

# Quantum Computing

Instructor: Alexander Razborov, University of Chicago.

razborov@cs.uchicago.edu

Course Homepage:

<http://people.cs.uchicago.edu/~razborov/teaching/winter11.html>

Winter Quarter, 2011

## Contents

<b>1</b>	<b>Classical and Quantum computation: circuit model</b>	<b>3</b>
1.1	Reversible Computation . . . . .	3
1.2	Probabilistic Computation . . . . .	4
1.3	Quantum Computation . . . . .	4
<b>2</b>	<b>Early Quantum Algorithms</b>	<b>5</b>
2.1	Deutsch algorithm (1985) . . . . .	5
2.2	Deutsch-Josza algorithm (1992) . . . . .	6
2.3	Simon's algorithm (1994) . . . . .	7
<b>3</b>	<b>BQP <math>\subseteq</math> PP</b>	<b>9</b>
<b>4</b>	<b>Famous Quantum Algorithms</b>	<b>11</b>
4.1	Grover's search algorithm (1996) . . . . .	11
4.1.1	A Geometrical Interpretation . . . . .	11
4.1.2	Some Details . . . . .	12
4.2	Factoring: Shor's Algorithm . . . . .	14
4.2.1	Reductions . . . . .	14
4.2.2	Linear Algebra . . . . .	16
4.2.3	Part 1: Phase Estimation Algorithm . . . . .	17
4.2.4	Part 2: How to Construct $ u_k\rangle$ ? . . . . .	19
4.3	Discrete Logarithm . . . . .	20
4.4	Hidden Subgroup Problem . . . . .	21
4.4.1	First Application - Symmetric Group . . . . .	21
4.4.2	Second Application - Dihedral Group . . . . .	22

<b>5</b>	<b>Quantum Probability</b>	<b>22</b>
5.1	“Tracing out” or “partial measurement” . . . . .	24
5.2	Superoperators . . . . .	25
<b>6</b>	<b>Quantum Complexity Theory: black-box model</b>	<b>27</b>
6.1	Hybrid method: optimality of Grover’s search . . . . .	27
6.2	Quantum Query Complexity vs. Other Complexity Measures	30
6.3	Ambainis’s Adversary Method . . . . .	35
6.4	Quantum Query Complexity and Formula Size . . . . .	37
<b>7</b>	<b>Quantum Communication Complexity</b>	<b>38</b>
7.1	Probabilistic Communication Complexity . . . . .	39
7.2	Quantum Communication Complexity . . . . .	39
7.3	Decomposition of quantum protocols . . . . .	42
7.4	Lower bound for $QC_2(IP_2)$ . . . . .	44
7.5	Lower bound for $QC_2(DISJ)$ . . . . .	45
7.6	Generalizations of the discrepancy method . . . . .	47
7.7	Direct products . . . . .	48
<b>8</b>	<b>Quantum Error-Correcting Codes</b>	<b>48</b>
8.1	Operator-sum representation of superoperators . . . . .	49
8.2	Projective Measurements . . . . .	49
8.3	Quantum Information Theory . . . . .	50
8.3.1	Error Correcting Codes in Classical Information Theory	50
8.3.2	Correcting Against Quantum Bit Flip . . . . .	50
8.3.3	Correcting Against Quantum Phase Flip . . . . .	51
8.3.4	Correcting Against Simultaneous Bit and Phase Flip .	51
8.4	Properties of the Recovery Operator . . . . .	51

## Lectures 1-3

Date: January 7, 11 and 13, 2011

*These lectures were not scribed; the brief digest below was kindly compiled a posteriori by Pratik Worah.*

The lecture topics in the course can be divided into three major parts:

1. Quantum Computations i.e. circuits and algorithms.
2. Quantum Complexity, in particular black-box models.
3. Quantum Communication Complexity.

## 1 Classical and Quantum computation: circuit model

### 1.1 Reversible Computation

Turing machines (TM) and circuits lie at the heart of classical computation. Similarly Quantum TMs and Quantum circuits lie at the heart of quantum computation. Although QTMs were defined by Bernstein and Vazirani (1997) the more popular (and equivalent) notion is to use quantum circuits to describe quantum computation. The complexity class QBP (uniform polysize quantum circuits) is the quantum analog of P (uniform polysize circuits) in this regard.

A classical circuit can compute any function  $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$  using a universal basis (say {NOT, AND} gates). However, such a computation is not always reversible since the basic logic gates are not reversible (i.e. a well defined inverse does not always exist). A *reversible circuit* can therefore be defined as one where the constituent gates are permutations (i.e. invertible). A *reversible computation* is therefore any computation that can be carried out by a reversible circuit. See the material on *Landauer's principle* in Wikipedia or [1] for the physics behind reversibility and reversible computation. Examples of reversible gates include the CNOT, Toffoli and Fredkin gates (the last gate is universal). By adding some *ancilla* bits (i.e. extra bits which remain unchanged at the end of the computation), one can effectively simulate classical computation using reversible circuits as the following theorem shows.

**Theorem 1.** *If  $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$  can be computed by any classical circuit with  $L$  gates then there exists a reversible circuit with  $O(L + m + n)$  gates that computes  $(\bar{x}, 0^{L+2m}) \rightarrow (\bar{x}, \overline{f(x)}, 0^{L+m})$ .*

*Proof.* The proof is given in [2]. The main complication is in the need to remove garbage after the computation.  $\square$

## 1.2 Probabilistic Computation

A *probabilistic circuit*  $C$  for computing  $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$  is a classical circuit with extra inputs  $\bar{y}$  chosen from the uniform distribution, the output of  $C$  is a random variable and  $\Pr(C(\bar{x}, \bar{y}) = f(\bar{x})) \geq \frac{2}{3}$ . One can now define *reversible probabilistic circuits* based on the above definitions.

## 1.3 Quantum Computation

Some necessary basic definitions from linear algebra (like *Hilbert spaces*, *tensor products*, *permutation matrices*, *unitary operators* etc) can be looked up from Wikipedia/standard texts (like [3]).

A *quantum circuit* computing  $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$  is just like a reversible circuit with quantum gates (main eg. the Hadamard gate). It is essentially a unitary operator in  $\mathbb{C}^{2^n \times 2^n}$ . Therefore instead of a bit we have qubits, where a qubit is just a unit vector in  $\mathbb{C}^2$ . Throughout the course we use *Dirac notation* to describe quantum states (defined below).

**Observation 2.**  $U$  unitary if and only if  $U$  preserves unit vectors.

**Observation 3.** The tensor product of unitary operators is unitary.

The above observations allow us to use bounded fan-in gates to describe the circuit and formally write the computation as a sequence of matrix multiplications.

**Theorem 4.** Any unitary operator on  $\mathbb{C}^{2^n}$  can be realized by a quantum circuit in which all gates act on at most 2 qubits.

A universal basis for quantum circuits would be any finite set of gates such that for any unitary operator  $U$  we can represent  $U$  with any given accuracy. The accuracy being measured by the spectral norm of the matrix. Example of a universal basis (standard basis):  $\{H_2, K, K^{-1}, T\}$  where  $H$  is the Hadamard matrix,  $T$  stands for the Toffoli gate and

$$K = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}.$$

In quantum circuits a *state* denotes a unit vector, for example:  $|0\rangle$ . An *entangled state* is a state that can not be represented in the form  $|\phi\rangle|\psi\rangle$ , for example  $\frac{|00\rangle+|11\rangle}{\sqrt{2}}$ .

