Hidden Subgroup Problem

Simon's problem and Period finding are two black-box problems for which quantum computers provide exponential speedups. What else can quantum computers do? These two problems have a similar structure, and it is useful to recognize this common ground, because it suggests further generalizations.

More specifically, Simon's problem and Period finding are both special cases of a problem that is naturally formulated in group-theoretic language: the Hidden Subgroup Problem (HSP). This is a black-box problem where we may regard the input to the function \( f \) to be an element of a group \( G \), which is mapped into a set \( X \), which we may take to be the set of \( m \)-bit strings:

\[
f : G \to X = \{0, 1\}^m
\]

The group \( G \) may be either finite or infinite, but we ordinarily assume it is finitely generated. That is, each element of \( G \) can be expressed as a product of a finite set of generating elements, where these generating elements may be used any number of times in the product, and in any order. The set \( X \) is finite.

We are promised that the function \( f \) is constant and distinct on the cosets of a subgroup \( H \leq G \). This means that
\[ f(g_1) = f(g_2) \text{ iff } g_2^{-1}g_1 \in H \]

(that is, \( g_1 = g_2h \) for some \( h \in H \)). The problem is to find \( H \) to list a set of elements of \( G \) that generate \( H \).

We may take the input size for the HSP to be an upper bound on \( \log(\#G/H) \), the number of cosets (which is finite because \( X \) is finite).

The promise may restrict the hidden subgroup further by specifying additional properties of \( H \). For example, in the case of Simon's problem,

\[ G = \mathbb{Z}_2 \times \mathbb{Z}_2, \quad H = \mathbb{Z}_2 = \{ 0, a \} \]

The group \( \mathbb{Z}_2 \) is the set \( \{ 0, 1 \} \), where the group operation is addition modulo 2. A product group \( G_1 \times G_2 \) is defined as the group of pairs of elements

\[ G_1 \times G_2 = \{ (g_1, g_2) \mid g_1 \in G_1, g_2 \in G_2 \} \]

where the group operations are performed "in parallel"

\[ (g_1, g_2) \circ (g_1', g_2') = (g_1g_1', g_2g_2') \]

Thus, the elements of \( \mathbb{Z}_2 \times \mathbb{Z}_2 \) are \( n \)-bit strings of 0's and 1's, where the group operation is bitwise XOR

\[ (x_{n-1}, x_0) \circ (y_{n-1}, y_0) = (x_{n-1} \oplus y_{n-1}, x_0 \oplus y_0) \]

Each element is its own inverse (i.e., is order 2)
The promise in Simon's problem is:

\[ f(x) = f(y) \text{ if } x \oplus y \in \{0, a\} = H = \mathbb{Z}_2 \]

Here \( \{0, a\} \) is isomorphic to \( \mathbb{Z}_2 \). The problem is to determine how this \( \mathbb{Z}_2 \) is embedded in \( G = \mathbb{Z}^n \) - i.e., to find its generator \( a \). The number of possible embeddings is exponential in \( n \). The number of cosets (and so the number of possible outputs in the set \( X \)) is \( 2^{n-1} \), and its log is the input size.

Another example is period finding, for which

\[ G = \mathbb{Z} \text{ and } H = r\mathbb{Z} = \{rk, k \in \mathbb{Z}\} \]

The group operation is addition, and we are promised that

\[ f(x) = f(y) \text{ if } x - y = r \text{ integer } \in H \]

The problem is to find the generator of \( H \), namely, the period \( r \). The number of cosets of \( H \) is \( |G/H| = r \), and an upper bound on its log is the input size.

Classically, the HSP has query complexity \( \Omega(\sqrt{|G/H|}) \); we need to query this many times in order to get the same output in response to two different queries with reasonable probability. This is exponential in the input size - the problem is hard classically.
But for any finitely generated abelian group, the problem is easy quantumply! It can be solved (with high success probability) using $O(\text{poly} \log |G/H|)$ queries and $O(\text{poly} \log |G/H|)$ additional computational steps.

