Quantum algorithms **Monsoon 2007**

Lecture 10

Lecturer: Pranab Sen *Date: August 30*

In the last lecture, we saw how characters of a finite abelian group G may play a role in identifying the hidden subgroup H from a uniform superposition over a coset of H. In this lecture, we build on this idea and show how to solve the HSP in G efficiently using the quantum Fourier transform QFT_G over G as a black box. In the next two lectures, we shall show how to implement good approximations to QFT_G in time $\mathrm{polylog}|G|$.

1 Fourier sampling the coset state

Let G be a finite abelian group given as a direct product of cyclic groups. The elements of G are represented as tuples according to its cyclic decomposition. Suppose we have a black box performing QFT_G. Let $f : G \rightarrow S$ be given via the standard reversible oracle $F(f) : |g\rangle|s\rangle \mapsto |g\rangle|s \oplus f(g)\rangle$ for all $g \in G$ and $s \in S$. Suppose f hides the subgroup $H \leq G$. The basic quantum algorithm for finding H is given below.

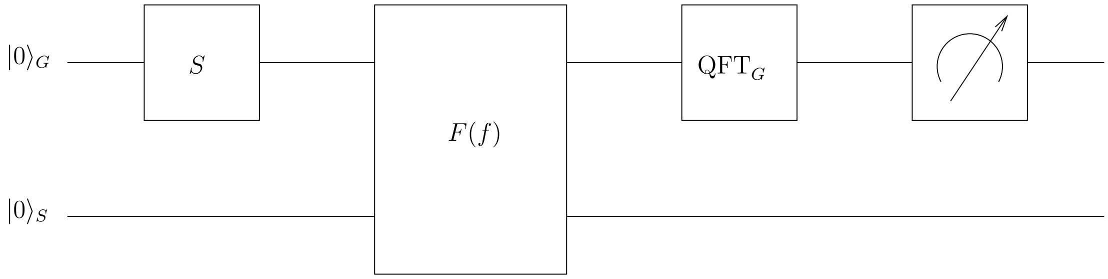

Figure 1. Fourier sampling the coset state of H in G

The transform S generates the uniform superposition over all elements of G, given the identity element $|0\rangle$ of G as input. We shall see below how S can be implemented with $\mathrm{polylog}|G|$ gates. The state of the first register after the application of $F(f)$ is nothing but the coset state σ_H^G of H in G. Consider the uniform superposition over a coset $x + H$, $x \in G$. Applying QFT_G to $|x + H\rangle$ gives

$$\begin{aligned}
\mathrm{QFT}_G(|x+H\rangle) &= \frac{1}{\sqrt{|H|}} \sum_{h \in H} |\chi_{x+h}\rangle = \frac{1}{\sqrt{|G| \cdot |H|}} \sum_{y \in G} \sum_{h \in H} \chi_{x+h}(y)|y\rangle \\
&= \frac{1}{\sqrt{|G| \cdot |H|}} \sum_{y \in G} \chi_x(y) \left(\sum_{h \in H} \chi_h(y) \right) |y\rangle \\
&= \sqrt{\frac{|H|}{|G|}} \sum_{y \in H^\perp} \chi_x(y)|y\rangle.
\end{aligned}$$

The third equality above follows because $\chi_{x+h}(y) = \chi_y(x+h) = \chi_y(x)\chi_y(h) = \chi_x(y)\chi_h(y)$. This shows that the Fourier transform moves the information about the particular coset into the phases of the basis states.