**Theorem 5.** Any basis consisting of all 1 qubit gates and a non-entangled 2 qubit gate is universal.

**Theorem 6.** Every quantum circuit of size  $L$  that uses gates from a fixed finite basis can be simulated with precision  $\delta$  by an  $O(L \log(\frac{L}{\delta}))$  size circuit over the standard basis using ancilla bits.

If one measures the vector  $y = \sum_i \alpha_i y_i$  at the end of the computation, then the vector is destroyed and one gets  $y_i$  with probability  $\alpha_i \alpha_i^*$  (cf. reversible probabilistic circuits). We will discuss later what will happen if we want to measure something *in the middle* of the computation.

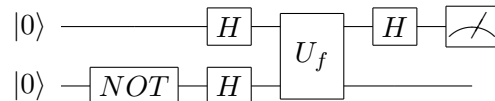
**Theorem 7.**  $BPP \subseteq BQP \subseteq PP$ .

## 2 Early Quantum Algorithms

### 2.1 Deutsch algorithm (1985)

This is our first quantum algorithm. Given a quantum circuit  $U_f : |xy\rangle \mapsto |x, y \oplus f(x)\rangle$  (as a blackbox) implementing  $f : \{0, 1\} \rightarrow \{0, 1\}$ , the task is to determine whether or not  $f$  is a constant function. The following is a brief description.

Start with input  $|00\rangle$  and apply the Hadamard gate to the first qubit. Next apply  $U_f$  and finally apply back the Hadamard gate to the first qubit.



Check that expression for the state of the first qubit is

$$\frac{(1 + (-1)^{f(0) \oplus f(1)})|0\rangle + (1 - (-1)^{f(0) \oplus f(1)})|1\rangle}{2}.$$

Note  $f$  is constant if and only if  $f(0) \oplus f(1) = 0$  i.e if and only if the qubit above is measured to be 0.

## Lectures 4-5

Scribe: Tatiana Orlova, University of Chicago.

Date: January 18 and 20, 2011

## 2.2 Deutsch-Josza algorithm (1992)

**Deutsch-Josza algorithm** is a generalization of Deutsch algorithm we studied in the previous lecture.

Suppose we are given a Boolean function in  $2^n$  inputs

$$f: \{0, 1\}^n \rightarrow \{0, 1\}.$$

We consider two types of Boolean functions. First, a constant function, as in the previous lecture. Second, a *balanced* function, which is a function that outputs as many zeros as ones ( $2^{n-1}$  zeros and  $2^{n-1}$  ones) over its input set. For example, in the simplest case  $n = 2$ , all Boolean functions are well defined by these two types, and, moreover, we are in the situation of Section 2.1.

Given  $f(x)$ , we want to determine whether it is constant or balanced. We can try to solve this problem deterministically. Clearly, if we make  $2^{n-1}$  queries we might get unlucky and get all 0s or all 1s. Thus, we need to know at least  $2^{n-1} + 1$  values of  $f(x)$  to decide whether it is constant or balanced, which simply means too many queries! We would like to see if using quantum computation can help us significantly reduce the number of queries.

Similar to the Deutsch algorithm we will first prepare the state  $|\phi\rangle$

$$|0\rangle \xrightarrow{\text{NOT}} |1\rangle \xrightarrow{H} \frac{|0\rangle - |1\rangle}{\sqrt{2}} \stackrel{df}{=} |\phi\rangle$$

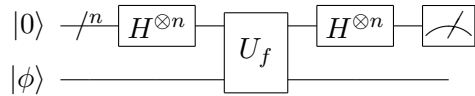
Recall, that the  $U_f$ -operator is defined by

$$U_f: |x, y\rangle \mapsto |x, y \oplus f(x)\rangle.$$

The initial state of the circuit is

$$|0^n\rangle |\phi\rangle.$$

We then perform the following transformations ( $/^n$  stands for duplicating a state  $n$  times)



$$\begin{aligned} |0^n\rangle |\phi\rangle &\xrightarrow{H^{\otimes n}} \frac{1}{2^{n/2}} \sum_x |x\rangle |\phi\rangle \xrightarrow{U_f} \frac{1}{2^{n/2}} \sum_x |x\rangle \frac{|f(x)\rangle - |f(x) \oplus 1\rangle}{\sqrt{2}} \\ &= \frac{1}{2^{n/2}} \sum_x (-1)^{f(x)} |x\rangle |\phi\rangle, \end{aligned}$$

where the sum is over all all binary strings  $x \in \{0, 1\}^n$ . Note, that

$$(-1)^{f(x)} \frac{|f(x)\rangle - |f(x) \oplus 1\rangle}{\sqrt{2}} \equiv |\phi\rangle$$

regardless of  $f(x)$ .

The transformation

$$|x\rangle \mapsto (-1)^{f(x)} |x\rangle$$

changes the phase of any vector according to  $f(x)$  and does not use ancilla bits. So, we simply moved from one representation to another, and, in fact, in future we will often be using this alternate form of feeding  $f$  to our algorithms.

### Discrete Fourier Transform

$$H^{\otimes n} |x\rangle = \frac{1}{2^{n/2}} \sum_y (-1)^{\langle x, y \rangle} |y\rangle,$$

where  $x, y$  are strings of length  $n$ , and  $\langle x, y \rangle$  is ordinary inner product (over  $\mathbb{Z}$  or  $\mathbb{F}_2$ ).

$$H^{\otimes n} \frac{1}{2^{n/2}} \sum_x (-1)^{f(x)} |x\rangle = \frac{1}{2^n} \sum_{x, y} (-1)^{f(x) + \langle x, y \rangle} |y\rangle.$$

We measure first  $n$  qubits and look at the coefficient  $\alpha_0$  corresponding to  $|0^n\rangle$ . The value  $\alpha_0 \alpha_0^*$  can be interpreted as the probability of getting  $0^n$  as the result of this measurement.

$$\alpha_0 = \frac{1}{2^n} \sum_x (-1)^{f(x)} = \begin{cases} \pm 1, & \text{if } f(x) \text{ is constant;} \\ 0, & \text{if } f(x) \text{ is balanced.} \end{cases}$$

Thus, the quantum Deutsch-Josza algorithm requires only a **single** query versus an **exponential** number of queries in classical deterministic case. It does not have this much advantage over the classical probabilistic computation, that provides a correct answer with probability  $1 - \frac{1}{2^n}$  by using only  $O(n)$  queries. This fact makes the result of this section a little bit less exciting :) In the next section we will discuss the first example of an exponential gap between classical and quantum computation.

### 2.3 Simon's algorithm (1994)

The problem can be formulated as follows.

**Input:** A function

$$f: \{0, 1\}^n \rightarrow \{0, 1\}^m,$$

where  $m$  can possibly be larger than  $n$ .

**Promise:** There exists  $s \in \{0, 1\}^n$  such that  $f(x) = f(y)$  if and only if  $x = y$  or  $x = y + s$ . (see Fig. 1 for an intuitive interpretation of  $s$ )

**Problem:** We want to determine this value of  $s$ .

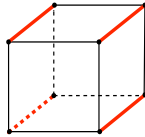


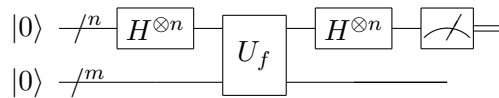
Figure 1: Function  $f(x)$  has the same values on the edges that share direction.

