Notes on quantum search algorithm

Zhongwei Jiang, Linfeng Zhao

August 25, 2024

1 Quantum Computation

1.1 Quantum Gate

1.1.1 Single-qubit gate

The number of qubits in a quantum state depends on the number of classical bits in its dimension, so we usually call a vector $|\psi\rangle = a|0\rangle + b|1\rangle$ parameterized by two complex numbers a and b satisfying $|a|^2 + |b|^2 = 1$ a qubit. According to the constraints and phase redundancy, the quantum state can be mapped to the Bloch sphere with two parameters:

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}\mathrm{e}^{\mathrm{i}\varphi}|1\rangle \tag{1.1.1}$$

Operations on a qubit must preserve the norm, and thus are described by 2×2 unitary matrices.By $\det|U| = e^{2i\alpha}$ Then

$$e^{-i\alpha}U \in SU(2) \tag{1.1.2}$$

They're only off by one global phase, so we usually ignore the global phase and only consider $SU(2)$.

The Lie group $SU(2)$ has three generators $i\sigma_j$, σ_j called Pauli matrices:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{1.1.3}$$

From Lie group elements, any single-qubit unitary can be written as a product of exponentials of Pauli matrices by a global phase:

$$U = e^{i\alpha}\exp\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right) \tag{1.1.4}$$

$\vec{n}$ is the coordinates of the rotation axis on the Bloch sphere. There is a homomorphism:

$$\begin{aligned} SU(2) &\to SO(3) \\ \exp\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right) &\mapsto R_{\vec{n}}(\theta) \end{aligned} \tag{1.1.5}$$

The former refers to single-qubit gates and the latter to transformations on the Bloch sphere. Since we often need transformations on the Bloch sphere to visualize quantum gates, we will use $R_{\vec{n}}(\theta)$ to refer to $\exp\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right)$ directly in the following. So $R_{\vec{n}}(\theta) \coloneqq \exp\left(-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}\right)$.

According to the lemma:

Lemma 1.1.1. *The real number x and the matrix A such that $A^2 = I$ give us*

$$e^{iAx} = \cos(x)I + i\sin(x)A \tag{1.1.6}$$

we know that:

$$R_{\vec{n}}(\theta) = \cos\left(\frac{\theta}{2}\right)I - i\sin\left(\frac{\theta}{2}\right)(\vec{n}\cdot\vec{\sigma}) \tag{1.1.7}$$

There is another form about $V \in SU(2)$:

$$V = R_{\vec{n}}(\beta)R_{\vec{m}}(\gamma)R_{\vec{n}}(\delta) \tag{1.1.8}$$

where $\vec{n} \neq k\vec{m}$,so we have:

$$U = e^{i\alpha}R_{\vec{n}}(\beta)R_{\vec{m}}(\gamma)R_{\vec{n}}(\delta) \tag{1.1.9}$$

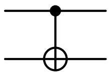
In the following, we use XYZ instead of $\sigma_x\sigma_y\sigma_z$, here are some other single-qubit gate, which are frequently used listed here:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \tag{1.1.10}$$

1.1.2 Controlled operation

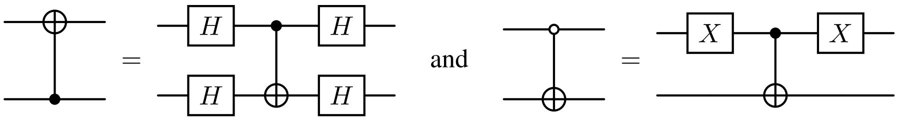
Analogous to the if statement of a classical circuit, we can use controlled-NOT gate to control target qubit with control qubit, its matrix form and circuit are shown as follows:

$$CNOT = \begin{pmatrix} I & 0 \\ 0 & X \end{pmatrix} \tag{1.1.11}$$



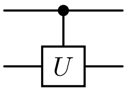
(1.1.12)

CNOT gate makes the state of the target qubit be flipped when the control qubit is 1 and remain unchanged when the control qubit is 0. That is $|c\rangle\,|t\rangle \to |c\rangle\, X^c\,|t\rangle$. We will also use two deformations of CNOT gates:



