IIT Madras
Physics

CS6841 Quantum Algorithms and Quantum Cryptography

Teacher: Shweta Agrawal

Chaganti Kamaraja Siddhartha

EP20B012

Homework I — July-Nov 2022

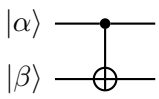
Due date: August 24th, 5pm

Contents

1 Building Gates

1.1 Building CNOT gate

CNOT gate changes the target qubit from state $|0\rangle$ to state $|1\rangle$ and vice-versa if control qubit is in state $|1\rangle$, no change in state otherwise.

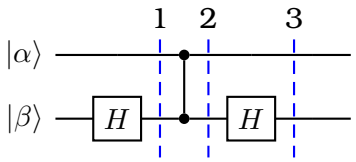


Initial State,

$$\text{Let, } |\alpha\rangle |\beta\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

Final state after **CNOT** is applied,

$$|\phi\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |11\rangle + a_3 |10\rangle$$



Initial State,

$$|\alpha\rangle |\beta\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

Hadamard gate applied on second qubit rotates state $|0\rangle$ to state $|+\rangle$ and state $|1\rangle$ to state $|-\rangle$

$$|\psi_1\rangle = \frac{a_0}{\sqrt{2}}(|00\rangle + |01\rangle) + \frac{a_1}{\sqrt{2}}(|00\rangle - |01\rangle) + \frac{a_2}{\sqrt{2}}(|10\rangle + |11\rangle) + \frac{a_3}{\sqrt{2}}(|10\rangle - |11\rangle)$$

Control-Z gate is applied between first and second qubit, it rotates the state $|11\rangle$ to state $-|11\rangle$ and vice-versa, others remain unaltered.

$$|\psi_2\rangle = \frac{a_0}{\sqrt{2}}(|00\rangle + |01\rangle) + \frac{a_1}{\sqrt{2}}(|00\rangle - |01\rangle) + \frac{a_2}{\sqrt{2}}(|10\rangle - |11\rangle) + \frac{a_3}{\sqrt{2}}(|10\rangle + |11\rangle)$$

re-writing state $|\psi_2\rangle$ by changing basis vector of second qubit,

$$|\psi_2\rangle = a_0 |0\rangle |+\rangle + a_1 |0\rangle |-\rangle + a_2 |1\rangle |-\rangle + a_3 |1\rangle |+\rangle$$

Hadamard gate is applied on second qubit which rotates state $|+\rangle$ to state $|0\rangle$ and state $|-\rangle$ to $|1\rangle$

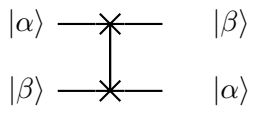
$$|\psi_3\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |11\rangle + a_3 |10\rangle$$

The final state $|\psi_3\rangle$ is equal to the final state $|\phi\rangle$ when **CNOT** gate is applied to initial state.

Therefore, we can build **CNOT** gate from **two Hadamard gates and one Control-Z gate** by first applying **Hadamard gate** on **target qubit** then applying **Control-Z gate** between **control and target qubits** and applying **Hadamard gate** on **target qubit**.

1.2 Building SWAP-gate from CNOT

SWAP gate swaps the two qubits.



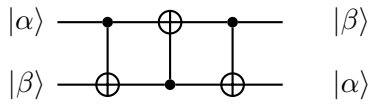
Initial State,

$$Let, |\alpha\rangle |\beta\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

Final state after **SWAP** is applied,

$$|\phi\rangle = a_0 |00\rangle + a_2 |01\rangle + a_1 |10\rangle + a_3 |11\rangle$$

1.2.1 Constructed SWAP gate



Initial State,

$$|\phi_0\rangle = |\alpha\rangle |\beta\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

CNOT gate changes the target qubit from state $|0\rangle$ to state $|1\rangle$ and vice-versa if control qubit is in state $|1\rangle$, no change in state otherwise.

$$|\phi_1\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |11\rangle + a_3 |10\rangle$$

Second CNOT gate applied, where second qubit is control and first qubit is target.

$$|\phi_2\rangle = a_0 |00\rangle + a_1 |11\rangle + a_2 |01\rangle + a_3 |10\rangle$$

Third CNOT gate applied, where first qubit is control and second qubit is target.

$$|\phi_3\rangle = a_0 |00\rangle + a_1 |10\rangle + a_2 |01\rangle + a_3 |11\rangle$$

The final state $|\phi_3\rangle$ is equal to final state $|\phi\rangle$ when SWAP gate is applied to initial state. Therefore, we can build **SWAP gate from three CNOT gates** by first applying **CNOT gate with first qubit as control and second qubit as target**, then **CNOT gate again with second qubit as control and first qubit as target** finally, **CNOT gate with first qubit as control qubit and second qubit as target qubit**. Then we can build SWAP gate from 3 CNOT gates.