The Simon's problem is a part of the class of problems known as the *hidden subgroup problems*. In order to demonstrate the significance of the problems in this class we will give a few examples. Solving this problem for  $\mathbb{Z}$ , which is an infinite abelian group, will imply solving integer factorization problem. For *Dihedral* group, the smallest non-abelian group, the result is unknown. For *symmetric* group the problem implies an efficient algorithm for the graph isomorphism problem. We will discuss all this in more details in Section 4.4 below.

The solution scheme will be extremely similar to what we already did. Now the initial state will be

$$|0^n, 0^m\rangle.$$

We then perform the following transformations



$$\begin{aligned}
|0^n, 0^m\rangle &\xrightarrow{H^{\otimes n}} \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x\rangle |0\rangle \xrightarrow{U_f} \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x, f(x)\rangle \xrightarrow{H^{\otimes n}} \\
&\xrightarrow{H^{\otimes n}} \frac{1}{2^n} \sum_{x,y \in \{0,1\}^n} (-1)^{\langle x,y \rangle} |y, f(x)\rangle \\
&= \frac{1}{2^n} \sum_{y \in \{0,1\}^n} \left( \sum_{x \in \{0,1\}^n} (-1)^{\langle x,y \rangle} |y, f(x)\rangle \right).
\end{aligned}$$

Note, that since  $f(x)$  is not injective, terms  $|y, f(x)\rangle$  in the final sum come in pairs and those with the opposite sign will cancel each other out. To show this, we consider coefficients of  $|y, z\rangle$  for all  $z \in \{0,1\}^m$ , such that  $f(x) = z$ , and  $f(x+s) = z$ . We have

$$(-1)^{\langle x,y \rangle} + (-1)^{\langle x+s,y \rangle} = (-1)^{\langle x,y \rangle} (1 + (-1)^{\langle s,y \rangle})$$

If  $\langle s, y \rangle = 1$ , i.e.  $s$  and  $y$  are not orthogonal

$$(-1)^{\langle x,y \rangle} (1 + (-1)^{\langle s,y \rangle}) = 0$$

At the end we have a uniform superposition  $|y, z\rangle$ , such that  $y \perp s$ . We then repeat the experiment from scratch multiple times to recover  $s$ .

The Simon's algorithm is a good example of "quantum magic" - *interference*, or cancelations. When we send our queries to  $f(x)$ , values that would appear with high probability might cancel out because of interference.

### 3 BQP $\subseteq$ PP

We want to show that

$$BQP \subseteq PP.$$

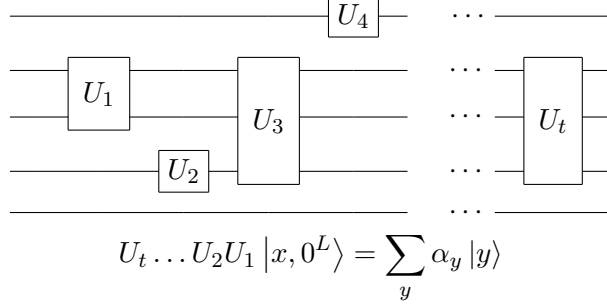
One characterization of languages in  $PP$  is this:  $L \in PP$  if and only if for all  $x \in L$

$$\#\{y: A(x, y)\} - \#\{y: R(x, y)\} \neq 0,$$

where  $x, y \in \{0, 1\}^n$  - strings of polynomial range,  $\{y: A(x, y)\}$  and  $\{y: R(x, y)\}$  are accepting and rejecting configurations respectively.

Consider a quantum circuit  $C$  with gates of the following types:  $\{H, K, K^{-1}, CNOT, T\}$ . At every particular time it executes one quantum operator. Thus, the whole computation is the product of the quantum

operators



We would like to compute coefficients  $\alpha_y$ . To do this we sum up the corresponding positions in quantum operators

$$\alpha_y = \sum_{|z_1\rangle, \dots, |z_{t-1}\rangle} \langle y|U_t|z_{t-1}\rangle \langle z_{t-1}|U_{t-1}|z_{t-2}\rangle \dots \langle z_1|U_1|x_1 0^L\rangle.$$

Note that if  $y$  and  $z_t$  are basis vectors

$$\langle y|U_t|z_t\rangle = U_t[y, z_t].$$

The matrix entries of our operators, except for Hadamard gate, are from the set

$$\{0, \pm 1, \pm i\}.$$

So that the resulting values remain in this set, since it is closed under multiplication. Only Hadamard gate creates nontrivial coefficients

$$\frac{1}{\sqrt{2}}\{\pm 1\}.$$

But the number of Hadamard gates in  $C$  is known in advance and does not depend on the input.

Denote this number by  $h$ . We have

$$\alpha_y = \sum_{|z_1\rangle, \dots, |z_{t-1}\rangle} \frac{1}{2^{h/2}} f(z_1, \dots, z_{t-1}, x, y),$$

where  $f \in \{0, \pm 1, \pm i\}$  is efficiently computed. The amplitude is

$$\alpha_y \alpha_y^* = \frac{1}{2^h} \sum_{\substack{|z_1\rangle, \dots, |z_{t-1}\rangle \\ |z'_1\rangle, \dots, |z'_{t-1}\rangle}} f(z_1, \dots, z_{t-1}, x, y) f^*(z'_1, \dots, z'_{t-1}, x, y).$$

We have

$$\alpha_y = \frac{1}{2^h} \left( \# \left\{ \vec{z}, \vec{z}' : f(z, x, y) f^*(z', x, y) = 1 \right\} - \# \left\{ \vec{z}, \vec{z}' : f(z, x, y) f^*(z', x, y) = -1 \right\} \right).$$

## 4 Famous Quantum Algorithms

### 4.1 Grover's search algorithm (1996)

We begin with a simple case (when the solution is known to be unique) that, however, already contains all essential ideas. The general case is sketched below in Section 4.1.2.

**Input:**  $f: [N] \rightarrow \{0, 1\}$ , where  $[N]$  stands for domain of size  $N$ , which in particular could be binary strings of length  $\log_2 N$ .

**Promise:** There exists a unique  $w$  such that  $f(w) = 1$ .

**Problem:** We want to find this  $w$ .

**Theorem 8.** *There exists a quantum algorithm that performs search in time  $O(\sqrt{N})$ .*

We will see in Section 6.1 that this bound is tight.

#### 4.1.1 A Geometrical Interpretation

Consider a standard superposition

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle.$$

Let  $|w\rangle$  be the unknown unit vector we want to find. Define

$$|\psi_{\text{Bad}}\rangle \stackrel{df}{=} \frac{1}{\sqrt{N-1}} \sum_{x \neq w} |x\rangle.$$

Vectors  $|\psi\rangle$  and  $|\psi_{\text{Bad}}\rangle$  generate a plane (2-dimensional subspace). Clearly,

$$|\psi\rangle = \sqrt{\frac{N-1}{N}} |\psi_{\text{Bad}}\rangle + \frac{1}{\sqrt{N}} |w\rangle,$$

and the vector

$$|\bar{\psi}\rangle = \sqrt{\frac{N-1}{N}} |w\rangle + \frac{1}{\sqrt{N}} |\psi_{\text{Bad}}\rangle.$$

is orthogonal to  $|\psi\rangle$ . In other words,  $\{|\psi\rangle, |\bar{\psi}\rangle\}$  is an orthogonal basis in the subspace generated by the vectors  $|\psi\rangle$  and  $|\psi_{\text{Bad}}\rangle$ . Denote the angle between  $|\psi\rangle$  and  $|\psi_{\text{Bad}}\rangle$  by  $\theta$ , then

$$|\psi_{\text{Bad}}\rangle = \cos \theta |\psi\rangle - \sin \theta |\bar{\psi}\rangle.$$

From the inner product  $\langle \psi_{\text{Bad}} | \psi \rangle$ , we find

$$\sin \theta = \frac{1}{\sqrt{N}},$$

and thus

$$\theta \approx \frac{1}{\sqrt{N}} + O\left(\frac{1}{N}\right).$$

Consider two geometrical transformations:

1. Reflection around the line defined by  $|\psi_{\text{Bad}}\rangle$  - this transformation corresponds to  $U_f$ ;
2. Reflection around the line defined by  $|\psi\rangle$  - we denote this transformation by  $V$ .

