Contents

1. Hidden subgroup problem

1.1. Review of Shor's algorithm

Problem 1.1 (Factoring)

Input a positive integer N.

Promise N is composite.

Task Find a non-trivial factor of N in $O(\text{poly}(n))$ time, where $n = \log N$.

Definition 1.2 An **efficient problem** is one that can be solved in polynomial time.

Remark 1.3 Clasically, the best known factoring algorithm runs in $e^{O\left(n^{1/3}(\log n)^{2/3}\right)}$. Shor's algorithm (quantum) runs in $O(n^3)$ by converting factoring into period finding:

- Given input N, choose $a < N$ which is coprime to N.
- Define $f : \mathbb{Z} \to \mathbb{Z}/N$, $f(x) = a^x \bmod N$. f is periodic with period r (the order of $a \bmod N$), i.e. $f(x + r) = f(x)$ for all $x \in \mathbb{Z}$. Finding r allows us to factor N.

1.2. Period finding

Problem 1.4 (Periodicity Determination Problem)

Input An oracle for a function $f : \mathbb{Z}/M \to \mathbb{Z}/N$.

Promise

- f is periodic with period $r < M$ (i.e. $\forall x \in \mathbb{Z}/M$, $f(x + r) = f(x)$), and
- f is injective in each period (i.e. if $0 \leq x < y < r$, then $f(x) \neq f(y)$).

Task Determine the period r.

Remark 1.5 Solving the periodicity determination problem classically requires takes time $O(\sqrt{M})$.

Definition 1.6 Let $f : \mathbb{Z}/M \to \mathbb{Z}/N$. Let H_M and H_N be quantum state spaces with orthonormal state bases $\{|i\rangle : i \in \mathbb{Z}/N\}$ and $\{|j\rangle : j \in \mathbb{Z}/M\}$. Define the unitary **quantum oracle** for f by U_f by

$$U_f|x\rangle|z\rangle = |x\rangle|z + f(x)\rangle.$$

The first register $|x\rangle$ is the **input register**, the last register $|z\rangle$ is the **output register**.

Definition 1.7 The **quantum query complexity** of an algorithm is the number of times it queries f (i.e. uses U_f).

Definition 1.8 The **quantum Fourier transform** over $\mathbb{Z}/M$ is the unitary QFT defined by its action on the computational basis:

$$\text{QFT}|x\rangle = \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} \omega^{xy}|y\rangle,$$

where $\omega = e^{2\pi i/M}$ is an M-th root of unity. Note that QFT requires only $O((\log M)^2)$ gates to implement, whereas a general $M \times M$ unitary requires $O(4^M/M)$ elementary gates.

Lemma 1.9 Let $\alpha = e^{2\pi i y/M}$. Then

$$\sum_{j=0}^{k-1} \alpha^j = \begin{cases} \frac{1-\alpha^k}{1-\alpha} = 0 & \text{if } \alpha \neq 1 \text{ i.e. } M \nmid y \\ k & \text{if } \alpha = 1 \text{ i.e. } M \mid y \end{cases}.$$

Proof (Hints). Trivial. □

Proof. The sum is a geometric series with common ratio α. □

Lemma 1.10 (Boosting success probability) If a process succeeds with probability p on one trial, then

$$\Pr(\text{at least one success in } t \text{ trials}) = 1 - (1-p)^t > 1 - \delta$$

for $t = \frac{\log(1/d)}{p}$.

Proof (Hints). Trivial. □

Proof. Trivial. □

Theorem 1.11 (Co-primality Theorem) The number of integers less than r that are coprime to r is $O(r/\log\log r)$.

Algorithm 1.12 (Quantum Period Finding) The algorithm solves the Periodicity Determination Problem: Let $f : \mathbb{Z}/M \to \mathbb{Z}/N$ be periodic with period $r < M$ and one-to-one in each period. Let $A = \frac{M}{r}$ be the number of periods. We work over the state space $H_M \otimes H_N$.

1. Construct the state $\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle|0\rangle$ and query U_f on it.
2. Measure second register in computational basis and discard the second register.
3. Apply the quantum Fourier transform to the input state.
4. Measure the input state, yielding outcome c.
5. Compute the denominator r_0 of the simplified fraction $\frac{c}{M}$.
6. Repeat the previous steps $O(\log\log r) = O(\log\log M) = O(\log m)$ times, halting if at any iteration, $f(0) = f(r_0)$.

Theorem 1.13 (Correctness of Quantum Period Finding Algorithm) When repeated, $O(\log\log r) = O(\log\log M)$ times, the quantum period finding algorithm obtains the correct value of r with high probability.

Proof. After querying U_f, we have the state $\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle|f(i)\rangle$. Upon measuring the second register in the computational basis, the input state collapses to $|\text{per}\rangle = \frac{1}{\sqrt{A}} \sum_{j=0}^{A-1} |x_0 + jr\rangle$, where $f(x_0) = y$ and $0 \leq x_0 < r$. Applying the quantum Fourier transform to $|\text{per}\rangle$ then gives Quantum Fourier Transform to $|\text{per}\rangle$:

$$\begin{aligned}\mathrm{QFT}|\mathrm{per}\rangle &= \frac{1}{\sqrt{M}}\sum_{y=0}^{M-1}\frac{1}{\sqrt{A}}\sum_{j=0}^{A-1}\omega^{(x_0+jr)y}|y\rangle \\ &= \frac{1}{\sqrt{MA}}\sum_{y=0}^{M-1}\omega^{x_0 y}\sum_{j=0}^{A-1}\omega^{jry}|y\rangle \\ &= \sqrt{\frac{A}{M}}\sum_{k=0}^{r-1}\omega^{x_0 kM/r}|kM/r\rangle\end{aligned}$$

Importantly, now the outcomes and probabilities are independent of x_0, so carry useful information about r. TODO add diagram showing amplitudes for this state. The outcome after the measuring the input state is $c = k_0 M/r$ for some $0 \leq k_0 < r$ (so $c/M = k_0/r$). If k_0 is coprime to r, then the denominator r_0 of the simplified fraction $\frac{c}{M}$ is equal to r. By the coprimality theorem, the probability that k_0 is coprime to r is $O(1/\log\log r)$. Checking if $f(0) = f(r_0)$ tells us if $r_0 = r$, since f is periodic and one-to-one in each period, and $r_0 \leq r$. □

1.3. Analysis of QFT part of period finding algorithm

Notation 1.14 For $R = \{0, r, ..., (A-1)r\} \subseteq \mathbb{Z}/M$ ($Ar = M$), write $|R\rangle$ for the uniform superposition of all computational basis states in R:

$$|R\rangle = \frac{1}{\sqrt{A}}\sum_{k=0}^{A-1}|kr\rangle.$$

Definition 1.15 For each $x_0 \in \mathbb{Z}/M$, define the lienar map by its action on the computational basis states:

$$\begin{aligned}U(x_0) : H_M &\to H_M, \\ |k\rangle &\mapsto |x_0 + k\rangle.\end{aligned}$$

Definition 1.16 Note that since $(\mathbb{Z}/M, +)$ is abelian, all $U(x_i)$ commute: $U(x_1)U(x_2) = U(x_1 + x_2) = U(x_2)U(x_1)$. Hence, they have a simultaneous basis of eigenvectors $\{|\chi_k\rangle : k \in \mathbb{Z}/M\}$, i.e. for all $k, x_0 \in \mathbb{Z}/M$, $U(x_0)|\chi_k\rangle = w(x_0, k)|\chi_k\rangle$, where $|w(x_0, k)| = 1$. The $|\chi_k\rangle$ are called **shift-invariant states** and form an orthonormal basis for H_M. The $|\chi_k\rangle$ are given explicitly by

$$|\chi_k\rangle = \frac{1}{\sqrt{M}}\sum_{\ell=0}^{M-1} e^{-2\pi i k\ell/M}|\ell\rangle.$$

Proposition 1.17 The explicit definition of the $|\chi_k\rangle$ indeed satisfies the property $\forall k, x_0 \in \mathbb{Z}/M$, $U(x_0)|\chi_k\rangle = w(x_0, k)|\chi_k\rangle$, and we have $w(x_0, k) = \omega^{kx_0}$, where $\omega = e^{2\pi i/M}$.

Proof (Hints). Straightforward. □

Proof. We have that

$$\begin{aligned} U(x_0)|\chi_k\rangle &= \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} e^{-2\pi i k\ell/M} |x_0 + \ell\rangle \\ &= \frac{1}{\sqrt{M}} \sum_{\tilde{l}=0}^{M-1} e^{-2\pi i(\tilde{l}-x_0)k/M} |\tilde{l}\rangle \\ &= e^{2\pi i k x_0/M} |\chi_k\rangle \\ &=: w(x_0,k)|\chi_k\rangle \end{aligned}$$

□

Remark 1.18 Let $U : H_M \to H_M$ be the unitary mapping the shift-invariant basis to the computational basis: $U : |\chi_k\rangle \mapsto |k\rangle$. The matrix representation of U^{-1} with respect to the computational basis has entries

$$\left(U^{-1}\right)_{jk} = \langle j|U^{-1}|k\rangle = \langle j|\chi_k\rangle = \frac{1}{\sqrt{M}} e^{-2\pi i jk/M}$$

So the matrix representation of U with respect to the same basis has entries $U_{kj} = \overline{\left(U^{-1}\right)_{jk}} = \frac{1}{\sqrt{M}} e^{2\pi i jk/M}$. Hence, we have

$$U|k\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} e^{2\pi i jk/M} |j\rangle,$$

and so U is precisely the QFT mod M.

1.4. The hidden subgroup problem (HSP)

Problem 1.19 (Discrete Logarithm Problem (DLP))

Input $g, x \in G$ for an abelian group G.

Promise g is a generator of G.

Task Find $\log_g x$, i.e. find $L \in \mathbb{Z}/|G|$ such that $x = g^L$.

Notation 1.20 Write $[n]$ for $\{1, ..., n\}$. Write e.g. ij for the set $\{i, j\}$.

Definition 1.21 Let $\Gamma_1 = ([n], E_1)$ and $\Gamma_2 = ([n], E_2)$ be (undirected) graphs. Γ_1 and Γ_2 are **isomorphic** if there exists a permutation $\pi \in S_n$ such that for all $1 \leq i, j < n$, $ij \in E_1$ iff $\pi(i)\pi(j) \in E_2$.

Definition 1.22 Let $\Gamma = ([n], E)$ be a graph. The **automorphism group** of Γ is

$$\mathrm{Aut}(\Gamma) = \{\pi \in S_n : ij \in E \text{ iff } \pi(i)\pi(j) \in E \quad \forall i, j \in [n]\}.$$

$\mathrm{Aut}(\Gamma)$ is a subgroup of S_n, and $\pi \in \mathrm{Aut}(\Gamma)$ iff π leaves Γ invariant as a labelled graph.

Definition 1.23 The **adjacency matrix** of a graph $\Gamma = (V, E)$ is the $n \times n$ matrix M_A defined by its entries:

$$\left(M_A\right)_{ij} := \begin{cases} 1 & \text{if } ij \in E \\ 0 & \text{otherwise} \end{cases}.$$

Problem 1.24 (Graph Isomorphism Problem)

Input Adjacency matrices M_1 and M_2 of graphs $\Gamma_1 = ([n], E_1)$ and $\Gamma_2 = ([n], E_2)$.

