Today's outline - February 16, 2023

- Mermin's interpretation of parallelism
- Simon's algorithm
- Distributed computation
- The Fourier transform

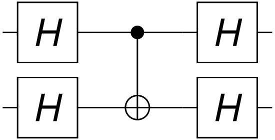
Reading Assignment: Reiffel: 7.8, 8.1–8.2 Wong: 7.6–7.7

Homework Assignment #04:
due Friday, February 17, 2023

Homework Assignment #05:
due Thursday, March 02, 2023

Mermin's interpretation

David Mermin proposed a simpler interpretation for how quantum algorithms and the solution to the Bernstein-Vazirani problem, in particular



Consider a C_{not} acting on the Hadamard basis

$$C_{not}|++\rangle = C_{not}\tfrac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \tfrac{1}{2}(|00\rangle + |01\rangle + |11\rangle + |10\rangle) = |++\rangle$$
$$C_{not}|+-\rangle = C_{not}\tfrac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle) = \tfrac{1}{2}(|00\rangle - |01\rangle + |11\rangle - |10\rangle) = |--\rangle$$
$$C_{not}|-+\rangle = C_{not}\tfrac{1}{2}(|00\rangle + |01\rangle - |10\rangle - |11\rangle) = \tfrac{1}{2}(|00\rangle + |01\rangle - |11\rangle - |10\rangle) = |-+\rangle$$
$$C_{not}|--\rangle = C_{not}\tfrac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle) = \tfrac{1}{2}(|00\rangle - |01\rangle - |11\rangle + |10\rangle) = |+-\rangle$$

If we then apply the Hadamard transform to each bit the resulting truth table becomes

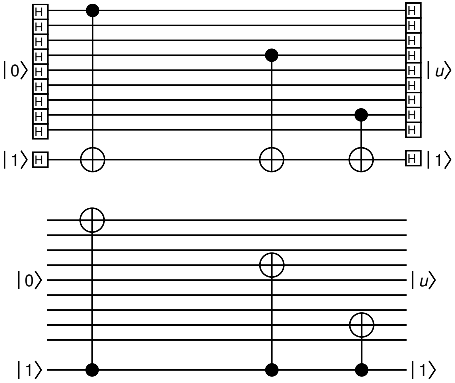
This is simply a C_{not} gate applied to the high order qubit controlled by the low order qubit

Initial			Final	
0	0	⟶	0	0
0	1	⟶	1	1
1	0	⟶	1	0
1	1	⟶	0	1

Mermin's interpretation

This insight leads to a simple way to look at the black box for U_{f_u}

1. Prepare an n-qubit register $|0\rangle_n$
2. Prepare an ancilla qubit $|a\rangle = |1\rangle$
3. Apply the Hadamard gate to all qubits
4. Place a $C_{not}|u_i\rangle|a\rangle$ for each $u_i = 1$
5. Apply the Hadamard gate to all qubits



The net effect is to have the ancilla bit "turn on" each qubit in the unknown, $C_{not}|a\rangle|u_i\rangle$ where $u_i = 1$

From this perspective there is no quantum parallelism but simply a discrete circuit which produces the desired outcome

Of course, this presupposes that one knows what $|u\rangle$ is so we are peering into the black box

Simon's problem – description

Suppose we have a 2-to-1 function $f(x)$ such that $f(x) = f(x \oplus a)$ where a is secret and both x and a are n bit strings

For example, when $n = 3$ we might have the table

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

There are 4 values for $f(x)$, each appearing twice, once in the top half of the table and once in the bottom

The goal of the algorithm is to find the the secret string a

Classically, this can be done by querying the function until we obtain two identical values for $f(x)$ and then calculate $a = x_0 \oplus x_1$