2 Understanding Unitary

2.1 Norm-Preserving

Since, A is norm-preserving:

$$||A(v+w)|| = ||v+w||$$

$$||A(v+w)||^2 = ||v+w||^2$$

$$||v+w||^2 = \langle v+w|v+w\rangle = \langle v|v\rangle + \langle v|w\rangle + \langle w|v\rangle + \langle w|w\rangle = ||v||^2 + \langle v|w\rangle + \langle w|v\rangle + ||w||^2$$

$$||A(v+w)||^2 = \langle A(v+w)|A(v+2)\rangle = \langle Av|Av\rangle + \langle Av|Aw\rangle + \langle Aw|Av\rangle + \langle Aw|Aw\rangle$$

$$\implies ||Av||^2 + \langle Av|Aw\rangle + \langle Aw|Av\rangle + ||Aw||^2$$

Since, A is norm preserving, $||Av|| = ||v||$.
Therefore,

$$\langle Av|Aw\rangle + \langle Aw|Av\rangle = \langle v|w\rangle + \langle w|v\rangle$$

$$Re(\langle Av|Aw\rangle) = Re(\langle Aw|Av\rangle) \ (\because (\langle Av|Aw\rangle)^\dagger = \langle Aw|Av\rangle)$$

similarly,

$$Re(\langle v|w\rangle) = Re(\langle w|v\rangle) \ (\because (\langle v|w\rangle)^\dagger = \langle w|v\rangle)$$

Therefore,

$$2Re(\langle Av|Aw\rangle) = 2Re(\langle v|w\rangle)$$

Substituting $w = iw$

$$i(\langle Av|Aw\rangle - \langle Aw|Av\rangle) = i(\langle v|w\rangle - \langle w|v\rangle)$$

hence, $-2i(\langle Av|Aw\rangle = -2i\langle v|w\rangle)$. Thus, the imaginary parts of $\langle Av|Aw\rangle$ and $\langle v|w\rangle$ are equal. Therfore,

$$\langle Av|Aw\rangle = \langle v|w\rangle$$

A is Norm-Preserving if and only if A is inner product Preserving.

2.2 Inner Product Preserving

Consider the operator,

$$I - A^\dagger A$$

$$\left\langle v\middle|I - A^\dagger A\middle|w\right\rangle = \langle v|I|w\rangle - \left\langle v\middle|A^\dagger A\middle|w\right\rangle$$

$$\left\langle v\middle|I - A^\dagger A\middle|w\right\rangle = \langle v|w\rangle - \langle Av|Aw\rangle$$

since, A is inner product preserving,

$$\langle Av|Aw\rangle = \langle v|w\rangle$$

Therefore,

$$\left\langle v\middle|I - A^\dagger A\middle|w\right\rangle = \langle v|w\rangle - \langle v|w\rangle = 0$$

For this to be true for all matrices in Vector space, $I - A^\dagger A = 0$.
Therefore,

$$A^\dagger A = I \implies AA^\dagger A = A \implies AA^\dagger = AA^{-1} = I$$

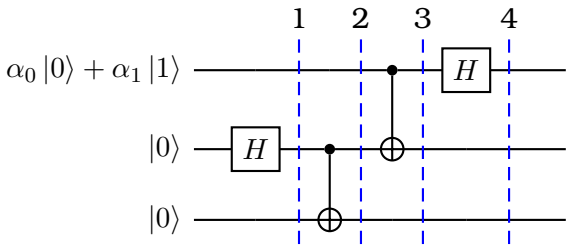
$$A^\dagger A = AA^\dagger = I$$

A is Inner product Preserving if and only if A is unitary.

