AQAlg: Advanced Quantum Algorithms **2025 - 2026**

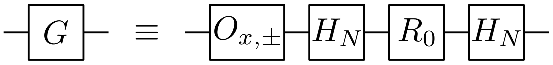
Lecture 2: Grover's algorithm and lower bounds

Lecturer: Simon Apers (apers@irif.fr)

1 Grover's algorithm

The "unstructured search" problem is defined as follows: given query access to a boolean input string $\{x_0, \dots, x_{N-1}\} \in \{0,1\}^N$, return a "marked" element (i.e., an index i such that $x_i = 1$), or decide that no marked element exists. How many queries does this take? Any classical algorithm trivially requires $\Omega(N)$ queries. On the other hand, Grover's quantum search algorithm solves this problem with $O(\sqrt{N})$ queries. In contrast to the exponential speedup in quantum period finding and Shor's algorithm, this "only" gives a quadratic speedup, but it has much wider applicability.

Assume $N = 2^n$. The following n-qubit circuit describes a single iteration G of Grover's algorithm:

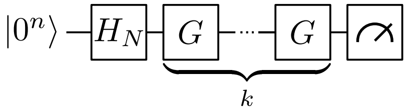


Here $O_{x,\pm}$ is the "phase oracle" defined by

$$O_{x,\pm} |i\rangle = (-1)^{x_i} |i\rangle ,$$

and R_0 is the reflection around $|0^n\rangle$ (i.e., $R_0 |0^n\rangle = |0^n\rangle$ and $R_0 |i\rangle = -|i\rangle$ if $i \neq 0$). We will prove the following proposition.

Proposition 1. *Consider input $\{x_0, \dots, x_{N-1}\} \in \{0,1\}^N$ and let $t = |x|$ denote the number of nonzero entries. There exists $k \in O(\sqrt{N/t})$ so that applying k iterations of the Grover operator to the initial state $\frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle$, and measuring the state (see circuit below), returns a marked element with constant probability.*



There is a nice geometric picture that proves this proposition. For this, we reinterpret the full Grover iteration as a product of two reflections. First, we can think about $H_N R_0 H_N$ as a reflection around the uniform superposition

$$|u\rangle = H_N |0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle .$$

Indeed, verify that $H_N R_0 H_N |u\rangle = |u\rangle$ while $H_N R_0 H_N |v\rangle = -|v\rangle$ for any $|v\rangle$ orthogonal to $|u\rangle$. Second, we interpret $O_{x,\pm}$ as a reflection around the "unmarked" superposition

$$|u_0\rangle = \frac{1}{\sqrt{N-t}} \sum_{i:x_i=0} |i\rangle .$$

If we also use the notation $|u_1\rangle = \frac{1}{\sqrt{t}} \sum_{i:x_i=1} |i\rangle$ for the "marked" superposition, then we can rewrite the initial state as

$$|u\rangle = \sin(\theta) |u_1\rangle + \cos(\theta) |u_0\rangle ,$$

with $\sin(\theta) = \sqrt{t/N}$. This corresponds to the left picture in Fig. 1. A Grover iteration first applies $O_{x,\pm}$, i.e., a reflection around the unmarked state $|u_0\rangle$. This leads to the middle picture. Then it applies $H_N R_0 H_N$, which is a reflection around the initial state $|u\rangle$. This leads to the right picture, which depicts the state after a single Grover iteration:

$$G|u\rangle = G(\sin(\theta) |u_1\rangle + \cos(\theta) |u_0\rangle) = \sin(3\theta) |u_1\rangle + \cos(3\theta) |u_0\rangle .$$

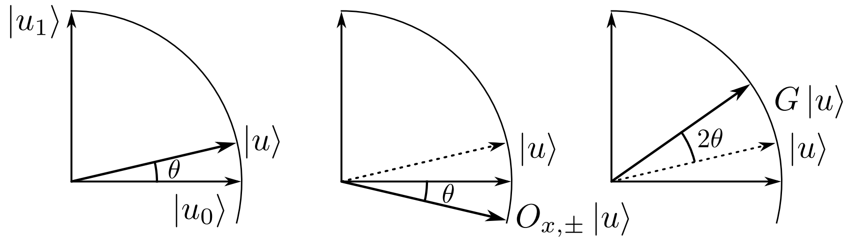


Figure 1: Depiction of one Grover iteration.

