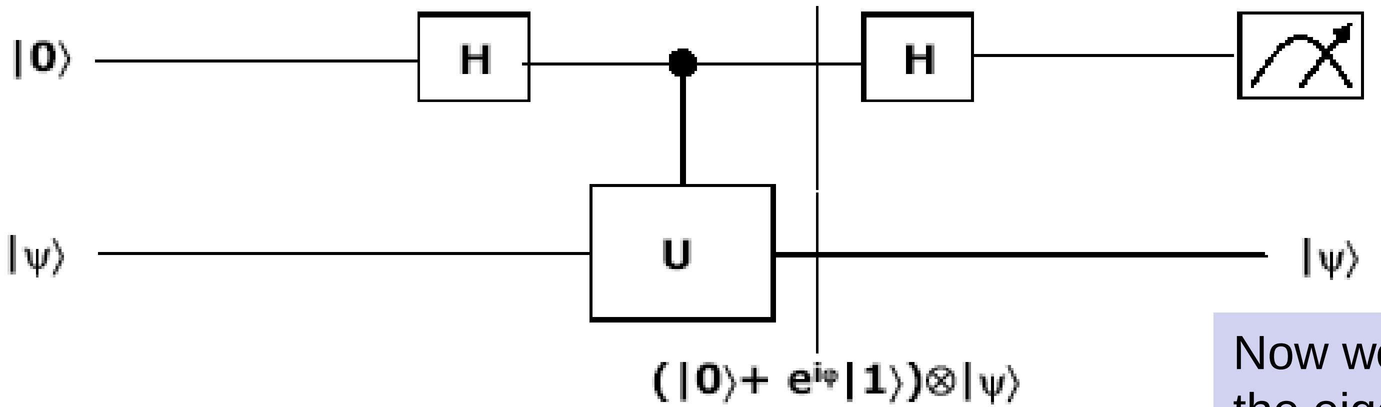


with respect to the top qubit; bottom qubit was unchanged...

The "eigenvalue kick-back" concept

Other ways to introduce a relative phase (2)

... more generally, the bottom qubit will "kick back" a *relative phase (eigenvalue)* in the top qubit if the bottom qubit is in an *eigenstate* of U:



Now we know that the eigenvalue is the same as relative phase

where

$$U|\psi\rangle = e^{i\varphi}|\psi\rangle$$

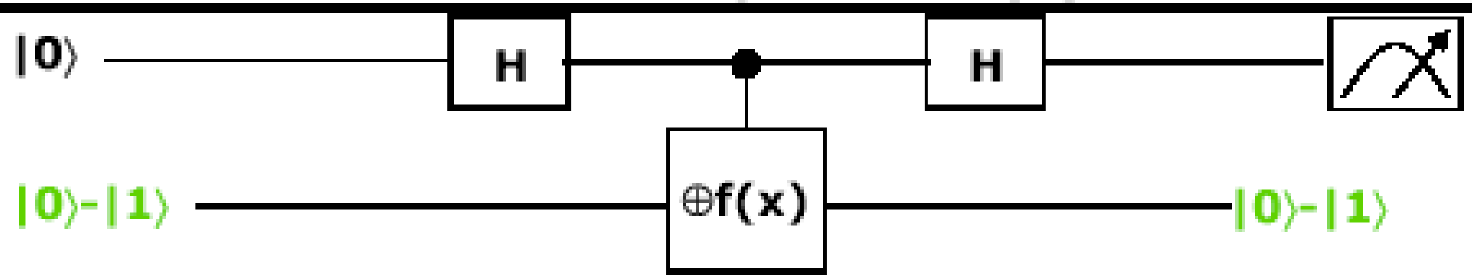
This so-called "eigenvalue kick-back" is a useful mechanism by which to *analyse* (though, not necessarily *implement*) quantum algorithms

The “eigenvalue kick-back” concept

illustrated for DEUTSCH

The "shift operation" as a generalization to Deutsch's Tricks

Deutsch's problem (7)



Deutsch's problem seems to be special: because of it's simplicity, the operation $|b\rangle \rightarrow |b\oplus f(x)\rangle$ can be analysed in a useful eigenbasis, namely $\{|0\rangle+|1\rangle, |0\rangle-|1\rangle\}$

But for more general problems, like period-finding (and the general *hidden subgroup problem*), we introduce the *shift operation*, $U_{sh(f)}$:

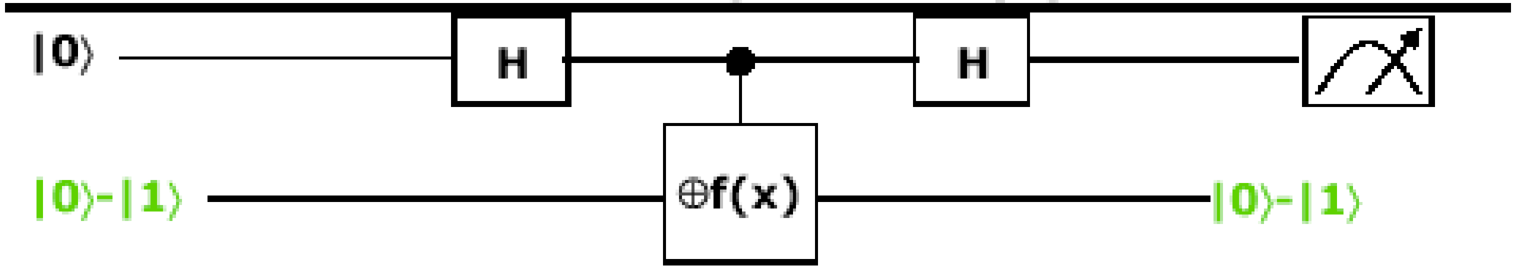
$$U_{sh(f)}: \quad |f(x)\rangle \rightarrow |f(x+1)\rangle$$

For general f, $U_{sh(f)}$ may not be implementable, because f is not necessarily one-to-one

However, $U_{sh(f)}$ is a powerful analysis tool…

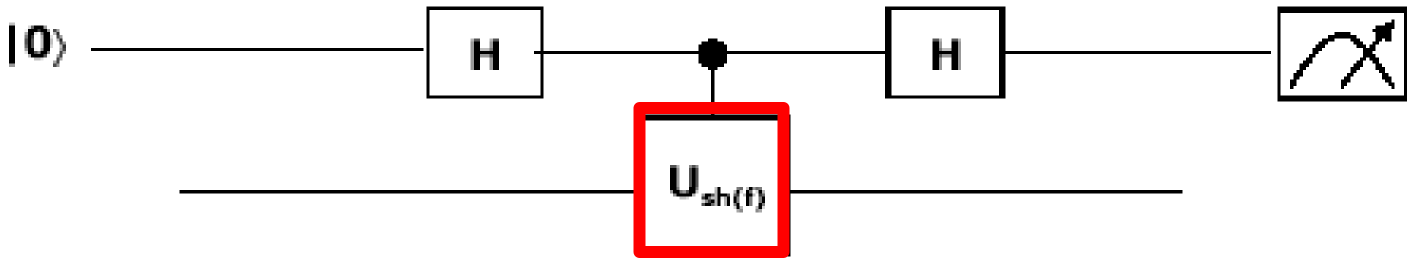
Change of controlled gate in Deutsch with Controlled-Ushift gate

Deutsch's problem (8)



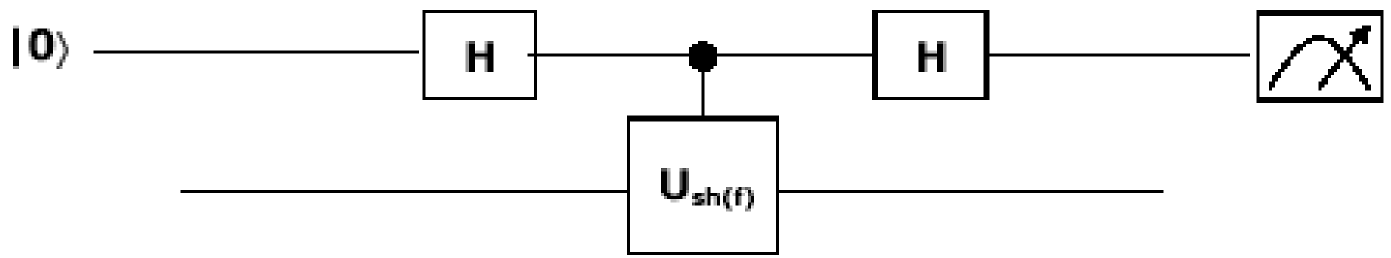
Instead of the controlled-⊕f(x) gate (above), assume we have a *controlled-*$U_{sh(f)}$ gate which maps

$$|0\rangle|f(x)\rangle \rightarrow |0\rangle|f(x)\rangle$$
$$|1\rangle|f(x)\rangle \rightarrow |0\rangle|f(x+1)\rangle$$