2.3 Norm-preserving and Unitary

A is Norm-preserving iff A is Inner product preserving. A is Inner product preserving iff A is Unitary.
Therefore, **A is Norm-Preserving iff A is Unitary**

3 Quantum Circuits



3.1 States of three Qubits

Initial state

$$|\psi_0\rangle = \alpha_0|000\rangle + \alpha_1|100\rangle$$

Hadamard gate is applied to second qubit, so the second state goes to $|+\rangle$ state.

$$|\psi_1\rangle = \frac{\alpha_0}{\sqrt{2}}(|000\rangle + |010\rangle) + \frac{\alpha_1}{\sqrt{2}}(|100\rangle + |110\rangle)$$

CNOT gate is applied to gates 2 and 3 with 2 as control and 3 as target. So, 3 rd qubit flips if 2nd qubit is 1 else no change.

$$|\psi_2\rangle = \frac{\alpha_0}{\sqrt{2}}(|000\rangle + |011\rangle) + \frac{\alpha_1}{\sqrt{2}}(|100\rangle + |111\rangle)$$

CNOT gate is applied to gates 1 and 2 with 1 as control and 2 as target. So, 2 nd qubit flips if 1st qubit is 1 else no change.

$$|\psi_3\rangle = \frac{\alpha_0}{\sqrt{2}}(|000\rangle + |011\rangle) + \frac{\alpha_1}{\sqrt{2}}(|110\rangle + |101\rangle)$$

Hadamard gate is applied to second qubit, so the $|0\rangle$ state goes to $|+\rangle$ state and $|1\rangle$ state goes to $|-\rangle$ state.

$$|\psi_4\rangle = \frac{\alpha_0}{2}(|000\rangle + |100\rangle + |011\rangle + |111\rangle) + \frac{\alpha_1}{2}(|010\rangle - |110\rangle + |001\rangle - |101\rangle)$$

State of the three qubits at the end of the circuit's operation are

$$\boxed{\frac{\alpha_0}{2}(|000\rangle + |100\rangle + |011\rangle + |111\rangle) + \frac{\alpha_1}{2}(|010\rangle - |110\rangle + |001\rangle - |101\rangle)}$$

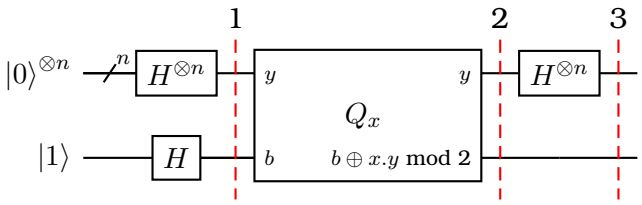
3.2 Probability of getting a state after the measurement

Probability of getting state $|\phi\rangle$ upon measuring state $|\psi\rangle$ is equal to $\|\langle\phi|\,|\psi\rangle\|^2$

State	Probability
$\|000\rangle$	$\frac{\alpha_0^2}{4}$
$\|001\rangle$	$\frac{\alpha_1^2}{4}$
$\|010\rangle$	$\frac{\alpha_1^2}{4}$
$\|011\rangle$	$\frac{\alpha_0^2}{4}$
$\|100\rangle$	$\frac{\alpha_0^2}{4}$
$\|101\rangle$	$\frac{\alpha_1^2}{4}$
$\|110\rangle$	$\frac{\alpha_1^2}{4}$
$\|111\rangle$	$\frac{\alpha_0^2}{4}$

4 Leveraging Parity

4.1 Bernstein-Vazirani Algorithm



4.2 Initial state

$$|\psi_0\rangle = |0\rangle^{\otimes n} |1\rangle$$

4.3 After application of Hadamard gates

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=\{0,1\}^n} |y\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

4.4 After passing through Oracle

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=\{0,1\}^n} (-1)^{x.y \bmod 2} |y\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=\{0,1\}^n} (-1)^{x.y} |y\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \left(\because (-1)^{x.y \bmod 2} = (-1)^{x.y} \right)$$

4.5 Hadamard gate on n-qubit register

we know,

$$\boxed{H^{\otimes n} |i\rangle = \sum_{j=\{0,1\}^n} (-1)^{i.j} |j\rangle}$$

If we apply Hadamard gates again on the register, since H is its own inverse,

$$H^{\otimes n} H^{\otimes n} |i\rangle = H^{\otimes n} \sum_{j=\{0,1\}^n} (-1)^{i.j} |j\rangle$$

$$\boxed{|i\rangle = H^{\otimes n} \sum_{j=\{0,1\}^n} (-1)^{i.j} |j\rangle}$$

Therefore,

$$\boxed{H^{\otimes n}\left|\psi_2\right\rangle = \left|x\right\rangle\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right)}$$

Now,

$$\left|\psi_3\right\rangle = \left|x\right\rangle\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right)$$

Since, we got $\left|x\right\rangle$, it is used to compute a fixed function f(x) which can be simulated using elementary gates.

