Basics of Quantum Algorithms

Projective Measurement

A PM is defined by $M = \sum_m \lambda_m |m\rangle \langle m|$ where $\{\lambda_m\}$ are the set of outcomes and $\{M_m\} = \{|m\rangle \langle m|\}$ are the set of measurement operators.

$$p(m) = \langle\psi|M_m^\dagger M_m|\psi\rangle = \langle\psi|m\rangle\langle m|\psi\rangle = \|\langle m|\psi\rangle\|^2$$

$$\sum_m p(m) = 1 \text{ and } |\psi\rangle \xrightarrow{\text{on measurement}} \frac{M_m |\psi\rangle}{\sqrt{p(m)}} = |m\rangle$$

Classical Circuit

$$f : \{0,1\}^n \to \{0,1\}^m$$

A function with an m-bit output is equivalent to m functions, each with a one-bit output, so we may just as well say that the basic task performed by a computer is the evaluation of $f : \{0,1\}^n \to \{0,1\}$.

$$f : \{0,1\}^n \to \{0,1\}, \text{ a boolean function}$$

The evaluation of a boolean function f can be reduced to a sequence of simple logical operations.

$$f^{(a)}(x) = \begin{cases} 1 & x = x^{(a)} \\ 0 & \text{otherwise} \end{cases}$$

where $f^{(a)}$ is obtained as the AND of n bits with NOT operation may or maynot present with the atomic variables acting as the n bits.

$$f(x) = f^{(1)}(x) \vee f^{(2)}(x) \vee \dots$$

Thus, the above form of $f(x)$ is referred to as the DNF form (disjunctive normal form). Thus, $\{AND, OR, NOT\}$ is universal.

Note: the gatesets $\{AND, OR, NOT\}$, $\{NAND\}$ etc. are universal.

Reversible Computation

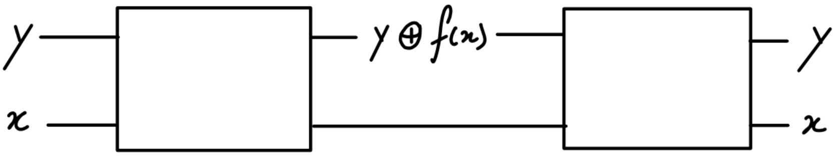
Landauer's Principle: erasure of information is a dissipative process.

According to Landauer’s principle, then, we need to do an amount of work at least $W = kT\ln 2$ to operate a 2-to-1 logic gate at temperature T.

But, if we only allow invertible functions from $\{0,1\}^n$ to $\{0,1\}^n$ then there need be no dissipation and no power requirement. We can compute for free!

Any irreversible computation can be packaged as an evaluation of an invertible function. For example, for any $f:\{0,1\}^n \to \{0,1\}$, we can construct $\hat{f}:\{0,1\}^{n+1} \to \{0,1\}^{n+1}$.

$$\hat{f}(x,y) = (x, y \oplus f(x))$$



Universal reversible gates: {Fredkin} and {Toffolli} for classical computation. {Toffoli, Hadamard} makes the set of universal quantum gates.

Uncomputation

Disadvantages of Reversible computing:

1. Garbage output bits: $(x,0) \to (f(x), g(x))$ where $f(x) =$ required output, $g(x) =$ garbage output. Now, $|f(x)\rangle\,|g(x)\rangle$ can be entangled.
2. Input dependent garbage qubits can interfere with your computation and give erroneous results.

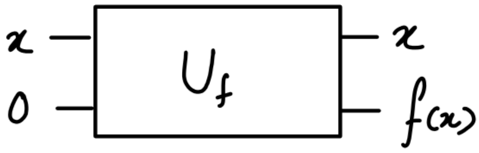
But using uncomputation, we can clean up the garbage at a small constant cost. We can convert all reversible circuits this way into $(x,y) \to (x, y \oplus f(x))$.