Before we explain the algorithm, let's discuss another application:

**Discrete Log Problem**

Recall that if $g$ is prime, then the group $\mathbb{Z}_g^*$ (multiplication mod $g$ with elements $\{1, 2, \ldots, g-1\}$) is cyclic. This means that $\mathbb{Z}_g^*$ is generated by a single element $\alpha$; thus,

$$\mathbb{Z}_g^* = \{ \alpha, \alpha^2, \alpha^3 - \alpha^{g-1} = 1 \}$$

Therefore any element $x \in \mathbb{Z}_g^*$ can be expressed in a unique way as the modular exponential $x = \alpha^y \pmod{g}$ where $y \in \{0, 1, 2, \ldots, g-2\}$

The discrete log mod $g$ with base $\alpha$ is the inverse of this function

$$x = \alpha^y \pmod{g} \iff y = \text{dislog}_{\mathbb{Z}_g^*}(x)$$

A discrete log can be defined this way for any cyclic group $G$ and any generating element of the group.
Example: $q = 7$, $\mathbb{Z}_q^* = \{1, 2, \ldots, 6\}$,

$\alpha = 5$ (or $\alpha = 3$) is generator

$x = 5^x \mod 7 = 1 5 4 6 2 3$

Inverse function is

$y = x^{log_5 5} = 0 4 5 2 1 3$

The modular exponential is easy to compute classically (by repeated squaring), but the discrete log seems to be hard to compute – the modular exponential is a candidate one-way function. It is hard to invert because $a^x$ jumps about in $\mathbb{Z}_q^*$ pseudorandomly as $x$ varies (for at least some values of $q$).

There are applications of this one-way function in cryptography; for example:

**Diffie-Hellman Key Exchange.**

This protocol's security rests on the presumed hardness of computing the discrete logarithm. The objective is for Alice and Bob to generate a shared secret key that is not known by their adversary Eve.
1. A prime number \( q \) and a generating element \( a \in \mathbb{Z}_q^\times \) are publicly announced.

2. Alice generates a random element \( x \in \mathbb{Z}_q^\times \) and keeps it secret. Bob generates random \( y \in \mathbb{Z}_q^\times \) and keeps it secret.

3. Alice computes and announces \( a^x \mod q \).
   Bob computes and announces \( a^y \mod q \).

4. Alice computes \((a^y)^x = a^{xy} \mod q\).
   Bob computes \((a^x)^y = a^{xy} \mod q\).
   This is their final shared key.

---

Alice and Bob can both compute the key because the modular exponential can be evaluated efficiently. The protocol is expected to be secure because even when \( a^x \) and \( a^y \) (but not \( x \) or \( y \)) are known, it is hard to compute \( a^{xy} \). Of course, if Eve can compute the discrete log, she could break the protocol. E.g., knowing \( a^x \) and \( a^y \) she could find \( x \) and then compute \((a^y)^x\).

And a quantum computer can evaluate a discrete log by solving a \( \text{HSP} \)! Here is how we would like to find

\[ y = \text{dlog}_a (x) - \text{the value of } r \text{ such that } x = a^r \mod q \]

We consider the function

\[ f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}_q^\times \]

\[ f(y_1, y_2) = a^{y_1} x^{-y_2} \mod q \]

When does \( f \) map two different inputs to the same output?
\[ f(y_1, y_2) = a_{y_1 - y_2} \mod q \]
\[ f(z_1, z_2) = a_{z_1 - z_2} \mod q \]

\[ y - y_2 = z_1 - z_2 \mod q - 1 \]
\[ \text{i.e.} \quad (y_1 - z_1) - R(y_2 - z_2) = 0 \mod q - 1 \]

This means that we may think of the input to \( f \) as an element of the additive group

\[ G = \mathbb{Z} \times \mathbb{Z} \]

where \( R \) is constant and distinct on the cosets of

\[ H = \{ (y_1, y_2) \mid y_1 = y_2 \mod q - 1 \} \]

\( H \) is generated by the elements \((1, 1), (q-1, 0)\). Since we find generators, we determine \( R \).

For an HSP problem with finitely generated \( G \), we may consider without loss of generality a corresponding problem with \( G = \mathbb{Z}^n \), since there is a homomorphism mapping \( G \) onto \( G \).

For example, in our original formulation of Simon's problem, we considered \( \tilde{G} = \mathbb{Z}^n \) where

\( \tilde{f}(x) = \tilde{f}(y) \) iff \( x \equiv y \mod 2 \), but instead we consider

\( G = \mathbb{Z}^n \) where

\[ x - y \equiv 0 \mod 2 \]
This means that the hidden subgroup is
\[ H = (2\mathbb{Z})^{n-1} \times (a\mathbb{Z}) \]

That is, the elements of \( H \) are
\[ \{ (m_1, a_1), (2m_1 + m_2, a_2), (2m_3 + m_4, a_3), \ldots, (2m_{n-1} + m_n, a_n) \} \]
where \( m_1, m_2, \ldots, m_n \in \mathbb{Z} \), and we assume
\[ m_1 \neq 0 \Rightarrow a_1 = 1 \]