Thus, applying QFT_G to the coset state ρ_H^G of H in G followed by measuring in the computational basis gives us a uniformly random element of $H^\perp$. Repeating this procedure $k := O(\log |G|)$ times gives independent uniform samples $y_1, \ldots, y_k \in H^\perp$. With high probability, $H^\perp = \langle y_1, \ldots, y_k \rangle$. In order to see this, define an ascending chain $Y_0 \leq Y_1 \leq \cdots Y_k$ of subgroups of G, where $Y_0 := \{0\}$ and $Y_i := \langle y_1, \ldots, y_i \rangle$ for $1 \leq i \leq k$. If $Y_i < H^\perp$ for some i, then with probability at least $1/2$, $y_{i+1} \notin Y_i$, in which case $|Y_{i+1}| \geq 2|Y_i|$. From this, it can be shown via elementary probability theory that for $k \geq \Omega(\log |H^\perp|)$, $Y_k = H^\perp$ with high probability. We shall see below how one can recover H from $H^\perp$.

2 Implementing S efficiently

We now see how to implement a $\mathrm{polylog}|G|$-sized quantum circuit for the unitary operator S that maps the identity element $|0\rangle$ to the uniform superposition over all group elements $|G\rangle$. Since G is given via a cyclic decomposition and its elements are represented as tuples according to this decomposition, it suffices to be able to implement S for the case of $G = \mathbb{Z}_n$. Our circuit will use single qubit unitaries whose entries depend on n; in fact, the entries can be estimated to within δ in deterministic time $\mathrm{polylog}(n, 1/\delta)$. These single qubit unitaries can be approximated by the Solovay-Kitaev theorem to within spectral distance ϵ by $\mathrm{polylog}(1/\epsilon)$-sized circuits made up of Hadamard and $\pi/8$-gates, and these approximating circuits can also be constructed in time $\mathrm{polylog}(n, 1/\delta, 1/\epsilon)$.

Our circuit for implementing S for $G = \mathbb{Z}_n$ is given below. The main idea behind our circuit for implementing S for $G = \mathbb{Z}_n$ is a special case of a more general principle called *exact amplitude amplification* which we shall study in a later lecture. Let $m := \lceil \log n \rceil$ be the number of qubits used to store elements of $\mathbb{Z}_n$; $2^{m-1} < n \leq 2^m$. Define the single qubit unitary R_n by

$$R_n : |0\rangle \mapsto \sqrt{\frac{2^{m-2}}{n}}|0\rangle + \sqrt{1 - \frac{2^{m-2}}{n}}|1\rangle, \quad |1\rangle \mapsto \sqrt{1 - \frac{2^{m-2}}{n}}|0\rangle - \sqrt{\frac{2^{m-2}}{n}}|1\rangle.$$

Let M be the unitary that flips the phase of its input basis state if the first qubit is $|0\rangle$ and the remaining m qubits encode a number less than n, that is,

$$M : |b\rangle|i\rangle = \left\{ \begin{array}{ll} -|b\rangle|i\rangle & \text{if } b = 0 \text{ and } i < n \\ |b\rangle|i\rangle & \text{otherwise} \end{array} \right. .$$

Let M' denote the unitary that flips the phase of its basis state input iff it is not all zeroes, that is,

$$M' : |b\rangle|i\rangle = \left\{ \begin{array}{ll} |b\rangle|i\rangle & \text{if } b = 0 \text{ and } i = 0 \\ -|b\rangle|i\rangle & \text{otherwise} \end{array} \right. .$$

Let $\mathcal{A} := R_n \otimes H^{\otimes m}$, where H is the single qubit Hadamard gate; observe that $\mathcal{A}^\dagger = \mathcal{A}$. The circuit for implementing S for $G = \mathbb{Z}_n$ is $\mathcal{A}M'\mathcal{A}M\mathcal{A}$ applied to the all-zeroes initial state.