Task Determine whether Γ_1 and Γ_2 are isomorphic.

Remark 1.25 The best known classical algorithm for solving the graph isomorphism problem has quasi-polynomial time complexity $n^{O((\log n)^2)}$.

Problem 1.26 (Hidden Subgroup Problem (HSP)) Let G be a finite group.

Input An oracle for a function $f : G \to X$.

Promise There is a subgroup $K < G$ such that:

1. f is constant on the (left) cosets of K in G.
2. f takes a different value on each coset.

Task Determine K.

Remark 1.27

- To find K, we either find a generating set for K, or sample uniformly random elements from K.
- We want to determine K with high probability in $O(\text{poly}\log|G|)$ queries. Using $O(|G|)$ queries is easy, as we just query all values $f(g)$ and find the "level sets" (sets where f is constant).

Example 1.28 The following problems are special cases of HSP:

- The Periodicity Determination Problem: $G = \mathbb{Z}/M$, $K = \langle r \rangle = \{0, r, ..., (A-1)r\}$. The cosets are $x_0 + K = \{x_0, x_0 + r, ..., x_0 + (A-1)r\}$ for each $0 \leq x_0 < r$.
- The DLP on $(\mathbb{Z}/p)^\times$: let $f : \mathbb{Z}/(p-1) \times \mathbb{Z}/(p-1) \to (\mathbb{Z}/p)^\times$ be defined by $f(a, b) = g^a x^{-b} = g^{a-Lb}$. $G = \mathbb{Z}/(p-1) \times \mathbb{Z}/(p-1)$, the hidden subgroup is $K = \{\lambda(L, 1) : \lambda \in \mathbb{Z}/(p-1)\}$. (Note that if we know K, we can pick any $(c, d) = (\lambda L, \lambda) \in G$ and compute $L = \frac{c}{d}$ to find L.)
- The Graph Isomorphism Problem: $G = S_n$, hidden subgroup is $K = \text{Aut}(G)$. Let $f_\Gamma : S_n \to X$ where X is set of adjacency matrices of labelled graphs on $[n]$, defined by $f_\Gamma(\pi) = \pi(A)$. Note $|S_n| = |G| = n!$, so $\log|G| \approx n \log n$, so $O(\text{poly}\log|G|) = O(\text{poly}\, n)$.

Definition 1.29 An **irreducible representation (irrep)** of a finite abelian group G is a homomorphism $\chi : G \to \mathbb{C}^\times$.

Theorem 1.30

- Let $\chi : G \to \mathbb{C}^\times$ be an irrep. For all $g \in G$, $\chi(g)$ is a $|G|$-th root of unity.
- There are always exactly $|G|$ distinct irreps. In particular, we can label each irrep uniquely by some $g \in G$.

Theorem 1.31 (Schur's Lemma) Let χ_i and χ_j be irreps of G. Then

$$\frac{1}{|G|} \sum_{g \in G} \chi_i(g) \overline{\chi_j(g)} = \delta_{ij}.$$

Example 1.32 $\chi_0 : G \to \mathbb{C}^\times$, $\chi_0(g) = 1$ is the **trivial irrep**. Note that for any $\chi_i \neq \chi_0$, $\sum_{g\in G} \chi_i(g) = 0$ by Schur's lemma.

Definition 1.33 For finite abelian G, we define the **shift operators** on $H_{|G|}$ for each $k \in G$ by

$$\begin{aligned} U(k) : H_{|G|} &\to H_{|G|}, \\ |g\rangle &\mapsto |k+g\rangle. \end{aligned}$$

Note that since G is abelian, the $U(k)$ commute: $U(k)U(l) = U(l)U(k)$ for all $k, l \in G$. Hence, they have simultaneous eigenstates, which gives an orthonormal basis for $H_{|G|}$.

Proposition 1.34 For each $k \in G$, consider the state

$$|\chi_k\rangle = \frac{1}{\sqrt{|G|}} \sum_{g\in G} \overline{\chi_k(g)}|g\rangle.$$

The $|\chi_k\rangle$ are shift-invariant (invariant up to a phase under the action of all $U(g)$, $g \in G$).

Proof (Hints). Straightforward. □

Proof. Since χ_k is a homomorphism, we have $\overline{\chi_k(g)} = \chi_k(-g)$. Now

$$\begin{aligned} U(g_0)|\chi_k\rangle &= \frac{1}{\sqrt{|G|}} \sum_{g\in G} \overline{\chi_k(g)}|g_0 + g\rangle \\ &= \frac{1}{\sqrt{|G|}} \sum_{g'\in G} \overline{\chi_k(g' - g_0)}|g'\rangle \\ &= \frac{1}{\sqrt{|G|}} \sum_{g'\in G} \overline{\chi_k(g')}\chi_k(g_0)|g'\rangle \\ &= \chi_k(g_0)|\chi_k\rangle. \end{aligned}$$

□

Definition 1.35 The **quantum Fourier transform (QFT)** on $H_{|G|}$ is the unitary implementing the change of basis from the shift-invariant states $\{|\chi_g\rangle : g \in G\}$ to the computational basis $\{|g\rangle : g \in G\}$.

Note that $\mathrm{QFT}^{-1}|g\rangle = |\chi_g\rangle$. So $\left(\mathrm{QFT}^{-1}\right)_{kg} = \langle k|\chi_g\rangle = \frac{1}{\sqrt{|G|}}\overline{\chi_g(k)}$, so $\mathrm{QFT}_{kg} = \frac{1}{\sqrt{|G|}}\chi_k(g)$. So the explicit form is

$$\mathrm{QFT}|g\rangle = \frac{1}{\sqrt{|G|}} \sum_{k\in G} \chi_k(g)|k\rangle.$$

Example 1.36

- For $G = \mathbb{Z}/M$, we can check that $\chi_a(b) = e^{2\pi i ab/M}$ are irreps. So the irreps of $\mathbb{Z}/M$ are naturally labelled by $a \in \mathbb{Z}/M$ and this gives the usual $\mathrm{QFT} \bmod M$ as defined earlier.

- Similarly, for $G = \mathbb{Z}/(M_1) \times \cdots \times \mathbb{Z}/(M_r)$, $\chi_g(h) = e^{2\pi i(g_1h_1/M_1+\cdots+g_rh_r/M_r)}$ are the irreps.

Algorithm 1.37 (Quantum HSP solver for finite abelian G) The algorithm solves the HSP for finite abelian G. We work in the state space $H_{|G|} \otimes H_{|X|}$.

1. Prepare the uniform superposition state
$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle|0\rangle$$
and query U_f on it.
2. Measure the output register, then discard this register.
3. Apply QFT mod $|G|$ to the input register, then measure this register.
4. Repeat the above steps $O(\log|G|)$ times.

Theorem 1.38 (Correctness of Quantum HSP Solver) The quantum HSP solver algorithm solves the HSP for finite abelian groups with high probability.

Proof. Querying U_f on the state gives
$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle|f(g)\rangle$$
Upon measurement of the output register, we obtain a uniformly random value $f(g_0)$ from $f(G)$, and the state collapses to a **coset state**
$$|g_0 + K\rangle = \frac{1}{\sqrt{|K|}} \sum_{k \in K} |g_0 + k\rangle.$$
We have $|K\rangle = \sum_{g \in G} a_g|\chi_g\rangle$, so $|g_0 + K\rangle = U(g_0)|K\rangle = \sum_{g \in G} a_g\chi_g(g_0)|\chi_g\rangle$. So applying QFT to the input state gives $\sum_{g \in G} a_g\chi_g(g_0)|g\rangle$, so the probability of measuring outcome k is $|a_k\chi_k(g_0)|^2 = |a_k|^2$. Now
$$\begin{aligned}\mathrm{QFT}|K\rangle &= \frac{1}{\sqrt{|K|}} \sum_{k \in K} \mathrm{QFT}|k\rangle \\ &= \frac{1}{\sqrt{|G||K|}} \sum_{g \in G} \left(\sum_{k \in K} \chi_g(k) \right) |g\rangle\end{aligned}$$
Note that irreps of G restricted to K are irreps of K. The trivial irrep $\chi_0 : G \to \mathbb{C}$ remains the trivial irrep χ_0 for K. But there may be other irreps that become the trivial irrep on restriction to K. Hence
$$\sum_{k \in K} \chi_g(k) = \begin{cases} |K| & \text{if } \chi_g|_K = \chi_0|_K \\ 0 & \text{otherwise} \end{cases}$$
Hence

$$\mathrm{QFT}|K\rangle = \sqrt{\frac{|K|}{|G|}} \sum_{\substack{g\in G \\ \chi_g|_K = \chi_0|_K}} |g\rangle$$

and measuring in the computational basis on this state yields random $g \in G$ such that $\forall k \in K$, $\chi_g(k) = 1$.

If K has generators $k_1, ..., k_m$ (note that for an arbitrary group, we have $m = O(\log|G|)$), then we have a set of equations $\chi_g(k_i) = 1$ for all $i \in [m]$. We can show that if $O(\log|G|)$ such g are drawn uniformly at random, then with probability at least $2/3$, we have enough equations to determine $k_1, ..., k_m$. □

Example 1.39 Let $G = \mathbb{Z}/M_1 \times \cdots \times \mathbb{Z}/M_r$. The irreps are

$$\chi_g(h) = e^{2\pi i(g_1h_1/M_1 + \cdots + g_rh_r/M_r)}.$$

For $k \in K$, $\chi_g(k) = 1$ iff $\frac{g_1k_1}{M_1} + \cdots + \frac{g_rk_r}{M_r} = 0 \bmod 1$. This is a homogenous linear equation in k, and $O(\log|G|)$ independent such equations determine K as the nullspace.

Remark 1.40 We can implement QFT over abelian groups (and some non-abelian groups, including S_n) using circuits with $O((\log|G|)^2)$ elementary gates.

In the non-abelian case, we can still easily prepare coset states with one query to f. But the shift operators $U(g_0)$ no longer commute, so we don't have a (canonical) shift-invariant basis.

Definition 1.41 A ***d*-dimensional unitary representation** of a finite group G is a homomorphism

$$\chi : G \to U(d)$$

where $U(d)$ is the group of $d \times d$ unitary matrices.

Definition 1.42 A d-dimensional unitary representation χ of G is **irreducible** if no non-trivial subspace of $\mathbb{C}^d$ is invariant under the action of $\{\chi(g_1), ..., \chi(g_{|G|})\}$ (i.e. we cannot simultaneously block diagonalise all the $\chi(g)$ matrices by a basis change).

Definition 1.43 A set of irreps $\{\chi_1, ..., \chi_m\}$ is a **complete set of irreps** for every irrep χ of G, there exists $1 \le i \le m$ such that χ is unitarily equivalent to χ_i, i.e. for some $V \in U(d)$, $\forall g \in G, \chi(g) = V\chi_i(g)V^\dagger$.

Theorem 1.44 Let the dimensions of a complete set of irreps $\chi_1, ..., \chi_m$ be $d_1, ..., d_m$. Then $d_1^2 + \cdots + d_m^2 = |G|$.

