Grover's Algorithm

Jiman Hwang (pingdummy1@gmail.com)

January 18, 2018

Disclaimer: This document is for self-study only and may contain false information. Mainly referenced from [Vaz]

1 Problem

Given function $f : \{0, \cdots, N-1\} \to \{0, 1\}$ such that

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

where $N = 2^n$. Find a.

2 Solution

2.1 Overview

1. Have a state of the same amplitude.
2. Negate the amplitude at $x = a$.
3. Flip all amplitude about their mean.
4. Repeat 2-3 enough and measure.

2.2 Amplitude inverter

Assuming we have the classical gate that performs $f(x)$, we can build the corresponding quantum gate U_f such that

$$U_f : |x\rangle |b\rangle \to |x\rangle |b \oplus f(x)\rangle$$

where $b \in \{0, 1\}$, $\oplus$ is XOR. To make a gate that negates the amplitude only if $x = a$, use U_f.

$$U_f |x\rangle |0\rangle = |x\rangle |f(x)\rangle \tag{1}$$
$$U_f |x\rangle |1\rangle = |x\rangle |1 \oplus f(x)\rangle \tag{2}$$

Managing two equations,

$$\frac{(1)-(2)}{\sqrt{2}} : U_f |x\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} = \frac{|f(x)\rangle - |1 \oplus f(x)\rangle}{\sqrt{2}}$$

which follows

$$U_f |x\rangle |-\rangle = (-1)^{f(x)} |x\rangle |-\rangle \quad \text{(phase kickback)}$$

Thus, by putting $|-\rangle$ at the last qubit to U_f, we meet the goal.

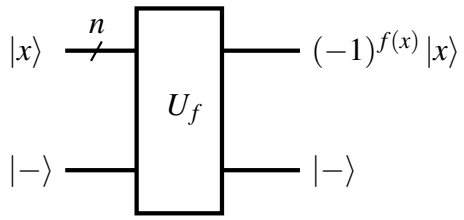

Figure 1: Amplitude inverter

2.3 Inversion about mean

Let R_M denote the gate that flips the input amplitudes about their mean. Then,

$$R_M : \sum_{x=0}^{N-1} \alpha_x |x\rangle \to \sum_{x=0}^{N-1} [-(\alpha_x - \mu) + \mu] |x\rangle = \sum_{x=0}^{N-1} (2\mu - \alpha_x) |x\rangle$$

where $\mu = \frac{1}{N} \sum_{x=0}^{N-1} \alpha_x$.
FYI, the output is a valid new state.

$$\left| \sum_{x=0}^{N-1} (2\mu - \alpha_x) |x\rangle \right|^2 = \sum_{x=0}^{N-1} (2\mu - \alpha_x)^2 = \sum_{x=0}^{N-1} \left(4\mu^2 - 4\mu\alpha_x + \alpha_x^2\right) = 4\mu^2 N - 4\mu^2 N + 1 = 1$$

To elicit R_M, let's input basis. That is, $|k\rangle$ or $\alpha_k = 1$ where $k \in \{0, \cdots, N-1\}$. Then, $\mu = 1/N$, and

$$R_M |k\rangle = \frac{2}{N} \sum_{x=0}^{N-1} |x\rangle - |k\rangle$$

Combining all k's,

$$\begin{bmatrix} R_M |0\rangle & \cdots & R_M |N-1\rangle \end{bmatrix} = \begin{bmatrix} \frac{2}{N} \sum_x |x\rangle - |0\rangle & \cdots & \frac{2}{N} \sum_x |x\rangle - |N-1\rangle \end{bmatrix}$$

or

$$R_M = \frac{2}{N} \begin{bmatrix} \sum_x |x\rangle & \cdots & \sum_x |x\rangle \end{bmatrix} - I$$