Universal Quantum Gates

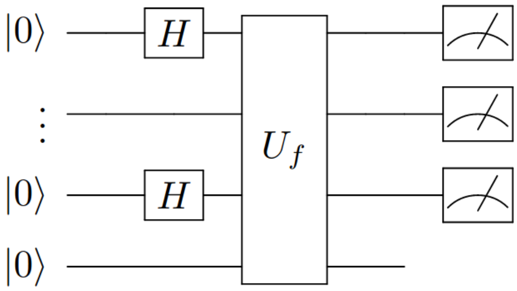
$$||U - U_t U_{t-1} \ldots U_1|| < \epsilon, \quad U_i \in G$$

Solovay Ketanov Theorem: Any general t-gate quantum circuit can be ϵ approximated using only $O(t\ \mathrm{polylog}(1/\epsilon))$ gates from $G = \{CNOT, H, R_{\pi/4}\}$. There are also other universal gate sets: some are efficient than others.

Quantum Parallelism



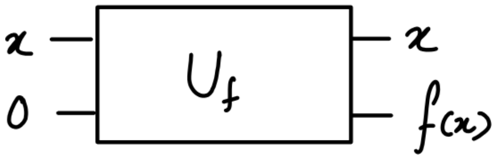
$$U_f |x\rangle|0\rangle = |x\rangle|f(x)\rangle$$



$$|0\rangle^{\otimes n}|0\rangle \xrightarrow{H^{\otimes n}\nVdash} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle\,|0\rangle \xrightarrow{U_f} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle\,|f(x)\rangle$$

- By applying U_f only once, we are able to obtain a quantum state that contains in it all 2^n possible values of $f(x)$ in superposition.
- This in itself is not very useful. If we make projective measurement, we will observe some $|z\rangle|f(z)\rangle$ with probability $1/2^n$.
- Quantum parallelism needs to be combined with interference, entanglement, to something better than classical computing.

Phase Kickback Oracle



If we substitute $|-\rangle$ for y we get:

if $f(x) = 0$,

$$|x\rangle \frac{|0\rangle - |1\rangle}{2} \xrightarrow{U_f} |x\rangle \frac{|0\rangle - |1\rangle}{2}$$

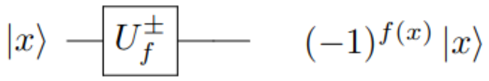
and if $f(x) = 1$, then we have

$$|x\rangle \frac{|0\rangle - |1\rangle}{2} \xrightarrow{U_f} -|x\rangle \frac{|0\rangle - |1\rangle}{2}$$

and thus we have the follwoing.

$$|x\rangle |-\rangle \xrightarrow{U_f} (-1)^{f(x)} |x\rangle |-\rangle$$

The second input and output lines can be dropped as they remain the same in another frequently used representation.



$$|x\rangle \xrightarrow{U_f^{\pm}} (-1)^{f(x)} |x\rangle$$

$$H^{\otimes n} |0\rangle^{\otimes n} \xrightarrow{U_f^{\pm}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}} |x\rangle \xrightarrow{U_f^{\pm}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}} (-1)^{f(x)} |x\rangle$$

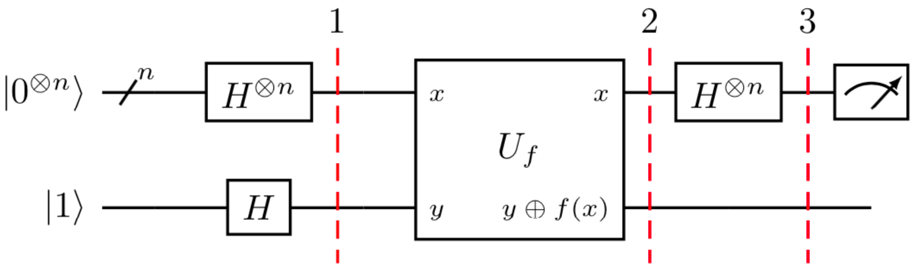
Deutsch-Josza Algorithm

In the Deutsch–Jozsa problem, we are given a black box quantum computer known as an oracle that implements some function $f : \{0,1\}^n \to \{0,1\}$. The function takes n-digit binary values as input and produces either a 0 or a 1 as output for each such value.