Notation 1.45 Write $\chi_{i,jk}(g)$ for the (j,k)-th entry of the matrix $\chi_i(g)$.

Theorem 1.46 (Schur Orthogonality) Let $\chi_1, ..., \chi_m$ be a complete set of irreps for G with respective dimensions $d_1, ..., d_m$, and let $i \in [m]$, $j, k \in [d_i]$. Then

$$\sum_{g\in G} \chi_{i,jk}(g)\overline{\chi_{i',j'k'}(g)} = |G|\delta_{ii'}\delta_{jj'}\delta_{kk'}.$$

Definition 1.47 The **Fourier basis** for a group G consists of

$$|\chi_{i,jk}\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} \overline{\chi_{i,jk}(g)} |g\rangle$$

for each $i \in [n]$ and $j, k \in [d_i]$. Note that by Schur orthogonality, this is an orthonormal basis.

Remark 1.48 Note that these states are not shift invariant for every $U(g_0) : |g\rangle \mapsto |g_0 g\rangle$. So measurement of the coset state $|g_0 K\rangle$ yields an output distribution that is not independent of g_0.

Definition 1.49 The **Quantum Fourier transform** over $H_{|G|}$ is the unitary mapping the Fourier basis to the computational basis:

$$\mathrm{QFT}|\chi_{i,jk}\rangle = |i, jk\rangle.$$

$|i, jk\rangle$ is a relabelling of the states $|g\rangle$ for $g \in G$ (note this is valid by Theorem 1.44).

Remark 1.50

- Measuring $\mathrm{QFT}|g_0 K\rangle$ does **not** give g_0-independent outcomes. A complete measurement in the computational basis gives an outcome i, j, k.
- However, there is an incomplete measurement which projects into the d_i^2-dimensional subspaces

  $$S_i = \mathrm{span}\{|\chi_{i,jk}\rangle : j, k \in [d_i]\}.$$

  for each $i \in [n]$. Call this measurement operator M_{rep}. Note that this distinguishes only between the irreps.
- Measuring only the representation labels of $\mathrm{QFT}|g_0 K\rangle$ gives an outcome distribution of the i values that i independent of the random shift g_0, since the χ_i are homomorphisms.
- Note this only gives partial information about K. If K is a normal subgroup, then in fact we can then determine K with $O(\log|G|)$ queries.

2. Quantum phase estimation (QPE)

Quantum phase estimation is a unifying algorithmic primitive, e.g. there is an alternative factoring algorithm based on QPE, and has many important applications in physics.

Problem 2.1 (Quantum Phase Estimation)

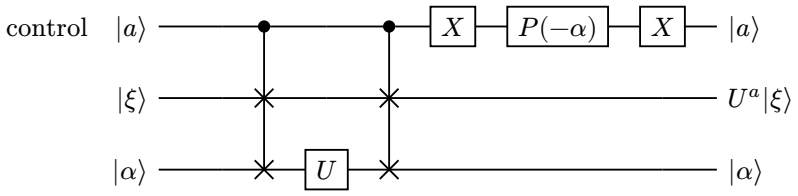
Input Unitary $U \in U(d)$ acting on $\mathbb{C}^d$; state $|v_\varphi\rangle \in \mathbb{C}^d$; level of precision $n \in \mathbb{N}$.

Promise $|v_\varphi\rangle$ is an eigenstate of U with **phase** (eigenvalue) $e^{2\pi i\varphi}$, $\varphi \in [0, 1)$ (i.e. $U|v_\varphi\rangle = e^{2\pi i\varphi}|v_\varphi\rangle$).

Task Output an estimate $\tilde{\varphi}$ of φ, accurate to n binary bits of precision.

Remark 2.2 If U is given as a circuit, we can implement the controlled-U operation, $C\text{–}U$, by controlling each elementary gate in the circuit of U.

If U is given as a black box, we need more information. Note that U is equivalent to $U' = e^{i\theta}U$ and $|\psi\rangle$ is equivalent to $e^{i\theta}|\psi\rangle$, but $C\text{–}U$ is not equivalent to $C\text{–}U'$. Given an eigenstate $|\alpha\rangle$ with known phase $e^{i\alpha}$ (so $U|\alpha\rangle = e^{i\alpha}|\alpha\rangle$), we have $U'|\alpha\rangle = e^{i(\theta+\alpha)}|\alpha\rangle$. so U and U' can be distinguished using this additional information. The following circuit implements $C\text{–}U$ (the top two lines end in state $C\text{–}U|a\rangle|\xi\rangle$):

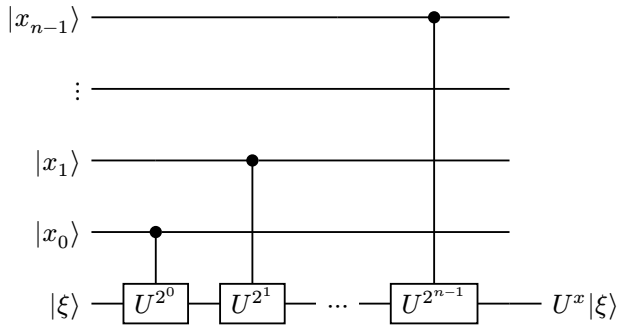


where $P(-\alpha) = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\alpha} \end{bmatrix}$, and $\bullet\text{–}\times\text{–}\times$ denotes the controlled SWAP operation.

Definition 2.3 For a unitary U, the **generalised control** unitary $C\text{–}U$ is defined linearly by

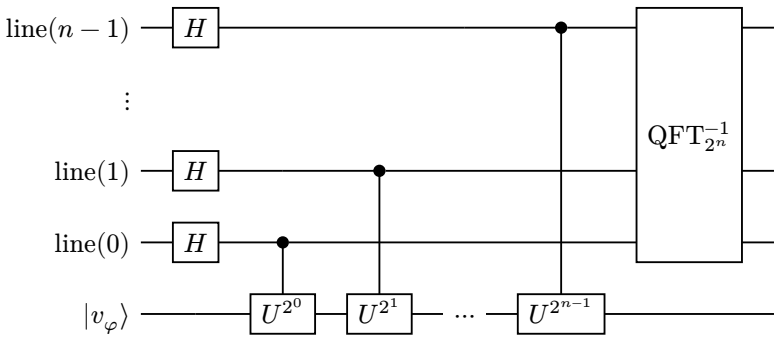
$$\forall x \in \{0,1\}^n, \quad C\text{–}U|x\rangle|\xi\rangle = |x\rangle U^x|\xi\rangle,$$

where U^x denotes U applied x times (e.g. $C\text{–}U|11\rangle|\xi\rangle = |11\rangle U^3|\xi\rangle$). Note that $C\text{–}U^k = (C\text{–}U)^k$. The following circuit implements $C\text{–}U$:



Algorithm 2.4 (Quantum Phase Estimation) Work over the space $(\mathbb{C}^2)^{\otimes n} \otimes \mathbb{C}^d$, where $(\mathbb{C}^2)^{\otimes n}$ is the n-qubit register, $\mathbb{C}^d$ is the "qudit" register.

1. Apply the following circuit to $|0...0\rangle|v_\varphi\rangle$:



2. Discard the qudit register holding $|v_\varphi\rangle$, and measure the input qubits, yielding outcome $y_0...y_{n-1}$ from lines $0, ..., n-1$.
3. The estimate of φ is $\tilde{\varphi} = y/2^n = y_0/2 + \cdots + y_{n-1}/2^n$.

Remark 2.5 After $C\text{–}U^{2^{n-1}}$, the input qubits are in the state

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} e^{2\pi i \varphi x} |x\rangle.$$

If φ had an exact n-bit expansion $0.i_1 i_2 ... i_n = (i_1 ... i_n)/2^n =: \varphi_n/2^n$, then this state is precisely $\mathrm{QFT}_{2^n} |\varphi_n\rangle$, in which case, after applying QFT^{-1}, we have $|\varphi_n\rangle$, so measuring the input bits gives φ_n, and so φ, exactly.

Lemma 2.6 For all $\alpha \in \mathbb{R}$,
1. If $|\alpha| \leq \pi$, then $\left|1 - e^{i\alpha}\right| = 2|\sin(\alpha/2)| \geq \frac{2}{\pi}|\alpha|$.
2. If $\alpha \geq 0$, then $\left|1 - e^{i\alpha}\right| \leq \alpha$.

Proof (Hints). For both, think graphically. □

Proof.
1. The line $y = \frac{2}{\pi}\alpha$ lies below $2\sin(\alpha/2)$ for $0 \leq \alpha \leq \pi$).
2. On the complex unit circle, the arc length α from 1 to $e^{i\alpha}$ is at least the chord length from 1 to $e^{i\alpha}$.

□

Theorem 2.7 (Phase Estimation Theorem) Let $\tilde{\varphi}$ be the estimate of φ from the quantum phase estimation algorithm. Then
1. $\Pr(\tilde{\varphi}$ is closest n-bit approximation of $\varphi) \geq \frac{4}{\pi^2} \approx 0.4$.
2. For all $\varepsilon > 0$, $\Pr(|\tilde{\varphi} - \varphi| > \varepsilon) = O\left(\frac{1}{2^n \varepsilon}\right)$. So for any desired accuracy ε, the probability of failure decays exponentially with the number of bits of precision (lines in the circuit).

Proof (Hints). Let $\delta(y) = \varphi - y/2^n = \varphi - \tilde{\varphi}$. Show the probability of the measuring yielding outcome y is

$$p_y = \frac{1}{2^{2n}} \left| \frac{1 - e^{2^n 2\pi i \delta(y)}}{1 - e^{2\pi i \delta(y)}} \right|^2.$$

1. Find an upper bound on $\delta(a)$ where a is the closest n-bit approximation of φ.
2. Show that

$$p_y \leq \frac{1}{2^{2n}} \left(\frac{2}{4\delta(y)} \right)^2 = \frac{1}{2^{2n+2}\delta(y)^2}.$$

Let $B = \{y \in \{0,1\}^n : |\delta(y)| > \varepsilon\}$. Show that for each $y \in B$, $|\delta(y)| \leq \varepsilon + k_y/2^n$ for some $k_y \in \mathbb{N}$, and that each k_y occurs at most twice here. Conclude the upper bound using an integral.

