PHYC90045 Introduction to Quantum Computing

Week 3

Lecture 5
Universality in quantum computing, Reversible computation, one qubit adder, the Deutsch-Josza algorithm

Lecture 6
Two basic quantum algorithms: Bernstein-Vazirani and Simon's Algorithms

Lab 3
Logical statements, Reversible logic, Adder, Deutsch-Josza algorithm

PHYC90045 Introduction to Quantum Computing

Overview

In this lecture we will discuss reversible logic and our first quantum algorithm – the Deutsch-Josza algorithm,

1. Reversible (classical) logic
2. The Deutsch-Josza algorithm
3. Aside: Universality in quantum computing

Along the way we will encounter common patterns often turn up in quantum algorithms, and will highlight them because they will help make sense of what of future quantum circuits.

See:

Kaye, 1.5, 6.1-6.4
Nielsen and Chuang, 1.4, 3.1
Reiffel, 6, 7.3-7.5

PHYC90045 Introduction to Quantum Computing

Universality in classical logic

A set of (classical) gates is said to be *functionally complete (or "universal")* if every possible truth table (ie. Boolean function) can be expressed using members of the set.

For example, in classical logic: {AND, OR, NOT} is functionally complete.

AND OR NOT

Every logical circuit can be made from combinations of these gates. In fact, the NAND gate alone is universal. However, we cannot implement these gates directly in a quantum computer. AND/OR are **not reversible**.

Irreversible Functions

We cannot implement AND or OR because these functions are irreversible.

A	B	A B
0	0	0
0	1	0
1	0	0
1	1	1

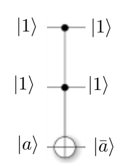
We cannot determine the inputs from the output.

Irreversible functions are *not unitary*.

Reversible Logic

Classically, if we would like to calculate some Boolean function, f, we can construct a circuit out of AND, OR and NOT gates. However, AND and OR gates are **not reversible**, and so can't be implemented in a quantum computer.

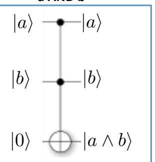
But by use of reversible circuits and additional bits/qubits, we can express everything in terms of reversible gates (such as Toffoli):



NOT a

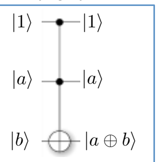
Reversible Logic and the Toffoli Gate

a AND b



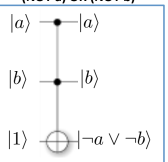
a	b	$a \wedge b$
0	0	0
0	1	0
1	0	0
1	1	1

a XOR b



a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

(NOT a) OR (NOT b)



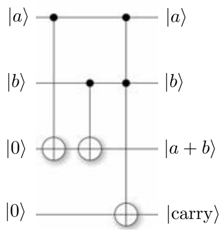
a	b	$\neg a \vee \neg b$
0	0	1
0	1	1
1	0	1
1	1	0

The Toffoli gate is **universal and reversible**. In principle every classical boolean function can be written in terms of reversible gates (such as Toffoli) which can be implemented on a quantum computer.

PHYC90045 Introduction to Quantum Computing

One Bit Adder

We can, for example, implement a one bit adder using only reversible gates:



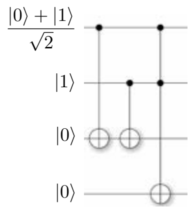
We will now explain this circuit and in the lab we will extend this to a two bit adder.

PHYC90045 Introduction to Quantum Computing

1+1 Quantum Style

Here is what happens when we add together numbers in superposition:

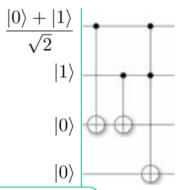
$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad \text{“+”} \quad |1\rangle$$