We are promised that the function is either constant (0 on all outputs or 1 on all outputs) or balanced (returns 1 for half of the input domain and 0 for the other half). The task then is to determine if f is constant or balanced by using the oracle.

In the Deutsch-Josza quantum algorithm the speedup is exponential.

- Classical Complexity: $2^{n-1} + 1$ queries
- Quantum Complexity: 1 query



$$|0^{\otimes n}\rangle \otimes |1\rangle \xrightarrow{H^{\otimes n+1}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |-\rangle$$

$$\xrightarrow{U_f} \frac{1}{\sqrt{2^{n+1}}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes \left(|f(x)\rangle - |1 \oplus f(x)\rangle\right) = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle \otimes |-\rangle$$

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{\forall x} (-1)^{f(x)} \left[\sum_{\forall y} (-1)^{x.y} |y\rangle \right]$$

$$= \frac{1}{\sqrt{2^n}} \sum_{\forall x} \left[\sum_{\forall y} (-1)^{f(x)\oplus x.y} |y\rangle \right] = \frac{1}{\sqrt{2^n}} \sum_{\forall y} \left[\sum_{\forall x} (-1)^{f(x)\oplus x.y} \right] |y\rangle$$

Now, the probability of measuring $|0^{\otimes n}\rangle$ is as given bellow.

$$\| \frac{1}{\sqrt{2^n}} \sum_{\forall x\in\{0,1\}^n} (-1)^{f(x)} \| = \begin{cases} 0 & f(x) \text{ is balanced} \\ 1 & f(x) \text{ is constant} \end{cases}$$

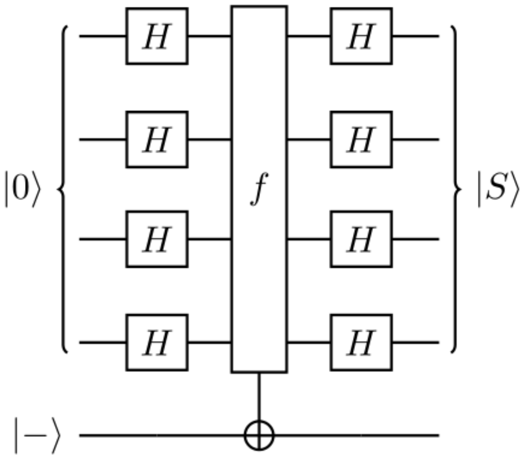
```
def oracle(circuit, n):
    case = np.random.randint(2)

    # consider the function to be constant
    if case == 1:
        print("Example Constant:")
        output = np.random.randint(2)
        if output == 1:
            circuit.x(n)
        return circuit

    # example of a balanced function
    else:
        print("Example Balanced:")
        circuit.cx(0, n)
        return circuit


def deutsch_josza(n):
    circuit = QuantumCircuit(n + 1, n + 1)
    for i in range(0, n):
        circuit.h(i)
    circuit.x(n)
    circuit.h(n)
    circuit = oracle(circuit, n)
    for i in range(0, n):
        circuit.h(i)

    for i in range(0, n):
        circuit.measure(i, i)

    print(circuit)
    return circuit
```

Bernstein-Vazirani Algorithm

Problem: Given a function $f : \{0,1\}^n \to \{0,1\}$ and a input x, $f(x) = (s.x) \mod 2$. We are expected to find s.