□

Proof. Let

$$|A\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n - 1} e^{2\pi i \varphi x} |x\rangle.$$

Let $\delta(y) = \varphi - y/2^n = \varphi - \tilde{\varphi}$. Since $\mathrm{QFT}^{-1}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{-2\pi i x y/2^n} |y\rangle$, we have

$$\mathrm{QFT}^{-1}|A\rangle = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} e^{2\pi i x \delta(y)} |y\rangle$$

so the probability of measuring outcome y is

$$p_y = \Pr\Big(\tilde{\varphi} = \frac{y}{2^n}\Big) = \frac{1}{2^{2n}} \left| \frac{1 - e^{2^n 2\pi i \delta(y)}}{1 - e^{2\pi i \delta(y)}} \right|^2.$$

1. Let $\alpha = 2^n 2\pi\delta(a)$, where a is the closest n-bit approximation of φ. Note we can imagine the possible values of $\tilde{\varphi}$ as lying on the unit circle, spaced by angle $\frac{2\pi}{2^n}$. This gives a visual intuition to the fact that $|\delta(a)| \leq \frac{1}{2^{n+1}}$. Hence $|\alpha| \leq \pi$, and so by the above lemma,

$$p_a = \Pr(\tilde{\varphi} = a) \geq \frac{1}{2^{2n}} \left(\frac{2^{n+2}\delta(a)}{2\pi\delta(a)} \right)^2 = \frac{4}{\pi^2}.$$

2. Note that $\left|1 - e^{2^n 2\pi i \delta(y)}\right| \leq 2$ by the triangle inequality. Let $B = \{y \in \{0,1\}^n : |\delta(y)| > \varepsilon\}$ denote the set of "bad" values of y. For all $y \in \{0,1\}^n$, we have $\delta(y) \in [-1, 1]$. If $|\delta(y)| \leq 1/2$, then, by the above lemma, we have $\left|1 - e^{2\pi i \delta(y)}\right| \geq 4|\delta(y)|$. If $\delta(y) > 1/2$, then $\delta(y) - 1 \in [-1/2, 1/2]$, so by the above lemma, $\left|1 - e^{2\pi i \delta(y)}\right| \geq 4|\delta(y) - 1|$ hence

$$p_y \leq \frac{1}{2^{2n}} \left(\frac{2}{4\delta(y)} \right)^2 = \frac{1}{2^{2n+2}\delta(y)^2}.$$

Let $\delta^+ = \min\{\delta(y) : y \in B, \delta(y) > 0\}$ be the smallest $\delta(y)$ such that $\delta(y) > \varepsilon$, and $\delta^- = \max\{\delta(y) : y \in B : \delta(y) < 0\}$ be the largest $\delta(y)$ such that $\delta(y) < -\varepsilon$. For all $y \in B$, we have $\delta(y) = \delta^+ + k_y/2^n$ or $\delta(y) = \delta^- - k_y/2^n$ for some $k_y \in \mathbb{N}$, so $|\delta(y)| > \varepsilon + k_y/2^n$. Note that each $k \in \mathbb{N}$, $k = k_y$ for at most 2 values of $y \in B$. Hence,

$$\begin{aligned}
\Pr(|\delta(y)| > \varepsilon) = \Pr(y \in B) &= \sum_{y \in B} p_y \\
&\leq \sum_{y \in B} \frac{1}{2^{2n+2}\big(\varepsilon + k_y/2^n\big)^2} \\
&< 2 \sum_{k=0}^{\infty} \frac{1}{2^{2n+2}} \frac{1}{(\varepsilon + k/2^n)^2} \\
&\leq \frac{1}{2^{2n+1}\varepsilon^2} + \sum_{k=1}^{\infty} \frac{1}{2^{2n+1}} \frac{1}{(\varepsilon + k/2^n)^2} \\
&= \frac{1}{2^{2n+1}\varepsilon^2} + \int_0^{\infty} \frac{1}{2^{2n+1}} \frac{1}{(\varepsilon + x/2^n)^2} \, \mathrm{d}x \\
&= \frac{1}{2^{2n+1}\varepsilon^2} + \int_{2^n\varepsilon}^{\infty} \frac{1}{2u^2} \, \mathrm{d}u = \frac{1}{2^{2n+1}\varepsilon^2} + \frac{1}{2^{n+1}\varepsilon}.
\end{aligned}$$

□

Remark 2.8 The QPE algorithm excluding the measurement is a unitary - call this unitary U_{PE}. If we apply U_{PE} to an arbitrary state $|\psi\rangle = \sum_j c_j|v_j\rangle$ where $|v_j\rangle$ are the eigenstates of U with eigenvalue $e^{2\pi i\varphi_j}$, then we have

$$U_{\text{PE}}|\psi\rangle = \sum_j c_j|\tilde{\varphi}_j\rangle|v_j\rangle$$

If every φ_j has an exact n-bit representation, then this is exact. Otherwise, we have $|\tilde{\varphi}_j\rangle = \sqrt{1-\eta}|\tilde{\varphi}_1\rangle + \sqrt{\eta}|\tilde{\varphi}_0\rangle$, where $|\tilde{\varphi}_1\rangle$ is a superposition of all n-bit strings that are correct to the first n-bits of φ, and $|\tilde{\varphi}_0\rangle$ is a superposition of strings with the first n bits not all correct.

Remark 2.9 Complexity of QPE: we use $C\text{–}U, ..., C\text{–}U^{2^{n-1}}$, so the number of uses of $C\text{–}U$ is $\approx 2^n$. So this initially looks like exponential time, but there are special cases of U where by repeated squaring, this can be implemented with $\text{poly}(n)$ gates.

If we want to estimate φ accurate to m bits of precision with probability $1-\eta$, then by the phase estimation theorem with $\varepsilon = \frac{1}{2^m}$, we need $n = O(m + \log(1/\eta))$ lines. Note this is a modest, polynomial increase in the number of lines of the circuit for an exponential reduction in η.

3. Amplitude amplification

Amplitude amplification is an extension of the key insights in Grover's algorithm (TODO: read part II notes for Grover's).

Notation 3.1 Given $|\alpha\rangle \in H_d$, write $L_{|\alpha\rangle} = \text{span}\{|\alpha\rangle\}$ for the one-dimensional subspace generated by $|\alpha\rangle$.

Notation 3.2 Given a subspace $A \leq H_d$, denote the projector onto A by P_A. Note that

$$P_A = \sum_{i=1}^{k} |a_i\rangle\langle a_i|$$

for any orthonormal basis $\{|a_1\rangle, ..., |a_k\rangle\}$ of A.

Notation 3.3 Given a subspace $A \leq H_d$, define the unitary $I_A = I - 2P_A$, which is the reflection in the "mirror" $A^\perp$: indeed, note that for all $|\varphi\rangle \in A$, $I_A = -|\varphi\rangle$, and for all $|\psi\rangle \in A^\perp$, $I_A|\psi\rangle = |\psi\rangle$, since $P_A|\psi\rangle = 0$.

In the case that A is one-dimensional and spanned by $|\alpha\rangle$, we have $P_A = |\alpha\rangle\langle\alpha|$, and write $I_{|\alpha\rangle} = I - 2|\alpha\rangle\langle\alpha|$.

Proposition 3.4 Let $|\alpha\rangle \in H_d$. For any unitary $U \in U(d)$, we have

$$UI_{|\alpha\rangle}U^\dagger = I_{U|\alpha\rangle}.$$

Proof (Hints). Trivial. □

Proof. $UI_{|\alpha\rangle}U^\dagger = UU^\dagger - 2U|\alpha\rangle\langle\alpha|U^\dagger = I_{U|\alpha\rangle}$. □

Problem 3.5 (Unstructured Search Problem)
Input An oracle for a function $f : \{0,1\}^n \to \{0,1\}$.
Promise There is a unique $x_0 \in \{0,1\}^n$ such that $f(x_0) = 1$.
Task Find x_0.

Remark 3.6 The unstructured search problem is closely related to the complexity class NP and to Boolean satisfiability.

Definition 3.7 For fixed $|x_0\rangle \in H_2^{\otimes n}$, the **Grover iteration operator** Q is defined as

$$Q := -H^{\otimes n} I_{|0\rangle} H^{\otimes n} I_{|x_0\rangle} = -I_{H^{\otimes n}|0\rangle} I_{|x_0\rangle}.$$

Remark 3.8 Note that for a function $f : \{0,1\}^n \to \{0,1\}$ fulfilling the promise of the Unstructured Search Problem, we can implement $I_{|x_0\rangle}$ without knowing x_0: we have $U_f|x\rangle\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = (-1)^{f(x)}|x\rangle\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = I_{|x_0\rangle}|x\rangle\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Hence, implementing Q requires only one query to f.

Theorem 3.9 (Grover) In the 2-dimensional subspace spanned by $|\psi\rangle = H^{\otimes n}|0\rangle$ and $|x_0\rangle$, the action of Q is a rotation by angle 2α, where $\sin(\alpha) = \frac{1}{\sqrt{2^n}} = \langle x_0|\psi\rangle$.

Algorithm 3.10 (Grover's Algorithm) Work in the state space $H_2^{\otimes n}$.
1. Prepare $|\psi\rangle = H^{\otimes n}|0\rangle$.
2. Apply Q^m to $|\psi\rangle$, where m is closest integer to $\frac{\arccos\left(1/\sqrt{N}\right)}{2\arcsin\left(1/\sqrt{N}\right)} = \frac{\theta}{2\alpha}$ and $\cos(\theta) = \sin(\alpha) = \langle x_0|\psi\rangle = 1/\sqrt{2^n}$. This rotates $|\psi\rangle$ to be close to $|x_0\rangle$ (within angle $\pm\alpha$ of $|x_0\rangle$).
3. Measure to get x_0 with probability $p = |\langle x_0|Q^m|\psi\rangle|^2 = 1 - \frac{1}{N}$. For large N, $\arccos\left(1/\sqrt{N}\right) \approx \frac{\pi}{2}$, and $\arcsin\left(1/\sqrt{N}\right) \approx 1/\sqrt{N}$. The number of iterations is $m = \frac{\pi}{4}\sqrt{N} = O\left(\sqrt{N}\right)$. So we need $O\left(\sqrt{N}\right)$ queries to U_f. In contrast, clasically we need $\Omega(N)$ queries to f to find x_0 with any desired constant probability. Note that $\Omega(N)$ queries are both necessary and sufficient.

Notation 3.11 Write G for the subspace of the state space H whose associated amplitudes in a given state we wish to amplify. G is called the "good" subspace. We call the subspace $G^\perp$ the "bad" subspace. Note that $H = G \oplus G^\perp$, and for any state $|\varphi\rangle \in H$, there is a unique decomposition with real, positive coefficients $|\varphi\rangle = \sin(\theta)|g\rangle + \cos(\theta)|b\rangle$, where $|g\rangle = P_G|\varphi\rangle$ and $|b\rangle = P_{G^\perp}|\varphi\rangle$.

Theorem 3.12 (Amplitude Amplification Theorem/2D-subspace Lemma) Let $G \leq H_2^{\otimes n}$ be a subspace and $|g\rangle = P_G|\psi\rangle$, $|b\rangle = P_{G^\perp}|\psi\rangle$. In the 2-dimensional subspace $\text{span}\{|\psi\rangle,|g\rangle\} = \text{span}\{|b\rangle,|g\rangle\}$, the unitary

$$Q = -I_{|\psi\rangle} I_G$$

is a rotation by angle 2θ, where $\sin(\theta) = \|P_G|\psi\rangle\|_2^2 = |\langle g|g\rangle|$ is the length of the "good" projection of $|\psi\rangle$.