**Key Idea:** Define a new transformation, **Grover Iterate**,  $G \stackrel{\text{df}}{=} VU_f$ . It is a composition of reflections  $U_f$  and  $V$ , and results in a rotation by  $2\theta$ . If we apply this operator to a unit vector precisely  $\lfloor \sqrt{N} \frac{\pi}{4} \rfloor$  times we will rotate this vector by  $\approx \frac{\pi}{2}$ . Thus, if we apply Grover Iterate  $\lfloor \sqrt{N} \frac{\pi}{4} \rfloor$  times to  $|\psi\rangle$  it will become “almost”  $|w\rangle$ .

#### 4.1.2 Some Details

We now fill in the details. Here we assume that all objects from  $[N] = \{1, 2, \dots, N\}$  are binary strings. We would like to construct a unitary operator  $V$  with the following properties

$$V |\psi\rangle = |\psi\rangle$$

and

$$V(|x\rangle - |y\rangle) = |y\rangle - |x\rangle.$$

We first apply DFT (or Hadamard matrices)

$$H^{\otimes n} |x\rangle = \frac{1}{2^{n/2}} \sum_y (-1)^{\langle x,y \rangle} |y\rangle$$

Let  $U_0$  be a unitary operator such that

$$U_0 |0\rangle = |0\rangle$$

and

$$U_0 |x\rangle = -|x\rangle.$$

Thus,  $U_0$  flips the phase of all non-zero vectors. We leave the explicit construction of  $U_0$  as an exercise.

We now apply  $U_0$  to the  $H^{\otimes n} |x\rangle$  and obtain

$$U_0 H^{\otimes n} |x\rangle = \frac{1}{2^{n/2}} (2|0\rangle - \sum_y (-1)^{\langle y,x \rangle} |y\rangle)$$

Finally, we apply  $n$  Hadamard gates again

$$H^{\otimes n} U_0 H^{\otimes n} |x\rangle = \frac{1}{2^n} \left( 2|\psi\rangle - \sum_{y,z} (-1)^{\langle y,x \rangle} (-1)^{\langle y,z \rangle} |z\rangle \right)$$

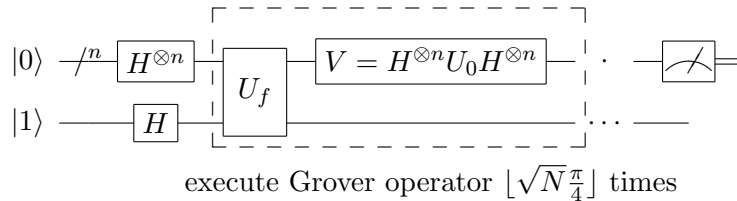
Terms where  $|z\rangle = |x\rangle$  will cancel out. So, we have

$$H^{\otimes n} U_0 H^{\otimes n} |x\rangle = \frac{1}{2^n} (2|\psi\rangle - 2^n |x\rangle) = -|x\rangle + \frac{2}{2^n} |\psi\rangle$$

Let us check

$$H^{\otimes n} U_0 H^{\otimes n} |\psi\rangle = (-|\psi\rangle + 2|\psi\rangle) = |\psi\rangle$$

$$H^{\otimes n} U_0 H^{\otimes n} (|x\rangle - |y\rangle) = |y\rangle - |x\rangle.$$



Now we briefly discuss what to do when there is more than one solution, and, moreover, their number

$$\ell = \#\{w: f(w) = 1\}$$

is not even known in advance.

But let us assume for a second that  $\ell$  is known. What should we do? It depends on the value of  $\ell$ . If  $\ell$  is really large ( $\ell \geq 10^{-2}N$ ), we apply the probabilistic algorithm. If  $\ell$  is small ( $\ell < 10^{-2}N$ ) then we apply straight-forward generalization. We replace  $w$  in  $(|w\rangle, |\psi\rangle)$  with the sum of all good values of  $w$ . It is easy to check that

$$\sin \theta = \sqrt{\frac{\ell}{N-1}}.$$

We then apply Grover Iterate  $\lfloor \sqrt{\frac{N}{\ell}} \frac{\pi}{4} \rfloor$  times.

In case the value of  $\ell$  is not known, we fix a constant  $C = 1.001 > 1$  and assume  $\ell = 1, \lceil C \rceil, \lceil C^2 \rceil, \dots, \lceil C^t \rceil$ , where  $t = O(\log N)$ . We then iterate Grover operator  $\lfloor \sqrt{\frac{N}{\ell}} \frac{\pi}{4} \rfloor$  times for each value of  $\ell$  and then distinguish between good and bad answers. Clearly, as  $\ell$  gets larger the number of iterations gets smaller,

$$\sqrt{\frac{N}{1}} + \sqrt{\frac{N}{C}} + \sqrt{\frac{N}{C^2}} + \dots = O(\sqrt{N}).$$

Even if we do not know the real value of  $\ell$ , in one of our  $O(\log N)$  experiments it will be guessed with sufficiently good accuracy.

## Lectures 6-7

Scribe: Denis Pankratov, University of Chicago.

Date: January 25 and 27, 2011

### 4.2 Factoring: Shor's Algorithm

#### 4.2.1 Reductions

In the *factoring problem* we are given a composite integer  $N \in \mathbb{Z}$ , and we are tasked with finding a nontrivial factor of  $N$ . Note that if we can solve this problem, we can completely factor any integer  $N$  in at most  $\log N$  steps. Recall that primality testing is in  $P$ , as well as computing gcd of two numbers, so we assume that these procedures are readily available to us. First we show how to reduce factoring to the *order finding problem*: given  $N, a \in \mathbb{N}$  such that  $(N, a) = 1$ , find minimum  $r \in \mathbb{N}$  such that  $r > 0$  and  $a^r \equiv 1(N)$ . In the rest of this section on Shor's Algorithm, the notation  $r, a$  and  $N$  will be always used in this sense.

Suppose  $N = p_1^{n_1} p_2^{n_2} \dots p_t^{n_t}$  for some primes  $p_i$ . We may assume that none of the  $p_i$  is 2, otherwise simply divide  $N$  by 2 as many times as possible.