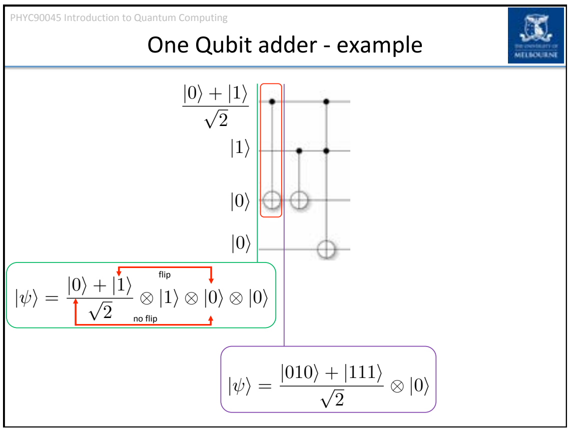
Let's do the walkthrough…

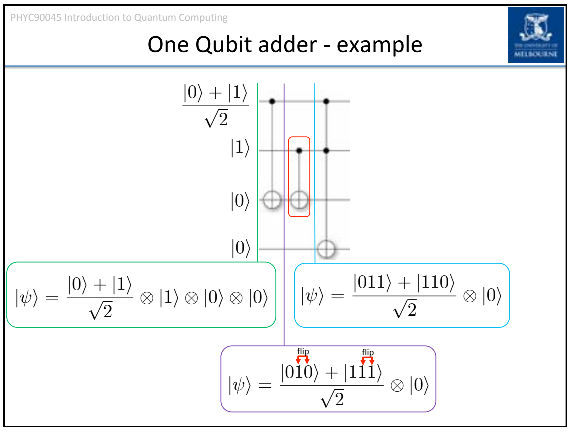
PHYC90045 Introduction to Quantum Computing

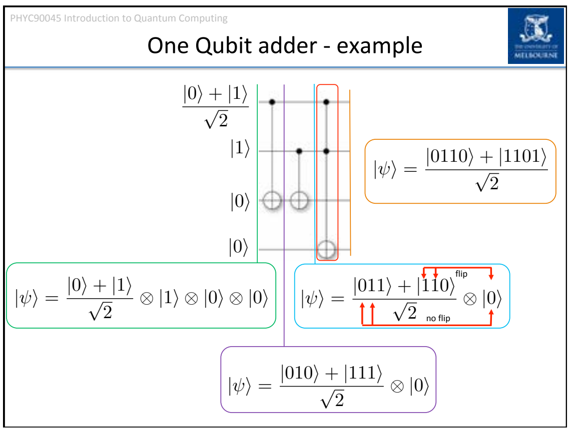
One Qubit adder - example



$$|\psi\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle$$

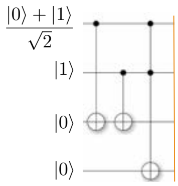

PHYC90045 Introduction to Quantum Computing
One Qubit adder - example
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|1\rangle$
$|0\rangle$
$|0\rangle$
$|\psi\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle$
flip
no flip
$|\psi\rangle = \frac{|010\rangle+|111\rangle}{\sqrt{2}} \otimes |0\rangle$


PHYC90045 Introduction to Quantum Computing
One Qubit adder - example
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|1\rangle$
$|0\rangle$
$|0\rangle$
$|\psi\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle$
$|\psi\rangle = \frac{|011\rangle+|110\rangle}{\sqrt{2}} \otimes |0\rangle$
flip
flip
$|\psi\rangle = \frac{|010\rangle+|111\rangle}{\sqrt{2}} \otimes |0\rangle$


PHYC90045 Introduction to Quantum Computing
One Qubit adder - example
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|1\rangle$
$|0\rangle$
$|0\rangle$
$|\psi\rangle = \frac{|0110\rangle+|1101\rangle}{\sqrt{2}}$
$|\psi\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle$
flip
no flip
$|\psi\rangle = \frac{|011\rangle+|110\rangle}{\sqrt{2}} \otimes |0\rangle$
$|\psi\rangle = \frac{|010\rangle+|111\rangle}{\sqrt{2}} \otimes |0\rangle$

PHYC90045 Introduction to Quantum Computing

One Qubit adder - example



$$|\psi\rangle = \frac{|0110\rangle + |1101\rangle}{\sqrt{2}}$$

Or, considering the last two registers as a two-qubit binary register (msb last):

1 = 10
2 = 01

$$|\psi\rangle = \frac{|0110\rangle + |1101\rangle}{\sqrt{2}}$$

$$|\psi\rangle = \frac{|0\rangle\,|1\rangle\,|1\rangle + |1\rangle\,|1\rangle\,|2\rangle}{\sqrt{2}}$$

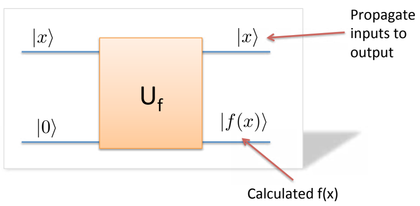
0+1=1 1+1=2

Note: this is an entangled state.

