Lecture 5: Grover's Algorithm & Amplitude Amplification

QFT-based algorithms are responsible for the known super-polynomial quantum speedups over (known) classical algorithms.

However, they are (so far?) only useful in a limited range of applications (mostly number-theoretic in nature, like order-finding & factoring).

Grover's algorithm (and its generalisation, amplitude amplification) offer only a quadratic speedup over classical. One can even prove quadratic is as good as it gets for unstructured search problems (which is where these techniques apply).

However, these techniques apply far more widely, giving quadratic speedups over exhaustive search for any NP-problem, quadratic speedup for any probabilistic quantum algorithm with known success probability (and other neat tricks like boosting the latter from probabilistic to deterministic).

Amplitude Amplification

Just as the QFT + phase estimation was the quantum subroutine at the heart of order-finding & factoring, amplitude amplification is the quantum subroutine at the heart of Grover's algorithm.

(And just as building the factoring algorithm from phase estimation was historically backwards, as Shor's alg. came first and the phase estimation technique was distilled out of it later, so building Grover's algorithm from amplitude amplification is historically backwards: Grover [1996] came first, and the amplitude amplitude amplification technique distilled from it soon after [Brassard & Hoyer 1997].)

Problem (Amplitude amplification)

Input: State $|\psi\rangle \in \mathbb{C}^N$

Black-box unitaries $Z_\psi = \mathbb{1} - 2|\psi\rangle\langle\psi|$ and $Z_G = \mathbb{1} - 2\Pi_G$ on $\mathbb{C}^N$

Π_G = projector onto "good" subspace G

Output: State $|\psi_{out}\rangle$ s.t. $\|\Pi_G|\psi_{out}\rangle\|^2 \geq 1-\varepsilon$

i.e. high probability of measuring $|\psi_{out}\rangle$ to be in "good" subspace

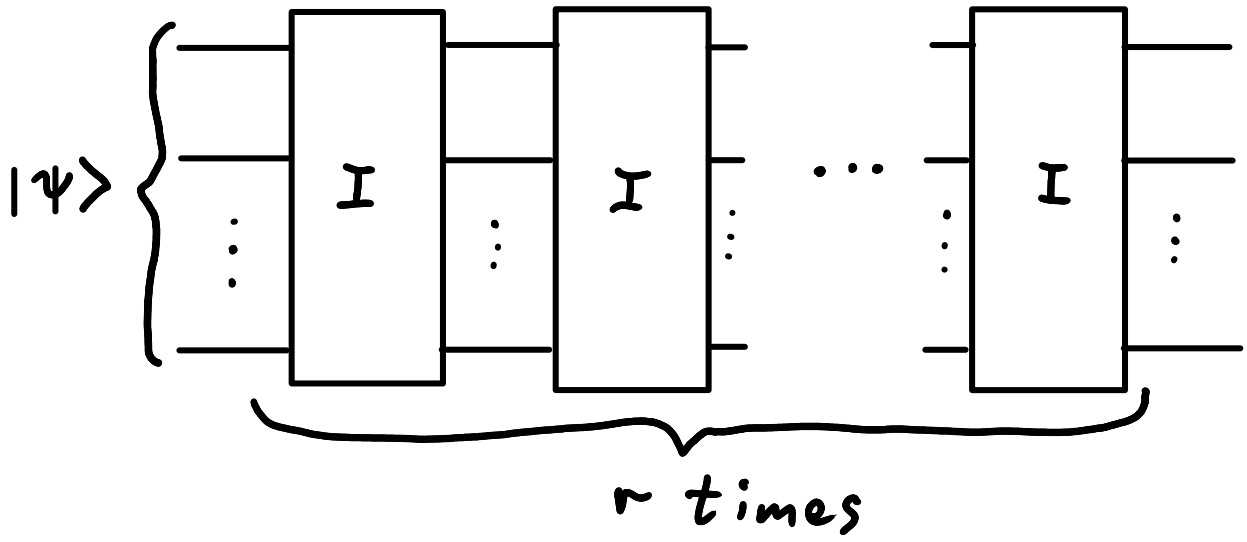
Algorithm

Algorithm is very simple:

Let $I = -Z_\psi Z_G$,

$r = \frac{\pi}{4\theta} - \frac{1}{2}$ (rounded to nearest integer)

where $\theta = \sin^{-1}\|\Pi_G|\psi\rangle\|$.