In general, it is useful that we can give a geometrical interpretation to \( G \) and \( H \). 
\( G \) is the \( n \)-dimensional hypercubic lattice containing all ordered \( n \)-tuples of integers. The subgroup \( H \) can be regarded as a sublattice of \( \mathbb{Z}^n \). This sublattice is spanned by a set of \( n \) linearly independent vectors \( \{ v_1, v_2, \ldots, v_n \} \), each an element of \( \mathbb{Z}^n \) (i.e., with integer entries). A general element of \( H \) is a linear combination of the generating vectors
\[ x = \sum_{a=1}^{n} a \alpha_a v_a \]

We may construct an \( n \times n \) generator matrix \( M \) for the lattice \( H \) whose rows are the generating vectors:
\[ M = \begin{pmatrix} \alpha_1^T \alpha_2^T \cdots \alpha_n^T \end{pmatrix} \]
And then
\[ H = \{ x = aM, \alpha \in \mathbb{Z}^n \} \]
where \( \alpha \) is the row vector \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n) \).
For fixed \( H \), the generator matrix is not unique. We may make the replacement...
M → RM

where \( R \) is an invertible integral matrix with \( \det R = \pm 1 \) (so that \( R^{-1} \) is also integral).

Both \( M \) and \( RM \) are generators of the same lattice \( \mathbb{Z}^n \).

The quotient space \( \mathbb{Z}^n \), may be called the "unit cell" of the lattice. It contains all the distinct ways to "shift" the lattice \( \mathbb{Z}^n \) by an element of \( \mathbb{Z}^n \). We may say that \( 1/|H| \) is the volume of the unit cell, the number of points it contains. Note that

\[
\frac{1}{|H|} = \det M
\]

The linear transformation \( M \) inflates the cube \( [0,1]^n \) to a region of volume \( \det M \).

Corresponding to the integral lattice \( H \) is its dual lattice, "denote \( H^* \): the elements of \( H^* \) are points in \( \mathbb{R}^n \) that are orthogonal to all the vectors in \( H \), modulo integers:

\[
H^* = \left\{ \mathbf{k} \in \mathbb{Z}^n \mid \mathbf{v} \cdot \mathbf{k} \in \mathbb{Z} \text{ for all } \mathbf{v} \in H \right\}
\]

Equivalently, \( \exp(2\pi i \mathbf{k} \cdot \mathbf{x}) = 1 \) for \( \mathbf{k} \in H^* \) and \( \mathbf{x} \in H \).

\( H^* \) is also a lattice (i.e., its elements are the span of a set of generating vectors with integer coefficients), but the components are not necessarily integer (although they are rational numbers). If \( H^* \) is generated by vectors \( v_1, v_2, \ldots, v_n \), then its generating matrix is...
\[ M^+ = \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_n \end{pmatrix} \quad \text{and} \quad H^+ = \{ k = \beta M^+, \beta \in \mathbb{Z}^n \} \]

We can choose the basis for the dual lattice such that \( (C_a \cdot C_b) = \delta_{ab} \), in which case

\[ M^+M^T = I. \]

That means that, once we have found \( M^+ \), an easy computation determines \( M \) (matrix inversion or transpose of \( M^+ \)). In the quantum algorithm for the abelian HSP, the quantum computation determines the generator of \( H^+ \) (i.e., the matrix \( M^+ \)) and then finding the generator of \( H \) is easy (by matrix inversion).

For example, in the case of period finding, we have \( G = \mathbb{Z} \), \( H = r \mathbb{Z} = \{ x = r \alpha, \alpha \in \mathbb{Z} \} \) and \( H^+ = \{ k = \frac{\beta}{r}, \beta \in \mathbb{Z} \} \). In the quantum algorithm, we are promised \( \exists \beta \in \mathbb{Z} \) such that \( r \beta \equiv 1 \pmod{N} \); thus, rather than \( \mathbb{Z} \), we used the quantum Fourier transform for \( \mathbb{C}^N \) where \( N \approx 2^k \). Fourier sampling then provided sufficient accuracy to determine an element \( \beta/r \in H^+ \) with high success prob. After a few samples, we could determine \( \frac{1}{r} \), the generator of \( H^+ \), and hence \( r \), the generator of \( H \). We want to extend this idea from subgroups of \( \mathbb{Z} \) to subgroups of \( \mathbb{Z}^n \).