(1.1.13)

We expect to extend this control from X to an arbitrary single-qubit gate U. That is, $|c\rangle\,|t\rangle \to |c\rangle\, U^c\,|t\rangle$ which is known as a controlled-U gate. Show that:



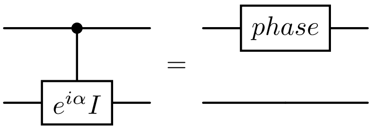
(1.1.14)

We wish to decompose this into a combination of single quantum bit gates and CNOT gates by introducing the theorem:

Lemma 1.1.2. *If U is a single quantum bit gate, then there exists $A, B, C \in SU(2)$, $ABC = I$, and then there is*

$$U = e^{i\alpha} AXBXC \tag{1.1.15}$$

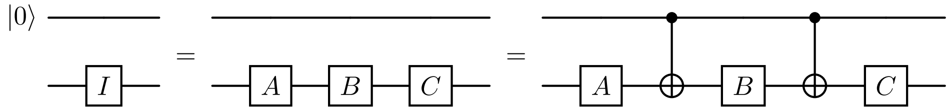
With this theorem, we can divide the controlled-U gate into several parts, looking first at the global phase part, which obviously has:



(1.1.16)

where $phase = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$.

Looking at the $AXBXC$ part again, we have the expectation that if the control qubit is $|0\rangle$, then we do nothing, and adding a CNOT gate anywhere doesn't change that, so we have.

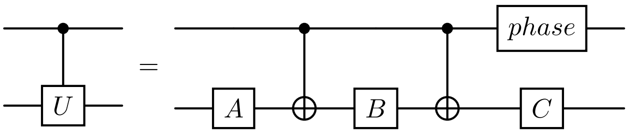


(1.1.17)

This construction also satisfies:if the control qubit is $|1\rangle$, then we apply $AXBXC$ to the target qubit.

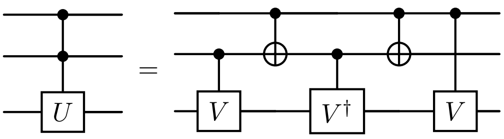
So we have that:

Theorem 1.1.1. *The controlled-U gate can be decomposed with CNOT gates and single qubit gates into this form:*



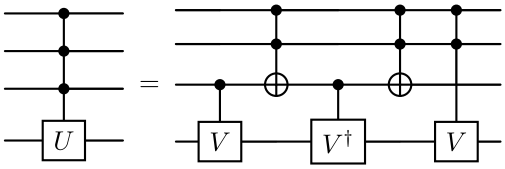
(1.1.18)

Considering the case of multiple control Qubits, first look at the case of two control Qubits, you can construct: if V is a single qubit gate satisfying $V^2 = U$, we have:



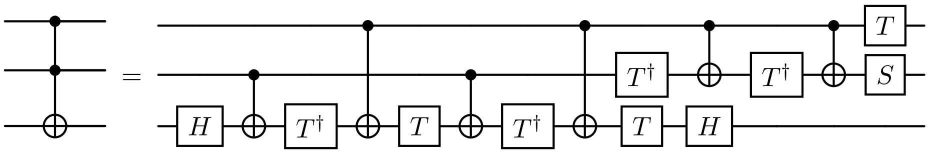
(1.1.19)

Then we can extend this to the case of n control qubits, there are many kinds of construction methods, here we use recursive method to construct:



(1.1.20)

Ultimately we will show that any unitary operation can be composed to an arbitrarily good approximation from just H, S, T, CNOT gates. Here is the construction of the Toffoli gate:



(1.1.21)

1.2 Universal Gate Set

1.2.1 Single-qubit and CNOT gates are universal

In the previous subsection, we showed that $C^n(U)$ gate can be decomposed into a combination of single-qubit gates and CNOT gates. In this section, we will show that any unitary gate can be combined from $C^n(U)$ gate.