We now trace through the operation of the circuit.

$$\begin{aligned}
\mathcal{A}M'\mathcal{A}M\mathcal{A}(|0\rangle|0\rangle) &= \mathcal{A}M'\mathcal{A}M\left(\left(\frac{1}{2\sqrt{n}}\,|0\rangle + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle\right) \otimes \sum_{i=0}^{2^m-1} |i\rangle\right) \\
&= \mathcal{A}M'\mathcal{A}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle\right) \\
&= \mathcal{A}M'\mathcal{A}\left(\frac{1}{2}\left(\frac{1}{2\sqrt{n}}\,|0\rangle + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle\right) \otimes \sum_{i=0}^{2^m-1} |i\rangle + \right. \\
&\qquad \frac{1}{2}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \right. \\
&\qquad \left.\left. \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle\right)\right) \\
&= \mathcal{A}M'\left(\frac{1}{2}|0\rangle|0\rangle + \mathcal{A}\left(\frac{1}{2}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \right.\right.\right. \\
&\qquad \left.\left.\left. \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle\right)\right)\right) \\
&= \mathcal{A}\left(\frac{1}{2}|0\rangle|0\rangle - \mathcal{A}\left(\frac{1}{2}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \right.\right.\right. \\
&\qquad \left.\left.\left. \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle\right)\right)\right) \\
&= \frac{1}{2}\left(\left(\frac{1}{2\sqrt{n}}\,|0\rangle + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle\right) \otimes \sum_{i=0}^{2^m-1} |i\rangle\right) - \\
&\qquad \frac{1}{2}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle\right) \\
&= \frac{1}{\sqrt{n}}\sum_{i=0}^{n-1} |i\rangle.
\end{aligned}$$

The fifth equality above follows from the fact that

$$\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1} |i\rangle + \sum_{i=n}^{2^m-1} |i\rangle\right) + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1} |i\rangle$$

is orthogonal to

$$\mathcal{A}(|0\rangle|0\rangle) = \left(\frac{1}{2\sqrt{n}}\,|0\rangle + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle\right) \otimes \sum_{i=0}^{2^m-1} |i\rangle,$$

so

$$\mathcal{A}\left(\frac{1}{2\sqrt{n}}\,|0\rangle \otimes \left(-3\sum_{i=0}^{n-1}|i\rangle + \sum_{i=n}^{2^m-1}|i\rangle\right) + \sqrt{\frac{1}{2^m} - \frac{1}{4n}}\,|1\rangle \otimes \sum_{i=0}^{2^m-1}|i\rangle\right)$$

is orthogonal to $|0\rangle|0\rangle$, that is, it is a superposition of basis states other than the all zeroes basis state.

Using standard techniques of clean deterministic reversible computation, we can implement the unitary operator using Toffoli gates and ancilla qubits. We can convert $\hat{M}$ to M by the standard trick of converting a bit flip function oracle into a phase flip oracle via Hadamard gates. A similar thing can be done in order to implement M'. The Toffoli gates can be replaced by the following circuit made up of single qubit gates and CNOT, which is an example of a general technique to implement multiply controlled unitaries by circuits of single qubit unitaries and CNOTs (see Chapter 4, Nielsen and Chuang's book for more details).