After k Grover iterations, we get a state

$$G^k |u\rangle = \sin((1+2k)\theta) |u_1\rangle + \cos((1+2k)\theta) |u_0\rangle .$$

Ideally, setting k to be $k^* = \pi/(4\theta) - 1/2$ would yield $G^k |u\rangle = |u_1\rangle$. Measuring this state returns a (uniformly random) marked element with certainty. However, k has to be an integer and so we set it to be the nearest integer to k^*. Assuming $\theta \leq 1/2$, we can bound the success probability by

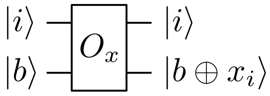
$$\sin^2((2k+1)\theta) = \sin^2(\pi/2 + 2(k-k^*)\theta) = \cos^2(2(k-k^*)\theta) \geq \cos^2(\theta) \geq 1 - \theta^2 \geq 3/4,$$

where we used that $|k - k^*| \leq 1/2$. The total complexity of the resulting algorithm is $O(k)$, which is $O(1/\theta) = O(\sqrt{N/t})$.

2 Quantum query complexity

We can think about Grover's algorithm as computing the OR-function on the N-bit input string x. The algorithm makes $O(\sqrt{N})$ quantum queries to the input, while any classical algorithm must make $\Omega(N)$ queries. Can we further improve on Grover's algorithm? Can we compute the OR-function with a single quantum query, as in the Deutsch-Jozsa algorithm? Quantum query complexity is the study of precisely these questions. For a general input $x = x_0 \ldots x_{N-1} \in \{0,1\}^N$ and a boolean function $f : \{0,1\}^N \to \{0,1\}$, we ask how many quantum queries an algorithm has to make to compute f. While quantum algorithms give upper bounds on the quantum query complexity, in this section we discuss *lower bounds*.

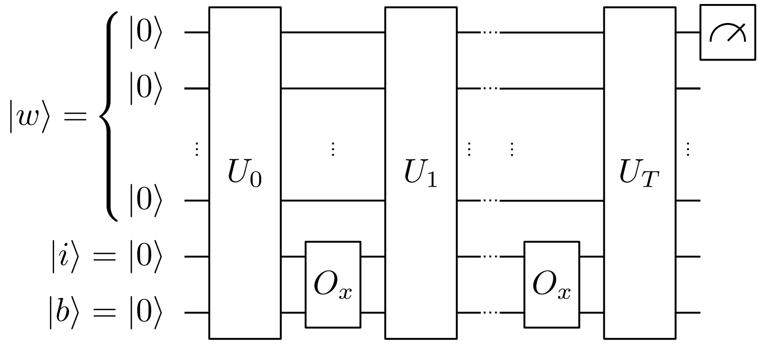
We consider quantum query access to the input through the unitary operation[1]



[1] This corresponds to the bit oracle. We could equivalently consider the phase oracle (see previous exercise session).

$$|i,b\rangle \mapsto O_x\,|i,b\rangle = |i, b\oplus x_i\rangle\,,$$

for $i \in \{0,1,\dots,N-1\}$, $b \in \{0,1\}$ and $\oplus$ addition mod 2. Now consider a general quantum algorithm for computing $f(x)$ of the input x. Apart from the $|i,b\rangle$ query-answer registers, the algorithm also has some workspace register $|w\rangle$ (to do calculations etc). If the algorithm makes T queries to the input, then we can describe it by a circuit of the following form:



We assume that the output of the algorithm corresponds to a measurement of the first qubit. Let p denote the probability of returning "1". For a fixed choice of unitaries $U_0,\dots,U_T$, we can interpret $p = p(x)$ as a function of (only) x. The algorithm correctly computes f if $p(x) = f(x)$. The algorithm computes f with probability at least 2/3 if $|p(x) - f(x)| \leq 1/3$ (and so $p(x) \geq 2/3$ if $f(x) = 1$ and $p(x) \leq 1/3$ if $f(x) = 0$).