Analysis of this algorithm is, as usual, not quite so simple...

Analysis

Recall Def. (orthogonal complement):

$G^\perp := \text{span}\{|\psi^\perp\rangle : \forall |\psi\rangle \in G, \langle\psi^\perp|\psi\rangle = 0\}$

Claim

$\forall$ subspaces A, states $|\psi\rangle$, $|\psi\rangle$ can be decomposed as a linear combination $|\psi\rangle = s|\varphi\rangle + c|\varphi^\perp\rangle$ of states $|\varphi\rangle \in A$, $|\varphi^\perp\rangle \in A^\perp$ with <u>real</u> coefficients $c, s \in \mathbb{R}$.

Proof:

Pick an orthonormal basis $\{|\varphi_i\rangle\}$ & $\{|\varphi_j^\perp\rangle\}$ for A & $A^\perp$. Then $\{|\varphi_i\rangle\} \cup \{|\varphi_j^\perp\rangle\}$ forms an orthonormal basis for the whole space, so

$$|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle + \sum_j \beta_j |\varphi_j^\perp\rangle \qquad \alpha_i, \beta_j \in \mathbb{C}$$

$$= \sqrt{\sum_i |\alpha_i|^2}\left(\sum_i \frac{\alpha_i}{\sqrt{\sum_i |\alpha_i|^2}}|\varphi_i\rangle\right) + \sqrt{\sum_j |\beta_j|^2}\left(\sum_j \frac{\beta_j}{\sqrt{\sum_j |\beta_j|^2}}|\varphi_j\rangle\right)$$

$$= s|\varphi\rangle + c|\varphi^\perp\rangle$$

$$|\varphi\rangle \in A, \quad |\varphi^\perp\rangle \in A^\perp$$

$$s = \sqrt{\sum_i |\alpha_i|^2}, \quad c = \sqrt{\sum_j |\beta_j|^2} \in \mathbb{R}.$$

Note: $s|\varphi\rangle = \pi_G|\psi\rangle$, $c|\varphi^\perp\rangle = \pi_G^\perp|\psi\rangle$.

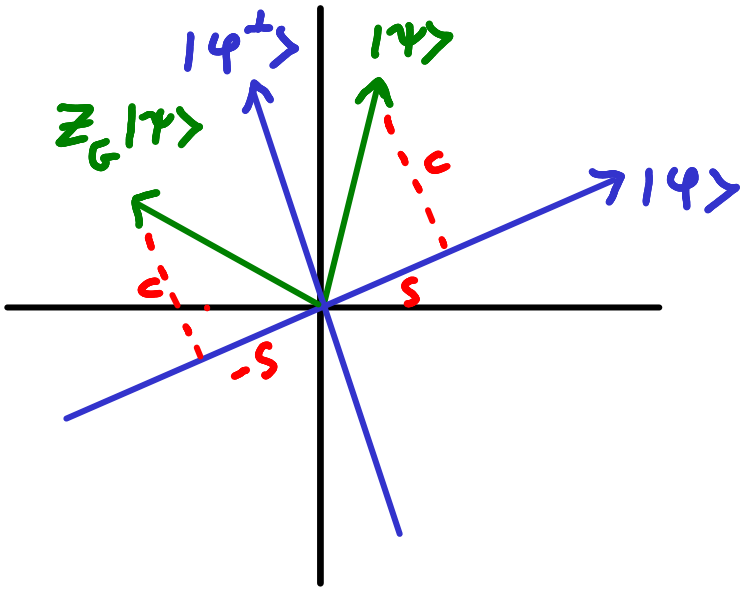
Can write $c = \cos\theta$, $s = \sin\theta$ since $c^2 + s^2 = 1$.