5 Classical Comparison

$P(A|B)$ = Probability of occurrence of event A given event B occurred.
For a balanced function there are $\frac{N}{2}$ 0's and $\frac{N}{2}$ 1's.

Let **"B"** denote the event that given function is a balanced function.
For a Constant function with constant value = 0 there are N 0's and zero 1's.

Let "$\mathbf{C_0}$" denote the event that given function is a constant function with constant value 0.
For a Constant function with constant value = 1 there are zero 0's and N 1's.

Let "$\mathbf{C_1}$" denote the event that given function is a constant function with constant value 1.

Probability of the output to be 00 and the function is a Balanced function

$$\boxed{P(00|B) = \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)}} \because \text{there are } \frac{N}{2} \text{ zeroes in Balanced Function.}$$

Probability of the output to be 01 and the function is a Balanced function

$$\boxed{P(01|B) = \frac{\left(\frac{N}{2}\right)^2}{N(N-1)}} \because \text{there are } \frac{N}{2} \text{ zeroes and } \frac{N}{2} \text{ ones in Balanced Function.}$$

Probability of the output to be 10 and the function is a Balanced function

$$\boxed{P(10|B) = \frac{\left(\frac{N}{2}\right)^2}{N(N-1)}} \because \text{there are } \frac{N}{2} \text{ zeroes and } \frac{N}{2} \text{ ones in Balanced Function.}$$

Probability of the output to be 11 and the function is a Balanced function

$$\boxed{P(11|B) = \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)}} \because \text{there are } \frac{N}{2} \text{ ones in Balanced Function.}$$

Probability of the output to be 00 and the function is a Constant function with constant value = 0

$$\boxed{P(00|C_0) = \frac{N(N-1)}{N(N-1)} = 1} \because \text{there are N zeroes in Constant Function.}$$

Probability of the output to be 01 and the function is a Constant function with constant value = 0

$$\boxed{P(01|C_0) = 0} \because \text{there are N zeroes and 0 ones in Constant Function.}$$

Probability of the output to be 10 and the function is a Constant function with constant value = 0

$$\boxed{P(10|C_0) = 0} \because \text{there are N zeroes and 0 ones in Constant Function.}$$

Probability of the output to be 11 and the function is a Constant function with constant value = 0

$\boxed{P(11|C_0) = 0}$ $\because$ there are N zeroes and 0 ones in Constant Function.

Probability of the output to be 00 and the function is a Constant function with constant value = 1

$\boxed{P(00|C_1) = 0}$ $\because$ there are 0 zeroes and N ones in Constant Function.

Probability of the output to be 01 and the function is a Constant function with constant value = 1

$\boxed{P(01|C_1) = 0}$ $\because$ there are 0 zeroes and N ones in Constant Function.

Probability of the output to be 10 and the function is a Constant function with constant value = 1

$\boxed{P(10|C_1) = 0}$ $\because$ there are 0 zeroes and N ones in Constant Function.

Probability of the output to be 11 and the function is a Constant function with constant value = 1

$\boxed{P(11|C_1) = \frac{N(N-1)}{N(N-1)} = 1}$ $\because$ there are 0 zeroes N ones in Constant Function.

Let us define our algorithm as following.

	Function
00	Constant
01	Balanced
10	Balanced
11	Constant

Note: The actual function for 00 and 11 can actually be constant or balanced but the algorithm will output it as constant.

Summary of Outputs and Probabilities

Output bits	**Function by algorithm**	**Actual Function**	**Probability**	**Remarks**
00	Constant	Balanced	$P(00\|B) = \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)}$	Incorrect output
01	Balanced	Balanced	$P(01\|B) = \frac{\left(\frac{N}{2}\right)^2}{N(N-1)}$	correct output
10	Balanced	Balanced	$P(10\|B) = \frac{\left(\frac{N}{2}\right)^2}{N(N-1)}$	correct output
11	Constant	Balanced	$P(11\|B) = \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)}$	Incorrect output
00	Constant	Constant 0	$P(00\|C_0) = 1$	correct output
11	Constant	Constant 1	$P(11\|C_1) = 1$	correct output

We ignored the zero Probability cases.

