Quantum Computing - Problem Set 2

Mateus de Oliveira Oliveira

School of Computer Science and Communication
KTH Royal Institute of Technology, 100-44, Stockholm, Sweden
mdeoliv@kth.se

1 QFT over $\mathbb{Z}_n$

Let $\mathbb{Z}_N$ denote the group of integers with addition modulo N. Then the Quantum Fourier Transform over $\mathbb{Z}_N$ is defined by the following action on basis states:

$$QFT_N|x\rangle = \frac{1}{\sqrt{N}} \sum_y e^{\frac{2\pi i y x}{N}} |y\rangle$$

1. Prove that the inverse Fourier transform is given by

$$QFT_N^{-1}|x\rangle = \frac{1}{\sqrt{N}} \sum_y e^{\frac{-2\pi i y x}{N}} |y\rangle$$

2. Prove that QFT is an unitary transformation.
3. Draw the circuit for the Fourier transform over Z_{16}. (Lecture 4).
4. Devise a quantum circuit that implements the operation $|x\rangle \rightarrow |x+1 \mod N\rangle$ for a fixed y, using only the Fourier transform, its inverse and phase gates.

2 Hidden Subgroup Problem Over Abelian Groups

Let $G = \mathbb{Z}_{N_1} \oplus \mathbb{Z}_{N_2} \oplus ... \oplus \mathbb{Z}_{N_k}$ be an Abelian group. Write $g = (g_1, g_2, ..., g_k)$ for an element of G and $g + h = (g_1 + h_1, g_2 + h_2, ..., g_k + h_k)$ where for each coordinate i the addition is done modulo N_i. Recall that

$$\chi_y(g) = e^{2\pi i \sum_{j=1}^{k} \frac{y_j g_j}{N_j}}$$

and that

$$QFT_G|g\rangle = \frac{1}{\sqrt{|G|}} \sum_{y \in G} \chi_y(g)|y\rangle$$

1. Write explicitly the action of the QFT over $Z_2 \oplus Z_3$ on each of the basis states $|0\rangle|0\rangle$, $|1\rangle|0\rangle$, $|0\rangle|1\rangle$, $|1\rangle|1\rangle$,$|0\rangle|2\rangle$,$|1\rangle|2\rangle$.
2. Draw the circuit for the QFT over $Z_2 \oplus Z_8$

As a crucial step in determining the Hidden subgroup H (Lecture 5) we needed to sample from the subgroup $H^\perp$ defined as

$$H^\perp = \{y \in G | \forall h \in H \sum_{i=1}^{k} \frac{y_i h_i}{N_i} \in \mathbb{Z}\} \tag{1}$$

3 Show that $H^\perp$ is indeed a subgroup of G.
4 Recall that Simon's algorithm can be modelled as a Hidden subgroup problem (Lecture 4). Who is $H^\perp$ in that case?
5 After having sampled enough elements to generate $H^\perp$ in Simon's algorithm we have a system of linear equations that will determine that will determine H. In the general Abelian HSP, suppose you have already sampled enough elements to generate $H^\perp$. How do you determine generators for the group H?

3 Grover's Algorithm

Let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function such that there are exactly m elements $a_1, ..., a_m$ for which $f(a_m) = 1$. We want to find some a_i in this list. We saw that to apply Grover's algorithm to this goal we need to estimate m so that we can determine the right number of times the iteration WV must be applied. (Lecture 6). Let $|\psi\rangle = \frac{1}{\sqrt{m}} \sum_{i=1}^{m} |a_i\rangle$ and $|\overline{\psi}\rangle = \frac{1}{\sqrt{2^n - m}} \sum_{x \notin \{a_1, \ldots, a_m\}} |x\rangle$.

1. Write the state $|\phi\rangle$ in terms of $|\psi\rangle$ and $|\overline{\psi}\rangle$.
2. WV has two eigenvectors. Write them in the basis $|\psi\rangle$ and $|\overline{\psi}\rangle$. What are their eigenvalues?
3. How the eigenvalue estimation algorithm can be used to determine the number of applications needed to rotate the state $|\phi\rangle$ close to $|\psi\rangle$?

4 Fourier Transform For General Groups

The goal of this question is to work out the fact that the hidden subgroup problem has polynomial time query complexity for any finite group G, i.e., not only for Abelian groups. That means that if we are given a function $f : G \to S$ with the promise that there is a subgroup K such that $f(x) = f(y)$ if and only if $Kx = Ky$ then we may determine K with polynomially many queries to f. This does not imply that the we can solve it in BQP (bounded quantum polynomial time) since an we will need to do an iteration over an exponential number of subgroups. Below we assume that G has N elements.

1. Describe how to create the state

$$|\psi\rangle = \frac{1}{\sqrt{|G|}} \sum_{g_1, g_2, \ldots, g_k \in G} |g_1 g_2 \ldots g_k\rangle |f(g_1) f(g_2) \ldots f(g_k)\rangle$$

(Observation: in the notation above $g_1 g_2 ... g_k$ denotes a sequence of distinct elements of G which can be non-Abelian, not the cyclic components of an element of G as in Question 2.)

2. Let $K_1, K_2, ..., K_s$ be an ordering of all subgroups of G satisfying $|K_i| \geq |K_{i+1}|$. Since G has N elements, any of its subgroups may be generated by $n = O(\log N)$ elements. In view of this fact, how many subgroups can G have? How many bits are needed to index all the subgroups unequivocally?
3. The idea of the proof is to show that for each subgroup K_i in the above list there is a function $Test_i$ such that if the hidden subgroup that we are searching for is K_i then $Test_i|0\rangle|0\rangle|\psi\rangle = |i\rangle|1\rangle|\psi\rangle$, i.e. the index of the subgroup will magically appear in the first register, while if K_i is not the hidden subgroup in question, then the state will be kept almost invariant, i.e., $Test_i|0\rangle|0\rangle|\psi\rangle$ will be almost parallel to $|0\rangle|0\rangle|\psi\rangle$. For the moment forget the "almost" and consider that in the latter case the application of $Test_i$ leaves the state completely invariant. What algorithm would you suggest to solve the HSP? (Don't try anything complicated. If you arrived here you already know the answer.)
4. Another caveat of $Test_i$ is the fact that it is never applied to the function register. Why does this imply that the query complexity of the HSP is polynomial?
5. Back to the "almost": Assume that if K_i is not the hidden subgroup, then

$$\|Test_i|0\rangle|0\rangle|\psi\rangle - |0\rangle|0\rangle|\psi\rangle\|_2 \leq \frac{1}{2^{O(s)}}$$

   where $\|\cdot\|$ denotes the L_2 norm of a vector. After having tested j subgroups and assuming that that nothing showed up in the first register, how far is the state $Test_j Test_{j-1} ... Test_1|0\rangle|0\rangle|\psi\rangle$ from the original state $|0\rangle|0\rangle|\psi\rangle$? In other words what is their distance in L_2 norm? (Your most obvious guess will be the correct. But you have to prove it formally.)
6. Show that $k = O(log|G|)$ is enough to make the algorithm you suggested in the previous item work.
7. (You don't need to solve this last item. It is here just to make you want to read the paper where this result first appeared.) Who is $Test_i$? □