What does the operator $Z_G = \mathbb{1} - 2\Pi_G$ do?

$$Z_G |\psi\rangle = (\mathbb{1} - 2\Pi_G)(s|\varphi\rangle + c|\varphi^\perp\rangle$$

$$= (s - 2s)|\varphi\rangle + c|\varphi^\perp\rangle$$

$$= -s|\varphi\rangle + c|\varphi^\perp\rangle$$



Z_G is a <u>reflection</u> in the hyperplane $G^\perp$ orthogonal to G.

Similarly, Z_ψ is a reflection in the hyperplane orthogonal to $|\psi\rangle$.

$-Z_G$ is a reflection in G instead of $G^\perp$ (inverts sign of $|\varphi^\perp\rangle$ coeffficient instead of $|\varphi\rangle$, above).

What does $I = -Z_\psi Z_G$ do?

By claim (above), can write $|\psi\rangle$ as

$$|\psi\rangle = s|\varphi\rangle + c|\varphi^\perp\rangle$$

where $|\varphi\rangle \in G$, $|\varphi^\perp\rangle \in G^\perp$, $c = \cos\theta$, $s = \sin\theta$.

Then

$$I|\varphi\rangle = -Z_\psi Z_G |\varphi\rangle$$

$$= \left[\mathbb{1} - 2(s|\varphi\rangle + c|\varphi^\perp\rangle)(s\langle\varphi| + c\langle\varphi^\perp|)\right]|\varphi\rangle$$

$$= \cos 2\theta\, |\varphi\rangle - \sin 2\theta\, |\varphi^\perp\rangle \qquad \langle\varphi|\varphi^\perp\rangle = 0$$

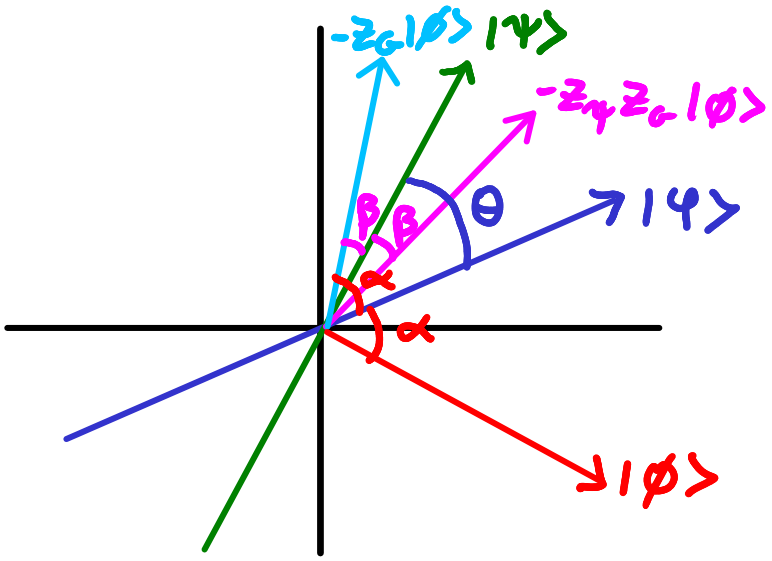
$$-Z_\psi Z_G |\varphi^\perp\rangle = \sin 2\theta\, |\varphi\rangle + \cos 2\theta\, |\varphi^\perp\rangle$$

So in the subspace spanned by

$$\{|\varphi\rangle = \Pi_G|\psi\rangle,\quad |\varphi^\perp\rangle = \Pi_G^\perp|\psi\rangle\}$$

$$I = -Z_\psi Z_G = \begin{pmatrix} \cos 2\theta & -\sin 2\theta \\ \sin 2\theta & \cos 2\theta \end{pmatrix} = \text{rotation by } 2\theta$$

where $\theta = \sin^{-1} \| \Pi_G |\varphi\rangle \|$:



$$|\psi_{out}\rangle = I^r |\psi\rangle$$

$$= \sin((2r+1)\theta)\,|q\rangle + \cos((2r+1)\theta)\,|q^\perp\rangle$$

rotate r times by angle 2θ
vector initially at angle θ.