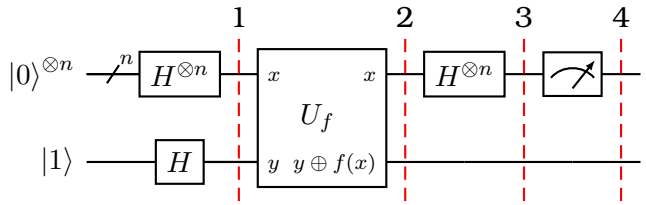
Probability of our algorithm being correct is Probability of correct output divided by Probability of all outputs.

$$P(correct) = \frac{P(01|B) + P(10|B) + P(00|C_0) + P(11|C_1)}{P(01|B) + P(10|B) + P(00|C_0) + P(11|C_1) + P(00|B) + P(11|B)}$$

$$P(correct) = \frac{\frac{\left(\frac{N}{2}\right)^2}{N(N-1)} + \frac{\left(\frac{N}{2}\right)^2}{N(N-1)} + 1 + 1}{\frac{\left(\frac{N}{2}\right)^2}{N(N-1)} + \frac{\left(\frac{N}{2}\right)^2}{N(N-1)} + 1 + 1 + \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)} + \frac{\frac{N}{2}\left(\frac{N}{2}-1\right)}{N(N-1)}}$$

$$\boxed{P(correct) = \frac{2}{3} + \frac{1}{6(1-\frac{1}{N})} \geq \frac{2}{3}}$$

6 Fun with Deutsch-Josza



6.1 Initial state

$$|\psi_0\rangle = |0\rangle^{\otimes n} |1\rangle$$

6.2 After application of Hadamard gates

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=\{0,1\}^n} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

6.3 After passing through Oracle

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=\{0,1\}^n} (-1)^{f(x)} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

6.4 After application of Hadamard gates

$$|\psi_3\rangle = \frac{1}{2^n} \sum_{x=\{0,1\}^n} \sum_{z=\{0,1\}^n} (-1)^{x.z+f(x)} |z\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

6.5 Measuring the First register

Let us consider the co-efficient of $|100\ldots0\rangle$ for two different cases.

$$x.z = x_1.z_1 + x_2.z_2 + \ldots + x_N.z_N$$
$$x.z = x_1.(1) + x_2.(0) + \ldots + x_N.0 = x_1$$
$$\frac{1}{2^n} \sum_{x=\{0,1\}^n} (-1)^{x_1+f(x)} |100\ldots0\rangle$$

6.5.1 Case 1: f(x) = 0 first N/2 bits and f(x) = 1 for Next N/2 bits.

since, $x_1 = 0$ for first $\frac{N}{2}$ bits $x_1 + f(x) = 0 + 0 = 0 \implies (-1)^{x_1+f(x)} = (-1)^0 = 1$ and $x_1 = 1$ for next $\frac{N}{2}$ bits $x_1 + f(x) = 1 + 1 = 2 \implies (-1)^{x_1+f(x)} = (-1)^2 = 1$

$$\text{Coefficient of } |100\ldots0\rangle = \frac{1}{2^n} \sum_{x=\{0,1\}^n} (1) = \frac{2^n}{2^n} = 1$$

$$\boxed{\text{Coefficient of } |100\ldots0\rangle = 1}$$

6.5.2 Case 2: Sum of 1's in the first N/2 bits and 0's in the next N/2 bits is equal to N/2.

Number of 1's in first half + Number of 0's in second half = N/2.
$\implies$ Number of 0's in second half = $\frac{N}{2}$ - Number of 1's in first half.
Also, Number of 0's in first half = $\frac{N}{2}$ - Number of 1's in first half.
Therefore, Number of 0's in first half = Number of 0's in second half.
And by similar arguments, Number of 1's in first half = Number of 1's in second half.
Let there be k number of zeroes in first half that implies there are k zeroes in second half and N/2 - k ones each in first and second halves.
For all terms in first half $x_1 = 0$ and for k terms in first half $f(x) = 0 \implies (-1)^{x_1+f(x)} = (-1)^0 = 1$
For all terms in first half $x_1 = 0$ and for N/2 - k terms in first half $f(x) = 1 \implies (-1)^{x_1+f(x)} = (-1)^1 = -1$.
For all terms in second half $x_1 = 1$ and for k terms in second half $f(x) = 0 \implies (-1)^{x_1+f(x)} = (-1)^1 = -1$.
For all terms in second half $x_1 = 1$ and for N/2 - k terms in second half $f(x) = 1 \implies (-1)^{x_1+f(x)} = (-1)^2 = 1$.
Summary

