Quantum Computing

1. Quantum States and Quantum Gates
2. Multiple Qubits and Entangled States
3. Quantum Gate Arrays
4. Quantum Parallelism
5. Examples of Quantum Algorithms
   1. Grover’s Unstructured Search Algorithm
   2. Schor’s Factoring Algorithm
   3. Quantum Error Correcting Codes
6. Quantum Computer Architecture
7. Quantum Computers and Complexity
Quantum States

For quantum computing we need only deal with finite quantum systems, and it suffices to consider only finite dimensional complex vector spaces with inner product.

A quantum bit (qubit) is a unit vector in a one-dimensional complex vector space.

We will use Bra/Ket notation (invented by Dirac) to represent these unit vectors.

\[ |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \]
# Single-Qubit State Transformations

## Identity Transformation

|   | $|0\rangle$  | $|1\rangle$  |
|---|---|---|
| I: | $|0\rangle$  | $|0\rangle$  | 1 0 |
|   | $|1\rangle$  | $|1\rangle$  | 0 1 |

## Complement

|   | $|0\rangle$  | $|1\rangle$  |
|---|---|---|
| X: | $|0\rangle$  | $|1\rangle$  | 0 1 |
|   | $|1\rangle$  | $|0\rangle$  | 1 0 |

## Negative Complement

|   | $|0\rangle$  | $|1\rangle$  |
|---|---|---|
| Y: | $|0\rangle$  | -$|1\rangle$  | 0 1 |
|   | $|1\rangle$  | $|0\rangle$  | -1 0 |

## XY Product (phase transformation)

|   | $|0\rangle$  | -$|1\rangle$  |
|---|---|---|
| Z: | $|0\rangle$  | $|0\rangle$  | 1 0 |
|   | $|1\rangle$  | -$|1\rangle$  | 0 -1 |
Reversible Gates

Controlled Not

\[\text{CNOT: } |00\rangle \rightarrow |00\rangle, \quad |01\rangle \rightarrow |01\rangle, \quad |10\rangle \rightarrow |11\rangle, \quad |11\rangle \rightarrow |10\rangle\]

\[\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 \\
\end{pmatrix}\]

\(\text{CNOT is unitary}\) \quad \(\text{CNOT CNOT}^* = I\)
Reversible Gates

Controlled-Controlled-Not (Toffoli Gate)

The third bit is negated only if the first two are both 1

\[
\begin{align*}
\text{CCNOT: } & |00x> \rightarrow |00x> \quad \text{where } |x> = |0> \text{ or } |x> = |1> \\
& |01x> \rightarrow |01x> \\
& |10x> \rightarrow |10x> \\
& |11x> \rightarrow |11y> \quad \text{where } |y> = X|x> \text{ complement of qubit}
\end{align*}
\]
Reversible Gates

Controlled-Not and Toffoli gates are both reversible and the unitary transformations that they implement cannot be decomposed into a tensor product of single qubit transformations.

The Toffoli Gate can be used to construct a complete set of Boolean connectives.

Negation

\[ T|1,1,x> = |1,1,-x> \]

Conjunction

\[ T|x,y,0> = |x,y,x\land y> \]
Walsh Hadamard Transformation

The single-qubit Hadamard transformation

\[ H: \begin{array}{c|c}
|0\rangle & \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \\
|1\rangle & \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)
\end{array} \]

When applied to \( n \) bits individually, \( H \) generates a superposition of all \( 2^n \) possible states – which can represent numbers from 0 to \( 2^n - 1 \).

\[
(H \otimes H \otimes \ldots \otimes H) |00\ldots00\rangle \\
= \frac{1}{\sqrt{2^n}}((|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \ldots \otimes (|0\rangle + |1\rangle)) \\
= \frac{1}{\sqrt{2^n}} \sum |x\rangle \quad \text{where the summation is from } x = 0 \text{ to } x = 2^n - 1
\]
Bra/Ket Notation