Let $B = \begin{bmatrix} \sum_x |x\rangle & \cdots & \sum_x |x\rangle \end{bmatrix}$, then

$$R_M = \frac{2}{N}B - I \tag{3}$$

In order to implement R_M as a quantum gate, R_M must be expressed as product of unitary matrices. Note that B is normal [Wei], thereby is unitarily diagonalizable by Spectral theorem (Spectral theorem at p.397 in [Lay11]). Suppose B is diagonalized into $PDP^\dagger$. Then,

$$(3) : R_M = \frac{2}{N}PDP^\dagger - PP^\dagger = P\left(\frac{2}{N}D - I\right)P^\dagger \tag{4}$$

which is the product of unitary matrices. Meanwhile P is composed of eigenvectors of B. If $\lambda, |v\rangle$ are eigenvalue and eigenvector respectively, $B|v\rangle = \lambda|v\rangle$. Let $|v\rangle = \sum_{x=0}^{N-1} \beta_x |x\rangle$, then

$$B|v\rangle = \sum_{x=0}^{N-1}\left(\sum_{y=0}^{N-1}\beta_y\right)|x\rangle \quad , \quad \lambda|v\rangle = \sum_{x=0}^{N-1}\lambda\beta_x|x\rangle$$

It follows

$$\forall x \quad \sum_{y=0}^{N-1}\beta_y = \lambda\beta_x$$

Let $p = \sum_x \beta_x$, then

$$\forall x \quad p = \lambda\beta_x \tag{5}$$

Also,

$$\sum_{x=0}^{N-1} p = \sum_{x=0}^{N-1}\lambda\beta_x \quad \Rightarrow \quad pN = p\lambda$$

We consider two cases, $p \neq 0$ and $p = 0$. If $p \neq 0$, then $\lambda = N$ and

$$\forall x \quad \beta_x = \frac{p}{N} \stackrel{\text{def}}{=} \beta$$

From quantum state condition,

$$\sum_{x=0}^{N-1}\beta_x^2 = N\beta^2 = 1$$

Satisfying this condition, we set β to $1/\sqrt{N}$, which produces $|v\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle$. Note that this eigenvector contributes one rank.

If $p = 0$, then $\lambda = 0$ from (5). Since

$$p = \sum_{x=0}^{N-1}\beta_x = 0 \tag{6}$$

is the only condition that binds $|v\rangle$, the eigenspace corresponding to $\lambda = 0$ has rank $N-1$.

Altogether, let $|v_1\rangle$ denote the eigenvector for $\lambda = N$, and $|v_k\rangle$ the eigenvector for $\lambda = 0$ where $k = 2, 3, \cdots, N$. With those, build P, D as follow.

$$P = \begin{bmatrix} 1/\sqrt{N} & \square & \cdots & \square \\ 1/\sqrt{N} & \square & \cdots & \square \\ \vdots & \vdots & \ddots & \vdots \\ 1/\sqrt{N} & \square & \cdots & \square \end{bmatrix} \quad , \quad D = \mathbf{diag}\{N, 0, 0, \cdots, 0\}$$

where empty columns in P satisfy (6). Since there is no further constraint, we can simply fill $\square$ with values as we like. To ease implementation, we adjust values so $P = H^{\otimes n}$. Using it,

$$(4) : R_M = H^{\otimes n} \left(\frac{2}{N} D - I \right) H^{\otimes n}$$

The left task is to implement $\frac{2}{N}D - I$ into the circuit. Note that

$$\frac{2}{N} D - I = \mathbf{diag}\{1, -1, -1, \cdots, -1\}$$

In other words,

$$\begin{cases} |0\rangle \to |0\rangle \\ |x\rangle \to -|x\rangle \quad \text{if } x \neq 0 \end{cases}$$

Reminding ourselves of Sec 2.2, we can take advantage of the phase kickback again. The phase kickback should be opened only if there's a non $|0\rangle$ qubit. We use a controlled not gate and X gates to distinguish this case and the others as follow.