$$\| \Pi_G |\psi_{out}\rangle \|^2 = \sin^2((2r+1)\theta)$$

maximized for $(2r+1)\theta = \frac{\pi}{2}$

$\rightarrow$ Take r = nearest integer to $\frac{\pi}{4\theta} - \frac{1}{2} = O\left(\frac{1}{\theta}\right)$

Let $r = \left(\frac{\pi}{4\theta} - \frac{1}{2}\right) + c$, $c < 1$

Grover speedup!

$$\| \Pi_G |\psi_{out}\rangle \|^2 = \sin^2\left(\frac{\pi}{2} + 2c\theta\right)$$

$$= \cos^2(2c\theta) = 1 - O(\theta^2)$$

Grover's Algorithm

Problem (Unstructured Search)

Input: Black-box function $f: \{0,1\}^n \rightarrow \{0,1\}$

Promise: $f(x) = 1$ on exactly k inputs x
i.e. $|\{x : f(x) = 1\}| = k$

Output: x s.t. $f(x) = 1$.

As usual, quantum black-box is
$U_f |x\rangle |y\rangle = |x\rangle |y \oplus f(x)\rangle$.

Think of f as identifying "good" x values. Task is to find a good x (search). Black-box nature of f models the fact we have no additional info on what values of x are likely to be good, other than what we get by querying f (unstructured).

Thinking of f as a "database" storing values that can be looked up is misleading (even if some of the literature discusses it in these tems!); actually implementing any such database would lose the quadratic advantage.

Grover is about searching in an abstract solution space.

E.g. consider the class NP. By definition, any NP problem has a verifier that outputs 1 on valid solutions
⟶ can take verifier as f in unstructured search. Since verifier for NP problems are poly-time (by def.), have efficient implementation of f in this case ⟶ de-oraclised problem.

However, # solutions k not usually known in this case (see later).

Classical lower bound

For once, proving a lower bound on the number of calls to the oracle f required to identify x with high probability is straightforward.

Theorem

Any classical algorithm solving Unstructured Search with success probability $\geq 1-\varepsilon$ requires $O((1-\varepsilon)\frac{N}{k})$ queries to f.

Proof

f can be any function satisfying promise, so have uniform prior on "good" x (i.e. $f(x)=1$).

Wlog each query to f is on distinct x (otherwise can improve algorithm by skipping duplicate queries and using already-known result).

After r queries, if haven't found $f(x)=1$ then best we can do is guess random x from remaining possibilities

$$\Pr(\text{fail using } r \text{ queries}) = \binom{N-r-1}{k} \Big/ \binom{N}{k}$$

$$= O\left(\frac{N^k - kr N^{k-1}}{N^k}\right) = O\left(1-\frac{kr}{N}\right) \leq \varepsilon$$

$$\Rightarrow r = O\left((1-\varepsilon)\frac{N}{k}\right). \quad \square$$

Quantum algorithm

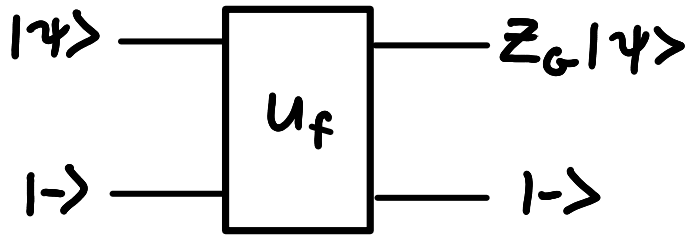
Let $|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{N-1} |x\rangle$, $N = 2^n$

$$Z_0 |x\rangle = \begin{cases} -|x\rangle & \text{if } x = 0^n \\ |x\rangle & \text{otherwise} \end{cases}$$