First or Second	No. of Terms	Value of $(-1)^{x_1+f(x)}$
First Half	k	1
First Half	N/2 - k	-1
Second Half	k	-1
Second Half	N/2 - k	1

Therefore,

$$\text{Coefficient of } |100\ldots0\rangle = \frac{1}{2^n} \sum_{x=\{0,1\}^n} (-1)^{x_1+f(x)}$$

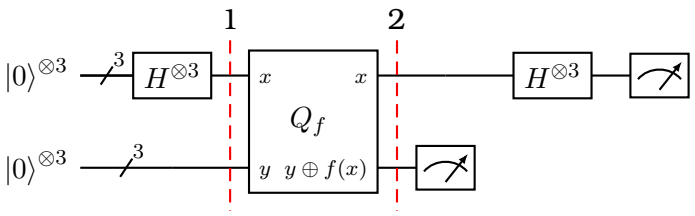
$$\text{Coefficient of } |100\ldots0\rangle = \frac{1}{2^n}\left((k)(1) + \left(\frac{N}{2}-k\right)(-1) + (k)(-1) + \left(\frac{N}{2}-k\right)(1)\right) = 0$$

$$\boxed{\text{Coefficient of } |100\ldots0\rangle = 0}$$

Case	Coefficient of $\lvert 100\ldots0\rangle$
Case 1	1
Case 2	0

If the measurement of the first register given state $|100\ldots0\rangle$ then the function $f(x)$ is case 1 and if it gives any state other than $|100\ldots0\rangle$ then $f(x)$ belongs to case 2.

7 Simon's Algorithm



7.1 Starting State

$$|\psi_0\rangle = |0\rangle^{\otimes 3} |0\rangle^{\otimes 3}$$

7.2 State after First Hadamard Transforms

$$|\psi_1\rangle = \frac{1}{\sqrt{2^3}} \sum_{x \in \{0,1\}^3} |x\rangle |0\rangle^{\otimes 3}$$

7.3 State after applying the oracle

$$|\psi_2\rangle = \frac{1}{\sqrt{2^3}} \sum_{x \in \{0,1\}^3} |x\rangle |f(x)\rangle$$

7.4 State after measuring the second register

The measurement gave $|001\rangle$

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}(|x\rangle + |x \oplus s\rangle)$$

where,

$$f(x) = f(x \oplus s) = 001$$

7.5 State after final Hadamard

$$|\psi_4\rangle = \frac{1}{\sqrt{2^7}} \sum_{z \in \{0,1\}^3} [(-1)^{x.z} + (-1)^{(x \oplus s).z}] |z\rangle$$

7.6 Measurement of first 3 qubits of final state

Measurement of first 3 qubits of final state give information about s because,
It will give output only if

$$(-1)^{x.z} = (-1)^{(x \oplus s).z}$$

which means:

$$\begin{aligned} x.z \bmod 2 &= (x \oplus s).z \bmod 2 \\ x.z \bmod 2 &= x.z \oplus s.z \bmod 2 \\ \implies s.z &= 0 \bmod 2 \end{aligned}$$

A string z will be measured, whose inner product with s = 0. Thus, repeating the algorithm $\approx$ n times, we will be able to obtain n different values of z and the following system of equation can be written:

$$\begin{cases} s.z_1 = 0 \bmod 2 \\ s.z_2 = 0 \bmod 2 \\ \vdots \\ s.z_n = 0 \bmod 2 \end{cases}$$

From which s can be determined, for example by Gaussian elimination.

7.7 Determining s

Let s = abc
$z_1 = 011$

$$s.z_1 = 0 \text{ mod } 2$$

$$b + c = 0 \text{ mod } 2$$

either, bc = 00 or bc = 11.
$z_2 = 101$

$$s.z_2 = 0 \text{ mod } 2$$

$$a + c = 0 \text{ mod } 2$$

either ac=00 or ac =11.
If c = 0 $\implies$ a = 0 and b = 0 then s = 000 but we know s $\neq$ 000 therefore, c = 1 $\implies$ a = 1, b = 1 $\implies$ s = 111.