Let $|0\rangle$ and $|1\rangle$ be any two basis vector (states) in the complex plane. And let $|x\rangle$ be any (unit) vector (state) in the plane. Then:

$$|x\rangle = a|0\rangle + b|1\rangle$$

$$\langle x| = |x\rangle^* = a^*\langle 0| + b^*\langle 1|$$

$$\langle x|x\rangle = 1 = (a^*\langle 0| + b^*\langle 1|)(a|0\rangle + b|1\rangle)$$

$$= a^*a \langle 0|0\rangle + a^*b\langle 0|1\rangle + b^*a\langle 1|0\rangle + b^*b \langle 1|1\rangle$$

$$= a^*a + b^*b = 1$$

Since $\langle 0|0\rangle = \langle 1|1\rangle = 1$ and $\langle 0|1\rangle = \langle 1|0\rangle = 0$
Bra/Ket Notation

Consider again the transformation $X$: 

\[
|0\rangle \rightarrow |1\rangle \\
|1\rangle \rightarrow |0\rangle 
\]

We can represent $X$ as

\[
X = |0\rangle\langle 1| + |1\rangle\langle 0| 
\]

$X|0\rangle = (|0\rangle\langle 1| + |1\rangle\langle 0|)|0\rangle = |0\rangle\langle 1|0\rangle + |1\rangle\langle 0|0\rangle = |1\rangle$

$X|1\rangle = (|0\rangle\langle 1| + |1\rangle\langle 0|)|1\rangle = |0\rangle\langle 1|1\rangle + |1\rangle\langle 0|1\rangle = |0\rangle$

All unitary transformations can be expressed in Bra/Ket notation.
Multiple Qubit States

Multiple Qubit states are formed from the tensor product of single Qubit states

\[ |x⟩ = a|0⟩ + b|1⟩ \]
\[ |y⟩ = c|0⟩ + d|1⟩ \]

\[ |x⟩ \otimes |y⟩ = (a|0⟩ + b|1⟩) \otimes (c|0⟩ + d|1⟩) \]

\[ = ac(|0⟩ \otimes |0⟩) + ad (|0⟩ \otimes |1⟩) + bc (|0⟩ \otimes |0⟩) + (bd(|1⟩ \otimes |1⟩) \]
\[ = ac|00⟩ + ad |01⟩ + bc|10⟩ + bd|11⟩ \]

The tensor product of an \( m \) dimension vector and a \( k \) dimension vector is an \( mk \) dimension vector
Entangled States

Consider the two 2-qubit states:

\[ |\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad \text{and} \quad |\varphi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle) \]

\[ |\varphi\rangle \text{ is composed by the tensor product } |0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \]

Measurement of the second bit will yield \(|0\rangle\) or \(|1\rangle\) with a probability of \(\frac{1}{2}\) for each result, independently of whether the first bit has been measured or not. Measurement of the first bit will always yield \(|0\rangle\).

\[ |\psi\rangle \text{ cannot be decomposed into a tensor product of two single qubits} \]

It is an entangled state.

The first measurement completely determines the outcome of the second.
Cloning

Quantum states cannot be copied or cloned!

Proof

Assume a unitary transformation $U$ such that $U|a0> = |aa>$

Let $|a>$ and $|b>$ be orthogonal states, and

$U|a0> = |aa>$ and $U|b0> = |bb>$

Now consider $|c> = \frac{1}{\sqrt{2}}(|a> + |b>)$

By linearity $U|c0> = \frac{1}{\sqrt{2}}(U|a0> + U|b0>) = \frac{1}{\sqrt{2}}(|aa> + |bb>)$

But if $U$ is a cloning transformation

$U|c0> = |cc> = \frac{1}{\sqrt{2}}(|a> + |b>) \otimes \frac{1}{\sqrt{2}}(|a> + |b>)$

$= \frac{1}{2}(|aa> + |ab> + |ba> + bb>)$ Contradiction!!
Quantum Gate Arrays

1-bit full adder

Let $|c\rangle = |1\rangle$, $|x\rangle = |0\rangle$, $|y\rangle = |1\rangle$, $|s\rangle = |0\rangle$, $|c'\rangle = |1\rangle$
Quantum Gate Arrays

It is possible to construct reversible quantum gates for any classical computable function $f$ with $m$ input and $k$ output bits.