Proof (Hints). Consider the matrix representation of Q in the $\text{span}\{|b\rangle,|g\rangle\}$ basis. □

Proof. By definition, we have $I_G|g\rangle = -|g\rangle$, and $I_G|b\rangle = |b\rangle$. Hence $Q|g\rangle = I_{|\psi\rangle}|g\rangle$ and $Q|b\rangle = -I_{|\psi\rangle}|b\rangle$. The matrix representation of $I_{|\psi\rangle}$ in the $\{|b\rangle,|g\rangle\}$ basis is

$$\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} - 2\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}[\cos(\theta)\ \ \sin(\theta)] = \begin{bmatrix}1-2\cos(\theta)^2 & -2\sin(\theta)\cos(\theta)\\ -2\sin(\theta)\cos(\theta) & 1-2\sin(\theta)^2\end{bmatrix}$$
$$= \begin{bmatrix}-\cos(2\theta) & -\sin(2\theta)\\ -\sin(2\theta) & \cos(2\theta)\end{bmatrix}.$$

So $Q|b\rangle = \cos(2\theta)|b\rangle + \sin(2\theta)|g\rangle$, and $Q|g\rangle = -\sin(2\theta)|b\rangle + \cos(2\theta)|g\rangle$. So the matrix representation of Q in the $\{|b\rangle,|g\rangle\}$ basis is

$$\begin{bmatrix}\cos(2\theta) & -\sin(2\theta)\\ \sin(2\theta) & \cos(2\theta)\end{bmatrix}$$

which indeed is a rotation by angle 2θ. □

Corollary 3.13 We have $Q^m|\psi\rangle = \cos((2m+1)\theta)|b\rangle + \sin((2m+1)\theta)|g\rangle$.

Proof (Hints). Trivial. □

Proof. Induction on m. □

Notation 3.14 Denote by m_{opt} the $m \in \mathbb{Z}$ which maximises the probability of measuring (in the $\{|b\rangle,|g\rangle\}$ basis) an outcome in G on the state $Q^m|\psi\rangle$. Note that this probability is equal to $\sin((2m+1)\theta)^2$, which is maximised when

$$(2m+1)\theta = \pi/2 \quad \Longrightarrow m = \pi/4\theta - 1/2.$$

So m_{opt} is the nearest integer to $\pi/4\theta - 1/2$.

Example 3.15 Let $\theta = \pi/6$, then $m_{\text{opt}} = 1$ and $Q|\psi\rangle = |g\rangle$. So we obtain a good outcome with certainty on measurement.

Remark 3.16 Note that since Q is a rotation by angle 2θ, $Q^{m_{\text{opt}}}|\psi\rangle$ lies within angle $\pm\theta$ of $|g\rangle$, hence the $|g\rangle$ component of $Q^{m_{\text{opt}}}|\psi\rangle$ has amplitude $\geq \cos(\theta)^2$. TODO: insert diagram. So for small θ,

$$\Pr(\text{measuring good outcome}) \geq \cos(\theta)^2 \approx 1 - O(\theta^2).$$

Also, for small θ,

$$m_{\text{opt}} = O(1/\theta) \approx O(1/\sin(\theta)).$$

Remark 3.17 Q can be implemented (efficiently) if $I_{|\psi\rangle}$ and I_G can be implemented (efficiently). For an efficient implementation of I_G, it suffices for G to be spanned by some subset of computational basis states, and the indicator function $\mathbb{1}_G$ is efficiently computable. In this case, we have

$$U_{\mathbb{1}_G}|x\rangle\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = (-1)^{\mathbb{1}_G(x)}\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Since I_G is defined by its action $|g\rangle \mapsto -|g\rangle$ for $g \in G$ and $|b\rangle \mapsto |b\rangle$ for $b \in G^\perp$, the first register holds the value $I_G|x\rangle$.

For an efficient implementation of $I_{|\psi\rangle}$, we usually have $|\psi\rangle = H^{\otimes n}|0...0\rangle$, and then $I_{|\psi\rangle} = H^{\otimes n} I_{|0\rangle} H^{\otimes n}$ can be implemented with $O(n)$ gates.

Remark 3.18 In the amplitude amplification process, the relative amplitudes of basis states inside $|g\rangle$ and $|b\rangle$ won't change. So amplitude amplification boosts the overall amplitude of $|g\rangle$ at the expense of the amplitude of $|b\rangle$.

3.1. Applications of amplitude amplification

Example 3.19 We can generalise Grover search from 1 marked item to k marked items out of $N = 2^n$ items. In this case,

$$\begin{aligned}|\psi\rangle &= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \\ &= \sqrt{\frac{k}{N}} \frac{1}{\sqrt{k}} \sum_{x\in G} |x\rangle + \sqrt{\frac{N-k}{N}} \frac{1}{\sqrt{N-k}} \sum_{x\in G^\perp} |x\rangle \\ &=: \sqrt{\frac{k}{N}} |g\rangle + \sqrt{\frac{N-k}{N}} |b\rangle\end{aligned}$$

so $\sin(\theta) = \sqrt{k/N}$. For $k \ll N$, $\sin(\theta) \approx \theta$, so $m_{\text{opt}} = O(\sqrt{N/k})$ uses of Q required. E.g. $N = 4 = 2^2$ items and $k = 1$ marked item, we have $\sin(\theta) = 1/2$, so $\theta = \pi/6$, so Grover search is exact, and requires exactly one application of Q.

Example 3.20 (Quadratic speedup of general quantum algorithms) Let U be a unitary representing a quantum algorithm/circuit, with $U|0...0\rangle = \alpha|g\rangle + \beta|b\rangle$ where $|g\rangle$ is a (generally non-uniform) superposition of good/correct outcomes, and $|b\rangle$ is a (generally non-uniform) superposition of bad/incorrect outcomes. Note $|g\rangle = \sum_{x\in\{0,1\}^n} c_x|x\rangle$ is generally a non-uniform superposition. We have

$$\Pr(\text{measuring } U|0...0\rangle \text{ yields good outcome}) = |\alpha|^2.$$

Thus, we need to run U about $O(1/|\alpha|^2)$ times to succeed with high probability.

Now define $|\psi\rangle = U|0...0\rangle$ and $Q = -I_{|\psi\rangle} I_G$. We can implement Q if we have a method to verify the output of U; so in particular, we can use this method for any NP problem. This will mean we can efficiently implement the indicator function $\mathbb{1}_G$ of good labels and therefore also I_G. So by the Amplitude Amplification Theorem, Q performs a rotation of 2θ where $\sin(\theta) = |\alpha|$. So after approximately

$$m_{\text{opt}} \approx \pi/4\theta = O(1/\theta) = O(1/\sin(\theta)) = O(1/|\alpha|)$$

(for θ small) uses of Q, we get a good outcome with high probability.

Problem 3.21 (Counting Problem)
Input $f : \{0,1\}^n \to \{0,1\}$.
Task Estimate the number $k = \left|f^{-1}(\{1\})\right|$ of inputs that evaluate to 1.

Example 3.22 (Quantum Counting) This combines amplitude amplification and quantum phase estimation. Let the "good" subspace G be the subspace with basis $f^{-1}(\{1\})$. As usual, let

$$|g\rangle := \frac{1}{\sqrt{k}} \sum_{x \in f^{-1}(\{1\})} |x\rangle, \quad |b\rangle := \frac{1}{\sqrt{2^n - k}} \sum_{x \in f^{-1}(\{0\})} |x\rangle,$$

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle = \sqrt{\frac{k}{N}} \frac{1}{\sqrt{k}} \sum_{x \in f^{-1}(\{1\})} |x\rangle + \sqrt{\frac{N-k}{N}} \frac{1}{\sqrt{N-k}} \sum_{x \in f^{-1}(\{0\})} |x\rangle.$$

Recall that Q has matrix representation

$$Q = \begin{bmatrix} \cos(2\theta) & -\sin(2\theta) \\ \sin(2\theta) & \cos(2\theta) \end{bmatrix}$$

in the orthonormal basis $\{|b\rangle, |g\rangle\}$ where $\sin(\theta) = \|P_G|\psi\rangle\|$. The eigenvalues and eigenstates of Q are $\lambda_\pm = e^{\pm 2i\theta}$ and $|e_\pm\rangle = \frac{1}{\sqrt{2}}(|b\rangle \mp i|g\rangle)$. So we can write $|\psi\rangle = \sin(\theta)|g\rangle + \cos(\theta)|b\rangle = \frac{1}{\sqrt{2}}\big(e^{-i\theta}|e_+\rangle + e^{i\theta}|e_-\rangle\big)$. So $|\psi\rangle$ is an equally-weighted superposition of eigenstates of Q. Write $e^{\pm 2i\theta} = e^{2\pi i \varphi_\pm}$ with $\varphi_\pm \in (0,1)$. We have $\varphi_+ = \theta/\pi$ and $\varphi_- = (-2\theta + 2\pi)/2\pi = 1 - \theta/\pi$. When $k \ll N$, $\sin(\theta) = \sqrt{k/N} \approx \theta$, so using U_{PE} with m qubits of precision

$$U_{\text{PE}}|\psi\rangle = \frac{1}{\sqrt{2}}\big(e^{-i\theta}|e_+\rangle|\tilde{\varphi}_+\rangle + e^{i\theta}|e_-\rangle|\tilde{\varphi}_-\rangle\big)$$

Measuring the QPE output gives (with probability 1/2) an estimate of $\varphi_+ = \theta/\pi \approx \frac{1}{\pi}\sqrt{k/N}$ or (with probability 1/2) an estimate of $\varphi_- = 1 - \theta/\pi \approx 1 - \frac{1}{\pi}\sqrt{k/N}$. So in either case, we get an estimate of $\sqrt{k/N}$ (since we can tell when $k \ll N$ which case we are in). By the Phase Estimation Theorem, with probability at least $4/\pi^2$, QPE with m lines gives us an approximation of $\sqrt{k/N}$ to precision $O(1/2^m)$, using $O(2^m)$ C–Q operations, each of which requires one query to f. Write $\delta/\sqrt{2^n} = 1/2^m$ for some $\delta > 0$. So we can estimate $\sqrt{k}$ to precision δ, and since $\Delta(x^2) = 2x\Delta(x)$, we estimate $\sqrt{k}$ to additive error (precision) $O\big(\delta\sqrt{k}\big)$ using $O(2^m) = O\big(\sqrt{N}/\delta\big)$ queries to f.

Remark 3.23 The quantum counting algorithm is quadratically faster than the best possible classical algorithm, which is:

- Sample random x from $\{0,1\}^n$, then $\Pr(f(x) = 1) = k/N$.
- Draw m samples $x_1, ..., x_m$, then the estimate is $\tilde{k} = \ell N/m$, where $m = |\{i \in [m] : f(x_i) = 1\}|$.

We need $m = O(N/\delta^2)$ to estimate k to high precision.

4. Hamiltonian simulation

We want to use a quantum system to simulate the evolution/dynamics of another quantum system, given its Hamiltonian H. For an n-qubit system, in general this

requires $O(2^n)$ time on a classical computer. For some physically interesting classes of H, we have quantum algorithms that run in $O(\text{poly}(n))$ time.

Proposition 4.1 The Schrodinger equation which governs the time evolution of a physical state $|\psi(t)\rangle$, which is given by (assuming $\hbar = 1$)

$$\frac{\mathrm{d}}{\mathrm{d}t}|\psi(t)\rangle = -iH|\psi(t)\rangle,$$

has solution

$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

when H is time-independent.