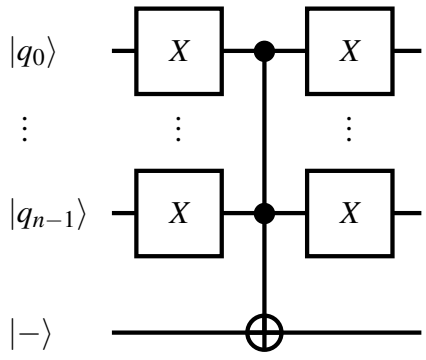

Figure 2: $\frac{2}{N}D - I$(upper n qubits)

or, equivalently,

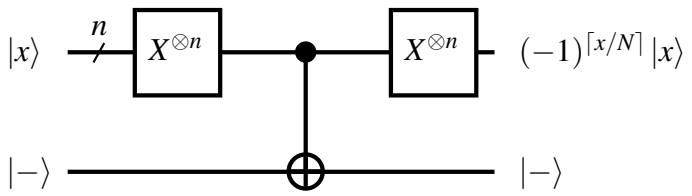

Figure 3: $\frac{2}{N}D - I$(upper n qubits, reduced expression)

Altogether, the circuit for inversion about mean looks like this.

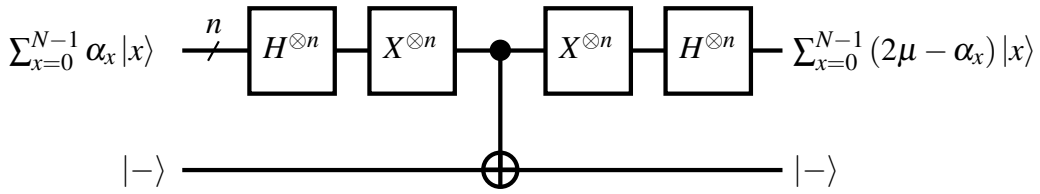

Figure 4: inversion about mean

2.4 The circuit

To sum up, enter $\sum_x |x\rangle$, $|-\rangle$, to U_f, R_M repeatedly and then measure.

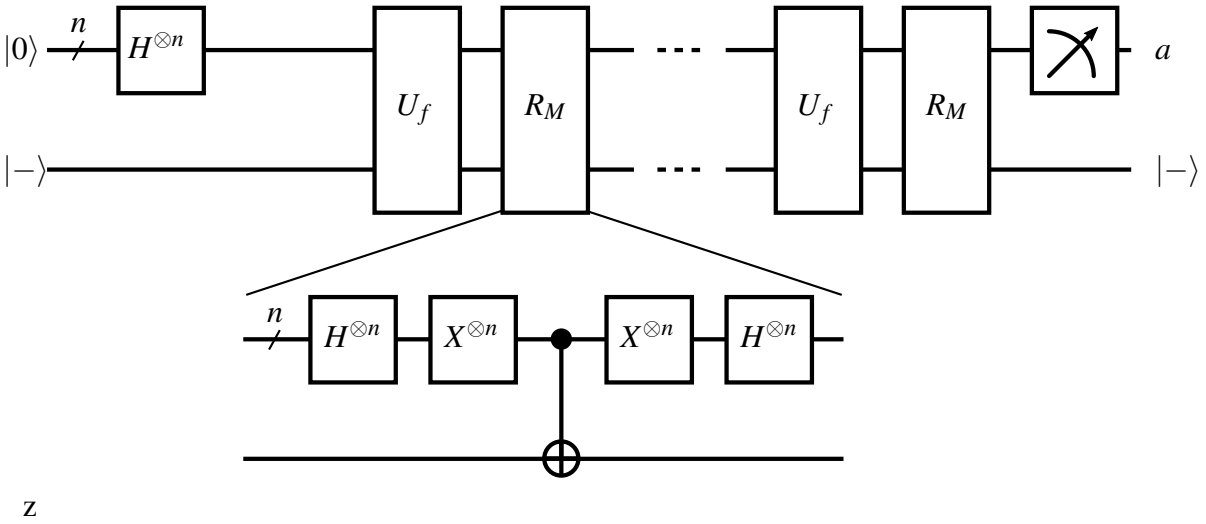

Figure 5: Grover's algorithm

2.5 Number of iterations

Suppose we repeat U_f, R_M h times. If h is too small or too big, the chance to get a from measurement may get small. Here, we aim the probability of obtaining a to $1/2$. Let α_k denote the amplitude of the state at $x = a$, β_k denote that at $x \neq a$ after k iteration.