Furthermore we can check if  $N$  has one prime divisor, i.e. if  $N = p^z$  for some odd prime  $p$  and integer  $z$ , since in this case  $z \leq \log N$  and we simply need to check that  $N^{1/z}$  is an integer. Hence, in the rest we assume that  $N$  has all odd prime factors, and at least two different primes. We have  $\mathbb{Z}_n^* \cong \mathbb{Z}_{p_1}^* \times \cdots \times \mathbb{Z}_{p_t}^*$  and  $\mathbb{Z}_{p_i}^* \cong \mathbb{Z}_{p_i^{n_i-1}(p_i-1)}$ . Therefore if  $a \in \mathbb{Z}_n^*$  we can write it as  $a = (a_1, \dots, a_t)$  and  $\text{ord}(a) = \text{lcm}(\text{ord}(a_1), \dots, \text{ord}(a_t))$ . Now, assume that we have access to an order finding black box  $B$ , which on input  $N, a$  outputs the minimal  $r$  such that  $a^r \equiv 1(N)$ . For an integer  $a$  chosen from  $\mathbb{Z}_N$  uniformly at random  $P(B(N, a) \text{ is even}) \geq 1/2$ . Keep picking  $a$  until we get an even  $r$ , i.e.  $r = 2s$  for some  $s \in \mathbb{Z}$ . (Observe that if we accidentally pick  $a$  such that  $(a, N) \neq 1$  we are done). Then we have  $a^{2s} \equiv 1(N)$  and  $(a^s - 1)(a^s + 1) \equiv 0(N)$ .  $a^s \not\equiv 1(N)$  since  $r$  is minimal, and if either  $(a^s - 1, N)$  or  $(a^s + 1, N)$  is a nontrivial factor of  $N$  we are done. The only problem occurs when  $a^s \equiv -1(N)$ , but probability that this happens is at most  $1/2^{t-1}$  (this is where we use the fact that  $N$  is not a prime power). This completes the reduction.

Instead of solving order finding problem directly, we will develop a quantum algorithm to output a rational  $\sigma$  such that for some  $k$  (unknown to us), we have

$$\left| \sigma - \frac{k}{r} \right| < \frac{1}{2N^2}. \quad (1)$$

**Claim 9.** *Once we have  $\sigma$  (as above) we can reconstruct  $r$ .*

*Proof.* Take  $\sigma - \lfloor \sigma \rfloor$ , invert it and repeat to get continued fraction expansion of  $\sigma$ :

$$\sigma = n_0 + \frac{1}{n_1 + \frac{1}{n_2 + \frac{1}{\dots}}}$$

Note that since  $\sigma \in \mathbb{Q}$ , the above procedure converges. If we truncate the continued fractions of  $\sigma$  at some level we obtain  $k/r$  (see e.g. [1, Theorem A4.16]).  $\square$

**Claim 10.** *There is at most one pair  $k', r'$  (in the reduced form) satisfying equation (1).*

*Proof.* Suppose that we have  $k', r'$  such that

$$\left| \sigma - \frac{k'}{r'} \right| < \frac{1}{2N^2}. \quad (2)$$

Then subtracting inequality (1) from (2), we obtain

$$\left| \frac{kr' - k'r}{rr'} \right| < \frac{1}{N^2}.$$

And consequently,

$$|kr' - k'r| < \frac{rr'}{N^2} \leq 1.$$

Since  $k, k', r$ , and  $r'$  are integers, we have  $kr' = k'r$ . It follows that  $k = k'$  and  $r = r'$ , since  $(k', r') = (k, r) = 1$ .  $\square$

## 4.2.2 Linear Algebra

First, we review some background from Linear Algebra.

**Definition 11.** A matrix  $H \in M_n(\mathbb{C})$  is called *Hermitian* if  $H = H^\dagger$ .

**Definition 12.** A matrix  $U \in M_n(\mathbb{C})$  is called *unitary* if  $U^\dagger = U^{-1}$ .

The above two notions are special cases of the following.

**Definition 13.** A matrix  $A \in M_n(\mathbb{C})$  is called *normal* if it commutes with its adjoint, i.e.  $AA^\dagger = A^\dagger A$ .

**Theorem 14** (Spectral Decomposition Theorem). *Any normal matrix  $A$  has a decomposition*

$$A = P\Lambda P^\dagger \tag{3}$$

where  $P$  is unitary and  $\Lambda$  is diagonal.

Observe that Spectral Decomposition Theorem implies that eigenvalues of Hermitian matrices are real, and eigenvalues of unitary matrices lie on a unit circle in the complex plane.

Given  $N \leq 2^n$  define operator  $U_a$  as follows.

$$U_a : |x\rangle \mapsto \begin{cases} |xa \pmod N\rangle & x \in [0, N-1] \\ |x\rangle & x \geq N \end{cases}$$

where  $x \in \{0, 1\}^n$ .

Observe that  $U_a$  is a permutation and is clearly computable in polynomial time. Since,  $U_a^r$  is an identity operator, all eigenvalues of  $U_a$  are  $r^{\text{th}}$  roots of unity, i.e. of the form  $e^{2\pi ik/r}$ . Now, we describe some of the eigenvectors of  $U_a$  that we will need for Shor's Algorithm (all others are obtained in a similar way by shifting this formula to cosets of the subgroup in  $\mathbb{Z}_N$  generated by  $a$ ).

$$|u_k\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} e^{-2\pi iks/r} |a^s \pmod N\rangle$$

It is straightforward to check that  $U_a|u_k\rangle = e^{2\pi ik/r}|u_k\rangle$ . The rest of Shor's Algorithm splits into 2 parts:

**Part 1:** If we have eigenvectors  $|u_k\rangle$ , what do we do with them?

**Part 2:** How do we get vectors  $|u_k\rangle$ ?

### 4.2.3 Part 1: Phase Estimation Algorithm

Consider a more general setting: given a unitary matrix  $U$  and an eigenvector  $|\psi\rangle$  such that  $U|\psi\rangle = e^{2\pi i\omega}|\psi\rangle$  for some  $\omega \in \mathbb{R}$ , estimate  $\omega$  to arbitrary accuracy (think:  $\omega = k/r, |\psi\rangle = |u_k\rangle, U = U_a$ ). It turns out we won't be able to solve it for arbitrary unitary operators. We need one more condition on  $U$ , which will come naturally as we develop the algorithm.

If  $U$  is a unitary operator acting on  $|y\rangle$  define its *controlled version*, denoted by  $c - U$  by

$$\begin{aligned} c - U|0\rangle|y\rangle &= |0\rangle|y\rangle \\ c - U|1\rangle|y\rangle &= |1\rangle U|y\rangle \end{aligned}$$

Note that this generalizes previously defined notion of a controlled  $f$  operator (in which case  $c - U$  is simply a permutation matrix).

**Observation 15.** *If  $U$  is computable by a small circuit then  $c - U$  is also computable by a small circuit.*

*Proof.* Note that for any two unitary operators  $U, V$ , we have  $c - UV = (c - U)(c - V)$ . Since our basis is universal, we can introduce new more complicated gates (controlled version of gates in the basis) and produce the desired circuit with a small increase in size.  $\square$

Now, we want to generalize it further and construct  $c - U^x$ , which given  $x \in \{0, 1\}^t$  (interpreted as an integer in binary) computes

$$c - U^x : |x\rangle|y\rangle \mapsto |x\rangle U^x|y\rangle.$$

The circuit shown in Figure 2 achieves this task (of all the gates shown on this picture, we keep those that correspond to 1 in the binary expansion of  $x$ ).

Observe that in order for the above circuit to be small, we need  $U^{2^t}$  be efficiently computable. This is the additional requirement on the unitary matrix  $U$  we mentioned at the beginning of the section. Observe that in our case,  $U_a^{2^t}|x\rangle = |xa^{2^t} \bmod N\rangle$  can be efficiently computed using repeated squaring algorithm.