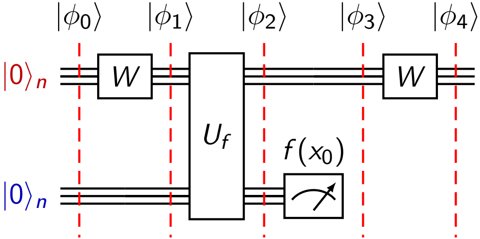
This can take up to $2^{n-1} + 1$ queries so the computation is $O(2^n)$

In contrast, Simon's quantum algorithm is a calculation which is $O(n)$

In this case, we can see that $a = 010 \oplus 111 = 101$ and this holds for all matched pairs in the table

Simon's algorithm – quantum circuit

The problem requires two registers of n bits each which we designate with $|0\rangle_n$ and $|0\rangle_n$ as input and output registers, respectively



$$|\phi_0\rangle = |0\rangle_n|0\rangle_n$$

$$|\phi_1\rangle = W \otimes I(|0\rangle_n|0\rangle_n) = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle|0\rangle_n$$

$$|\phi_2\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle|f(x)\rangle$$

$$|\phi_3\rangle = \frac{1}{\sqrt{2}}\left(|x_0\rangle + |x_0 \oplus a\rangle\right)|f(x_0)\rangle$$

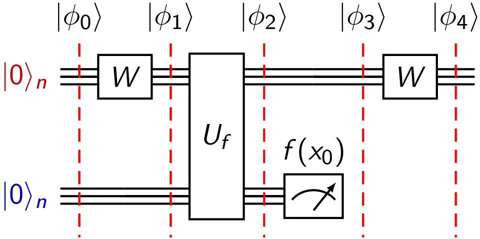
$$|\phi_4\rangle = W \otimes I\left[\frac{1}{\sqrt{2}}\left(|x_0\rangle + |x_0 \oplus a\rangle\right)|f(x_0)\rangle\right] = \frac{1}{\sqrt{2^n}}\frac{1}{\sqrt{2}}\sum_{y=0}^{2^n-1}\left[(-1)^{x_0\cdot y} + (-1)^{(x_0\oplus a)\cdot y}\right]|y\rangle|f(x_0)\rangle$$

$$= \frac{1}{\sqrt{2^{n+1}}}\sum_{y=0}^{2^n-1}(-1)^{x_0\cdot y}\left[1 + (-1)^{a\cdot y}\right]|y\rangle|f(x_0)\rangle$$

Simon's algorithm – quantum circuit

Dropping the $|f(x_0)\rangle$ as it has already been measured, we have

$$|\phi_4\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} (-1)^{x_0 \cdot y} \left[1 + (-1)^{a \cdot y}\right] |y\rangle$$



There are two cases to consider for the modulo 2 scalar product $a \cdot y$

$$y \cdot a \neq 0 \quad \longrightarrow \quad |\phi_4\rangle \equiv 0$$

The second case is for $a \cdot y = 0$, in which case

$$|\phi_4\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} (-1)^{x_0 \cdot y} [1 + 1] |y\rangle = \frac{1}{\sqrt{2^{n-1}}} \sum_{y=0}^{2^n-1} (-1)^{x_0 \cdot y} |y\rangle$$

This is a superposition of 2^n possible states, one of which will be observed when $|\phi_4\rangle$ is measured

If $n-1$ linearly independent $|y\rangle$ are measured, it is possible to solve $y \cdot a = 0$

Simon's algorithm – example

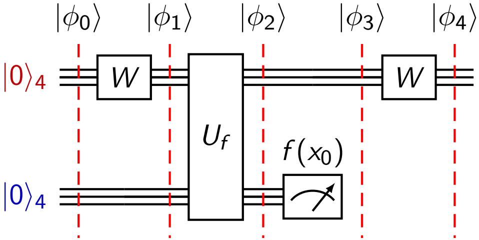
Suppose a system with $n = 4$ and $a = 1001$, $f(x)$ has the truth table