PHYC90045 Introduction to Quantum Computing

Implementing irreversible functions

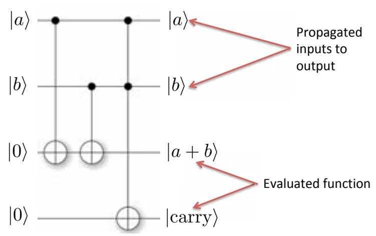
We cannot compute **irreversible** functions directly (not unitary). One strategy which you often see to make an irreversible function reversible is simply to propagate the input to the output:



PHYC90045 Introduction to Quantum Computing

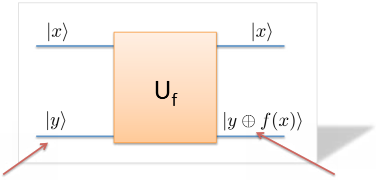
One Bit Adder

The adder is an example. Given $a+b$, we can't uniquely determine a and b
So we used this trick:



Implementing irreversible functions

One strategy which you often see to make an irreversible function reversible is simply to propagate the inputs:



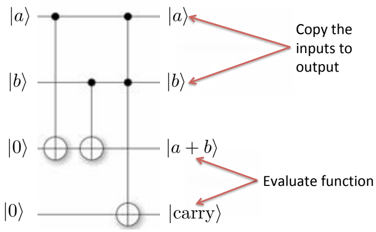
Must deal with all possible inputs, not just 0.

Add (bit-by-bit modulo 2) input and calculated f(x)

You will see this pattern in several of the quantum algorithms we will study.

One Bit Adder

The adder is an example. Given a+b, we can't uniquely determine a and b. So we used this trick:

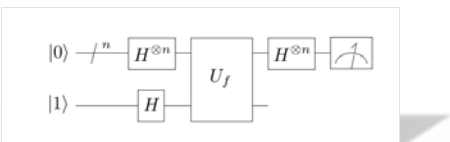


In the lab we will extend this to a two bit adder.

Deutsch-Josza algorithm

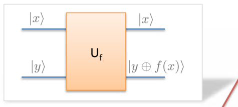
- Given a boolean function, ***f***, determine if:
  f is constant (always gives the same result), or
  f is balanced (gives equal numbers of 0s and 1s)

- **Classical algorithm** *(worst case) needs $2^n/2+1$ queries*
- ***Quantum algorithm*** *needs just 1 query.*

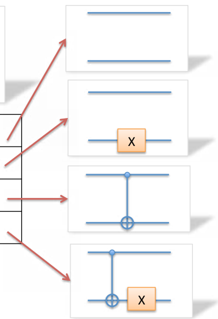


Deutsch-Josza algorithm (2)

Let's take the example with just one bit/qubit input. There are 4 choices of function:

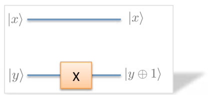


CONSTANT	$f_1(0) = 0$ $f_1(1) = 0$
	$f_2(0) = 1$ $f_2(1) = 1$
BALANCED	$f_3(0) = 0$ $f_3(1) = 1$
	$f_4(0) = 1$ $f_4(1) = 0$



PHYC90045 Introduction to Quantum Computing

Example of a constant function

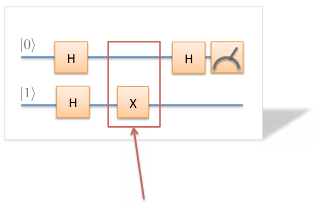


$$\begin{cases} f(0) = 1 \\ f(1) = 1 \end{cases}$$

The circuit always flips (ie. NOT) the second qubit, regardless of the input.
This function is **constant**, since the output is always 1.