Now we deal with new types of eigenvalues and eigenvectors

Deutsch's problem (9)



Note: $f(0)=f(1)$ implies $U_{sh(f)} = I$

$f(0) \neq f(1)$ implies $U_{sh(f)} = X$

Both I and X have eigenvectors $\{|0\rangle+|1\rangle, |0\rangle-|1\rangle\}$, but I has eigenvalues $\{1,1\}$ whereas X has eigenvalues $\{1,-1\}$

So, $U_{sh(f)}$ has eigenvectors $\{|0\rangle+|1\rangle, |0\rangle-|1\rangle\}$ with eigenvalues

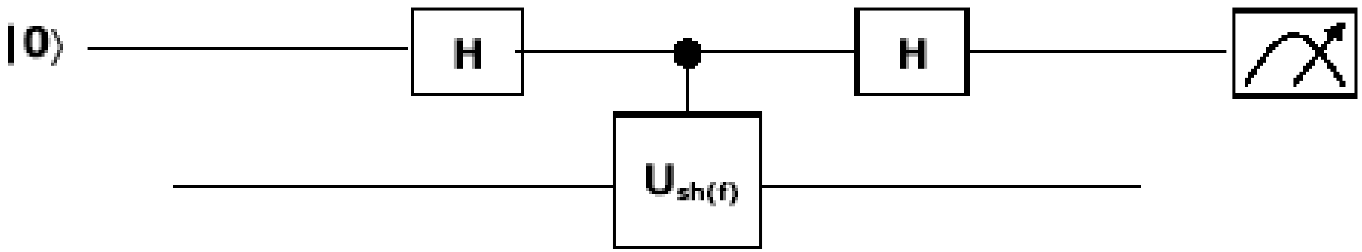
$$\{1, (-1)^{f(0)\oplus f(1)}\}$$

or, writing the eigenvalues another way,

$$\{1, e^{i\pi f(0)\oplus f(1)}\}$$

The general concept of the answer encoded in phase

Deutsch's problem (10)



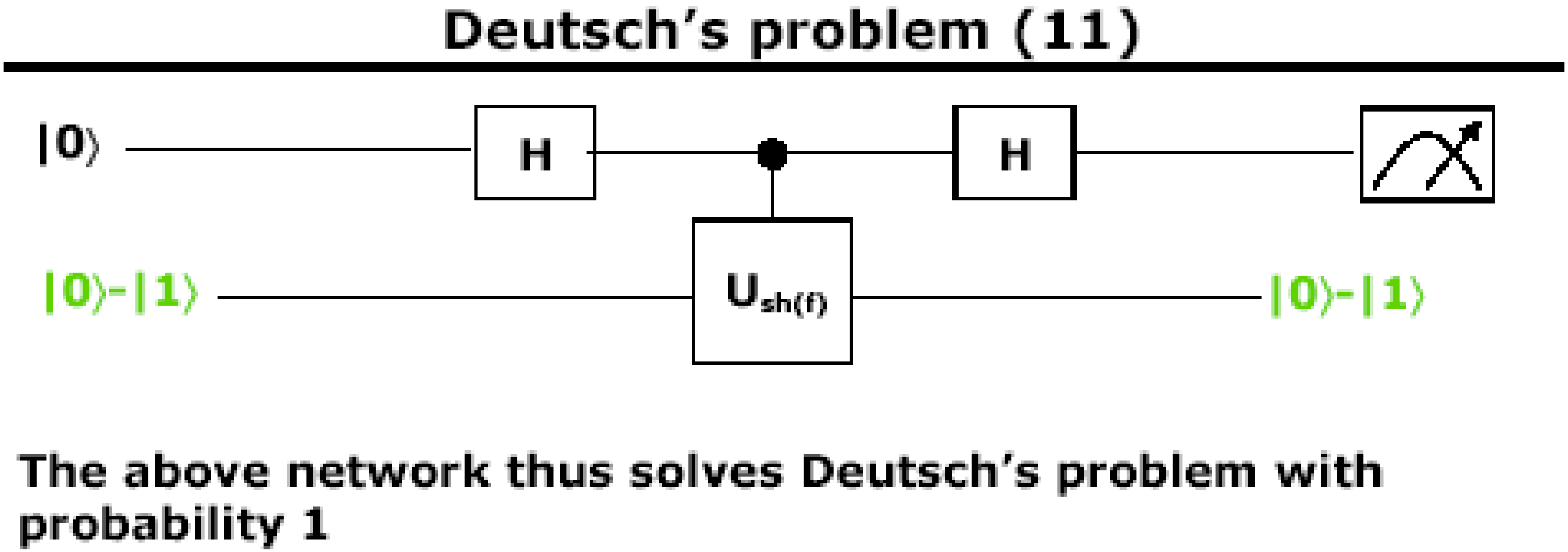
Eigenvalues of $U_{sh(f)}$ are $\{1, e^{i\pi f(0)\oplus f(1)}\}$