First, we introduce the theorem:

Lemma 1.2.1. *Any D-dimensional unitary transformation U can always be decomposed into the product of $d(d-1)/2$ two-level unitary transformations under natural basis.*

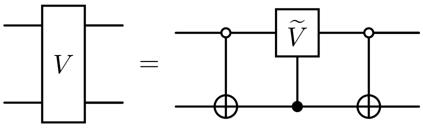
So exists two-level unitary transformations V_j made $U = \prod_{j=1}^{d(d-1)/2} V_j$[1]. Two-level unitary transformations are unitary matrices which act non-trivially only on two-or-fewer vector components.The proof idea of this theorem is simple[2], let's focus on the two-level unitary transformations V.

Consider the binary expansion of the two bases $|s\rangle$ and $|t\rangle$ on which V operates, where $s = s_1 \dots s_n$ and $t = t_1 \dots t_n$. We can use Toffoli gate to turn s_j into t_j, after $n-1$ steps $|s_1 \dots s_{k-1} s_k s_{k+1} \dots s_n\rangle$ into $|t_1 \dots t_{k-1} s_k t_{k+1} \dots t_n\rangle$. It puts $|s\rangle$ and $|t\rangle$in the same qubit and makes V into a $C^n(\widetilde{V})$ gate which $\widetilde{V}$ is the non-trivial 2×2 unitary submatrix of V.

Let's take an example to illustrate this theorem, consider the U gate:

$$V = \begin{pmatrix} a & 0 & 0 & c \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ b & 0 & 0 & d \end{pmatrix} \quad \text{and} \quad \widetilde{V} = \begin{pmatrix} a & c \\ b & d \end{pmatrix} \tag{1.2.1}$$

Notice that V acts non-trivially only on the states $|00\rangle$ and $|11\rangle$, so we can use the CNOT gate to turn $|00\rangle$ into $|01\rangle$, then we can get the $C(\widetilde{V})$ gate. Here is the circuit of V:



(1.2.2)

So we have:

Theorem 1.2.1. *Any unitary gate can be combined from single-qubit gates and CNOT gates.*

[1] Unless otherwise specified, the conjunction sign in this document is calculated according to the composite order of the map, that is $\prod_{j=1}^{n} a_j = a_n a_{n-1} \cdots a_1$.

[2] Each single-qubit gates has $d(d-1)/2$ degrees of freedom and each two-level unitary transformations turn one of single-qubit gates elements into zero in order. So exists two-level unitary transformations V_j made $\left(\prod_{j=1}^{d(d-1)/2} V_j \right) U = I$.

1.2.2 Approximate universal operators with a discrete set

It is hard to implement all single-qubit, so we consider to approximate it with a discrete set which we can implement. We introduce the induced norm of operators to measure the degree of approximation:

$$E(U,V) = \|U - V\| := \sup_{\substack{|\psi\rangle \in \mathbb{C}^n \\ \||\psi\rangle\|=1}} \|(U-V)|\psi\rangle\| \tag{1.2.3}$$

Where U is the target unitary operator that we wish to implement, V is the unitary operator that is actually implemented in practice, and $E(U,V)$ is the error when V is implemented instead of U. It is natural that:

Theorem 1.2.2. *in the approximation of m gates, the error add at most linearly:*

$$E\left(\prod_{j=1}^{m} U_j, \prod_{j=1}^{m} V_j\right) \leq \sum_{j=1}^{m} E(U_j, V_j) \tag{1.2.4}$$

Proof. Considering $|\psi_0\rangle$ which maximizes $\left\|\left(\prod_{i=1}^{m} U_i - \prod_{i=1}^{m} V_i\right)|\psi\rangle\right\|$ and definition $|\psi_i\rangle = V_i|\psi_{i-1}\rangle$, $|\Delta_i\rangle = U_i|\psi_{i-1}\rangle - |\psi_i\rangle$, we can see that

$$E\left(\prod_{j=1}^{m} U_j, \prod_{j=1}^{m} V_j\right) = \left\| |\Delta_m\rangle + \sum_{j=1}^{m-1}\left(\prod_{k=j+1}^{m} U_k\right)|\Delta_j\rangle \right\| \leq \sum_{j=1}^{m} \||\Delta_j\rangle\| \leq \sum_{j=1}^{m} E(U_j, V_j) \tag{1.2.5}$$