So, instead of \( \mathbb{Z}^n \), suppose we consider \( \mathbb{Z}^N \) for some sufficiently larger \( N \). And to keep the discussion simple at first, suppose that \( H \) is actually a subgroup of \( \mathbb{Z}^N \) rather than of \( \mathbb{Z}^n \).
As in the period finding algorithm, we query the black box with
\[ \frac{1}{\sqrt{|G|}} \sum_{x \in G} |x\rangle \] and so obtain \[ \frac{1}{\sqrt{|G|}} \sum_{x \in G} |x\rangle \otimes |f(x)\rangle \]
where \( f \) is a constant and distinct on the coset \( y \in G \). We would prepare in the input register the uniform superposition of elements in the same coset as \( x_0 \), which is
\[ |1_{H}, x_0\rangle = \frac{1}{\sqrt{|H|}} \sum_{x \in H} |x + x_0\rangle \]
This state has an important property: it is \( H \)-invariant. We may consider the unitary transformation \( U_y \) associated with an element \( y \in G \) whose action is
\[ U_y: |x\rangle \mapsto |x + y\rangle \]
We note that, for \( y \in H \), the "coast state" \( |1_{H}, x_0\rangle \) is invariant under \( U_y \), because
\[ U_y |1_{H}, x_0\rangle = \frac{1}{\sqrt{|H|}} \sum_{x \in H} |y + x + x_0\rangle \]
and we may reparametrize the sum over \( x \) by replacing \( x \mapsto x - y \), thus obtaining
\[ \frac{1}{\sqrt{|H|}} \sum_{x \in H} |x' + x_0\rangle = |1_{H}, x_0\rangle \]
To appreciate the significance of \( H \)-invariance, note that if \( |14\rangle \) obeys \( U |14\rangle = |14\rangle \), then
\[ U |14\rangle = (V U V^{-1}) |14\rangle = |14\rangle \]
where \( |14\rangle = V |14\rangle \)
Now apply this identity to \( U = U_0 \) and \( V = F \), that is

\[
V: |x\rangle \mapsto \frac{1}{\sqrt{161}} \sum_{k \in \mathbb{Z}} e^{2\pi i k \cdot x / N} |k\rangle
\]

\[
V^{-1}: |k\rangle \mapsto \frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} e^{-2\pi i k \cdot x / N} |x\rangle
\]

Then \( U_0 |k\rangle = e^{2\pi i k \cdot y / N} |k\rangle \)

\[
V^{-1} |k\rangle \mapsto \frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} e^{-2\pi i k \cdot x / N} |x+y\rangle
\]

\[
\frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} e^{-2\pi i k \cdot x / N} |x\rangle
\]

\[
= e^{2\pi i k \cdot y / N} \frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} e^{-2\pi i k \cdot x / N} |x\rangle
\]

\[
\frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} e^{-2\pi i k \cdot x / N} |1k\rangle
\]

Therefore, the state \(|k\rangle\) is invariant under \( U_0 \) if \( \frac{1}{N} k \cdot y = \text{integer} \) — or, for \( y \in \mathbb{H} \), if \( \frac{k}{N} \in \mathbb{H}^+ \).

Thus, if a state is \( H\)-invariant, then in the Fourier basis its expansion contains \( k \) \( \frac{1}{N} \) \( k \cdot H \) with a nonzero coefficient only if

\[
\frac{k}{N} \in \mathbb{H}^+
\]

More explicitly, we compute

\[
|H, x\rangle = \frac{1}{\sqrt{161}} \sum_{x \in \mathbb{H}} |x\rangle
\]

\[
\frac{1}{\sqrt{161}} \sum_{x \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} e^{2\pi i k \cdot x / N} |k\rangle
\]

Because of \( N\)-invariance, only \( \frac{k}{N} \in \mathbb{H}^+ \) survives in the sum,

and, for such \( k \), \( e^{2\pi i (k \cdot x) / N} = 1 \), and we obtain
\[
\frac{1}{\sqrt{\det H^+}} \sum_{k \in H^+} e^{2\pi i \langle x, k \rangle} |k> 
\]

Therefore, if we "Fourier sample" - i.e. Fourier transform and then measure, the prob. distribution that governs the outcome is the uniform distribution on \( H^+ \) once we have sampled from \( H^+ \) enough times, with high probability a generating set for \( H^+ \) will be found.

How many samples are enough (assuming now \( \det G \) is finite - e.g. \( G = \mathbb{Z}_N \) - and \( H \leq G \)?)

Suppose \( K \) is a group (e.g. abelian or not), and \( m \) elements of \( K \) are chosen uniformly at random. If these \( m \) elements do not generate \( K \), then they must be contained in some maximal proper subgroup \( S \subseteq K \).