8 Superdense Coding

Initial State,

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

If NOT gate is applied on Alice's qubit $|\phi\rangle$.

$$|\alpha\rangle = \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle)$$

If Z gate is applied on Alice's qubit $|\phi\rangle$.

$$|\beta\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$$

If both NOT gate and Z gate are applied on Alice's qubit $|\phi\rangle$

$$|\gamma\rangle = \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle)$$

Message bits	Gates Applied	Final state of qubits
00	No gates	$\frac{1}{\sqrt{2}}(\|00\rangle + \|11\rangle)$
01	X-gate	$\frac{1}{\sqrt{2}}(\|10\rangle + \|01\rangle)$
10	Z-gate	$\frac{1}{\sqrt{2}}(\|00\rangle - \|11\rangle)$
11	X-gate and Z-gate	$\frac{1}{\sqrt{2}}(\|10\rangle - \|01\rangle)$

The qubits $|\phi\rangle$, $|\alpha\rangle$, $|\beta\rangle$, $|\gamma\rangle$ are bell states which are a set of basis states. Therefore, if we measure the states in bell basis Bob can exactly determine Alice's message.

9 Deffered Measurements

9.1 Randomized Circuit C(x)

1. Random circuit takes input with n bits and a random number α with r bits.
2. r $COIN_{\frac{1}{2}}$ gates are applied to r bits to generate a random number.
3. The Randomized circuit is now fed wit the inputs and the circuit contains s CCNOT gates which apply on the input.
4. The circuit outputs m output bits.

9.2 Quantum Circuit C'(x)

1. Creating the corresponding reversible circuit with inputs n, α and a ancilla bits.
2. To randomize α, we initialize $|0\rangle$ to r bits and apply r Hadamard gates to them. Their state now becomes $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.
3. Apply r CNOT gates where above r qubits are control and other r qubits as target which make them entangled. $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
4. Since, they are entangled measuring the bits into which we copied each computational basis state of α is equivalent to measuring the bits of α itself.
5. Now, send the n bits and r bits into the circuit with s CCNOT gates arranged similar to the random circuit. This now gives the output m +r bits. The m bits are similar to m bits in the above randomized circuit output m bits because we used the same circuit.
6. In fact, though, it doesn't matter whether wer are measure the fresh qubits before or after running the quantum circuit. In fact, we can delay their measurement arbitrarily long, or just avoid it altogether. This is known as the "principle of deferred measurement". Measurement is equivalent to entanglement of the system with its environment.

10 Amplifying Success

P(C(x)=f(x)) = $\frac{2}{3}$, $P(C(x) \neq f(x)) = \frac{1}{3}$ Let us repeat the experiment m times. Probability of getting success l times = ${}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l}$ let us define our C'(x) to give the maximum of 0 and 1. Therefore, we need $l \geq \frac{m}{2}$.

$$P(C'(x) = f(x)) = \sum_{l=\frac{m}{2}}^{\infty} {}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l}$$

$$P(C'(x) = f(x)) = 1 - \sum_{l=0}^{\frac{m}{2}} {}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l}$$

Since,

$$P(C'(x) = f(x)) = 1 - P(C'(x) \neq f(x)) = 1 - \sum_{l=0}^{\frac{m}{2}} {}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l}$$

$${}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l} \leq {}^{m}C_l \left(\frac{2}{3}\right)^{\frac{m}{2}} \left(\frac{1}{3}\right)^{\frac{m}{2}}$$

we know, $\sum_{i=0}^{\frac{k}{2}} {}^{m}C_l(\frac{2}{3}) = 2^{m-1}$ Hence,

$${}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l} \leq \frac{1}{2}\left(\frac{8}{9}\right)^{\frac{m}{2}}$$

$$P(C'(x) = f(x)) = 1 - {}^{m}C_l \left(\frac{2}{3}\right)^l \left(\frac{1}{3}\right)^{m-l} \geq 1 - \frac{1}{2}\left(\frac{8}{9}\right)^{\frac{m}{2}}$$

If $n = \frac{m}{2}(\log_{\frac{9}{8}} 2 - 1) \implies P(C'(x) = f(x)) \geq 1 - 2^{-n}$

$$\boxed{P(C'(x) = f(x)) \geq 1 - 2^{-n}}$$