Definition 4.2 The **exponential** of a matrix $A \in \mathbb{C}^{n\times n}$ is defined as

$$\exp(A) = e^A := \sum_{k=0}^{\infty} \frac{A^k}{k!}.$$

Note that if $[A, B] = 0$, then $\exp(A)\exp(B) = \exp(A + B)$, but generally this does not hold when $[A, B] \neq 0$.

Theorem 4.3 If H is Hermitian, then e^{-iHt} is unitary for all $t \in \mathbb{R}$.

Definition 4.4 $U(t) = e^{-iHt}$ is called the **evolution operator**. Given H and $t > 0$, we want to simulate $U(t)$ accurately.

Proposition 4.5 If $A \in \mathbb{C}^{n\times n}$ has spectrum $\{(\lambda_i, |e_i\rangle) : i \in [n]\}$, then

$$\exp(A) = \sum_{i=1}^{n} \exp(\lambda_i)|e_i\rangle.$$

Definition 4.6 The **operator norm** (**spectral norm**) of an operator $A : H \to H$ acting on the space H of states is

$$\|A\| := \max\{\|A|\psi\rangle\| : \psi \in H, \|\psi\| = 1\}.$$

Theorem 4.7 If A is diagonalisable with eigenvalues $\lambda_1, ..., \lambda_n$, then

$$\|A\| = \max\{|\lambda_1|, ..., |\lambda_n|\}.$$

Proposition 4.8 The operator norm satisfies the following properties:
1. **Submultiplicative**: $\|AB\| \leq \|A\|\|B\|$
2. **Triangle inequality**: $\|A + B\| \leq \|A\| + \|B\|$.

Definition 4.9 Let $U, \tilde{U} : H \to H$ be operators. $\tilde{U}$ **ε-approximates** U if

$$\left\|U - \tilde{U}\right\| \leq \varepsilon,$$

i.e. for all normalised states $|\psi\rangle$, $\left\|U|\psi\rangle - \tilde{U}|\psi\rangle\right\| \leq \varepsilon$.

Lemma 4.10 Let $U_1, ..., U_m, \tilde{U}_1, ..., \tilde{U}_m$ be unitaries. Suppose $\tilde{U}_i$ ε-approximates U_i for each $1 \leq i \leq m$. Then

$$\left\|U_n \cdots U_1 - \tilde{U}_n \cdots \tilde{U}_1\right\| \leq n\varepsilon$$

So the error increases at most linearly.

Definition 4.11 H is a ***k*-local Hamiltonian on *n* qubits** if we can write

$$H = \sum_{j=1}^{m} H_j$$

where each H_j acts non-trivially on at most k qubits, in which case we write $H_j = \widetilde{H}_j \otimes I$ (note these qubits need not be adjacent).

Note that $m \leq \binom{n}{k} = O\big(n^k\big)$, and we usually take k to be a constant.

Notation 4.12 Write $U_{(i)}$ for the unitary

$$I \otimes \cdots \otimes I \otimes U \otimes I \otimes \cdots \otimes I$$

where U is in the i-th position, i.e. $U_{(i)}$ is the unitary acting on the i-th qubit on n-qubits.

Example 4.13

- $H = X \otimes I \otimes I - 5Z \otimes Y \otimes I$ is 2-local on 3 qubits.
- For the **Ising model** on an $n \times n$ grid, where each qubit acts non-trivially only with its neighbours, the Hamiltonian is

$$H = J \sum_{i,j=1}^{n} \left(Z_{(i,j)}Z_{(i,j+1)} + Z_{(i,j)}Z_{(i+1,j)}\right)$$

  where $J \in \mathbb{R}$ is a coupling constant.
- For the **Heisenberg model** on a line, the Hamiltonain is

$$H = \sum_{i=1}^{n-1}\Big(J_X X_{(i)}X_{(i+1)} + J_Y Y_{(i)}Y_{(i+1)} + J_Z Z_{(i)}Z_{(i+1)}\Big),$$

  where $J_X, J_Y, J_Z \in \mathbb{R}$ are constants.

Theorem 4.14 (Solovay-Kitaev) Let U be a unitary on k-qubits, and S be a universal set of elementary gates. Then U can be ε-approximated using $O((\log(1/\varepsilon))^c)$ gates from S, where $c < 4$ is a constant.

Proof. Omitted. □

Proposition 4.15 Let $H = \sum_{j=1}^{m} H_j$ be a k-local Hamiltonian where all the local terms H_j commute. Then for all $t > 0$ and $\varepsilon > 0$, the evolution operator $U(t) = e^{-iHt}$ can be ε-approximated by a circuit with $O(m\,\mathrm{polylog}(m/\varepsilon))$ gates from any universal gate set.

Note that $m = O\big(n^k\big)$, so the time-complexity is polynomial in n.

Proof (Hints). Straightforward. □

Proof. Fix $t > 0$ and $\varepsilon > 0$. We have

$$U(t) = e^{-iHt} = e^{-i\sum_{j=1}^{m} H_j t} = \prod_{j=1}^{m} e^{-iH_j t}.$$

Each $e^{-iH_j t}$ is a unitary that acts non-trivially on at most k qubits. So we have a circuit for e^{-iHt} in terms of some set of k-qubit gates. By Solovay-Kitaev, each $e^{-iH_j t}$ can be δ-approximated by a unitary $\tilde{U}_j(t)$ circuit with $O(\text{polylog}(1/\delta))$ gates. By Lemma 4.10, we have

$$\left\| U(t) - \prod_{i=1}^{m} \tilde{U}_j(t) \right\| < m\delta.$$

So choosing $\delta = \varepsilon/m$, we obtain a circuit of size $O(m \,\text{polylog}(m/\varepsilon))$ which ε-approximates $U(t)$. □

Notation 4.16 For $N \times N$ matrices X and Y, write $X = Y + O(\varepsilon)$ to mean $X = Y + E$ where $\|E\| \leq \varepsilon$.

Lemma 4.17 (Lie-Trotter Product Formula) Let A and B be $N \times N$ matrices with $\|A\|, \|B\| \leq \delta < 1$. Then

$$e^{-iA}e^{-iB} = e^{-i(A+B)} + O(\delta^2).$$

Proof (Hints). Write $e^{-iA} = I - iA + E_A$ and show that $\|E_A\| = O(\delta^2)$, do the same for two other matrices.b □

Proof. We have

$$e^{-iA} = I - iA + \sum_{j=2}^{\infty} \frac{(-iA)^j}{j!} =: I - iA + E_A.$$

Now

$$\begin{aligned}
\|E_A\| &= \left\| (-iA)^2 \sum_{j=0}^{\infty} \frac{(-iA)^j}{(j+2)!} \right\| \\
&\leq \|(-iA)^2\| \cdot \left\| \sum_{j=0}^{\infty} \frac{(-iA)^j}{(j+2)!} \right\| && \text{by submultiplicativity} \\
&\leq \|(-iA)^2\| \cdot \sum_{j=0}^{\infty} \frac{\|(-iA)^j\|}{(j+2)!} && \text{by triangle inequality and continuity} \\
&\leq \|A\|^2 \sum_{j=0}^{\infty} \frac{\|(-iA)\|^j}{j!} && \text{by submultiplicativity} \\
&= \delta^2 e^{\delta} \leq \delta^2.
\end{aligned}$$

So $e^{-iA} = I - iA + O(\delta^2)$. By the same argument, we have

$$\begin{aligned}
e^{-iB} &= I - iB + O(\delta^2), \\
e^{-i(A+B)} &= I - i(A+B) + O(2\delta^2) = I - i(A+B) + O(\delta^2)
\end{aligned}$$

since $\|A+B\| \le \|A\| + \|B\| = 2\delta$. Hence,

$$\begin{aligned} e^{-iA}e^{-iB} &= (I - iA + O(\delta^2))(I - iB + O(\delta^2)) \\ &= I - i(A+B) + O(\delta^2) \\ &= e^{-i(A+B)} + O(\delta^2), \end{aligned}$$

since $\|AB\| \le \delta^2$ by submultiplicativity. □

Proposition 4.18 There is a $\mathrm{poly}(n, 1/\varepsilon, t)$-time quantum algorithm for simulating the evolution operators of k-local Hamiltonians.

Proof. Let $H = \sum_{j=1}^{m} H_j$ be a k-local Hamiltonian and $U(t) = e^{-iHt}$ be its evolution operator. We can assume that not all the H_j commute, otherwise we are done by Proposition 4.15. Assume $t = 1$ and each $\|H_j\| \le \delta$ with $\delta \le 1/m$, since then $\|H_1 + \cdots + H_\ell\| \le \ell\delta$, and we need the Lie-Trotter approximation to hold for all $\ell \in [m]$. We have

$$\begin{aligned} &\left(e^{-iH_1}e^{-iH_2}\right)\cdots e^{-iH_m} \\ &= \left(e^{-i(H_1+H_2)} + O(\delta^2)\right)e^{-iH_3}\cdots e^{-iH_m} && \text{by Lie-Trotter} \\ &= e^{-i(H_1+H_2)}e^{-iH_3}\cdots e^{-iH_m} + O(\delta^2) && \text{by submultiplicativity} \\ &= \left(e^{-i(H_1+H_2+H_3)} + O((2\delta)^2)\right)e^{-iH_4}\cdots e^{-iH_m} + O(\delta^2). \end{aligned}$$

since each e^{-iH_j} is unitary, so has unit norm. Repeatedly applying Lie-Trotter, we obtain

$$\begin{aligned} e^{-iH_1}\cdots e^{-iH_m} &= e^{-i(H_1+\cdots+H_m)} + O(\delta^2) + \cdots + O(((m-1)\delta)^2) \\ &= e^{-i(H_1+\cdots+H_m)} + O(m^3\lambda^2). \end{aligned}$$

Let the $O(m^3\lambda^2)$ error be $Em^3\lambda^2$. For general $\|H_i\|$ and $t > 0$, introduce M large (to be chosen later), and define $\widetilde{H}_j = H_j t/M$. Note that $\|\widetilde{H}_j\| \le \delta t/M =: \tilde{\delta}$. Now

$$U(t) = e^{-i(H_1+\cdots+H_m)t} = \left(e^{-i(H_1+\cdots+H_m)t/M}\right)^M.$$

So we need the error in approximating $e^{-iHt/M}$ to be at most ε/M. So using the above error bound, we want $Em^3\tilde{\delta}^2 < \varepsilon/M$, i.e. $M > Em^3(\delta t)^2/\varepsilon$. With this choice of M, we have

$$\left\|e^{-iH_1 t/M}\cdots e^{-iH_m t/M} - e^{-i(H_1+\cdots+H_m)t/M}\right\| \le \varepsilon/M.$$

Hence by Lemma 4.10,

$$\left\|e^{-iH_1 t}\cdots e^{-iH_m t} - e^{-i(H_1+\cdots+H_m)t}\right\| \le \varepsilon.$$

The circuit is composed of Mm gates of the form $e^{-iH_j t/M}$, so the entire circuit consists of $O(m^4(\delta t)^2/\varepsilon)$ of these gates. Recall that if H is k-local, then $m \le \binom{n}{k} = O(n^k)$. So we have a circuit with $C = O(n^{4k}(\delta t)^2/\varepsilon)$ gates of the form $e^{-iH_j t/M}$ approximating e^{-iHt} to precision ε. By Solovay-Kitaev, each of these gates can be

(ε/C)-approximated by $O(\log^4(C/\varepsilon))$ gates from an elementary universal gate set. So the final complexity is $\tilde{O}\big(n^{4k}(\delta t)^2/\varepsilon\big)$ which is $\mathrm{poly}(n, 1/\varepsilon, t)$. □

Remark 4.19

- The time dependence is quadratic, but there are improved product formulae that allow the dependence of the circuit size on t to be $O(t^{1+\alpha})$ for any $\alpha > 0$.
- The ε-dependence is $\mathrm{poly}(1/\varepsilon)$ whereas in the commuting case it was $\mathrm{polylog}(1/\varepsilon)$. However, there are methods that decrease this to $(1/\varepsilon)^{1/2k}$.