Thus, the answer $f(0)\oplus f(1)$ is encoded in the "phase" of an eigenvalue of the shift operator!

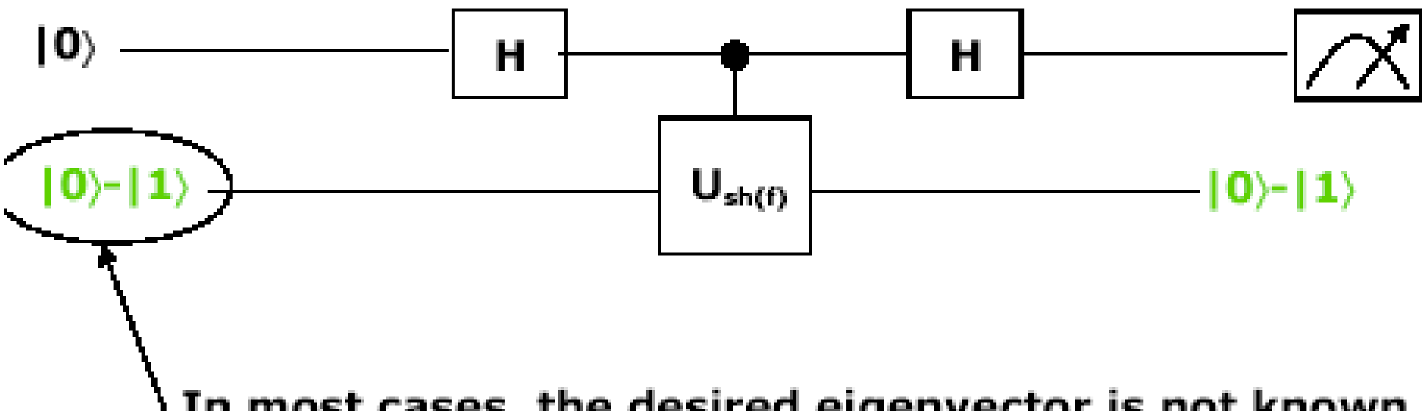
We know that if we input the eigenvector $|0\rangle-|1\rangle$ in the bottom register, the controlled-shift gate will kick back this relative phase into the top qubit; the phase $\varphi=\pi f(0)\oplus f(1)$ is either 0 or π

From our interferometry experiment, we know we can distinguish the two cases $\varphi=0$ or $\varphi=\pi$...

Shift operator allows to solve Deutsch's problem with certainty



Controlling amplitude versus controlling phase



In most cases, the desired eigenvector is not known

However $|0\rangle = (|0\rangle+|1\rangle) + (|0\rangle-|1\rangle)$

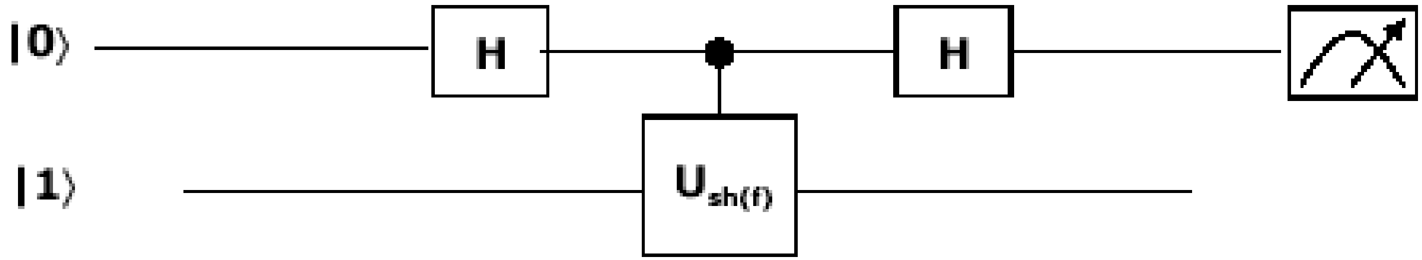
$|1\rangle = (|0\rangle+|1\rangle) - (|0\rangle-|1\rangle)$