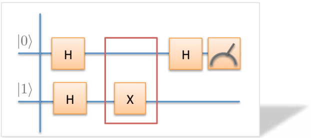
PHYC90045 Introduction to Quantum Computing

Deutsch algorithm: constant function



The function, U, is implemented by the gates inside this box

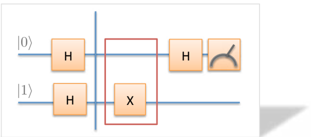
Deutsch algorithm: walkthrough



$$|\psi\rangle = |0\rangle \otimes |1\rangle$$

PHYC90045 Introduction to Quantum Computing

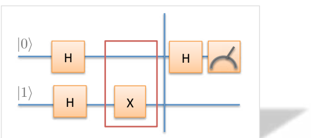
Deutsch algorithm: walkthrough



$$\begin{aligned} |\psi\rangle &= H|0\rangle \otimes H|1\rangle \\ &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\ &= |+\rangle \otimes |-\rangle \end{aligned}$$

PHYC90045 Introduction to Quantum Computing

Deutsch algorithm: walkthrough

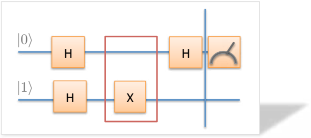


$$\begin{aligned} |\psi\rangle &= |+\rangle \otimes X|-\rangle \\ &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes X\frac{|0\rangle - |1\rangle}{\sqrt{2}} \\ &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|1\rangle - |0\rangle}{\sqrt{2}} \\ &= -|+\rangle \otimes |-\rangle \end{aligned}$$

Global phase is unmeasurable

PHYC90045 Introduction to Quantum Computing

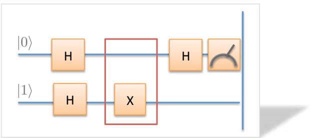
Deutsch algorithm: walkthrough



$$
\begin{aligned}
|\psi\rangle &= -H\,|+\rangle \otimes |-\rangle \\
&= -\,|0\rangle \otimes |-\rangle
\end{aligned}
$$

PHYC90045 Introduction to Quantum Computing

Deutsch algorithm: walkthrough

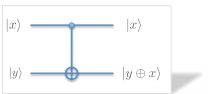


$$
\begin{aligned}
|\psi\rangle &= -\,|0\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&= -\,|0\rangle \otimes |-\rangle
\end{aligned}
$$

We will measure "0" with 100% probability. This indicates (with a single evaluation of the function) that the function is **constant.**

PHYC90045 Introduction to Quantum Computing

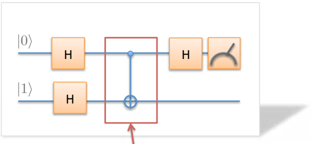
Example of a balanced function



$$
\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}
$$

The circuit only flips (ie. NOT) the second qubit, if the input is a 1.
This function is **balanced**, since the output has equal numbers of 0 and 1 output.

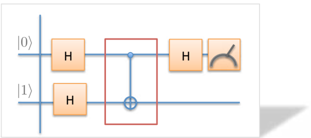
Deutsch algorithm: balanced function



The function, U, is implemented by the gates inside this box

PHYC90045 Introduction to Quantum Computing

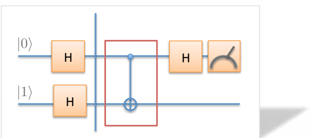
Deutsch algorithm: walkthrough



$$|\psi\rangle = |0\rangle \otimes |1\rangle$$

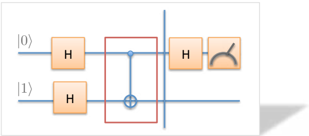
PHYC90045 Introduction to Quantum Computing

Deutsch algorithm: walkthrough



$$\begin{aligned} |\psi\rangle &= H|0\rangle \otimes H|1\rangle \\ &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\ &= |+\rangle \otimes |-\rangle \end{aligned}$$

Deutsch algorithm: walkthrough



$$|\psi\rangle = CNOT\,|+\rangle\,|-\rangle$$

This is a common pattern called "phase kickback"

Phase kickback

Consider X applied to the output register:

$$X\,|-\rangle = X\frac{|0\rangle - |1\rangle}{\sqrt{2}}$$
$$= \frac{|1\rangle - |0\rangle}{\sqrt{2}}$$
$$= -\,|-\rangle$$

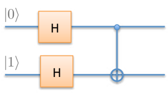
If we were to apply an X gate to the target qubit we get a *global phase* change.

Otherwise the state is unchanged.

What would happen when apply a *control*-X gate?