There exists a quantum gate array that implements the unitary transformation $U_f : |x, y> \rightarrow |x, y \oplus f(x)>, \text{ where } \oplus \text{ indicates bitwise xor.}$
Quantum Gate Arrays

The previous transformation $U_f$ is reversible

$$U_f^+ = U_f \quad \quad U_f^+ U_f = U_f U_f = I$$

But $|y \oplus f(x) \oplus f(x)\rangle = |y\rangle$
Superposition of Quantum States

Consider the Tofolli Gate

\[ |x\rangle \quad \quad \quad \quad |x\rangle \]
\[ |y\rangle \quad \quad \quad \quad |y\rangle \]
\[ |0\rangle \quad \quad \quad \quad |x\rangle \land |y\rangle \]

Apply \( T \), the Tofolli transform, to the superposition of all inputs.

\( T(H|0\rangle \otimes H|0\rangle \otimes |0\rangle) = \frac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |111\rangle) \)

Quantum parallelism – Applying the Tofolli transform to a superposition of all of the input states produces a superposition of all of the states in the “truth table”
Superposition of Quantum States

**BUT** Only one of the superposed states can be extracted by measurement

\[ T(H|0> \otimes H|0> \otimes |0> ) = \frac{1}{2}(|000> + |010> + |100> + |111>) \]

Measurement of the output projects the superposition onto the set of states consistent with the result
Quantum Parallelism

In order to take advantage of quantum parallelism one must:

1. Transform the state in such a way as to amplify the values of interest – so that they have a higher probability of being selected during measurement

   Grover’s unstructured search algorithm

2. Find common properties of ALL the states of \( f(x) \)

   Shor’s factoring algorithm
Grover’s Unstructured Search

Given an unstructured list of size N and a proposition P, find an x in the list such that P(x) is true.

Let $U_P$ be the Quantum gate that implements the classical function $P(x)$, and let $n$ be such that $2^n \geq N$.

$$U_P : |x,0> \rightarrow |x,P(x)>$$

$U_P$ operating on a superposition of all input states yields:

$$\frac{1}{\sqrt{2^n}} \sum_{i=0}^{N-1} |x,P(x)>$$

If there is a single state where $P(x_i) = 1$ (true), the probability of reading that state in a measurement is just $\frac{1}{\sqrt{2^n}}$

We need to increase the probability of selecting this state.
Grover’s Search Algorithm

1. Prepare a register containing the superposition of all values 0…2^n-1

2. Compute P(x_i) on this register.

3. Change the amplitude a_i to -a_i for x_i such that P(x_i) = 1

4. Apply inversion about the average to increase the amplitude of x_i with P(x_i) = 1.

5. Repeat steps 2 through 4 \(\pi/4(\sqrt{2^n})\) times.

6. Read the result
Step 3. Changing the sign

Let $U_P$ be the unitary transformation that implements

$$U_P |x, b> = |x, b \oplus P(x)>$$

Apply $U_P$ to the superposition $|\Psi> = (1/\sqrt{2^n}) \sum_{x=0..n-1} |x>$ and

Choose $b = (1/\sqrt{2})(|0> - |1>)$

Let $X_0 = \{x | P(x) = 0\}$, and $X_1 = \{x | P(x) = 1\}$ then

$$U_P |\Psi, b> = (1/\sqrt{2^n}) U_P \left( \sum_{x \in X_0} |x, b> + \sum_{x \in X_1} |x, b> \right)$$
Grover’s Quantum Search Algorithm

\[ U_P |\Psi, b> = (1/\sqrt{2^n}) U_P (\sum |x, b> + \sum |x, b>) \]
\[ \quad \quad \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \]

\[ = (1/\sqrt{2^{n+1}}) U_P (\sum |x, 0> + \sum |x, 0> - \sum |x, 1> - \sum |x, 1>) \]
\[ \quad \quad \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \]

\[ = (1/\sqrt{2^{n+1}}) (\sum |x, 0 \oplus 0> + \sum |x, 0 \oplus 1> - \sum |x, 1 \oplus 0> - \sum |x, 1 \oplus 1>) \]
\[ \quad \quad \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \]

\[ = (1/\sqrt{2^{n+1}}) (\sum |x, 0 > + \sum |x, 1> - \sum |x, 1 > - \sum |x, 0>) \]
\[ \quad \quad \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \]

\[ = (1/\sqrt{2^n}) (\sum |x > - \sum |x>) \otimes |b> \]
\[ \quad \quad \quad \quad \quad x \in X_0 \quad \quad \quad x \in X_1 \]