x	$f(x)$
0000	1111
0001	0001
0010	1110
0011	1101
0100	0000
0101	0101
0110	1010
0111	1001
1000	0001
1001	1111
1010	1101
1011	1110
1100	0101
1101	0000
1110	1001
1111	1010

$$|\phi_0\rangle = |0\rangle|0\rangle = |0000\rangle|0000\rangle$$

$$|\phi_1\rangle = \frac{1}{4}\sum_{x=0}^{15}|x\rangle|0000\rangle$$

$$|\phi_2\rangle = \frac{1}{4}\sum_{x=0}^{15}|x\rangle|f(x)\rangle$$



$$|\phi_3\rangle = \frac{1}{\sqrt{2}}\left[|x_0\rangle + |x_0 \oplus a\rangle\right]|f(x_0)\rangle$$

For example, suppose $f(x_0) = 1010$

$$|\phi_3\rangle = \frac{\left[|0110\rangle + |1111\rangle\right]}{\sqrt{2}}|f(x_0)\rangle$$

now apply the Walsh transformation

$$|\phi_4\rangle = \frac{\left[|0000\rangle - |0010\rangle - |0100\rangle + |0110\rangle + |1001\rangle - |1011\rangle - |1101\rangle + |1111\rangle\right]}{\sqrt{8}}$$

Note that any value of $|f(x_0)\rangle$ measured will result in these 8 $|x_0\rangle$

Simon's algorithm – example

$$|\phi_4\rangle = \frac{1}{\sqrt{8}}\left[|0000\rangle - |0010\rangle - |0100\rangle + |0110\rangle + |1001\rangle - |1011\rangle - |1101\rangle + |1111\rangle\right]$$

The result of the final measurement, $|y\rangle$ will be one of these eight values and each of them should satisfy the linear equation $a \cdot y = 0$

Since we know that $a = |1001\rangle$ for this example, we can check this identity

and the other 6 have the same properties

$$|1001\rangle\cdot|0000\rangle = 1\cdot0 + 0\cdot0 + 0\cdot0 + 1\cdot0 = 0$$

$$|1001\rangle\cdot|1001\rangle = 1\cdot1 + 0\cdot0 + 0\cdot0 + 1\cdot1 = 2 = 0$$

It is now necessary to collect $n - 1 = 3$ independent values of $|y\rangle$ to solve for a

Trial	$\|y\rangle$	Indep.?
1	$\|0000\rangle$	No
1	$\|0010\rangle$	Yes
1	$\|0100\rangle$	Yes
1	$\|0110\rangle$	No
1	$\|1001\rangle$	Yes

Create a matrix from the $y \cdot a = 0$ equation and the three independent values obtained

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Simon's algorithm – example

Solve this matrix equation by Gaussian elimination

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Convert the matrix to an upper triangular form by swapping rows 1 and 3

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Since the bottom row of the matrix is all zeros, a_0 can be either 0 or 1

$$a_0 = 0$$

$$a_1 = 0, \quad a_2 = 0, \quad a_3 + a_0 = 0$$

$$a_3 = 0 \quad \longrightarrow \quad a = |0000\rangle$$

trivial, incorrect solution

$$a_0 = 1$$

$$a_1 = 0, \quad a_2 = 0, \quad a_3 + a_0 = 0$$

$$a_3 = -1 = 1 \quad \longrightarrow \quad a = |1001\rangle$$

correct solution

Distributed computation

Alice and Bob are each provided with an $N = 2^n$ bit number, u and v respectively

Alice must compute an n-bit number a and Bob must compute an n-bit number b such that

$$\begin{array}{ccl} d_H(u,v) = 0 & \longrightarrow & a = b \\ d_H(u,v) = N/2 & \longrightarrow & a \neq b \\ \text{else} & \longrightarrow & \text{no condition on } a \text{ and } b \end{array}$$

This is a challenging problem because u and v are exponentially larger than a and b

A classical solution requires a communication of at least $N/2$ bits but with enough entangled pairs, no additional communication is needed in a quantum solution