Phase kickback

Now with a control-X gate:

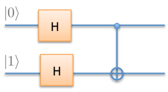


$$\mathrm{CNOT}\,|+\rangle\,|-\rangle = \mathrm{CNOT}\frac{|0\rangle + |1\rangle}{\sqrt{2}}\frac{|0\rangle - |1\rangle}{\sqrt{2}} = \frac{|0\rangle}{\sqrt{2}}\frac{|0\rangle - |1\rangle}{\sqrt{2}} + \frac{|1\rangle}{\sqrt{2}}\frac{|1\rangle - |0\rangle}{\sqrt{2}}$$

$$CNOT\,|+\rangle\,|-\rangle = \frac{|0\rangle\,|-\rangle - |1\rangle\,|-\rangle}{\sqrt{2}}$$

If we were to apply a control-X gate then any control states which apply the X-gate receive a phase change. The "1" state receives the (relative) phase change in this case.

Phase kickback

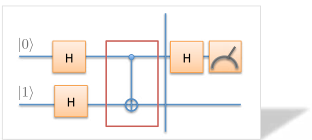


$$CNOT\,|+\rangle\,|-\rangle = \frac{|0\rangle\,|-\rangle - |1\rangle\,|-\rangle}{\sqrt{2}}$$
$$= \frac{|0\rangle - |1\rangle}{\sqrt{2}}\,|-\rangle$$
$$= |-\rangle\,|-\rangle$$

This causes the **phase** to be applied to the *control qubit*. This is known as phase kickback.

The state of the target qubit remains unchanged.

Deutsch algorithm: walkthrough

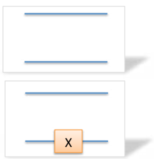


$$|\psi\rangle = CNOT\,|+\rangle\,|-\rangle = |-\rangle\,|-\rangle$$

Phase kickback changes the state of the *control* qubit.

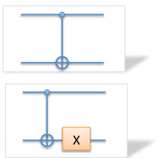
Phase kickback for balanced functions

Constant functions



The control qubit is unchanged by the **constant** functions.

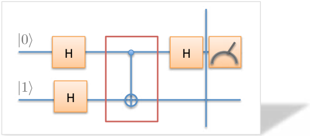
Balanced functions



The same phase kickback pattern that we just saw applies to both **balanced** functions

PHYC90045 Introduction to Quantum Computing

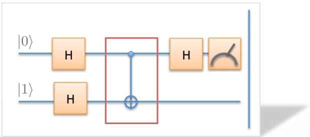
Deutsch algorithm: walkthrough



$$|\psi\rangle = H\,|-\rangle \otimes |-\rangle$$
$$= |1\rangle \otimes |-\rangle$$

PHYC90045 Introduction to Quantum Computing

Deutsch algorithm: walkthrough

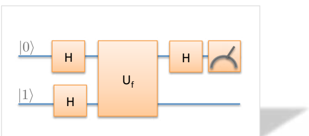


$$|\psi\rangle = |1\rangle \otimes |-\rangle$$

We will measure "1" with 100% probability. This indicates (with a single evaluation of the function) that the function is **balanced.**

PHYC90045 Introduction to Quantum Computing

Deutsch-Josza (3)

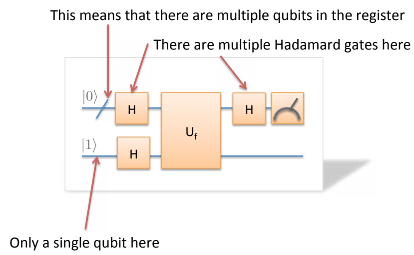


		Measured
CONSTANT	$f_1(0) = 0$	0
	$f_1(1) = 0$	
	$f_2(0) = 1$	0
	$f_2(1) = 1$	
BALANCED	$f_3(0) = 0$	1
	$f_3(1) = 1$	
	$f_4(0) = 1$	1
	$f_4(1) = 0$	

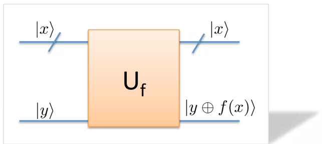
The Deutsch-Josza algorithm determines in just *one query* whether the function is constant or balanced.