The amplitude of the states in \( X_1 \) have been inverted
Grover’s Quantum Search Algorithm

Step 4. Apply inversion about the average

\[
\sum_{i=0}^{N-1} a_i |x_i\rangle \rightarrow \sum_{i=0}^{N-1} (2A - a_i) |x_i\rangle
\]

Where \( A \) denotes the average of the \( a_i \)'s

This inversion can be performed by the \( N \times N \) matrix

\[
D = \begin{pmatrix}
(2/N) - 1 & 2/N & 2/N & \ldots & 2/N \\
2/N & (2/N) - 1 & 2/N & \ldots & 2/N \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
2/N & 2/N & 2/N & \ldots & 2/N
\end{pmatrix}
\]
Grover’s Quantum Search Algorithm

The transformation $D$ is unitary ($D^*D = I$) and can be decomposed into $O(n) = O(\log N)$ quantum gates.

$$D = WRW$$

where $W$ is the Walsh-Hadamard transformation and

$$W = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 0 & -1 & 0 & \ldots & 0 \\ 0 & 0 & -1 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & -1 \end{pmatrix}$$

$$R = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 0 & -1 & 0 & \ldots & 0 \\ 0 & 0 & -1 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & -1 \end{pmatrix}$$
Grover’s Quantum Search Algorithm

Consider $R = R' - I$, where $I$ is the identity matrix and $R'$ is

$$R' = \begin{pmatrix}
2 & 0 & 0 & \ldots & 0 \\
0 & 0 & 0 & \ldots & 0 \\
0 &           &           & \ldots & 0 \\
0 &           &           & \ldots & 2/N
\end{pmatrix}$$

Then $WRW = W(R' - I)W$, and it is easily shown that

$$WR'W = \begin{pmatrix}
2/N & 2/N & 2/N & \ldots & 2/N \\
2/N & 2/N & 2/N & \ldots & 2/N \\
2/N &           &           & \ldots & 2/N \\
2/N &           &           & \ldots & 2/N
\end{pmatrix}$$

And $WR'W - I = D$
Schor’s Factoring Algorithm

Factor an integer M

Step 1 – choose arbitrarily an integer \( a \)

if \( a \) is not relatively prime to \( M \) we have found a factor
else apply the rest of the algorithm.

Step 2 – choose \( m \) such that \( M \leq 2^m < M^2 \)

Use quantum parallelism to compute \( f(x) = a^x \mod M \)
for all integers from 0 to \( 2^m - 1 \)

The function is encoded in the quantum state

\[
\frac{1}{\sqrt{2^m}} \sum |x, f(x)\rangle \quad \text{where } x \text{ ranges from } 0 \text{ to } 2^m - 1
\]
Step 2 -- details

Repeatedly square $a \pmod{M}$ {classically} to get $a^{2^i}$ for $i \leq m$

This requires $O(m)$ squarings and multiplications of $m$-bit numbers (mod $M$)

This gives way to a gate array that uses $O(m \log(m) \log\log (m))$ gates to multiply two $m$-bit numbers (using Schönhage-Strassen)

The multiplication algorithm then becomes:

\[
\begin{align*}
  \text{f(x)} & := 1 \\
  \text{for } i = 0 \text{ to } m - 1 & \\
  & \quad \text{if } (x_i == 1) \text{ then} \\
  & \quad \quad \text{f(x)} := \text{f(x)} \ast a^{2^i} \pmod{M} \quad \text{//subroutine 2} \\
  & \quad \text{endif} \\
  \text{endfor}
\end{align*}
\]
Step 2 – details (cont.)

Subroutine 2

let $c = a^i \pmod{M}$, and $b = \text{power}_i$, then

result := 0

for $j = 0$ to $m - 1$

if ($b_i == 1$) then

result := result + $2^ic \pmod{M}$

endif

endfor

Where $2^ic$ is precomputed and built into the gate structure. The inverse of this operation is then used to erase $b$. 
Schor’s Factoring Algorithm

Step 3 – Construct a state that has the same period as \( f \)

Measure the last \( \lceil \lg M \rceil \) bits of \( (1/\sqrt{2^m}) \sum |x,f(x)> \)

The measurement projects the state space onto the subspace that is compatible with the measured value.