As it turns out, the number of queries T puts strong constraints on the polynomial p. First of all, recall the notion of a multilinear polynomial $q : \{0,1\}^N \to \mathbb{C}$, which is a function of the form

$$q(x) = \sum_{S\subseteq\{0,\dots,N-1\}} c_S \prod_{i\in S} x_i, \qquad c_S \in \mathbb{C}.$$

The degree of q is $\deg(q) = \max\{|S| \mid c_S \neq 0\}$. In the exercises we will prove that any function $f : \{0,1\}^N \to \mathbb{C}$ has a unique representation as such a multilinear polynomial (of degree at most N). The following claim, which we prove later, shows that the polynomial p is even further constrained.

Claim 1. *The output probability $p : \{0,1\}^N \to [0,1]$ of a quantum circuit making T queries is a multilinear polynomial of degree at most $2T$.*

As a consequence, if f is a polynomial of degree d, then any quantum query algorithm for which $p(x) = f(x)$ must make $T \geq d/2$ queries.

Exercise 1. *Write the OR-function for $N = 3$ bits as a multilinear polynomial. Conclude that there is no 1-query quantum algorithm to compute the OR-function on 3 bits.*

More generally, it can be shown that the OR-function on N bits has degree N, and so any quantum algorithm that computes OR with success probability 1 must make at least $N/2$ queries.

If we only need to be correct with probability at least 2/3, then it suffices that $|p(x)-f(x)| \leq 1/3$. The *approximate degree* $\widetilde{\deg}(f)$ of f is the lowest degree of a polynomial that approximates f in such a way. It follows that any quantum algorithm that computes f with probability at least 2/3 must make $T \geq \widetilde{\deg}(f)/2$ quantum queries. Turning to the OR function, it is nontrivial but elementary to show that $\widetilde{\deg}(\mathrm{OR}) \in \Omega(\sqrt{N})$ (see e.g. lecture notes Childs). This implies an $\Omega(\sqrt{N})$ lower bound on the bounded error quantum query complexity of the OR function, and this proves that Grover's algorithm is optimal.

2.1 Proof of Claim 1

Let $|\psi_T\rangle$ denote the output state of the T-query algorithm (before measurement). We expand it as

$$|\psi_t\rangle = \sum_{z=(w,i,b)} \alpha_z |z\rangle = \sum_{z=(w,i,b)} \alpha_z(x) |z\rangle ,$$

where we observed that the amplitudes $\alpha_z(x) \in \mathbb{C}$ are functions of the input x. The output is obtained from measuring the first qubit of the final state $|\psi_T\rangle$. If z_1 denotes the first bit of z, then the probability of outputting "1" is

$$\sum_{z:z_1=1} |\alpha_z(x)|^2 = p(x).$$

We will prove that the functions $\alpha_z(x)$ are multilinear polynomials of degree at most T. Claim 1 directly follows from that.

The proof is by induction. If $T = 0$ then the claim is trivially satisfied, as the state $|\psi_0\rangle$ does not depend on x. Now, assuming that the claim holds for $|\psi_T\rangle$, let us prove it for $|\psi_{T+1}\rangle = U_{T+1}O_x |\psi_T\rangle$. For a basis state $|z\rangle = |w, i, b\rangle$, we rewrite the oracle action

$$O_x |w, i, b\rangle = |w, i, b \oplus x_i\rangle = x_i |w, i, b \oplus 1\rangle + (1 - x_i) |w, i, b\rangle .$$

Applying this to $|\psi_T\rangle$ yields

$$O_x |\psi_T\rangle = \sum_{z=(w,i,b)} \alpha_z(x) O_x |w, i, b\rangle = \sum_{z=(w,i,b)} \alpha_z(x)\big(x_i |w, i, b \oplus 1\rangle + (1 - x_i) |w, i, b\rangle\big).$$

This shows that the new amplitudes are linear combinations of terms of the form $\alpha_z(x)x_i$ or $\alpha_z(x)(1-x_i)$. Hence, the gate O_x only increase the degree of the α_z's by 1.

Then, note that applying the unitary U_{T+1} to $O_x |\psi_T\rangle$ only forms linear combinations of the amplitudes. This cannot further increase the degree of the coefficients, and hence we proved that the amplitudes $\alpha_z(x)$ of $|\psi_{T+1}\rangle$ have degree at most $T + 1$.