Classically, this would require *two queries.*

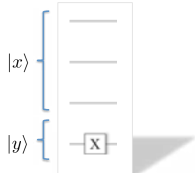
Multiple qubits: Deutsch-*Josza*



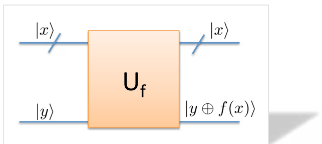
Example of multi-qubit constant function



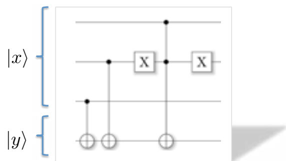
x	f(x)
000	1
001	1
010	1
011	1
100	1
101	1
110	1
111	1



Example of multi-qubit balanced function

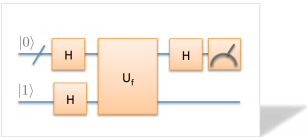


x	f(x)
000	0
001	1
010	1
011	0
100	1
101	0
110	1
111	0



PHYC90045 Introduction to Quantum Computing

Multiple qubits: Deutsch-*Josza*

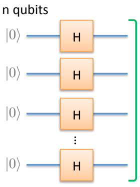


Let's walkthrough this circuit…

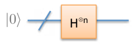
PHYC90045 Introduction to Quantum Computing

Recap: binary and decimal representations

n qubits



shorthand notation



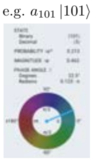
$$|\psi\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}} ... \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$|\psi\rangle = \left[\frac{1}{\sqrt{2}}\right]^n (|00...0\rangle + ... + |11...1\rangle)$$

i.e. even superposition over binary rep of integers: i = 0 to $2^n - 1$

In general we use two representations in the QUI ($N = 2^n$):

e.g. $a_{101}\,|101\rangle$



"binary"

$$|\psi\rangle = a_{0...00}\,|0...00\rangle + a_{0...01}\,|0...01\rangle + a_{0...10}\,|0...10\rangle + ... + a_{1...1}\,|1...1\rangle$$

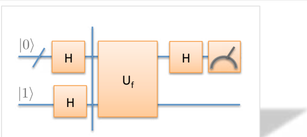
"decimal"

$$|\psi\rangle = a_0\,|0\rangle + a_1\,|1\rangle + a_2\,|2\rangle + ... + a_{N-1}\,|N-1\rangle$$

$$\boxed{|\psi\rangle = \sum_\phi a_\phi\,|\phi\rangle \quad a_\phi = |a_\phi| e^{i\theta_\phi}}$$

PHYC90045 Introduction to Quantum Computing

Deutsch-Josza Walkthrough

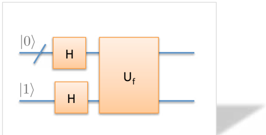


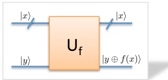
After the initial Hadamard gates, the state is (n qubits, N = 2^n):

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Equal (even) superposition of states

General Function Phase Kickback



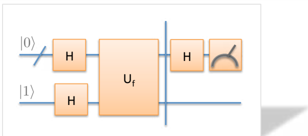


Using phase kickback, after the function has been applied:

$$|\psi\rangle = \frac{1}{\sqrt{N}}\sum_x (-1)^{f(x)}|x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

If the function evaluates to "1" then the target qubit is flipped, and we pick up a phase. Otherwise, there is no phase applied. This is a simple way to write that.

Deutsch-Josza Walkthrough

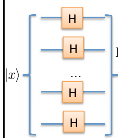


Using phase kickback, after the function has been applied:

$$|\psi\rangle = \frac{1}{\sqrt{N}}\sum_x (-1)^{f(x)}|x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

If the function evaluates to "1" then the target qubit is flipped, and we pick up a phase. Otherwise, there is no phase applied. This is a simple way to write that.

Hadamard applied to a general state



$$\mathrm{H}^{\otimes n}|x\rangle = \frac{1}{\sqrt{N}}\sum_{z=0}^{N-1} a_z|z\rangle$$

Amplitude a_z -> how many times does the binary representation of z and x have 1's in the same location?

$$x_0 z_0 + x_1 z_1 + x_2 z_2 + \ldots + x_n z_n$$

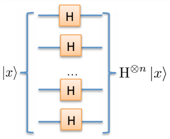
Shorthand for the bitwise dot product is: $x \cdot z = \sum_{j=0}^{n} x_j z_j$

