Lecture & Proseminar 250078/250042
"Quantum Information, Quantum Computation, and Quantum Algorithms" WS 2021/22

— Exercise Sheet #11 —

Problem 28: Factoring 15

Verify the factoring algorithm (i.e., the reduction to period finding described in the lecture – subection 3.c) for $N = 15$ – i.e., consider all $a = 2, \ldots, N-1$, check wether $\gcd(a, N) = 1$, find r s.th. $a^r \bmod N = 1$ (you don't have to use a quantum computer), and check if this can be used to compute a non-trivial factor of N. How many different cases do you find? What possible periods r appear?

Problem 29: Grover's algorithm with multiple marked elements.

Consider the Grover search problem of finding x_0 such that $f(x_0) = 1$ for a given function $f : \{0, N-1\} \to \{0, 1\}$. In the lecture, we derived Grover's algorithm which finds x_0 given that it is unique. In this problem, we will derive a generalization of Grover's algorithm which allows to tackle the search problem in the case where there are $K > 1$ solutions x to the equation $f(x) = 1$. The goal is to find one x with $f(x) = 1$ with high probability.
The oracle is constructed the same way as before, i.e., it acts as

$$O_f = \mathbb{I} - 2 \sum_{x:f(x)=1} |x\rangle\langle x| \ .$$

The algorithm proceeds the same way as before, namely, by starting in the state $|\omega\rangle$ (given in the lecture), repeatedly applying Grover iterations $G = -O_\omega O_f$ (with O_ω as in the lecture), and finally measuring in the computational basis.

1. Show that O_f can be obtained from $U_f : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$.
2. Show that the Grover iteration G leaves the space $\mathcal{S} = \mathrm{span}(|\omega\rangle, |x_0\rangle)$ invariant, where $|\omega\rangle$ is as in the lecture, and
$$|x_0\rangle \propto \sum_{x:f(x)=1} |x\rangle \ .$$
3. What is the action of G on a state in $\mathcal{S}$?
4. For a given number of solutions K, how many times do we have to apply G to get a good overlap with $|x_0\rangle$? What result will we get when measuring in the computational basis?
5. Compare this to the scaling of the classical algorithm (i.e. trying random x until a solution is found).

Problem 30: Quantum counting.

Consider the same setting and notation as in Problem 29. Here, we will use a combination of Grover iterations G and phase estimation (Problem 27 on Sheet #10) to estimate ("count") the number K of solutions up to some error δK. Our goal will be to understand how the accuracy δK scales with the number Q of queries to f (or U_f).

1. First, determine the scaling δK for classical counting: Since we assume that f is a black-box function, the best we can classically do is to sample Q random values x_i, $i = 1, \ldots, Q$, compute $f(x_i)$, and use this to estimate K. What is the error δK as a function of Q (and K, N)?
2. We will now construct a quantum algorithm for estimating K. First, determine the eigenvalues $e^{i\theta_k}$, $k = 1, 2$, of G restricted to the subspace $\mathcal{S}$. (This is most easily done by observing that G is a rotation by an angle 2ϕ with $\sin\phi = \sqrt{K/N}$ – cf. Problem 29 – in this two-dimensional space.)

3. Now assume we are given one of the corresponding eigenvectors $|\theta_k\rangle$. We can now use the phase estimation algorithm to determine the phase $\theta_k/2\pi$ corresponding this eigenvector up to some number d of digits. What is the number of queries to O_f required for that? What is the resulting accuracy of θ_k? (You can assume that the phase estimation is exact, i.e. neglect the additional error arising from the fact that $\theta_k/2\pi$ does not stop after d digits.)

4. From θ_k, we can estimate K. What is the error δK as a function of Q (and K, N)?

5. Show that this algorithm can be adapted to work also if we cannot prepare the state $|\theta_k\rangle$, but rather start in some other easy-to-prepare state in the subspace $\mathcal{S}$.