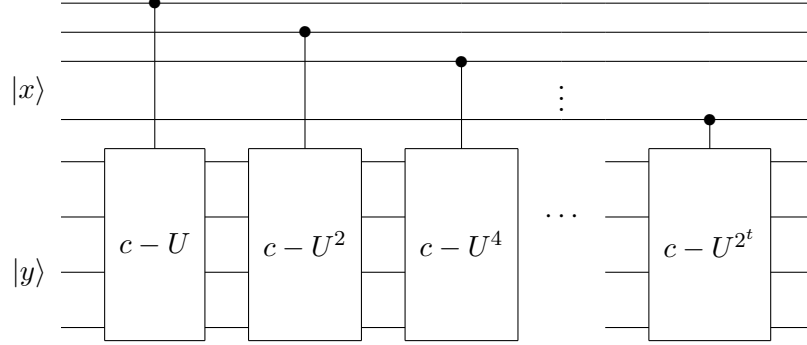


Figure 2: Circuit to compute  $c - U^x$ .

We will need one last ingredient for the phase estimation algorithm. It is *quantum Fourier transform*, defined as follows.

$$\text{QFT}_m : |x\rangle \mapsto \frac{1}{\sqrt{m}} \sum_{y=0}^{m-1} e^{2\pi i xy/m} |y\rangle$$

where  $x, y \in \mathbb{Z}_m$  and  $m \gg N$ . It is easy to check that the inverse of this operator is defined in the following manner.

$$\text{QFT}_m^{-1} : |x\rangle \mapsto \frac{1}{\sqrt{m}} \sum_{y=0}^{m-1} e^{-2\pi i xy/m} |y\rangle.$$

In these notes we are omitting how to prepare  $\text{QFT}_m$ .

The circuit representing the phase estimation algorithm is shown in Figure 3.

Performing the computation, we obtain

$$\begin{aligned} |0\rangle|\psi\rangle &\mapsto \frac{1}{\sqrt{m}} \sum_{x=0}^{m-1} |x\rangle|\psi\rangle \\ &\mapsto \frac{1}{\sqrt{m}} \sum_x |x\rangle U^x |\psi\rangle \\ &= \frac{1}{\sqrt{m}} \sum_x e^{2\pi i \omega x} |x\rangle|\psi\rangle \end{aligned}$$

Finally, we obtain

$$|0\rangle|\psi\rangle \mapsto \frac{1}{m} \sum_{x,y \in \mathbb{Z}_m} e^{2\pi i \omega x - 2\pi i xy/m} |y\rangle|\psi\rangle \quad (4)$$

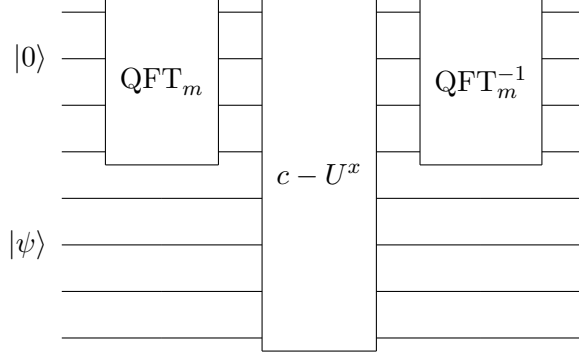


Figure 3: Phase estimation algorithm.

We measure the first register. Let  $p(y)$  be the amplitude in front of  $|y\rangle|\psi\rangle$ , i.e.

$$p(y) = \frac{1}{m} \sum_x e^{2\pi i x(\omega - y/m)}.$$

Clearly, there exists at least one  $y$  such that  $|\omega - y/m| \leq 1/2m$ . We provide an informal argument that for such  $y$  we have  $p(y) > 0.1$  (an analytical expression in a closed form can be found e.g. in [4, Section 7.1.1]). Assume that  $0 \leq \omega - y/m \leq 1/2m$  and consider a unit circle on the complex plane. Each term in the expression for  $p(y)$  represents a unit vector rotated counter clockwise by an angle  $\omega - y/m$ . So after  $m$  steps, we'll move by an angle at most  $\pi$ . If it is in fact less than  $\pi/2$ , then the average of the terms will have a large real part. If it is greater than  $\pi/2$ , then the average of the terms will have a large imaginary part.

By choosing  $m = N^2$  we obtain the desired result.

#### 4.2.4 Part 2: How to Construct $|u_k\rangle$ ?

Recall, that the eigenvectors of interest to us are of the form

$$|u_k\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} e^{2\pi i sk/r} |a^s \pmod N\rangle.$$

We cannot construct individual vectors  $|u_k\rangle$ , but we can construct their uniform superposition  $1/\sqrt{r} \sum_{k=0}^{r-1} |u_k\rangle = |1\rangle$ . We will get a uniform superposition of expressions in (4). There will be no cancellation between different

values of  $k$ , because we measure only the first register. So we can use the Phase Estimation Algorithm with  $|\psi\rangle = |1\rangle$ , and we will get an estimate of a phase for a random value of  $k$ , but this is all we need for our purposes.

### 4.3 Discrete Logarithm

Another problem from the domain of cryptography is a so-called “Discrete Logarithm problem”: given  $N, a, b \in \mathbb{N}$  with  $N$  - prime, find  $t$  such that  $a^t \equiv b \pmod{N}$ . Unlike factoring, this problem is believed to be hard even against Boolean circuits.

To solve this problem efficiently on a quantum computer we will apply Shor’s order-finding algorithm (even though in this case the order can be easily computed as  $r = N - 1$ ). Here we have two operators

$$U_a : |x\rangle \mapsto |xa \pmod{N}\rangle, \text{ and } U_b : |x\rangle \mapsto |xb \pmod{N}\rangle.$$

Note that if we apply order-finding algorithm to both  $U_a$  and  $U_b$  acting on a specific vector  $|u_k\rangle$  then we get a good estimate  $\sigma \approx k/(N - 1)$  and  $\sigma' \approx \ell/(N - 1)$ . Since  $b \equiv a^t \pmod{N}$  we have  $U_b = U_a^t$  and  $\ell = kt$ . Consequently we can estimate  $t \approx \sigma'/\sigma$ . The only problem is that if we apply  $U_b$  after  $U_a$  we lose the vector  $|u_k\rangle$ . The solution to this problem is to apply  $U_a$  and  $U_b$  in parallel.

There is a physical justification for the validity of this argument. Namely, we can measure information partially. The part we measure gets destroyed, but we can continue with the rest as if nothing happened. This idea will be developed later.

The validity of the circuit solving the discrete logarithm problem can be also confirmed with the following direct computation:

$$|0\rangle|\psi\rangle|0\rangle \mapsto \frac{1}{m^2} \sum_{x_1, x_2, y_1, y_2 \in \mathbb{Z}} e^{2\pi i \omega_1 x_1 - 2\pi i x_1 y_1 / m} e^{2\pi i \omega_2 x_2 - 2\pi i x_2 y_2 / m} |y_1\rangle|\psi\rangle|y_2\rangle.$$

Here,  $\omega_1 = k/(N - 1)$  and  $\omega_2 = kt/(N - 1)$ . Then the amplitude in front of  $|y_1\rangle|\psi\rangle|y_2\rangle$  is

$$p(y_1, y_2) = \frac{1}{m^2} \sum_{x_1, x_2} p(y_1)p(y_2),$$

where  $p(y_1)$  and  $p(y_2)$  are as in Shor’s algorithm. Thus if  $p(y_1) > 0.1$  and  $p(y_2) > 0.1$  then  $p(y_1, y_2) > 0.01$ , so we can measure  $y_1, y_2$ , take rounded value of  $y_2/y_1$  as  $t$ , check if it works, and repeat if needed.