When 1's in the same location, we get a sign change -> $(-1)^{x \cdot z}$

Hadamards applied to a general state (n qubits, N = 2^n):

$$\mathrm{H}^{\otimes n}|x\rangle = \frac{1}{\sqrt{N}}\sum_{z=0}^{N-1}(-1)^{x \cdot z}|z\rangle$$

e.g. Hadamard applied to a general state



$$\mathrm{H}^{\otimes 3}\overset{x=0}{|000\rangle} = \frac{|0\rangle+|1\rangle}{\sqrt{2}}\frac{|0\rangle+|1\rangle}{\sqrt{2}}\frac{|0\rangle+|1\rangle}{\sqrt{2}}$$

$$\mathrm{H}^{\otimes 3}|000\rangle = \frac{1}{\sqrt{2^3}}\underbrace{(\overset{z=0}{|000\rangle}+\overset{z=1}{|001\rangle}+\ldots+\overset{z=7}{|111\rangle})}_{\text{bitwise: } x\cdot z=0}$$

$$\mathrm{H}^{\otimes 3}|x=0\rangle = \frac{1}{\sqrt{2^3}}\sum_{z=0}^{2^3-1}(-1)^{x\cdot z}|z\rangle$$

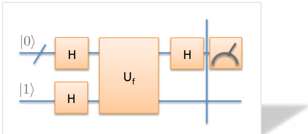
$$\mathrm{H}^{\otimes 3}\overset{x=4}{|100\rangle} = \frac{1}{\sqrt{2^3}}\frac{|0\rangle-|1\rangle}{\sqrt{2}}\frac{|0\rangle+|1\rangle}{\sqrt{2}}\frac{|0\rangle+|1\rangle}{\sqrt{2}}$$

$$\mathrm{H}^{\otimes 3}|100\rangle = \frac{1}{\sqrt{2^3}}(\underbrace{|000\rangle+|010\rangle+|001\rangle+|011\rangle}_{x\cdot z=0}-\underbrace{|100\rangle-|110\rangle-|101\rangle-|111\rangle)}_{x\cdot z=1}$$

$$\mathrm{H}^{\otimes 3}|x=4\rangle = \frac{1}{\sqrt{2^3}}\sum_{z=0}^{2^3-1}(-1)^{x\cdot z}|z\rangle \qquad \boxed{\mathrm{H}^{\otimes 3}|x\rangle = \frac{1}{\sqrt{2^3}}\sum_{z=0}^{2^3-1}(-1)^{x\cdot z}|z\rangle}$$

General for n=3

Deutsch-Josza Walkthrough



$$\mathrm{H}^{\otimes n}|x\rangle = \frac{1}{\sqrt{N}}\sum_{z=0}^{N-1}(-1)^{x\cdot z}|z\rangle$$

And after the final Hadamard gates:

$$|\psi\rangle = \mathrm{H}^{\otimes n}\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}(-1)^{f(x)}|x\rangle$$

$$|\psi\rangle = \frac{1}{N}\sum_{x=0}^{N-1}(-1)^{f(x)}\sum_{z=0}^{N-1}(-1)^{x\cdot z}|z\rangle$$

Constant function

For a constant function (f(x) = 0 for all x, or f(x) = 1 for all x):

$$\begin{aligned}|\psi\rangle &= \frac{1}{N}\sum_{x=0}^{N-1}(-1)^{f(x)}\sum_{z=0}^{N-1}(-1)^{x\cdot z}|z\rangle\\ &= \frac{(-1)^{f(0)}}{N}\sum_{x=0}^{N-1}\sum_{z=0}^{N-1}(-1)^{x\cdot z}|z\rangle\\ &= \frac{(-1)^{f(0)}}{N}\sum_{z=0}^{N-1}\left(\sum_{x=0}^{N-1}(-1)^{x\cdot z}\right)|z\rangle\\ &= (-1)^{f(0)}|z=0\rangle\end{aligned}$$

$$\sum_{x=0}^{N-1}(-1)^{x\cdot z} = \begin{cases} N, & z=0\\ 0, & z\neq 0\end{cases}$$

So for a constant function “0” will always be measured (global phase is unimportant).

PHYC90045 Introduction to Quantum Computing

Balanced Function

$$|\psi\rangle = \mathrm{H}^{\otimes n}\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}(-1)^{f(x)}|x\rangle$$

$$|\psi\rangle = \frac{1}{N}\sum_{x=0}^{N-1}(-1)^{f(x)}\sum_{z=0}^{N-1}(-1)^{x\cdot z}|z\rangle$$

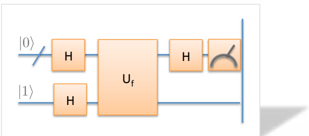
For a balanced function (equal number of f(x) = 0 and f(x) = 1):

$$|\psi\rangle = \frac{1}{N}\sum_{z=0}^{N-1}\left(\sum_{x,f(x)=0}(-1)^{x\cdot z} - \sum_{x,f(x)=1}(-1)^{x\cdot z}\right)|z\rangle$$

Which has zero amplitude for the $|z=0\rangle$ state, and non-zero for other states.