It turns out we can *always* resort to inputting an equal superposition of eigenvectors of the shift operator, which will give the desired eigenvalue kick back in the top qubit with some reasonable probability (in this case ½)

(We actually already saw this effect in the controlled-⊕f(x) solution to Deutsch's problem)

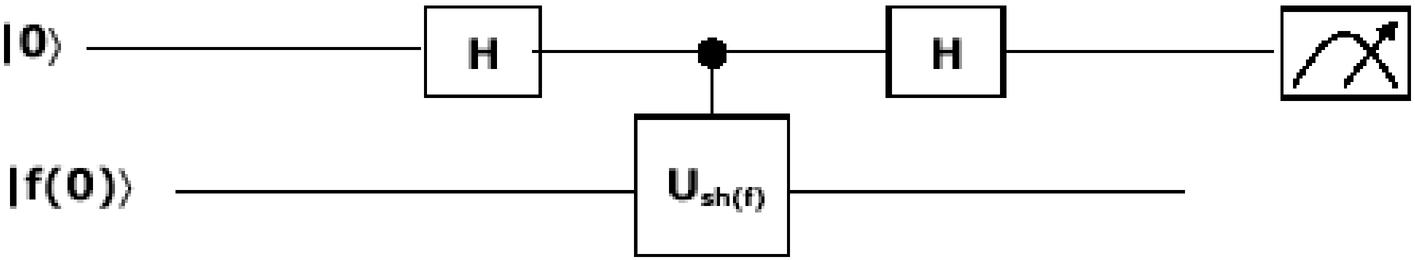
Controlling amplitude versus controlling phase

Thus, the following network solves Deutsch's problem with probability 1/2



Suppose we were given the state $|f(0)\rangle$, which is a uniform superposition of the eigenvectors of $U_{sh(f)}$ (in general, as we'll see later!)

Then, the following network solves Deutsch's problem with probability 1/2

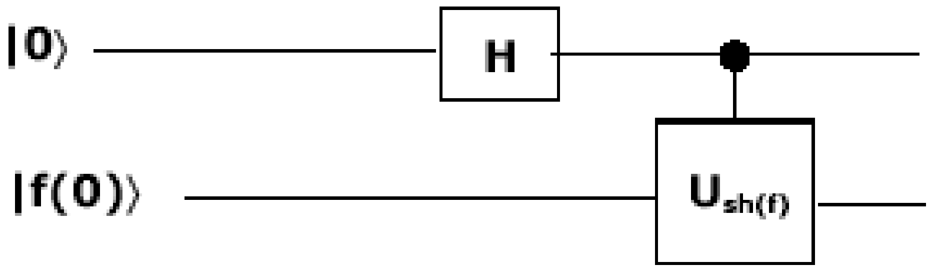


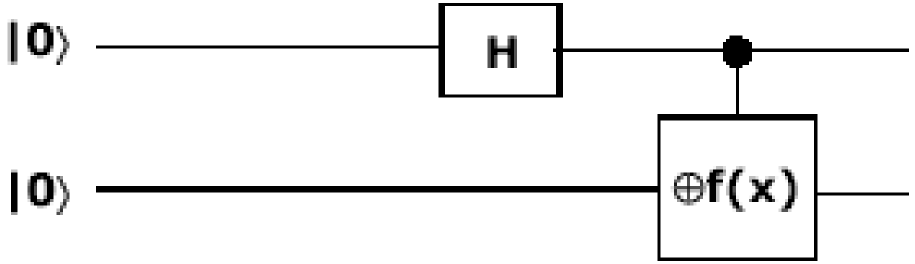
Exercise for students

Exercise for students

Controlled-$\oplus f(x)$ v. Controlled-$U_{sh(f)}$ (3)

Exercise: Compare the states produced by the following networks





The equivalence of these two states is the fundamental link between the shift operator (eigenvalue-estimation approach to quantum algorithms) and the controlled-$\oplus f(x)$ operator (standard approach)

Sources used

Dave Bacon
Lawrence Ioannou