## 4.4 Hidden Subgroup Problem

The problems solved by Simon's Algorithm, Shor's Algorithm, and Discrete Logarithm Algorithm can be phrased as instances of a more general *Hidden Subgroup Problem*.

**Simon's Algorithm** Given a finite abelian group  $G = \mathbb{Z}_2^k$  and some function  $f$  such that for a subgroup  $H \leq G$  of index 2 we have  $f(x) = f(y)$  if and only if  $x \in yH$ , the goal is to find  $H$ . ( $yH$  denotes a  $y$ -coset of  $H$ ).

**Shor's Algorithm** Given  $G = \mathbb{Z}$  and  $f(x) = a^x \pmod N$  the goal is to find a hidden subgroup  $H = r\mathbb{Z}$ . Again,  $f(x) = f(y)$  if and only if  $x \in yH$ .

**Discrete Logarithm** Given  $G = \mathbb{Z}_r \times \mathbb{Z}_r$  and  $f(x, y) = a^x b^y \pmod N$  the goal is to find a hidden subgroup  $H = \{(x, y) \mid x + ty = 0\}$  generated by  $(-t, 1)$ .

**Theorem 16** (Kitaev, 95). *If  $G$  is a finitely generated abelian group and  $H \leq G$  is a finite index subgroup then Hidden Subgroup Problem (HSP) for  $G$  is solvable by a polytime quantum algorithm.*

We will not give a proof of this theorem, it can be found e.g. in [2, Section 13.8].

A major open problem is to solve HSP for non-abelian groups. The progress for non-abelian case has been rather limited, but the motivation for studying non-abelian case is quite compelling. There are two great applications.

### 4.4.1 First Application - Symmetric Group

If HSP were solved for  $S_n$  (symmetric group of order  $n!$ ), we could solve the graph isomorphism problem as follows. Given graphs  $G_1$  and  $G_2$ , each on  $n$  variables, consider group  $S_{2n}$ . For  $\sigma \in S_{2n}$  define  $f(\sigma) = \sigma(G_1 \cup G_2)$ , i.e.  $\sigma$  acts on the vertices of  $G_1 \cup G_2$  by permuting them. Then  $f(\sigma_1) = f(\sigma_2)$  if and only if  $\sigma_2^{-1}\sigma_1 \in \text{Aut}(G_1 \cup G_2)$ . Once we know  $\text{Aut}(G_1 \cup G_2)$  (say, by a list of generators  $L$ ) we can decide if two graphs are isomorphic by checking whether there exists a permutation in  $L$  that moves all vertices from  $G_1$  to  $G_2$ .

### 4.4.2 Second Application - Dihedral Group

Dihedral group, denoted  $D_{2n}$ , is defined as a group of symmetries of a regular  $n$ -gon. The order of  $D_{2n}$  is  $2n$ . Let  $r$  be a rotation by  $2\pi/n$  counter clockwise, and  $s$  be a reflection through vertex 1 and  $n/2$  if  $n$  is even or the center of the opposite edge if  $n$  is odd. Then  $D_{2n}$  consists of  $r^i$  and  $sr^i$  for  $0 \leq i \leq n-1$ . In a sense,  $D_{2n}$  is very close to being abelian, since  $[D_{2n} : \mathbb{Z}_n] = 2$ . Observe that  $D_{2n}$  contains many involutions (subgroups of order 2), and it is not known if one can detect a subgroup of order 2.

Shortest Vector Problem (SVP) in lattices  $\mathbb{Z}^n \subset \mathbb{R}^n$  is to find a shortest non-zero vector in a lattice, i.e.  $\min\{|v| \mid v \in \mathbb{Z}^n \setminus \{0\}\}$ . Ajtai and Dwork [5] showed how to create a public-key cryptographic system whose security could be proven using only worst-case hardness of a certain version of SVP. This was the first result that used worst-case hardness to create secure systems. However, if you can solve HSP (in a sense) for  $D_{2n}$  then you can break SVP (almost) [6]. Let us describe one of the technicalities here.

Most of the current approaches to HSP for non-abelian groups use the operator  $U_f$  only via the following algorithm known as the *coset sampling algorithm* (that is a reasonable assumption due to the absolutely generic nature of the function  $f$ ). Consider  $f : G \rightarrow Z$ , where  $G$  is a group and  $Z$  is an arbitrary set,  $|Z| = N$ , and a subgroup  $H \leq G$ ,  $L = |H|$ . Since  $U_f : |x, 0\rangle \mapsto |x, f(x)\rangle$ , we have  $U_f : 1/\sqrt{N} \sum_x |x, 0\rangle \mapsto 1/\sqrt{N} \sum_x |x, f(x)\rangle$ . We “measure the second register” and obtain value of  $f(x) = y$ , we then continue the computation. Intuitively, we expect to obtain a uniform superposition of all  $x$  in a “random” coset of  $H$ , i.e.  $1/\sqrt{L} \sum_{f(x)=y} |x\rangle$ .

In the next section we show how to make these notions precise.

## Lecture 8

Scribe: Kenley Pelzer, University of Chicago.

Date: February 1, 2011

## 5 Quantum Probability

Deficiencies of the current formalism of unitary operators:

1. Probability distributions (dealing with randomness) over pure states need to be considered.

2. A problem with building quantum computers is the issues with noise and decoherence; we need a way to describe quantum noise (because no system is completely isolated from the environment). The unitary model is not up to the challenge; we need to consider mixed states.

3. We need to consider partial measurement (tracing out).

We start with a set of unitary (pure) states and their probabilities:

$$(p_1, |\psi_1\rangle) (p_2, |\psi_2\rangle) \\ (p_1, |\psi_1\rangle) (p_2, |\psi_2\rangle) \dots$$

Each  $|\psi\rangle$  is also an exponential sum:

$$|\psi\rangle = \sum_x \alpha_x |x\rangle \tag{5}$$

This is messy, so we want something more concise: there is an invariant that we can work with.

If two (possibly mixed) states have the same invariant, they are physically indistinguishable (in our world, this means that they are *computationally* indistinguishable).

A density matrix is such an invariant.

$|\psi\rangle|\psi\rangle \leftarrow$  tensor product (unit vector in a larger space)

$\langle\psi|\psi\rangle \leftarrow$  scalar product

$\rho_\psi = |\psi\rangle\langle\psi| =$  density operator for state  $\psi$  (also called the “outer product”)

$$\rho_\psi(x, y) = \alpha_x \alpha_y^* \text{ if } |\psi\rangle = \sum_x \alpha_x |x\rangle.$$

Three important properties of density matrices:

- (a) Density matrix is Hermitian.
- (b) Trace of density matrix is 1.
- (c) Matrix is positive semidefinite (its eigenvalues are non-negative).

Definition: A density matrix is any square matrix that satisfies all of the conditions (a)-(c) listed above.

If we take a convex combination of density matrices, we get another density matrix.

So we can sum density matrices with corresponding probabilities:

$$\rho = p_1 |\psi_1\rangle\langle\psi_1| + \dots + p_t |\psi_t\rangle\langle\psi_t|.$$

$\rho$  is a density matrix. We apply a unitary operation on both sides:

$$U\rho_\psi U^\dagger = U|\psi\rangle\langle\psi|U^\dagger = |U\psi\rangle\langle\psi U| \text{ (by definition of bra-ket rules)}$$

We can do many more great things now, like half of a unitary operation, giving

$$\rho \rightarrow \frac{1}{2}\rho + \frac{1}{2}U\rho U^\dagger$$

Another thing is *depolarization at the rate  $\eta$*  defined as

$$\mathcal{E}_\eta(\rho) = (1 - \eta)\rho + \frac{\eta}{N}I_N,$$

where  $I_N$  is the identity matrix.