The state after measurement is:

\[
C \sum_x g(x)|x,u>
\]

where \( g(x) = \begin{cases} 
1 & \text{if } f(x) = u \\
0 & \text{otherwise}
\end{cases} \)

and \( C \) is a complex scale factor
Schor’s Factoring Algorithm

\[ |x\rangle \quad \rightarrow \quad A|x, 0\rangle \rightarrow |x, f(x)\rangle \]

where

\[ f(x) = a^x \mod M \]

Measure last \( \log M \) bits

State \( |x, f(x)\rangle \) is collapsed to \( g(x) |x, u\rangle \) where

\( u \) is the measured state, and \( g(x) = 1 \) if \( f(x) = u \), else 0.
Schor’s Factoring Algorithm

The above graph displays the values of x consistent with the measurement on $|f(x)>$

**If only we could measure the separation between adjacent values, our problem would be solved!**

**But alas, quantum mechanics only allows a single measurement!!**

We must find a way of determining the periodicity of these states from a single measurement of the first $m$ bits of our register.
Schor’s Factoring **Algorithm**

The Fourier transform of this superposition of states that give the measured \( u \) as a value of \( f(x) \) maps into states in a frequency domain.

\[
\begin{align*}
|x> & \quad G(c)|c> \\
A|x,0> & \quad U_{QFT} \\
g(x)|x> & \quad \text{Measurement of } f(x) \\
G(c)|c> & \quad \text{Measurement of transformed states}
\end{align*}
\]
Schor’s Factoring Algorithm

Step 4 – Perform a quantum Fourier transform on the resulting state

\[ U_{QFT} : \sum_x g(x)|x,u> \rightarrow \sum_x G(c)|c,u> \]

When the period \( r \) of the function \( g(x) \) is a power of 2, the resulting Fourier transform is

\[ \sum_j c_j |j(2^m/r),u> \]

where the amplitude is 0 except at multiples of \( 2^m /r \). When the period \( r \) does not divide \( 2^m \), the transform approximates the exact case and most of the amplitude is attached to integers close to \( 2^m /r \)
Schor’s Factoring Algorithm

Step 4 – details

Let \( S_{j,k} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & e^{i\theta_{k-j}}
\end{bmatrix} \)

where \( \theta_{k-j} = \pi/2^{k-j} \)

The quantum Fourier transform is then given by

\[
H_{m-1}S_{m-2,m-1}H_{m-2}S_{m-3,m-1}S_{m-3,m-2}H_{m-3} \cdots H_1S_{0,m-1}S_{0,m-2} \cdots S_{0,1}H_0
\]

Where the matrices are applied in reverse order (from left to right) and the transformed state is the bit-reversed state of state \(|c\rangle\)

The quantum Fourier transform requires \( m(m-1)/2 \) gates.
Schor’s Factoring Algorithm

Step 5 – Extract the period

Measure the state, and call the result $v$

$$v = j(2^m/r)$$

In general $j$ and $r$ will be relatively prime and the period is not a power of 2. A continued fraction expansion (on a classical computer) will produce an integer value $q$, that is likely to be the period.

Step 6 – Find a factor of $M$

If $q$ is even, use the Euclidean algorithm to check whether $(a^{q/2} + 1)$ or $(a^{q/2} - 1)$ has a non-trivial factor with $M$
Schor’s Factoring Algorithm

The reason why \((a^{q/2} + 1)\) or \((a^{q/2} - 1)\) is likely to have a non-trivial common factor with \(M\) is as follows:

If \(q\) is the period of \(f(x) = a^x \mod M\) then

\[a^q = 1 \mod M\] since

\[a^q a^x = a^x \mod M\] for all \(x\).