□

We are more concerned with the error of the approximation in the measurement. considering M is a POVM element in an arbitrary measurement POVM, and P_U(or P_V) is the probability of obtaining this outcome if U(or V) were performed with the state $|\psi\rangle$, We can prove[1] that:

$$|P_U - P_V| \leq 2E(U,V) \tag{1.2.6}$$

This makes it possible that if we want the probability difference between the approximate line and the ideal line on a certain outcome to be within a tolerance $\Delta > 0$, we only need to ensure that $E(U_j, V_j) \leq \Delta/(2m)$.

From the decomposition given by 1.1.9, we can further care about the degree of approximation of $R_{\vec{n}}(\theta)$. Let's start by introducing a neat theorem:

Lemma 1.2.2. *Considering α, $\theta \in \mathbb{R}/2\pi\mathbb{Z}$, if $\theta/\pi \in \mathbb{R}\backslash\mathbb{Q}$, we can find a subsequences $\{x_n\}$ of sequences $\{n\}$ makes $\lim\limits_{n\to\infty} x_n\theta = \alpha$.*

Proof. $\forall\epsilon > 0$, $\exists N = \frac{2\pi}{\epsilon}$, when $n > N$, we can find $|j\theta - k\theta| \leq \frac{2\pi}{n}$ which $j, k \in \{n\}, j > k$, let $x_n = (j-k)\left\lfloor\frac{\alpha}{|j\theta - k\theta|}\right\rfloor$, so we have $|x_n\theta - \alpha| < |j\theta - k\theta| \leq \frac{2\pi}{n} < \epsilon$. □

Further we have[2]:

$$\begin{aligned} &E(R_{\vec{n}}(\alpha), R_{\vec{n}}(\theta)^{x_n}) \\ =\ &E(R_{\vec{n}}(\alpha), R_{\vec{n}}(\alpha + (x_n\theta - \alpha))) \\ =\ &|1 - \exp((x_n\theta - \alpha)/2)| \leq\ \epsilon/2 \end{aligned} \tag{1.2.7}$$

According the proof, we just need to find the gate that has the Angle that the theorem requires. Fortunately, $THTH$ gate has the angel we need. It is a rotation of the Bloch sphere about an axis along $\vec{n} = (\cos\frac{\pi}{8}, \sin\frac{\pi}{8}, \cos\frac{\pi}{8})$ and through an angle θ defined by $\cos\frac{\theta}{2} = \cos^2\frac{\pi}{8}$, so we just let $R_{\vec{n}}(\theta) = THTH$. To construct unitary operator U, we still need to approximate the rotation of the other axis, but even more fortunate is that $HR_{\vec{n}}(\theta)H$ is exactly what we need, has axis $\vec{m} = (\cos\frac{\pi}{8}, -\sin\frac{\pi}{8}, \cos\frac{\pi}{8})$ and the Angle, like $R_{\vec{n}}(\theta)$, is an irrational multiple of π. After layers and layers of preparation, let's list the final approximation with suitable positive integers x_n, y_n, z_n to U:

$$E(U, R_{\vec{n}}(\theta)^{x_n} H R_{\vec{n}}(\theta)^{y_n} H R_{\vec{n}}(\theta)^{z_n}) \leq \frac{3}{2}\epsilon \tag{1.2.8}$$

Theorem 1.2.3. *Given any single qubit unitary operator U and any $\epsilon > 0$, it is possible to approximate U to within ϵ [3]using a circuit composed of H gates and T gates alone.*

[1] Let $|\Delta\rangle = (U-V)|\psi\rangle$, notice that $\langle\psi|U^\dagger MU|\psi\rangle - \langle\psi|V^\dagger MV|\psi\rangle = \langle\psi|U^\dagger M|\Delta\rangle + \langle\Delta|MV|\psi\rangle$.