$$Z_\psi = H^{\otimes n} Z_0 H^{\otimes n}$$

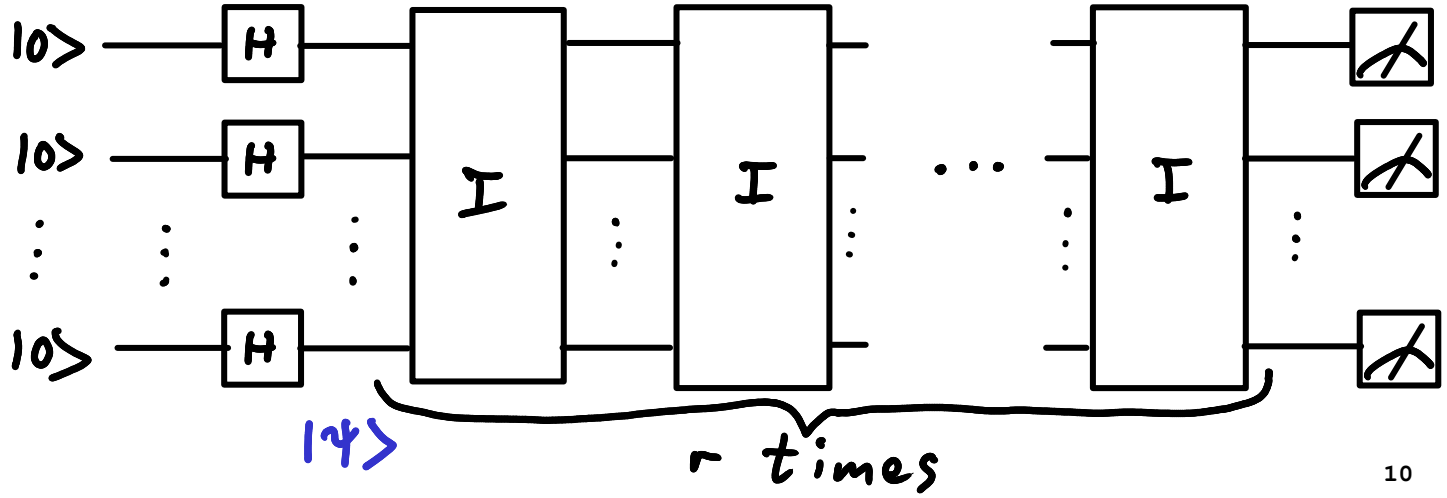
$$Z_G |x\rangle = (-1)^{f(x)} |x\rangle, \quad x \in \{0,1\}^n$$

Note that Z_G can be implemented using one call to U_f using phase-kickback:



Z_0 can be implemented efficienty by exactly the same method applied to the circuit for reversibly computing $g: x \to NOT(AND(x_0, x_1, \ldots, x_{n-1}))$.

Grover's algorithm is amplitude amplication applied to this particular $|\psi\rangle$, Z_G, Z_ψ:



Analysis

Note $Z_0 = \mathbb{1} - 2|0^n\rangle\langle 0^n|$

since $Z_0|x\rangle = |x\rangle - 2\langle 0^n|x\rangle |0^n\rangle$

$$= \begin{cases} -|0^n\rangle & \text{if } x = 0^n \\ |x\rangle & \text{otherwise} \end{cases}$$

$$Z_\psi = H^{\otimes n} Z_0 H^{\otimes n} = \mathbb{1} - 2|\psi\rangle\langle\psi|$$

where $|\psi\rangle = H^{\otimes n}|0^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{N-1} |x\rangle.$

$$Z_G = \sum_{x: f(x)=0} |x\rangle\langle x| - \sum_{x: f(x)=1} |x\rangle\langle x| = \mathbb{1} - 2\Pi_G$$

where $G = \text{span}\{|x\rangle : f(x)=1\}.$

$$\theta = \sin^{-1} \| \Pi_G |\psi\rangle \| = \sin^{-1} \| \Pi_G \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \|$$

$$= \sin^{-1}\left(\frac{1}{\sqrt{N}} \left\| \sum_{x: f(x)=1} |x\rangle \right\| \right) = \sin^{-1} \sqrt{\frac{k}{N}}$$

$$\approx \sqrt{\frac{k}{N}} \quad \text{for } k \text{ const, } N \text{ growing}$$

$\Rightarrow r = O\left(\frac{1}{\theta}\right) = O\left(\sqrt{\frac{N}{k}}\right)$ in amplitude amplification alg.