If \(q\) is even we can write

\[(a^{q/2} + 1) (a^{q/2} - 1) = 0 \mod M\]

thus, if neither of these factors is a multiple of \(M\), either has a non-trivial factor with \(M\). (found by using Euclid’s algorithm on a classical computer)
Schor’s Factoring Algorithm

Step 7 – Repeat the algorithm, if necessary

Reasons for having to repeat the algorithm

1. The value of \( v \) was not close enough to a multiple of \( 2^m/r \)
2. The period \( r \) and the multiplier \( j \) have a common factor – so that the denominator \( q \) was a factor of the period and not the period itself
3. Step 6 yields \( M \) as \( M \)’s factor
4. The period of \( f(x) = a^x \mod M \) is odd

Schor shows (claims) that this procedure, when applied to a random \( a \) (mod \( M \)) yields a non-trivial factor of \( M \) with probability at least \( 1 - 1/2^{k-1} \) where \( k \) is the number of distinct odd prime factors of \( M \).
Schor’s Factoring Algorithm

Example

Let $M = 21$

Step 1 – choose $a = 11$ which is relatively prime to $M$

Step 2 – let $M^2 \leq 2^m < 2M^2$  

$441 \leq 2^m < 882$ then $m = 9$

find $a^x \mod M$ on the superposition of states  $x = 0, \ldots, 2^m - 1$

Step 3 – Measure the last $\lceil \log M \rceil$ qubits of the state

$(1/\sqrt{2^m}) \sum_x |x, a^x \mod M>$

$\rightarrow C \sum_x g(x)|x, u>$

Step 4 – Apply Fourier transform to collapsed state

$U_{QFT} : \sum_x g(x)|x, u> \rightarrow \sum G(c)|c>$
Schor’s Factoring Algorithm

Step 5. Measure the transformed state

Assume measurement gives $v = 427$

Since $v$ and $2^m$ are relatively prime, the period $r$ will probably not divide $2^m$, and continued fraction expansion of $v/2^m$ yields $q = 6$. (This is probably the period of $f$)

Step 6. Since $q$ is even, $a^{q/2} + 1$ or $a^{q/2} - 1$ is likely to have a non-trivial common factor with $M$. (On a classical computer)

$a = 11$, $q/2 = 3$, $M = 21$

$(11^3 + 1) = 1332$  
$(11^3 - 1) = 1330$

$\gcd(1332, 21) = 7$  
$\gcd(1330, 21) = 3$
Quantum Error Correction

One problem in constructing quantum computers is the problem of decoherence – the interaction of qubits with the external environment causing them to transform in an unintended (and non-unitary) fashion.

Steane estimates the decoherence of any system likely to be built to be $10^7$ times too large to run Shor’s algorithm on a 130 digit number.*

Characterization of Errors

$$|\Psi\rangle \rightarrow (e_1 I + e_2 X + e_3 Y + e_4 Z)|\Psi\rangle = \sum_i e_i E_i |\Psi\rangle$$

The possible errors for each single qubit consist of no errors (I), bit-flip errors (X), phase errors (Z), and bit-flip phase errors (Y).

In the general case, possible errors are expressed as linear combinations of unitary error operators $E_i$.

Quantum Error Correction

Given – an error correcting code, $C$, with a syndrome extraction operator $S_C$—

An $n$-bit quantum state $|\psi\rangle$ is encoded in an $n + k$ bit quantum state $|\varphi\rangle$.

$$|\varphi\rangle = C|\psi\rangle$$

Assume decoherence leads to an error state $\sum_i e_i E_i |\varphi\rangle$

The signal encoded state $|\varphi\rangle$ can be recovered as follows:

1. Apply the syndrome extraction operator $S_C$ to the quantum state (padded with sufficient 0 bits.)

$$S_C(\sum_i e_i E_i |\varphi\rangle) \otimes |0\rangle = \sum_i e_i E_i |\varphi\rangle \otimes |i\rangle$$

2. Measure the $|i\rangle$ component of the result. This yields some random value $i_0$ and projects the state to $E_{i_0} |\varphi, i_0\rangle$

3. Apply the inverse error $E_{i_0}^{-1}$ to the first $n + k$ qubits of $E_{i_0} |\varphi, i_0\rangle$ to get the corrected state $|\varphi\rangle$
Quantum Error Correction

Example

Consider an error correcting code $C$ that maps

$|0\rangle \rightarrow |000\rangle$, and

$|1\rangle \rightarrow |111\rangle$

$C$ can correct single bit-flip errors

$E = \{I \otimes I \otimes I, X \otimes I \otimes I, I \otimes X \otimes I, I \otimes I \otimes X\}$