(Proper means \( S \) is smaller than \( K \), and "maximal" means we cannot add another element of \( K \) to \( S \) without generating all of \( K \). Any proper subgroup has order \( |S| \leq |K|/2 \), because the order of the subgroup must divide the order of \( K \), and the probability that all \( m \) elements are in \( S \) is

\[
\text{Prob (all in } S) = \left( \frac{|S|}{|K|} \right)^m 
\]

and therefore the prob. that the \( m \) elements generate \( K \) is

\[
\text{Prob (\text{elements generate } K)} \geq 1 - \sum_{S \text{max}} \left( \frac{|S|}{|K|} \right)^m 
\]

where the sum is over maximal proper \( S \)

\[
\geq 1 - (\# \text{max}) \cdot 2^{-m}
\]
That can be expressed as \( \frac{\text{integer}}{\text{odd } N} \), where \\
\[ \det M = \frac{16}{|H|}, \] the number of cosets. In the formulation of the HSP, we are provided with an upper bound \( |G/H| \leq R \), and \( N \) needs to be large enough to point to a unique rational number with denominator \( \leq R \), with reasonable success probability.

In our discussion of period finding, \( (H^+ = \frac{1}{k}) \), we noted that Fourier sampling yields a rational number \( y/N \) close to \( \text{integer}/k \) with high prob:

\[ \sum_{k} \text{Prob} \left( \left| \frac{y}{N} - \frac{k}{N} \right| \leq \frac{s}{2N} \right) \geq \frac{4}{4e} \]

so that choosing \( N > R \) was good enough to determine a rational number with denominator \( \leq R \). If we fix the desired accuracy \( s \), then the proportion of the distribution lying outside the peaks decreases as \( N \) increases (an exercise):

\[ \text{Prob} \left( \forall k, \left| \frac{y}{N} - \frac{k}{N} \right| > s \right) \leq \frac{1}{NS} \]

The peak of the Fourier transform shrinks with increasing \( N \), so that the prob of lying outside all peaks with half width \( s \) scales like \( 1/N \).

When we sample \( H^+ \), we find an \( n \)-component vector, where each component should be determined to accuracy \( \sqrt{R} \) (where \( |G/H| \leq R \)). The probability of success in finding all \( n \) components is then accuracy \( 1/R^m \).
where \((H_{\text{max}})\) denotes the total # of maximal proper subgroups.

If \(K\) is abelian, we can count the maximal proper subgroups. \(S\) is a sublattice of \(K\), and if \(S\) is a maximal proper subgroup, then its dual lattice \(S^\perp\) contains a vector not in \(K^\perp\), there is only one such (linearly independent) vector if \(S\) is maximal, so if there were two then we could remove one, obtaining a smaller \(S^\perp\) and such a larger proper subgroup. Any nontrivial vector not in \(K^\perp\) determines such a subgroup, so there are \(|G/K^\perp| - 1\) choices (where e.g. \(G = \mathbb{Z}_p^n\)), and

\[
\text{Prob (element generates \(S\))} \approx 1 - 2^{-m}|G/K^\perp|
\]

So in the case of the hidden subgroup problem where \(H \leq G = \mathbb{Z}_p^n\), we are sampling \(K = H^\perp\) and \(16/K^\perp\) becomes \(16/|H|\), the number of cosets, to have constant success probability, how do we choose \(m\) such that e.g.

\[
2^{-m} \frac{16/|H|}{2} \leq \frac{1}{2}
\]

or \(m \geq \log(16/|H|)\) (compare the conclusion for Simon's problem) and \(16/|H| < N^4\), so it suffices if \(m = O(\log \log N)\).

How large should \(N\) be? For period finding with \(m \geq \log \log N\), choosing \(N \geq 2^\sqrt{\log \log N}\) provides adequate precision for finding \(r\). For an integral lattice with generating matrix \(M\), its inverse matrix (transpose of \(M^{-1}\)) has entries
\[ P(\text{success}) \geq (1 - \frac{1}{NS})^n \]

and the prob of being successful in each of \( n \) consecutive samplings is

\[ P(\text{success in Time}) \geq (1 - \frac{1}{NS})^nm \]

For \( S = \frac{1}{R^2} \), the success probability is a constant for

\[ \frac{mnR^2}{N} < \text{constant} \quad \text{or} \quad N = O(mnR^2) \]

Since \( m = O(n \log N) \) samples are sufficient to find generators of \( H^\perp \), we conclude it suffices to choose \( N < \frac{15}{16} \)

\[ N = O(n^2 R^2 \log N) = O(n^2 R^2 \log(nR)) \]

This is good enough to determine \( H^\perp \) in

\[ m = O(n \log N) = O(n \log(nR)) \]

queries, and the generators of \( H \) are found by inverting the matrix \( M^\perp \) that generates \( H^\perp \).

The algorithm is efficient to both the number of queries and the number of steps in the quantum Fourier transform. The number of cosets \( |\Gamma/H| \)