[2] The construction of x_n in this part is only to prove the existence, without considering the complexity. There are actually far less complex constructs.

[3] The difference between Equation 1.2.8 and here by one coefficient can be solved by the setting of ϵ in the proof of Theorem 1.2.2, but it does not matter.

1.2.3 Circuit size for approximation

It makes no sense to talk about its existence without giving its size, so let's estimate the number of gates needed for the approximation in the previous section. For convenience of stating the theorem, we say that S is an ϵ-net in W, if every point in W is within a distance ϵ of some point in S, where S, $W \in SU(2)$ and $\epsilon > 0$ and the distance is $D(U,V) \coloneqq \mathrm{tr}|U-V|$. And we define $\mathcal{G}_l$ to be the set of all words of length at most l.

Theorem 1.2.4 (Solovay-Kitaev theorem)**.** *Let $\mathcal{G}$ be a fitite set of elements in $SU(2)$ conatining its own inverses, such that $\langle\mathcal{G}\rangle$ is dense in $SU(2)$. Let $\epsilon > 0$ be given. Then $\mathcal{G}_l$ is an ϵ-net in $SU(2)$ for $l = O(\ln^c(1/\epsilon))$, where $c = \ln 5/\ln(3/2)$.*

According to the theorem, we can konw that to approximate a circuit containing m single qubit unitaries to an accuracy ϵ requires $O(m\ln^c(m/\epsilon))$ gates from the discrete set. The proof of this theorem is quite long, so only the main ideas are given here. Let's first introduce the lamma:

Lemma 1.2.3. *Let $\mathcal{G}$ be a fitite set of elements in $SU(2)$ conatining its own inverses, such that $\langle\mathcal{G}\rangle$ is dense in $SU(2)$. For some $\epsilon > 0$ and any $k \in \mathbb{N}$, if $\mathcal{G}_l$ is an ϵ^2-net for S_ϵ, then $\mathcal{G}_{5^k l}$ is an $\epsilon(k)^2$-net for $S_{\epsilon(k)}$. where $\epsilon(k) = (C\epsilon)^{(3/2)^k}/C$, $\epsilon(k)^2 < \epsilon(k+1)$ and $S_\epsilon \coloneqq \{U \in SU(2) | D(U,I) \le \epsilon\}$.*

Proof. We know the element of $SU(2)$ can be written as $U = \exp(-i\vec{a}\cdot\vec{\sigma}/2)$, we write $U = u(\vec{a})$.And introduce the symbol $[U,V]_{gp} = UVU^\dagger V^\dagger$. With constant d, we have two conclusions:

$$D([u(\vec{a}),u(\vec{b})]_{gp}, u(\vec{a}\times\vec{b})) \le d\epsilon^3 \quad \text{and} \quad D(u(\vec{a}),u(\vec{b})) = \left\|\vec{a}-\vec{b}\right\| + O(\epsilon^3) \tag{1.2.9}$$

If $U = u(\vec{x}) \in S_{\epsilon^2}$, we can find $\vec{y}\times\vec{z} = \vec{x}$ which $u(\vec{y}), u(\vec{z}) \in S_\epsilon$, so that we can find $u(\vec{y_0}), u(\vec{z_0}) \in \mathcal{G}_l \cap S_\epsilon$ make $D(u(\vec{y_0}), u(\vec{y}))$, $D(u(\vec{z_0}), u(\vec{z})) \le \epsilon$. Notice that

$$\begin{aligned} D(U,[u(\vec{y_0}),u(\vec{z_0})]_{gp}) &\le D(U,u(\vec{y_0}\times\vec{z_0})) + D(u(\vec{y_0}\times\vec{z_0}),[u(\vec{y_0}),u(\vec{z_0})]_{gp}) \\ &= \|\vec{y}\times\vec{z} - \vec{y_0}\times\vec{z_0}\| + d\epsilon^3 \\ &\le (d+2)\epsilon^3 + O(\epsilon^4) \\ &\le C\epsilon^3 = \epsilon(1)^2 \end{aligned} \tag{1.2.10}$$