"# iterations

$\rightarrow$ $\Pr(\text{measure } x : f(x)=1) = \| \Pi_G |\psi_{out}\rangle \|^2$

$= 1 - O\left(\frac{k}{N}\right)$ in time $O\left(\sqrt{\frac{N}{k}}\right)$

A quadratic speedup over classical $O\left(\frac{N}{k}\right)$.

Quantum Lower-Bound

Theorem

Any quantum algorithm for unstructured search requires $O(\sqrt{N})$ queries to U_f

Proof: Not too difficult, but beyond the scope of this course. Interestingly the lower bound and the algorithm achieving it were discovered completely independently at around the same time [Bennett, Brassard, Bernstein, Vazirani 1997].

Quadratic Speedup of many quantum algorithms

Consider any quantum algorithm for a decision problem (output 0 or 1) that does not use measurements except final measurement of output qubit.

Let success probability = p

$\Rightarrow$ output state before final measurement

$$|\Psi_{out}\rangle = \sqrt{p}\,|1\rangle|q\rangle + \sqrt{1-p}\,|0\rangle|\phi\rangle$$

$\rightarrow$ Need $O(\frac{1}{p})$ repetitions to get right answer with probability $1-\delta$.

Apply amplitude amplification with $\Pi_G = 11\times 11 \otimes \mathbb{1}$

$\rightarrow$ Amplify success probability to $1-\delta$ in time $O\left(\sqrt{\frac{1}{p}}\right)$.

Unknown k

In the above algorithm, the number of Grover iterations, r, depends on the number of "good" x's, k. Setting r to the wrong value will either over- or under-rotate, and the success probability $\sin\theta$ falls off very quickly. So k must be known in advance.

Not the case for many applications, e.g. to NP problems. (N.B. For NP there is a randomised poly-time reduction to unique-SAT, but the poly-time reduction will destroy the quadratic advantage.)

Simple strategy:

Choose k uniformly at random in $\{1, \ldots, \sqrt{N}+1\}$, run Grover for $\sqrt{\frac{N}{k}}$ iterations, check if $f(x) = 1$, rinse, repeat until success.

$\Pr(\text{succeed in one Grover run}) \geq \frac{1}{4}$ (Exercise)

$\Pr(\text{succeed within } n \text{ Grover runs}) \geq 1 - \left(\frac{1}{4}\right)^n$

$\Rightarrow$ For success probability $\geq 1-\delta$, need $n = O(\log \frac{1}{\delta})$

$\rightarrow$ total run-time $O(\sqrt{N} \log \frac{1}{\delta})$.

There are more sophisticated strategies that do somewhat better.

Exact Grover search

Consider special case $\frac{k}{N} = \frac{1}{4}$:

$$\sin\theta = \| \pi_G |\psi\rangle \| = \sqrt{\frac{1}{4}}$$

$$\Rightarrow \theta = \sin^{-1}\left(\frac{1}{2}\right) = \frac{\pi}{6}$$

$$\text{Optimal } r = \frac{\pi}{4\theta} - \frac{1}{2} = 1$$

$$\Pr(\text{succeed after } r = 1 \text{ iterations}) = 1 \; !$$

I.e. when searching for 1 item out of 4, a single Grover iteration is guaranteed to find matching item deterministically.

In fact, Grover's algorithm can be made exact for <u>any</u> known value of $\frac{k}{N}$.

(<u>Exercise</u>)

There have been many other improvements to Grover's algorithm since the original, of which we list a few without proof:

Oblivious Amplitude Amplification
[Grover 2005]
Works even when $|\psi\rangle$ is unknown.

Fixed-point Amplitude Amplification
[Yoder, Low, Chuang 2014]

Overlap with "good" subspace increases monotonically with # iterations r
I.e. increasing r always increases success probability.

Oblivious Fixed-point Amplitude Amplification
[Guerreschi 2018]

Combines both of the above.

ALL the above results retain the quadratic speedup.

There also many other applications of Grover search & amplitude amplification, of which we list none! (See reseach literature.)