The syndrome extraction operator is:

$S: |x_0, x_1, x_2, 0, 0, 0\rangle \rightarrow |x_0, x_1, x_2, x_0 \text{ xor } x_1, x_0 \text{ xor } x_2, x_1 \text{ xor } x_2\rangle$
Quantum Error Correction

Consider the qubit

\[ |\psi> = \frac{1}{\sqrt{2}}(|0> - |1>) \]

\[ C|\psi> = |\varphi> = \frac{1}{\sqrt{2}}(|000> - |111>) \]

And the error is (two single bit-flip errors)

\[ E = (\frac{4}{5}) X \otimes I \otimes I + (\frac{3}{5}) I \otimes X \otimes I \]

The resulting error state is

\[ E |\varphi> = (\frac{4}{5} X \otimes I \otimes I + 3/5 I \otimes X \otimes I ) (1/\sqrt{2}) (|000> - |111>) \]
Quantum Error Correction

continuing

\[ E |\varphi> = \left(\frac{4}{5}\sqrt{2}\right)(|100> - |011>) + \left(\frac{3}{5}\sqrt{2}\right)(|010> - |101>) \]

Now apply the syndrome extraction operator to

\[ S_c E |\varphi> \otimes |000> \]

\[ = S_c (\left(\frac{4}{5}\sqrt{2}\right)(|100000> - |011000>) + \left(\frac{3}{5}\sqrt{2}\right)(|010000> - |101000>)) \]

\[ = (\frac{4}{5}\sqrt{2})(|100110> - |011110>) + (\frac{3}{5}\sqrt{2})(|010101> - |101101>) \]

\[ = (\frac{4}{5}\sqrt{2})(|100> - |011>) \otimes |110> + (\frac{3}{5}\sqrt{2})(|010> - |101>) \otimes |101> \]
Quantum Error Correction

Measurement of the last three bits yields either state

\[ |110\rangle \quad \text{or} \quad |101\rangle \]

Assume measurement yields \( |110\rangle \)

then the state collapses to

\[ \frac{1}{\sqrt{2}}(|100\rangle - |011\rangle) \otimes |110\rangle \]

The error can now be removed by applying \( E_i^{-1} = X \otimes I \otimes I \) to the first three bits of the state.

\[ E_i^{-1} \left( \frac{1}{\sqrt{2}}(|100\rangle - |011\rangle) \right) = \frac{1}{\sqrt{2}}(|000\rangle - |111\rangle) \]
The Architecture of a Quantum Computer

Preliminaries

Define an operator \( q^\dagger = |1><0| \)

\[ q^\dagger|0\rangle = |1><0| 0\rangle = |1\rangle \]
\[ q^\dagger|1\rangle = |1><0| 1\rangle = 0 \]

Similarly, \( q \), the complex conjugate of \( q^\dagger \), is defined

\[ q = |0><1| \]

\[ q|0\rangle = |0><1| 0\rangle = 0 \]
\[ q|1\rangle = |0><1| 1\rangle = |0\rangle \]

Note that

\[ q^\dagger + q = X \quad \text{and} \quad q^\dagger q + q q^\dagger = I \]
The Architecture of Quantum Computers

Time evolution of a quantum system

Let $|\Psi_{in}\rangle$ be the initial state of a quantum system, then the time evolution of this system is given by the equation

$$|\Psi_{out}\rangle = e^{iHt} |\Psi_{in}\rangle$$

where $H$ is the time evolution operator (Hamiltonian) and $e^{iHt}$ can be expanded as

$$e^{iHt} = 1 + iHt - H^2t^2/2! + \ldots.$$

The operator $H$ is operating an innumerable number of times, and our output state is the superposition of all these possibilities.
The Architecture for a Quantum Computer

The basic Computer Architecture

Program counter

Memory

logic

program
The Architecture of a Quantum Computer

\[ H = \sum q^\dagger_{i+1} q_i A_{i+1} + \text{c.c} \]

where summation is from \( i = 0 \) to \( i = k-1 \)

Qubit registers

PC

Are we there yet?
References