Clearly, $\alpha_0 = \beta_0 = 1/\sqrt{N}$. Drawing out the recurrence,

$$\begin{cases} a_k = -(-a_{k-1} - \mu_{k-1}) + \mu_{k-1} = 2\mu_{k-1} + a_{k-1} \\ b_k = -(b_{k-1} - \mu_{k-1}) + \mu_{k-1} = 2\mu_{k-1} - b_{k-1} \end{cases}$$

where $\mu_{k-1} = [-a_{k-1} + (N-1)\, b_{k-1}]/N$, or

$$\begin{cases} a_k = \left(1 - \frac{2}{N}\right) a_{k-1} + \frac{2(N-1)}{N} b_{k-1} \\ b_k = -\frac{2}{N} a_{k-1} + \frac{N-2}{N} b_{k-1} \end{cases}$$

In matrix expression,

$$\begin{bmatrix} a_k \\ b_k \end{bmatrix} = \frac{1}{N} \begin{bmatrix} N-2 & 2N-2 \\ -2 & N-2 \end{bmatrix} \begin{bmatrix} a_{k-1} \\ b_{k-1} \end{bmatrix}$$

Modifying the right hand side,

$$\begin{bmatrix} a_k \\ b_k \end{bmatrix} \leq \frac{1}{N} \begin{bmatrix} N & 2N \\ 0 & N \end{bmatrix} \begin{bmatrix} a_{k-1} \\ b_{k-1} \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_{k-1} \\ b_{k-1} \end{bmatrix}$$

where the inequality sign applies to each elements of matrices. Solving the recurrence,

$$\begin{bmatrix} a_k \\ b_k \end{bmatrix} \leq \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}^k \begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$$

or

$$\begin{bmatrix} a_k \\ b_k \end{bmatrix} \leq \begin{bmatrix} 1 & 2k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{N} \\ 1/\sqrt{N} \end{bmatrix}$$

Solving for a_k,

$$a_k \leq \frac{1+2k}{\sqrt{N}}$$

Since $a_h = 1/\sqrt{2}$,

$$\frac{1}{\sqrt{2}} \leq \frac{1+2h}{\sqrt{N}}$$

Asymptotically,

$$h = \Omega\left(\sqrt{N}\right)$$

Unfortunately, I couldn't figure out how to prove $h = O\left(\sqrt{N}\right)$. For the rest of document we assume $h = \Theta\left(\sqrt{N}\right)$. :-(

2.6 Cost of algorithm

Let C denote the random variable for the cost of the Grover's algorithm. The range is $\left\{\Theta\left(\sqrt{N}\right), \Theta\left(2\sqrt{N}\right), \cdots\right\}$, and Probability Mass Function(PMF) is

$$p_C(k) = \begin{cases} \left(\frac{1}{2}\right)^k & \text{if } k = 1, 2, \cdots \\ 0 & \text{otherwise} \end{cases}$$

Calculating the expectation,

$$E(C) = \sum_{k=1}^{\infty} \Theta\left(k\sqrt{N}\right)\left(\frac{1}{2}\right)^k = \Theta\left(\sqrt{N}\sum_{k=1}^{\infty}\frac{k}{2^k}\right) = \Theta\left(\sqrt{N}\cdot 2\right) = \Theta\left(\sqrt{N}\right)$$

References

[Lay11] David C. Lay. *Linear Algebra and Its Applications*. Pearson, 4 edition, 2011.

[Vaz] Umesh Vazirani. Grover's search algorithm. `courses.edx.org/courses/BerkeleyX/CS-191x/2013_August/courseware/c4ebc1bd43144a0395eca18d4810ed31/2fd0c74e8c794b42bf1904167cc839a5/`.

[Wei] Eric W. Weisstein. Normal matrix. `mathworld.wolfram.com/NormalMatrix.html`.