5. The Harrow-Hassidim-Lloyd (HHL) algorithm

Problem 5.1 (Linear System Solution Problem)
Input matrix $A \in \mathbb{C}^{N\times N}$, vector $b \in \mathbb{C}^N$.
Task find a vector $x \in \mathbb{C}^N$ such that $Ax = b$.

Remark 5.2 The best known classical algorithms for solving linear systems require $O(\mathrm{poly}(N) \cdot \log(1/\varepsilon))$ time. Note that just reading the inputs A and b, or writing the solution x requires $O(\mathrm{poly}(N))$ time.

Instead of computing the full solution x, the HHL algorithm estimates properties of x of the form $\mu = x^T M x$ (i.e. quadratic forms), where M is Hermitian, e.g. the total weight assigned by x to a subset of indices/components. Classically, there is no better known way of doing this than computing the entire solution first. HHL can solve such tasks in $O\big(\mathrm{polylog}(N) \cdot \frac{1}{\varepsilon} \cdot \kappa^2\big)$ time, where κ is the condition number of A. When $\kappa = \mathrm{polylog}(N)$, this is an exponential speedup over the best known classical algorithms.

Definition 5.3 The **condition number** of a square matrix $A \in \mathbb{C}^{N\times N}$ is defined as

$$\kappa(A) := \begin{cases} \|A^{-1}\| \cdot \|A\| & \text{if } A \text{ is invertible} \\ \infty & \text{otherwise} \end{cases}.$$

$\kappa(A)$ can be thought of a measure of "how invertible" A is. We say A is **well-conditioned** if $\kappa(A)$ is small.

Proposition 5.4 If $A \in \mathbb{C}^{N\times N}$ is Hermitian with eigenvalues $\lambda_1, ..., \lambda_N$, then

$$\kappa(A) = \frac{\max\{|\lambda_i| : i \in [N]\}}{\min\{|\lambda_i| : i \in [N]\}}.$$

Proof (Hints). Straightforward. □

Proof. We have

$$A = \sum_{i=1}^{N} \lambda_i |v_i\rangle\langle v_i|$$

where $|v_1\rangle, ..., |v_n\rangle$ are the eigenvalues of A. Hence,

$$A^{-1} = \sum_{i=1}^{N} \lambda_i^{-1} |v_i\rangle\langle v_i|,$$

so A^{-1} has eigenvalues $\lambda_1^{-1}, ..., \lambda_N^{-1}$. So by Theorem 4.7, we have

$$\kappa(A) = \left\|A^{-1}\right\| \cdot \|A\| = \max\left\{\left|\lambda_i^{-1}\right| : i \in [N]\right\} \cdot \max\{|\lambda_i| : i \in [N]\}$$
$$= \frac{\max\{|\lambda_i| : i \in [N]\}}{\min\{|\lambda_i| : i \in [N]\}}.$$

□

Preliminary requirements for HHL algorithm to be applicable:

- We also assume that $\boldsymbol{b}$ is normalised, (or that $\|b\|_2$ is efficiently computable), and that the state $|b\rangle$ can be efficiently prepared on a quantum computer.

The quantum algorithm will work on $n = \log N$ qubits and will never need to "write down" A, b, or $x = A^{-1}b$ as lists of numbers. It will output a state $|\hat{x}'\rangle$ that is ε-close to $|\hat{x}\rangle$, and $|\hat{x}\rangle$ is proportional to $A^{-1}|b\rangle$ in $O\big(\text{poly}(n) \cdot \kappa^2 \cdot \frac{1}{\varepsilon}\big)$. Using $|\hat{x}'\rangle$ a further $O(\text{poly}(n)\kappa^2/\varepsilon)$ times, we can estimate any $\mu = x^T M x$.

The best known classical algorithm requires $O\big(\text{poly}(N) \cdot \kappa \cdot \log\big(\frac{1}{\varepsilon}\big)\big)$ time, even with assumptions comparable to our assumptions for HHL.

Note that when ε is constant (or even $\varepsilon = O\big(\frac{1}{\text{poly}(n)}\big)$) and for well-conditioned A, we have an exponential speedup.

Definition 5.5 Given an angle c, define the **controlled-rotation** unitary $C-\text{Rot}$ linearly by

$$C-\text{Rot}|x\rangle|0\rangle = |x\rangle(\sqrt{1 - c^2/x^2}|0\rangle + \frac{c}{)}$$

Algorithm 5.6 (HHL) We are given $|b\rangle = \frac{1}{\|b\|_2}\sum_{i=0}^{N-1} b_i|i\rangle$.

1. Apply U_{PE} for the unitary e^{-iA} (this is implemented by Hamiltonian simulation) with m bits of precision on the state $|b\rangle|0\rangle^{\otimes m}$.
2. Apply $C-\text{Rot}$ to the state.
3. Perform a **post-selection** step: measure the last qubit, and if the outcome is 0, reject and go back to step 1, otherwise accept.
4. Perform a measurement in the M basis on the resulting state.
5. Repeat all of the above $O\big(\log\big(\frac{1}{\eta}\big)/\delta^2\big)$ times and compute the empirical mean of the measurements.

Theorem 5.7 Under the following assumptions, the HHL algorithm computes a estimate $\hat{\mu}$ of $\mu = x^T M x$ to accuracy δ, with probability at least $1 - \eta$, in $O(...)$ time:

- The state $|b\rangle$ can be prepared exactly and efficiently.
- The unitary $C-\text{Rot}$ can be implemented exactly and efficiently.
- Measurements in the M basis can be performed efficiently.
- There is an efficient Hamiltonian simulation algorithm for A, i.e. $U(t) = e^{-iAt}$ can be implemented with $O(\text{poly}(n) \cdot t)$ gates.

Remark 5.8 The condition that there is an efficient Hamiltonian simulation algorithm for A holds for local Hamiltonians, but also for the larger class, which

naturally occurs in applications, of locally-computable and row-sparse (every row contains at most $O(\text{polylog}(N))$ non-zero entries) matrices.

Remark 5.9

- We can also use HHL for non-Hermitian A: double the system size and set

$$\tilde{A} = \begin{bmatrix} 0 & A^\dagger \\ A & 0 \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} 0 \\ b \end{bmatrix}.$$

  Then run HHL on $\tilde{A}$ and $\tilde{b}$: if $Ax = b$, then $\tilde{A}\tilde{x} = \tilde{b}$ where $\tilde{x} = \begin{bmatrix} x \\ 0 \end{bmatrix}$.
- We can also use HHL for non-Hermitian M: run HHL on $M_1 = \frac{1}{2}(M + M^\dagger)$ and $M_2 = \frac{1}{2i}(M - M^\dagger)$ (which are Hermitian) to give estimates $\hat{\mu}_1$ and $\hat{\mu}_2$, then combine to give $\hat{\mu} := \hat{\mu}_1 + i\hat{\mu}_2$.

[Assume that Hamiltonian simulation and phase estimation are exact, let $A = \sum_{i=1}^N \lambda_i |v_i\rangle\langle v_i|$, assume $\lambda_{\max} = 1$, and assume that $\kappa(A)$ is known or bounded $\leq \kappa_{\max}$. This means $|\lambda_i| \in [1/\kappa_{\max}, 1]$ for each i. Work in the n-qubit Hilbert space spanned by $\{|0\rangle, ..., |N-1\rangle\}$.

Write

$$|b\rangle = \sum_{i=1}^N b_i |i\rangle = \sum_{j=1}^N \beta_j |v_j\rangle$$

The solution vector to $Ax = b$ is $|\hat{x}\rangle := A^{-1}|b\rangle = \sum_{j=1}^N \beta_j \cdot \frac{1}{\lambda_j}|v_j\rangle$ (since $A^{-1} = \sum_{i=1}^N \frac{1}{\lambda_j}|v_j\rangle\langle v_j|$). The transformation $|b\rangle \mapsto A^{-1}|b\rangle$ is linear but not unitary so cannot be directly implemented. Instead, we implemented it probabilistically using QPE, performed on $U = e^{-iA}$, which in turn is implemented by Hamiltonian simulation. At the end, we'll have a measurement step that introduces the non-unitarity. Apply U_{PE} for e^{-iA} with m lines on the state $|b\rangle|0\rangle^{\otimes m}$, which gives $\sum_i \beta_i |v_i\rangle|\lambda_i\rangle$ (assuming e^{-iA} and U_{PE} are exact and error-free). Consider the controlled rotation $C\text{–Rot}$ acting on $n+1$ qubits: $C\text{–Rot}|\lambda\rangle|0\rangle = |\lambda\rangle(\cos(\theta)|0\rangle + \sin(\theta)|1\rangle) = |\lambda\rangle\left(\sqrt{1 - c^2/\lambda^2}|0\rangle + \frac{c}{\lambda}|1\rangle\right)$, with $\theta = \arcsin(c/\lambda)$ and $c \leq \min\{|\lambda_i| : i \in [N]\}$. Since $\lambda \in [1/\kappa, 1]$, $1/\lambda \in [1, \kappa]$ is larger than 1, so can choose $c = 1/\kappa$. So in $C\text{–Rot}$, the angle depends on the first register but not on A or b (so we're not sneaking extra info in here). $C\text{–Rot}$ can be implemented efficiently using $O(\text{poly}(n))$ one and two-qubit gates (by e.g. Solovay-Kitaev). Assume we can implement $C\text{–Rot}$ efficiently and exactly.

