# Quantum Computing (Fall 2013) Instructor: Shengyu Zhang.

# Lecture 5 Quantum searching algorithms and lower bounds

## Grover’s search

After learning quantum algorithms for algebraic problems, we have a feeling that quantum speedup (over classical algorithms) needs a strong algebraic structure in the problem. Thus it was a surprise when Grover discovered that searching, one of the most fundamental primitives, in a totally unstructured set, provides quantum speedup.

Consider a set of elements, say {22,8,3,12,…}. We want to see whether there is a target element, say 11. Suppose that once given any particular element, it is easy for us to check whether it’s our target. Classically, to solve the problem, we clearly need to scan the whole list, yielding a complexity of . Using Grover’s search, we can finish the job by looking at the list only times.

To make it easier to state, suppose that we are given a string and we want to decide whether . We have an oracle to tell us whether a particular position is 1 or 0. That is, if we make a query “”, then the oracle gives the answer. Such algorithms are called query algorithms. You must have seen these when learning sorting, where the oracle answers the question “” and it is well-known that the optimal sorting algorithm needs queries. In a quantum algorithm, we can make queries in superposition, and get corresponding answers in superposition as well. For example, if we make a query by inputting to the oracle, then we’ll get in the answer. We can also assume that the oracle takes a query and answers . Can you see why?

Assuming the latter oracle format, we now give the Grover’s algorithm.

…

Here is the reflection about the state , namely, and for all . The oracle can also be viewed as a reflection about the state , where . These two reflections combined have the effect of rotating the current state towards a target one by where . So it takes iterations. Since in each iteration the algorithm only takes 1 query, the algorithm needs queries.

(In class, I’ll show a geometrical interpretation of the algorithm, and things will be very clear.)

After the rotations, measure in the computational basis and then verify it by one more query. In this way we’ll find an with .

Question: What if we don’t know in advance? We can try until we succeed.

## Quantum query algorithms

Grover’s searching algorithm belongs to the general class of quantum query algorithms. Suppose and an oracle O has the following effect:

where the addition is. That is, the oracle tells the value of input variables in superposition. Note that it doesn’t let us to get all information of input variables in one query, since the answers are in a superposition and we cannot extract the bits from it.

Recall that even the last algorithm we talked about in last lecture (for HSP for non-Abelian group) falls in the framework of query algorithm.

A general question is how to design query efficient algorithms and how to prove lower bounds for quantum query complexity. Namely, we want to pin down the quantum query complexity , the smallest number of queries needed to compute with error probability at most on any input. Next we introduce one powerful method for lower bound proving.

## Quantum adversary method

To use the quantum adversary method, we need to first find a matrix where . Index the rows by and columns by . The matrix needs to satisfy as long as . Define by if and otherwise. Suppose is nonzero, and consider the quantity . This is a lower bound of the quantum query complexity! Taking the best and denote . The lower bound goes like the following.

The proof is in paper [HLS07], which is well written and self-contained enough to be easily read. Thus there is no point of copying the proof here. I’ll show the whole proof in class.

## Note

Grover’s search first appeared in [Gro96]. The quantum adversary method was originally discovered in a weaker form by Ambainis [Amb02], and various equivalent forms were founded; see [SS06]. The final form as we presented was proposed in [HLS07], which is stronger than previous ones.

## Reference

[Amb02] Andris Ambainis, **Quantum Lower Bounds by Quantum Arguments**, *Journal of Computer and System Sciences*, Volume 64, Issue 4, pages 750–767, 2002. (Earlier at STOC’00.)

[Gro96] Lov Grover, **A fast quantum mechanical algorithm for database search**, in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212-219, 1996.

[HLS07] Peter Hoyer, Troy Lee, Robert Spalek, **Negative weights make adversaries stronger**, in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 526 – 535, 2007.

[SS06] Robert Spalek, Mario Szegedy, **All Quantum Adversary Methods are Equivalent**, *Theory of Computing* 2(1): 1-18, 2006.