Start with n entangled pairs of particles, (a_i, b_i) in states $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ with

$$a_0, a_1, \ldots, a_{n-1}, b_0, b_1, \ldots, b_{n-1} \longrightarrow |\psi\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i, i\rangle$$

Distributed computation

Alice uses the phase change subroutine with $f(i) = u_i$

$$\sum_{i=0}^{N-1} |i\rangle \longrightarrow \sum_{i=0}^{N-1} (-1)^{u_i} |i\rangle$$

Bob uses the phase change subroutine with $f(i) = v_i$

$$\sum_{i=0}^{N-1} |i\rangle \longrightarrow \sum_{i=0}^{N-1} (-1)^{v_i} |i\rangle$$

They each apply the Walsh transformation to get a common global state

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{u_i \oplus v_i} \left(W|i\rangle \otimes W|i\rangle \right) = \frac{1}{N\sqrt{N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} (-1)^{u_i \oplus v_i} (-1)^{i \cdot j} (-1)^{i \cdot k} |jk\rangle$$

The probability that the measurement results in $a = x = b$ is the modulus squared of $\langle x, x|\psi\rangle$

$$\langle x, x|\psi\rangle = \frac{1}{N\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{u_i \oplus v_i} (-1)^{i \cdot x} (-1)^{i \cdot x} = \frac{1}{N\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{u_i \oplus v_i}$$

Distributed computation

The probability that Alice and Bob measure the same n bit value, x, is given by

$$P_{xx} = |\langle x, x|\psi\rangle|^2 = \left| \frac{1}{N\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{u_i \oplus v_i} \right|^2$$

If $u = v$, then $(-1)^{u_i \oplus v_i} = 1$ so when summed over the N possible values of x, $P_{xx} = 1$ and Alice and Bob will measure $a = b$ with probability 1

$$\langle x, x|\psi\rangle = \frac{1}{\sqrt{N}}$$

For $d_H(u, v) = N/2$ there will be exactly the same number of 1 and -1 values in the sum so $P_{xx} = 0$ and Alice and Bob will measure $a = b$ with probability 0

$$\langle x, x|\psi\rangle = 0$$

Quantum parallelism

The action of the quantum operator U_f on a maximally superposed state appears to do more computation than a classical computation of $f(x)$

$$U_f|x,0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x,f(x)\rangle$$

This is not the case, as a single measurement of an m qubit system can only result in a single m-qubit value

Exponential speedups through the use of quantum computation are not possible in general and in many cases there is no speedup at all

Even an efficient quantum algorithm cannot probe the vast space of an $n \otimes m$ qubit system

The value of quantum computing lies in two general techniques

Amplification of outputs of interest: By transforming the state in a way that the output values of interest have a higher probability of being measured

Measuring properties of the set of all $f(x)$: An example is using the quantum Fourier Transform to determine the periodicity of $f(x)$

Discrete Fourier transform

The quantum Fourier transform is an important building block for many quantum algorithms

In order to develop the efficient implementation of the quantum Fourier transform, it is useful to start with the classical discrete and fast Fourier transforms

The discrete Fourier transform (DFT) is a linear transformation which takes a discrete column vector $a(k)$ to a column vector of Fourier coefficients, $A(x)$, where $0 \leq k, x \leq N-1$

$$A(x) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a(k) e^{2\pi i kx/N}$$

The DFT operator is an $N \times N$ matrix with elements

$$F_{xk} = \tfrac{1}{\sqrt{N}} e^{2\pi i kx/N}$$

Assume $a(k) = e^{-2\pi i uk/N}$ is a function of frequency $u < N$ which evenly divides N

Computing the Fourier coefficients,

$$A(x) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a(k) e^{2\pi i kx/N} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{-2\pi i uk/N} e^{2\pi i kx/N} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i k(x-u)/N}$$

All are zero except for when $x - u = 0 \mod N$ so the only term which survives is $A(u)$