Applying $C\text{–Rot}$ to the state $U_{\text{PE}}|b\rangle|0\rangle$, we get

$$\sum_{j=1}^N \beta_j \sqrt{1 - c^2/\lambda_j^2}|v_j\rangle|\lambda_j\rangle|0\rangle + \beta_j \frac{c}{\lambda_j}|v_j\rangle|\lambda_j\rangle|1\rangle$$

Now we measure the last qubit, and accept if outcome is 1. This is called a post-selection step. The state collapses to a state proportional to

$$\sum_{j=1}^N \frac{c}{\lambda_j}\beta_j |v_j\rangle|\lambda_j\rangle|1\rangle$$

Probability of successfully preparing state $|x\rangle$ which is proportional to $A^{-1}|b\rangle$ is equal to probability of measurement outcome being 1, which is

$$\begin{aligned} p &= \left\| \sum_{j=1}^{N} \beta_j \frac{c}{\lambda_j} |v_j\rangle |\lambda_j\rangle \right\|^2 \\ &= \sum_{j=1}^{N} \left| \beta_j c/\lambda_j \right|^2 \\ &= \frac{1}{\kappa^2} \sum_{j=1}^{N} \left| \beta_j/\lambda_j \right|^2 \\ &\geq \frac{1}{\kappa^2} \sum_{j=1}^{N} \left| \beta_j \right|^2 = \frac{1}{\kappa^2} \end{aligned}$$

The post measurement state is

$$|\hat{x}\rangle = \frac{1}{\sqrt{p}} \sum_{j=1}^{N} \beta_j \frac{c}{\lambda_j} |v_j\rangle |\lambda_j\rangle$$

To boost the success probability to at least $1-\eta$, we can either repeat until the post selection step $O\Big(\frac{\log(1/\eta)}{p}\Big) = O(\log(1/\eta)\kappa^2)$ times, or use amplitude amplification:

$$U_{\text{HHL}}|b\rangle = \sqrt{1-p}|\text{junk}\rangle|0\rangle + \sqrt{p}|\hat{x}\rangle|1\rangle$$

Assuming we can also efficiently prepare $|b\rangle$, we can apply amplitude amplification to obtain a quadratic improvement: $O(\log(1/\eta)/\sqrt{p}) = O(\log(1/\eta)\kappa)$ measurements.

We can now estimate $\mu = x^T M x = \langle x|M|x\rangle$ by performing measurements on $\hat{x}$ in the M basis, and computing empirical averages: $\hat{\mu} = \langle \hat{x}|M|\hat{x}\rangle = \frac{c^2}{p}\langle x|M|x\rangle$.

By the Chernoff-Hoeffding bound, to estimate the mean $\hat{\mu}$ with probability at least $1-\eta$ to accuracy δ, we need $O\Big(\frac{\log(1/\eta)}{\delta^2}\Big)$ measurements.

To estimate p: the post-selection step is a Bernoulli trial, with probability of outcome 1 being p. So the mean is $0 \cdot (1-p) + 1 \cdot p = p$. This can also be estimated by the empirical average, and we use the Chernoff-Hoeffding bound. Alternatively, we can use amplitude amplification in the form of quantum counting (see find example) - this gives a quadratic improvement over the above.]

Remark 5.10 Runtime of HHL:

- The precision dependence in Hamiltonian simulation can be made $O(\log(1/\varepsilon))$, so we neglect it.
- Most of the complexity of HHL comes from QPE. Suppose we do QPE with m qubits of precision $\alpha = 1/2^m$. We need to figure out how to choose m and t to get ε-precision at the end of HHL, i.e. ε-precision for estimating μ.
- QPE uses $C\text{–}U$, ..., $C\text{–}U^{2^{m-1}}$. Can implemented $C\text{–}U = \begin{bmatrix} I & 0 \\ 0 & U \end{bmatrix}$ on $\tilde{A} = \begin{bmatrix} 0 & 0 \\ 0 & A \end{bmatrix}$. So we need to implement e^{-iAt} for $t = 1, 2, 4, ..., 2^{m-1}$. Since $(C\text{–}U)^{2^j} = e^{-iA\cdot 2^j}$, the

total time of evolution is $t_0 = 1 + 2 + \cdots + 2^{m-1} = O(2^m) = O(1/\alpha)$. After controlled rotation and post selection, we get

$$|\hat{x}'\rangle = \frac{1}{D'} \sum_{j=1}^{N} \beta_j \frac{c}{\lambda'_j} |v_j\rangle |\lambda_{j'}\rangle$$

where $D'^2 = p = \sum_{j=1}^{N} \left|\beta_j\right|^2 \left|c/\lambda'_j\right|$. We want $|\hat{x}'\rangle$ to be within ε of $|\hat{x}\rangle$. To establish dependence of t_0 on ε, we use two facts about relative errors:

1. $\mathrm{d}(1/\lambda) \approx \frac{1}{\lambda^2}\,\mathrm{d}\lambda$, so $\frac{\mathrm{d}(1/\lambda)}{1/\lambda} = \frac{\mathrm{d}\lambda}{\lambda}$
2. If A' and B' approximate A and B to relative error ξ, then $\frac{A'}{B'}$ approximates $\frac{A}{B}$ to relative error ξ: $\frac{A'}{B'} = \frac{A(1+\xi)}{B(1+\xi)} = \frac{A}{B}(1 + O(\xi))$.

So $\frac{1}{\lambda'_j}$ approximates $\frac{1}{\lambda_j}$ to relative error $O\left(\frac{\eta}{\lambda_j}\right) = O(\kappa\eta)$ since $\left|1/\lambda_j\right| \leq \kappa$. D' approximates D to relative error $O(\kappa\eta)$ as well, because D is homogenous in $\frac{1}{\lambda_j}$. Using (2) with $A' = \beta_j c/\lambda'_j$ and $B' = D'$, the amplitudes of $|\hat{x}'\rangle$ approximate those of $|\hat{x}\rangle$ to $O(\kappa\eta)$. Hence,

$$\| |\hat{x}'\rangle - |\hat{x}\rangle \| = \| (1 + O(\kappa\eta))|\hat{x}\rangle - |\hat{x}\rangle \| = O(\kappa\eta).$$

So we want $O(\kappa\eta) = \varepsilon$ so take $\eta = \frac{\varepsilon}{\kappa}$, so $t_0 = \frac{\kappa}{\varepsilon}$.

So the overall complexity is $O(\mathrm{poly}(n) \cdot t) = O\big(\mathrm{poly}(n) \cdot \kappa \cdot \frac{1}{\varepsilon}\big)$ (we would get κ^2 without using amplitude amplification).2

Theorem 5.11 (Chernoff-Hoeffding) For a random variable X on $[a, b]$ with mean μ, define the RV $\overline{X} = \frac{1}{k}(X_1 + \cdots + X_k)$, where the X_i are IID with same distribution as X. Then

$$\Pr(\left|\overline{X} - \mu) > \varepsilon\right| \leq e^{-2k\delta^2/(b-a)^2}.$$

Remark 5.12 HHL is an important algorithm as there are many applications of solving linear systems, for example:

- Numerical solutions of PDEs using discretisation leads to linear systems of size far larger than original problem description.
- Machine learning, pattern matching, etc.

6. Clifford computations and classical simulation of quantum computation

We want to know whether there is a "key quantum effect or resource" that gives quantum computing its (potential) benefits over classical computing?

To formalise this comparison of quantum vs classical computing, we will define a precise mathematical framework of classical simulation of quantum computation.

Problem 6.1 (Classical Simulation)

Input

- Description of a quantum circuit C as a list of 1- and 2- qubit gates acting on n qubit lines.

- Description of an input product state $|\alpha_1\rangle \otimes \cdots \otimes |\alpha_n\rangle$ (note this also has a poly(n)-sized classical description).
- Designated output qubit(s) line(s).

Promise

- C has size $N = |C| = \text{poly}(n)$.
- We only measure one qubit, for a decision answer.

Task By (randomised) classical means only, perform in poly(n) time either:

- **Weak simulation**: sample a bit from the output distribution of $C|\alpha_1\rangle ... |\alpha_n\rangle$ with the output qubit measured in the computational basis.
- **Strong simulation**: calculate the output probabilities $\Pr(\text{output is } 0) = p$. We want the ability to perform strong simulation to imply the ability to perform weak simulation, for multiple output qubits.

Definition 6.2 If C is classical simulable (in poly(n) time), then we say there is no **quantum advantage** (up to polynomial overheads).

Remark 6.3

- Any quantum process performs a weak simulation of itself, i.e. the final measurement gives a sample from the output distribution.
- Strong simulability is a much stronger property.
- "Direct" strong simulation is always possible, but generally not in polynomial time: the action of successive gates is simply matrix-vector multiplication in a 2^n-dimensional space, and so we can compute all amplitudes of the output state in $O(2^n)$ time. Although this direct simulation isn't efficient, it shows that any function computable by a quantum computer is also classically computable.

Theorem 6.4 If the state is promised to be a product of (single-qubit) states at each stage of the circuit (i.e. after each gate), then the direct strong simulation can be efficiently performed.

Proof. At each stage in the circuit, the state is of the form $|\psi\rangle = |\alpha_1\rangle ... |\alpha_n\rangle = \sum_{i_1,...,i_n \in \{0,1\}} c_{i_1 ... i_n} |i_1 ... i_n\rangle$, and each gate C acts on 2-qubits (some trivially on one of the qubits). Suppose the gate $U_{i_1 i_2}$ acts on the i_1-th and i_2-th qubits. The action of $U_{i_1 i_2}|\psi\rangle$ has amplitudes

$$\tilde{c}_{i_1,...,i_n} = \sum_{k_1,k_2 \in \{0,1\}} U_{i_1 i_2}^{k_1 k_2} c_{k_1 k_2 i_3 ... i_n}$$

But $|\psi\rangle$ is a product so $c_{i_1 ... i_n} = a_{i_1} b_{i_2} c_{i_3} \cdots x_{i_n}$, $i_1, ..., i_n \in \{0, 1\}$ and

$$\tilde{c}_{i_1,...,i_n} = \underbrace{\left(\sum_{k_1,k_2 \in \{0,1\}} U_{i_1 i_2}^{k_1 k_2} a_{k_1} b_{k_2} \right)}_{4\times 4 \text{ matrix multiplication}} \underbrace{c_{i_3} ... c_{i_n}}_{\substack{\text{all amplitudes of qubits} \\ \text{not affected by } U_{i_1 i_2} \text{ remain unchanged}}}$$

By assumption, the state $U_{i_1 i_2}|\psi\rangle$ is also a product state. So $\tilde{c}_{i_1,...,i_n} = \tilde{a}_{i_1} \tilde{b}_{i_2} c_{i_3} ... x_{i_n}$ factorises again. This can be done in constant time (since it's just a 4-dimensional vector). □

Remark 6.5 Note a state is a product state iff it has no entanglement. So sometimes claimed that entanglement is the source of quantum speedups. However, the converse is not true, i.e. entanglement is necessary but not sufficient.

6.1. Clifford computations

Definition 6.6 The 1-qubit Pauli group $\mathcal{P}_1$ is $\{\pm i, \pm 1\} \times \{I, X, Y, Z\}$ with $XY = iZ$ etc. The n-qubit Pauli group is $\mathcal{P}_n = \mathcal{P}_1 \otimes \cdots \otimes \mathcal{P}_n$.

Definition 6.7 A **Clifford operation** on n-qubits is a unitary $C \in U(2^n)$ which preserves the Pauli group under conjugation, i.e.

$$\forall P \in \mathcal{P}_n, \quad C^\dagger P C \in \mathcal{P}_n.$$

The **Clifford group** is the set of all Clifford operations - it is the normaliser of the subgroup $\mathcal{P}_n$.

Remark 6.8 Clifford groups are important in applications:

- Quantum error correction (e.g. stabiliser codes) and fault-tolerance.
- They give insights into the power of quantum vs classical computing.
- They form a metaplectic representation of symplectic groups.