Specifically, given $U \in S_{\epsilon(1)}$, we can find $V \in \mathcal{G}_l$ such that $D(U,V) \le \epsilon(0)^2$, and thus $UV^\dagger \in S_{\epsilon(0)^2}$, so

$$D([u(\vec{y_0}),u(\vec{z_0})]_{gp}, UV^\dagger) = D([u(\vec{y_0}),u(\vec{z_0})]_{gp}V, U) \le \epsilon(1)^2 \tag{1.2.11}$$

that is, $\mathcal{G}_{5l}$ is an $\epsilon(1)^2$-net for $S_{\epsilon(1)}$. Recursively, $\mathcal{G}_{5^k l}$ is an $\epsilon(k)^2$-net for $S_{\epsilon(k)}$. □

For $U \in SU(2)$ we can find U_0 make $D(U_0,U) < \epsilon(0)^2 < \epsilon(1)$, so we have a first order approximation U_0 of U. We can go on to approximate their difference $V = UU_0^\dagger$, With the lemma, we can find U_1 make $D(U_1,V) < \epsilon(1)^2 < \epsilon(2)$, so we have a second order approximation U_1U_0 of U. Continue in this way, we can approximate to the accuracy we want which $\epsilon(k+1) < \epsilon$.

Although the approximation of a set of single-qubit gates is polynomial, approximating arbitrary unitary gates is actually hard. Considering the normalization of n qubit states $\||\psi\rangle\| = 1$ gives that the state space is an unit $(2^{n+1}-1)$-dimensional unit sphere. Given the error ϵ, a state approximation gives an $(2^{n+1}-2)$-dimensional sphere of radius ϵ. So in order to cover the state space, we need about[1]

$$\frac{S_{2^{n+1}-1}(1)}{V_{2^{n+1}-2}(\epsilon)} = \frac{\sqrt{\pi}\Gamma(2^n-\frac{1}{2})(2^{n+1}-1)}{\Gamma(2^n)\epsilon^{2^{n+1}-1}} = \Omega(\epsilon^{-2^{n+1}+1}) \tag{1.2.12}$$

states, where[2] $S_d(r)$, $V_d(r)$ is the surface area, volume of a d-dimensional sphere of radius r. But for a fixed initial state, m gates can only compute $O(n^{km})$ different states at most, where k is a constant. We must have:

$$O(n^{km}) \ge \Omega(\epsilon^{-2^{n+1}+1}) \tag{1.2.13}$$

which gives us

$$m = \Omega\left(\frac{2^n\ln(1/\epsilon)}{\ln(n)}\right). \tag{1.2.14}$$

This is exponential in n, so it is hard to approximate arbitrary unitary gates.

[1] O is for upper bounds, Ω is for lower bounds and Θ is for tight bounds.

[2] $S_k(r) = 2\pi^{(k+1)/2}r^k/\Gamma((k+1)/2)$, $V_k(r) = 2\pi^{(k+1)/2}r^{k+1}/(k+1)\Gamma((k+1)/2)$, where $\Gamma(s) = \int_0^\infty t^{s-1}e^{-t}\mathrm{d}t$.

1.3 Quantum Circuit Model

Before discussing the algorithm, let's make some basic assumptions about quantum computers (quantum circuit model) clear[1]:

- A quantum computer consists of a classical part and a quantum part: The classical part is unnecessary but can simplify many tasks.
- A quantum circuit operates on n qubits, so the state space is 2^n-dimensional complex Hilbert space.
- It is assumed that any computational basis state $|x_1, \cdots, x_n\rangle$ can be prepared in at most n steps.
- Gates can be applied to any subset of qubits as desired, and a universal family of gates can be implemented.
- Measurements may be performed in the computational basis of one or more of the qubits in the computer.