PHYC90045 Introduction to Quantum Computing

Deutsch-Josza Walkthrough



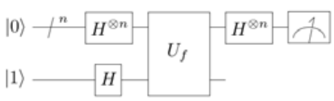
If 0 is measured, then the function is constant.
If *any other* value is measured, then the function is balanced.

The Deutsch-Josza algorithm evaluates if a function is constant or balanced with a single query. Classically we would require $\mathrm{O}(2^n)$ queries.

Of course, there are classical probabilistic algorithms with establish with high probability in few queries, but only with high probability of success not with certainty.

PHYC90045 Introduction to Quantum Computing

Deutsch-Josza algorithm

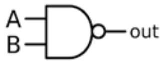


- Given a Boolean function, ***f***, determine if:
  - *f* is constant (always gives the same result)
  - *f* is balanced (gives equal numbers of 0s and 1s)

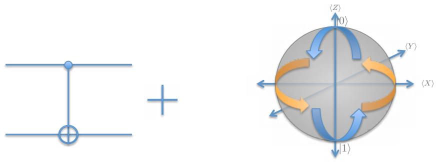
- **Classical algorithm** *(worst case) needs $2^n/2+1$ queries*
- ***Quantum algorithm*** *needs just 1 query.*

Aside: Universality in Quantum Computing

In classical computing the NAND gate is universal: every Boolean function can be implemented as a sequence of NAND (NOT AND) gates



In quantum computing every quantum circuit can be expressed as a sequence of:

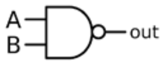


CNOT

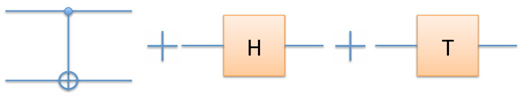
Single Qubit Rotations

Aside: Universality in Quantum Computing

In classical computing the NAND gate is universal: every Boolean function can be implemented as a sequence of NAND (NOT AND) gates



In quantum computing every quantum circuit can be approximated as a sequence of:



CNOT

Hadamard

T Gate
(Z axis, pi/4)

Aside: Outline of Universal Gate Set Proof

(1) The following sequence of gates creates an irrational fraction of 2π angle rotation gate:

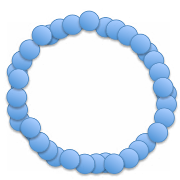
$$THTH$$

Specifically, you can show by direct multiplication that it is a rotation around

$$\vec{n} = \left(\cos\frac{\pi}{8}, \sin\frac{\pi}{8}, \cos\frac{\pi}{8}\right)$$

By an angle defined by: $\cos\frac{\theta}{2} = \cos^2\frac{\pi}{8}$

(2) You can use this to approximate (within some error, δ) **every rotation around n**. This takes ~2π/δ applications of THTH.



Aside: Outline of Universal Gate Set Proof

(3) The following sequence of gates creates an irrational fraction of 2π

$$HTHT$$

Around a different axis:

$$\vec{m} = \left(\cos\frac{\pi}{8}, -\sin\frac{\pi}{8}, \cos\frac{\pi}{8}\right)$$

By an angle defined by: $\cos\frac{\theta}{2} = \cos^2\frac{\pi}{8}$

(4) You can use this to approximate (within some error, ε) **every rotation around m**. This takes ~2π/ε applications of HTHT.

Aside: Outline of Universal Gate Set Proof

(5) These gates can (approximately) implement any rotation around two different axes, you can use this to approximate any single qubit rotation.

(6) Single qubit rotations plus the CNOT gate is universal

(7) Therefore {H, T, CNOT} is universal.

Full proof (carefully keeping track of approximations)
is found in Nielsen and Chuang 4.5.3

Week 3

Lecture 5
Universality in quantum computing, Reversible computation, one qubit adder, the Deutsch-Josza algorithm

Lecture 6
Two basic quantum algorithms: Bernstein-Vazirani and Simon's Algorithms

Lab 3
Logical statements, Reversible logic, Adder, Deutsch-Josza algorithm