Classical Complexity: n queries

Quantum Complexity: 1 queries



$$|0^{\otimes n}\rangle \otimes |1\rangle \xrightarrow{H^{\otimes n+1}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |-\rangle \xrightarrow{U_f} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle \otimes |-\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{s\cdot x} |x\rangle \otimes |-\rangle \xrightarrow{H^{\otimes n}\otimes\mathbb{I}} \frac{1}{\sqrt{2^n}} \sum_{x,y\in\{0,1\}^n} (-1)^{s\cdot x+x\cdot y} |y\rangle \otimes |-\rangle$$

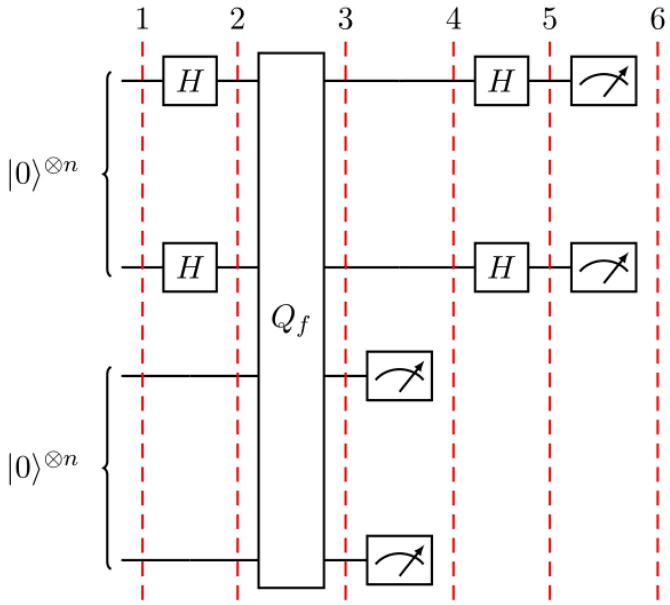
The above evaluates to $|s\rangle$ because of the reason mentioned below.

$$|s\rangle \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{s.x} |x\rangle \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{x,y\in\{0,1\}^n} (-1)^{s\cdot x+x\cdot y} |y\rangle$$

Simon’s Algorithm

Problem: Given a function $f : \{0,1\}^n \to \{0,1\}^n$ and some x and y, $f(x) = f(y)$ if and only if $x = y \oplus s$.

Problem	Classical Complexity	Quantum Complexity
Deutsch-Josza	$2^{n-1}+1$	1
Bernstein-Vazirani	n	1
Simon’s Algorithm	$O(\sqrt{2^n})$	$O(n)$



$$(H^{\otimes n}|0^n\rangle)\otimes|0^n\rangle = \frac{1}{2^{n/2}}\sum_{x\in 0,1^n}(|x\rangle\otimes|0^n\rangle)$$

$$\xrightarrow{U_f}\frac{1}{2^{n/2}}\sum_{x\in\{0,1\}^n}|x\rangle\otimes|f(x)\rangle = \frac{1}{2^{(n-1)/2}}\sum_{x\in\{0,1\}^n}\frac{(|x\rangle+|x\oplus s\rangle)}{\sqrt{2}}\otimes|f(x)\rangle$$

Once we measure the last n registers, we obtain some $|f(x)\rangle$. Then the state on our above register converges to some $(|x\rangle+|x\oplus s\rangle)/\sqrt{2}$.

$$\frac{|x\rangle+|x\oplus s\rangle}{\sqrt{2}}\xrightarrow{H^{\otimes n}}\frac{1}{\sqrt{2^{n+1}}}\sum_z[(-1)^{x\cdot z}+(-1)^{(x\oplus s)\cdot z}]\,|z\rangle$$

Thus, we finally obtain the following state in our first n registers.

$$\frac{1}{\sqrt{2^{n+1}}}\sum_z(-1)^x[1+(-1)^{s\cdot z}]\,|z\rangle$$

Now, upon measurement we will observe $|z\rangle$ if and only if $(s\cdot z) \mod 2 = 0$.

$$\begin{gathered} s\cdot z_1 = 0 \\ \vdots \\ s\cdot z_n = 0 \end{gathered}$$

If $\{z_1\ldots z_2\}$ are linearly independent then we have s after $O(4n)$. The probability that they are independent is at least $> 1/4$.