2 Quantum Search Algorithm

2.1 Principles

In the search algorithm, we have the $N = 2^n$ elements in the search space $S = \{0,1\}^n$ and the target element $T \subseteq S$ which has M elements. The output of the algorithm should be the state containing all the target elements, so we let

$$|\beta\rangle := \frac{1}{\sqrt{M}} \sum_{x \in T} |x\rangle \quad \text{and} \quad |\alpha\rangle := \frac{1}{\sqrt{N-M}} \sum_{x \notin T} |x\rangle . \tag{2.1.1}$$

And the input of the algorithm should be a trivial state, may as well let[2]

$$|\psi\rangle := H^{\otimes n} |0\rangle \tag{2.1.2}$$

which use $|0\rangle$ to refer to $|0\rangle^{\otimes n}$ for convenience. There is

$$\langle \alpha | \beta \rangle = 0 \quad \text{and} \quad |\psi\rangle = \sqrt{\frac{N-M}{N}} |\alpha\rangle + \sqrt{\frac{M}{N}} |\beta\rangle . \tag{2.1.3}$$

We need to find some operation makes $|\psi\rangle$ into $|\beta\rangle$. The operation of turning some non-trivial Angle is hard, so we consider the operation reflecting across some axis. Define two operations here, O is a reflection about $|\alpha\rangle$ and GO^{-1} is a reflection about $|\psi\rangle$.

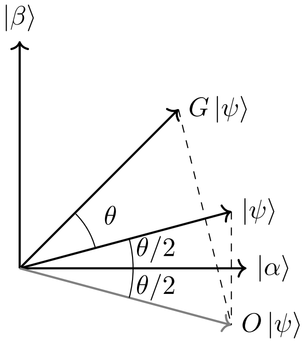


Figure 1

Let $\sin(\theta/2) = \sqrt{M/N}$, the angle between $|\psi\rangle$ and $|\alpha\rangle$ is $\theta/2$. As shown in the Figure 1, operation G gives us a rotation of θ, we call G the Grover iteration. So we just need applying $R := \lfloor \frac{\pi-\theta}{2\theta} \rceil$[3] times Grover iteration on $|\psi\rangle$ to approximate $|\beta\rangle$, if $M \ll N$ we have

$$G^k |\psi\rangle = \cos(\frac{2R+1}{2}\theta) |\alpha\rangle + \sin(\frac{2R+1}{2}\theta) |\beta\rangle \approx |\beta\rangle . \tag{2.1.4}$$

The angular error is at most $\theta/2 \approx \sqrt{M/N}$.

[1] This part is completely taken from the book[2].

[2] $|\psi\rangle$ is a commonly used state, called the equal superposition state.

[3] The actual function about $\lfloor x \rceil$ is to denote the integer closest to the real number x.

2.2 Oracle

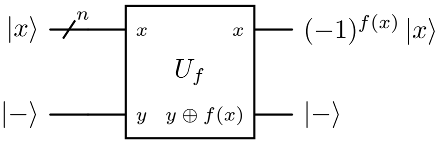
Let's construct the reflection about $|\alpha\rangle$, we call it oracle. We know that states in the space spanned by $|\alpha\rangle$ and $|\beta\rangle$ can be decomposed into

$$|x\rangle = |\beta\rangle \langle\beta|x\rangle + |\alpha\rangle \langle\alpha|x\rangle \tag{2.2.1}$$

and we wish

$$O\,|x\rangle = |\beta\rangle \langle\beta|x\rangle - |\alpha\rangle \langle\alpha|x\rangle \,. \tag{2.2.2}$$

But we don't know the exact form of $|\alpha\rangle$ and $|\beta\rangle$. To construct this, we need a special gate[1]



(2.2.3)

which gives us a operation $|x\rangle \xrightarrow{U_f} (-1)^{f(x)}\,|x\rangle$. Then we just need to construct the mapping

$$\omega(x) = \begin{cases} 1 & x \in T \\ 0 & x \notin T \end{cases} \tag{2.2.4}$$

such that U_ω satisfies the properties we need:

$$\begin{aligned} U_\omega\,|\beta\rangle &= |\beta\rangle \\ U_\omega\,|\alpha\rangle &= -\,|\alpha\rangle \,. \end{aligned} \tag{2.2.5}$$

So the U_ω is the Oracle.