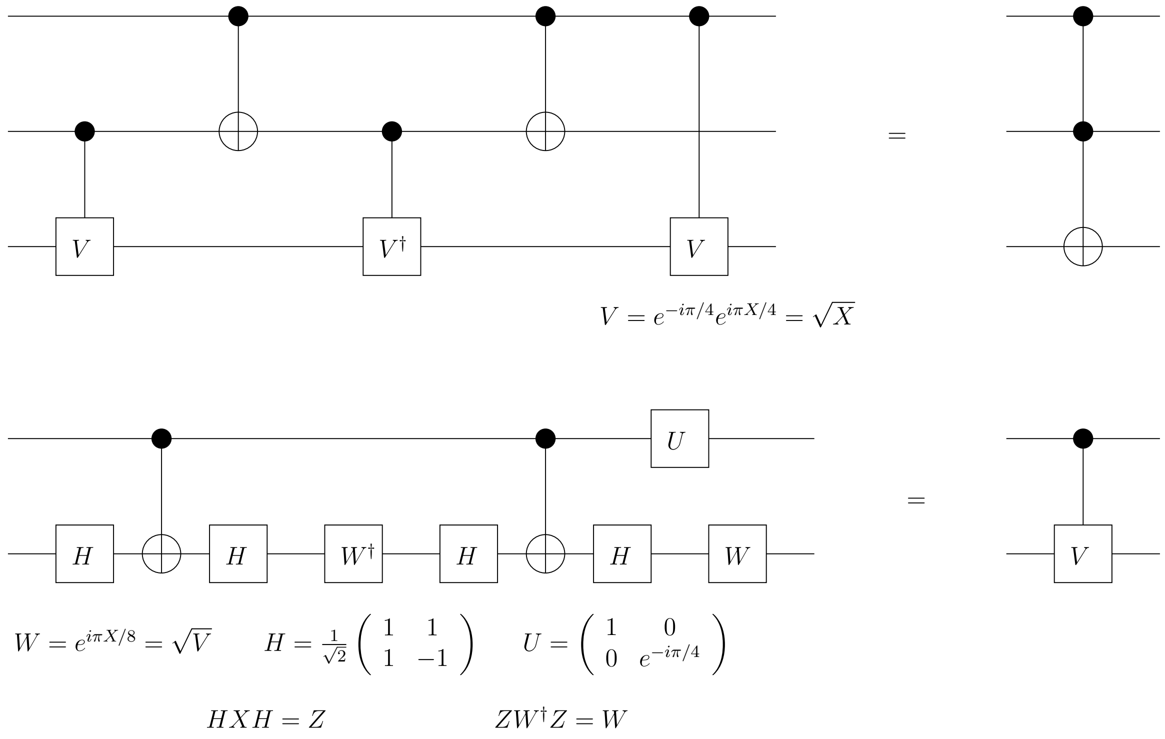
V
$V^\dagger$
V
=
$V = e^{-i\pi/4}e^{i\pi X/4} = \sqrt{X}$
U
H
H
$W^\dagger$
H
H
W
=
V
$W = e^{i\pi X/8} = \sqrt{V}$
$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
$U = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{pmatrix}$
$HXH = Z$
$ZW^\dagger Z = W$

Figure 2. Implementing Toffoli by single qubit gates and CNOTs

Overall, M, M' can be implemented using $O(m) = O(\log n)$ single qubit gates and CNOTs, and as a result, S can be implemented with a similar amount of resources.

3 Recovering H from $H^\perp$

We now see how to recover H from a generating set $y_1, \ldots, y_k$ of $H^\perp$ by a polynomial time deterministic algorithm. Suppose G is given via the cyclic decomposition $G = Z_{n_1} \times \cdots \times Z_{n_l}$ and elements of G

are represented as l-tuples. Evaluating a character χ_y, $y = (y_1, \ldots, y_l) \in G$ at a group element $x = (x_1, \ldots, x_l) \in G$ reduces to computing $\chi_y(x) = \exp(\frac{2\pi i}{|G|} \sum_{j=1}^{l} \frac{|G|}{n_j} x_j y_j)$. Then, recovering $H = (H^{\perp})^{\perp}$ from $H^{\perp} = \langle y_1, \ldots, y_k \rangle$ amounts to solving the set of equations

$$|G|\, z_i + \sum_{j=1}^{l} \frac{|G|\, y_{i,j}}{n_j}\, x_j = 0, \quad 1 \leq i \leq k,$$

where $(x_1, \ldots, x_l)$ is the l-tuple of integer valued unknowns representing elements of H, $(y_{i,1}, \ldots, y_{i,l}$ is the l-tuple representation of y_i, $1 \leq i \leq k$ and $z_1, \ldots, z_k$ are additional integer valued unknowns. A integer basis for the solution space of this system can be obtained by putting the constraint matrix into the so-called *Smith normal form* by elementary row and column operations over the integers. Thus, every integer solution to this system is an integer linear combination of the basis solutions. It is easy to see that the integer solutions to this system are in bijection with the elements of H. Hence, we obtain a generating set for H.