Quick notes on Grover's algorithm

Peter Brown

June 22, 2022

1 Grover's algorithm

Suppose you have a list of N items and you need to find the item in the list with a particular property. For the remainder of this we'll take $N = 2^n$ because we'll be considering running the algorithm on n qubits. I.e., you have some query function $f : \{0,1\}^N \to \{0,1\}$ that satisfies

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} = \mathbf{w} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Here the function is defined so the item we are looking for is the bitstring $\mathbf{w}$. In Grover's algorithm we'll use a phase oracle

$$U_{\mathbf{w}} |\mathbf{x}\rangle = (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle \ , \tag{2}$$

where $|\mathbf{x}\rangle = |x_0 x_1 \dots x_n\rangle = |x_0\rangle \otimes |x_1\rangle \otimes \cdots \otimes |x_n\rangle$ is an n-qubit state. You could implement this unitary by using a phase-kickback like in the Deutsch-Josza algorithm, but we can also just view the oracle as the diagonal unitary

$$U_{\mathbf{w}} = \begin{pmatrix} (-1)^{f(0\dots0)} & 0 & \cdots & 0 \\ 0 & (-1)^{f(0\dots1)} & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & (-1)^{f(1\dots1)} \end{pmatrix} \tag{3}$$

that is applied only to the first n qubits (no extra qubits). Effectively, it leaves all bitstrings alone except $|\mathbf{w}\rangle$ which is applies a (-1) phase to.

It feels like from the implementation perspective that we're cheating by defining this unitary oracle as it looks like we already know the answer. However, in a useful real world problem the idea would be to build the oracle out of a circuit that somehow checks that our input satisfies a desired property. For example, we might check that a hash of the input evaluates to a certain value, $\text{SHA}(\mathbf{x}) = \mathbf{y}$. We can then implement the oracle by building a circuit that will check the value of a hash and apply a phase if it is equal to $\mathbf{y}$. In this way we would end up with an oracle like $U_{\mathbf{w}}$ but we wouldn't a priori know the solution. Nevertheless, complexity theorists are still interested in query complexity of some oracle even if we don't know how to necessarily implement it in practice.

We define a couple more gates, let $R = 2|\mathbf{0}\rangle\langle\mathbf{0}| - I$ be an n-qubit unitary. This gate has the following action on the n-qubit computational basis states

$$R|\mathbf{x}\rangle = \begin{cases} |\mathbf{x}\rangle & \text{if } \mathbf{x} = \mathbf{0} \\ -|\mathbf{x}\rangle & \text{otherwise} \end{cases} \tag{4}$$

Now let $S = H^{\otimes n} R H^{\otimes n}$ where H is the Hadamard gate. By the definition of R we have that

$$S = 2|+\rangle\langle+|^{\otimes n} - I \, . \tag{5}$$

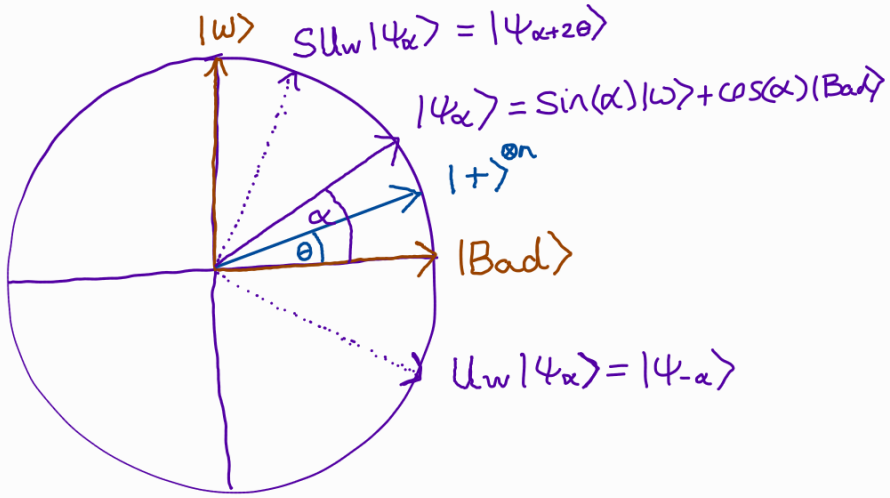
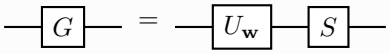


Figure 1: A geometric depiction of the action of the Grover G gate. The state of the system at some iteration can be thought of as a superposition $|\psi_\alpha\rangle = \sin(\varphi)\,|\mathbf{w}\rangle + \cos(\alpha)\,|Bad\rangle$. The action of the oracle gate $U_\mathbf{w}$ is then a reflection about the vector $|Bad\rangle$ and the S gate is a reflection about the vector $|+\rangle^{\otimes n}$.

Now we define a subcircuit which we call G which is just



(6)

i.e., as a matrix $G = S\,U_\mathbf{w}$.

Grover's algorithm circuit

The circuit used in Grover's algorithm is the following

(7)

where the G gate is repeated $O(\sqrt{N})$ times.

1.1 The geometric intuition

Recall that $H^{\otimes n}\,|\mathbf{0}\rangle = 2^{n/2}\sum_\mathbf{x}|\mathbf{x}\rangle$ is an equal superposition over all bitstrings. There is a single bitstring in this set that we actually want, namely $|\mathbf{w}\rangle$. Define the vector

$$|Bad\rangle = \frac{1}{\sqrt{2^n-1}}\sum_{\mathbf{x}\neq\mathbf{w}}|\mathbf{x}\rangle \tag{8}$$

to be an equal superposition of every bitstring that is not $|\mathbf{w}\rangle$. Notice that we have

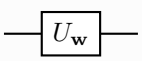
$$|+\rangle^{\otimes n} = \sin(\theta)\,|\mathbf{w}\rangle + \cos(\theta)\,|Bad\rangle\ . \tag{9}$$

We can think about these three vectors as lying in some circle (where orthogonality is denoted by perpendicular vectors), see Fig. 1.

Now consider any other vector

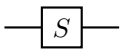
$$|\psi(\alpha)\rangle = \sin(\alpha)\,|\mathbf{w}\rangle + \cos(\alpha)\,|Bad\rangle \tag{10}$$

which lies elsewhere in this circle. The main geometrical intuition behind Grover's algorithm is the action of the gates on vectors in this circle. In particular,



reflects about the line $\alpha = 0$ (about the vector $|Bad\rangle$) (11)

and



reflects about the line $\alpha = \theta$ (about the vector $|+\rangle^{\otimes n}$) (12)

So overall the Gate G is the composition of the two reflections, there overall action on the state $|\psi\rangle$ is

$$G|\psi\rangle = \sin(\alpha + 2\theta)|\mathbf{w}\rangle + \cos(\alpha + 2\theta)|Bad\rangle \ . \tag{13}$$

So the G gate is rotating the vector anticlockwise by an angle 2θ each time we apply it.

In Grover's Algorithm we start by preparing the state

$$\sin(\theta)|\mathbf{w}\rangle + \cos(\theta)|Bad\rangle \tag{14}$$

and after applying G k-times we will have a state

$$\sin(\theta(2k+1))|\mathbf{w}\rangle + \cos(\theta(2k+1))|Bad\rangle \tag{15}$$

we can then choose k, the number of times we apply the gate G, to be such that $\sin(\theta(2k+1))^2 \approx 1$ and we'll have a large probability of measuring the correct solution! It can be shown that k scales like $O(\sqrt{N})$!