2.3 Grover iteration

Then, let's construct the reflection about $|\psi\rangle$. Same as the oracle, we decompose states into

$$|x\rangle = |\psi_\perp\rangle \langle\psi_\perp|x\rangle + |\psi\rangle \langle\psi|x\rangle \tag{2.3.1}$$

which $|\psi_\perp\rangle = \sqrt{\frac{N-M}{N}}\,|\alpha\rangle - \sqrt{\frac{M}{N}}\,|\beta\rangle$, and we wish

$$GO^{-1}\,|x\rangle = |\psi_\perp\rangle \langle\psi_\perp|x\rangle - |\psi\rangle \langle\psi|x\rangle \,. \tag{2.3.2}$$

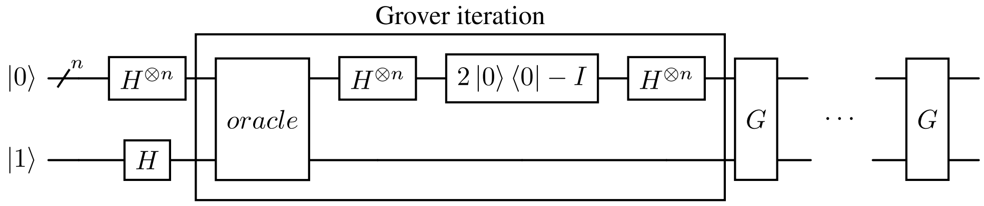
But this time, we know $|\psi\rangle$ specifically. Operation $2\,|\psi\rangle\langle\psi| - I$ is what we need which satisfies the properties:

$$\begin{aligned} (2\,|\psi\rangle\langle\psi| - I)\,|\psi\rangle &= -\,|\psi\rangle \\ (2\,|\psi\rangle\langle\psi| - I)\,|\psi_\perp\rangle &= |\psi_\perp\rangle \,. \end{aligned} \tag{2.3.3}$$

So we get the construction of the Grover iteration:

$$G = (2\,|\psi\rangle\langle\psi| - I)O. \tag{2.3.4}$$

After constructing each sub-operation, the circuit of the algorithm is given here[2]:



(2.3.5)

The ellipsis indicates that it is repeated R times.

[1] Note that the mapping $|x\rangle \xrightarrow{U_f} (-1)^{f(x)}\,|x\rangle$ holds only if the controlled qubit is $|-\rangle$.

[2] The oracle workspace are omitted here.

2.4 Performance

Each of the operations in the Grover iteration may be efficiently implemented on a quantum computer. So we just need to focus on the number of Grover iteration.

We already know, in order to rotate $|\psi\rangle$ near $|\beta\rangle$, we need to repeat the Grover iteration

$$R = \left\lfloor \frac{2\arccos\sqrt{M/N}}{\arcsin\sqrt{M/N}} \right\rceil \tag{2.4.1}$$

times. This is not intuitive enough, so let's sacrifice some precision to find a simpler expression. We know that $R \leq \lceil \pi/2\theta \rceil$, and if $M \leq N/2$ we have

$$\frac{\theta}{2} \geq \sin\frac{\theta}{2} = \sqrt{\frac{N}{M}}, \tag{2.4.2}$$

from which we obtain an elegant upper bound on the number of iterations required,

$$R \leq \left\lceil \frac{\pi}{4}\sqrt{\frac{N}{M}} \right\rceil \tag{2.4.3}$$

That is, $R = O(\sqrt{N/M})$ Grover iterations must be performed in order to obtain a solution to the search problem with high probability.[1]

If $M > N/2$, the quantum search algorithm is not necessary. We can just randomly pick an item from the search space, and then check that it is a solution using the oracle. This approach has a success probability at least one-half, and only requires one consultation with the oracle.

3 Bounds for Grover's algorithm

Show the proof of Theorem 8 and 9 in [1].

References

[1] Catalin Dohotaru and Peter Hoyer. Exact quantum lower bound for grover's problem. *Quantum Information & Computation*, 9(5):533–540, 2009.

[2] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, December 2010.

[1] The classical algorithm requires $O(N/M)$ calls to the oracle.