Many more noise channels can be found in [1, Section 8.3]; some of them will also be considered in Section 8 below.

Another way to create the identity matrix is to say that each state occurs with probability  $\frac{1}{N}$  if there are  $N$  states. This is the mathematical equivalent of a “completely depolarized” (or “totally random”) state.

Two things to NOT mix up:

$$|x_1\rangle, p = \frac{1}{N}, |x_2\rangle, p = \frac{1}{N}, \dots, |x_N\rangle, p = \frac{1}{N} \text{ with density matrix } \frac{1}{N}I_N$$

versus the *uniform superposition*,

$$\psi = \frac{1}{\sqrt{N}} \sum_x |x\rangle \tag{6}$$

with density matrix:

$$\frac{1}{N} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

If you apply a measurement immediately, these two density matrices give equivalent results, but after applying a unitary operator, we get very different results (as we have already seen many times before).

### 5.1 “Tracing out” or “partial measurement”

Take state  $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \leftarrow$  pure state. Upon measuring (and then discarding) the second register, we intuitively should get the mixed state

$$(|0\rangle, \frac{1}{2}), (|1\rangle, \frac{1}{2})$$

In vector space  $\mathcal{H}_A \otimes \mathcal{H}_B$ ,

$$\psi = \sum_{a,b} \alpha_{ab} |a\rangle |b\rangle \quad (7)$$

$$\sum_{a,b} |\alpha_{ab}|^2 = 1. \quad (8)$$

Denoting  $\sum_a |\alpha_{ab}|^2$  by  $p_b$ , with probability  $p_b$  we get

$$\frac{1}{\sqrt{p_b}} \sum_a \alpha_{ab} |a\rangle \quad (9)$$

$$\sum_b \left[ p_b \sum_{a_1, a_2} \left( \frac{1}{p_b} \alpha_{a_1, b} \alpha_{a_2, b}^* |a_1\rangle \langle a_2| \right) \right] = \quad (10)$$

$$\sum_{a_1, a_2, b} \alpha_{a_1, b} \alpha_{a_2, b}^* |a_1\rangle \langle a_2| = \quad (11)$$

$$\sum_{a_1, a_2, b_1, b_2} (\alpha_{a_1, b_1} \alpha_{a_2, b_2}^* |a_1\rangle \langle a_2| \langle b_1| \langle b_2|) \quad (12)$$

Thus,  $Tr_B (|a_1\rangle \langle a_2| \otimes |b_1\rangle \langle b_2|)$  “should” be defined as

$$|a_1\rangle \langle a_2| \cdot \langle b_2| \langle b_1|. \quad (13)$$

This operation is called “tracing out”. It is a good exercise to check that this operator indeed takes density matrices to density matrices.

## 5.2 Superoperators

All examples of quantum operations we have seen so far share the following properties: they are linear operators that act on matrices, take matrices of one size to matrices of another (possibly, different) size and take density matrices to density matrices. This is “almost” the right definition of a *superoperator* or an operator “physically realizable” in nature, for the completely right one see e.g. [1, Section 8.2.4]. We will see one more (and sometimes more useful) definition in Section 8.

A superoperator is not necessarily reversible.

## Lecture 9

Scribes: Kenley Pelzer and Tatiana Orlova, University of Chicago.

Date: February 8, 2011

If you want to measure noise, we need to know distance between two different states.

In unitary world, pure states are just unit vectors, so "distance" is angle between them (there's hardly any other choice).

Probability distributions:

Statistical difference ( $l_1$ ) is represented by a diagonal matrix:

$$\begin{pmatrix} p_1 & 0 & 0 & 0 \\ 0 & p_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & p_n \end{pmatrix} - \begin{pmatrix} q_1 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & q_n \end{pmatrix} = \begin{pmatrix} a_1 & 0 & 0 & 0 \\ 0 & a_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & a_n \end{pmatrix}$$

Given event  $E$ , the difference between the probability of event  $E$  given two different distributions can be bounded by:

$$|P_p(E) - P_q(E)| \leq \frac{1}{2} \sum_{i=1}^N |a_i| \quad (14)$$

This is the statistical distance between two distributions; note that we must take absolute value since

$$\sum_{i=1}^N a_i = 0. \quad (15)$$

Assume now that  $\varrho$  and  $\varsigma$  are density matrices. It would be tempting to define the *trace distance* between them simply as

$$D(\varrho, \varsigma) = \text{Tr}(|\varrho - \varsigma|), \quad (16)$$

but we want to be able to measure in an arbitrary basis. Thus, we let

$$D(\varrho, \varsigma) = \max_U \text{Tr}(|U(\varrho - \varsigma)U^\dagger|). \quad (17)$$

**Theorem 17.** *If  $T$  is an arbitrary superoperator, then for any pair of density matrices:*

$$D(T(\varrho), T(\varsigma)) \leq D(\varrho, \varsigma) \quad (18)$$

The proof is omitted and can be found e.g. in [1, Section 9.2.1].

## 6 Quantum Complexity Theory: black-box model

In the general black-box problem, we are typically given a function  $f : [N] \rightarrow \{0, 1\}$ . While the set  $[N]$  can actually be of arbitrary nature, in many interesting cases it is comprised of binary strings. To commemorate this fact, we use lower case letters  $x, y, z$  etc. for its elements (and we will typically represent the function  $f$  by its truth-table  $X = (X_1, \dots, X_N)$ , where  $X_x = f(x)$ ).

### 6.1 Hybrid method: optimality of Grover's search

In the search problem we want to find  $x \in \{1, 2, \dots, N\}$  such that  $f(x) = 1$ . We have shown that Grover's search algorithm solves this problem by making  $O(\sqrt{N})$  queries to the black-box  $U_f$  (see Theorem 8 in Section 4). We now show that this result is the best possible.

**Theorem 18.** *Grover's search algorithm is optimal, i.e. every quantum black-box search algorithm requires  $\Omega(\sqrt{N})$  queries.*

The proof of the above theorem follows directly from the same lower bound for the corresponding decision problem. We now state the decision problem and then prove the lower bound for it.

Let  $X \stackrel{df}{=} (X_1, \dots, X_N)$ , where  $X_x \in \{0, 1\}$ , such that  $X_x = f(x)$  for all  $x \in [N]$ . We will denote by  $X_0$  an all-zero string, i.e. the string  $X = (X_1, \dots, X_N)$ , such that  $X_x = 0$  for all  $x \in [N]$ , and by  $\mathbf{X}_x$  the string  $X = (X_1, \dots, X_N)$ , such that  $X_y = \begin{cases} 1, & \text{if } y = x; \\ 0, & \text{otherwise} \end{cases}$ . In other words,  $\mathbf{X}_x$  is the string that contains precisely one 1 in the  $x$ th place. We want to compute the following function

$$F(X) \stackrel{df}{=} \begin{cases} 0, & \text{if } X \equiv X_0; \\ 1, & \text{if } X \in \{\mathbf{X}_x\}_{x \in [N]}. \end{cases}$$

**Theorem 19.** *Computing  $F(X)$  requires  $\Omega(\sqrt{N})$  queries to the black box  $U_f$ .*

*Proof.* Let

$$|\psi_j^x\rangle \stackrel{df}{=} U_j U_{\mathbf{X}_x} U_{j-1} \dots U_1 U_{\mathbf{X}_x} |\psi\rangle,$$

and

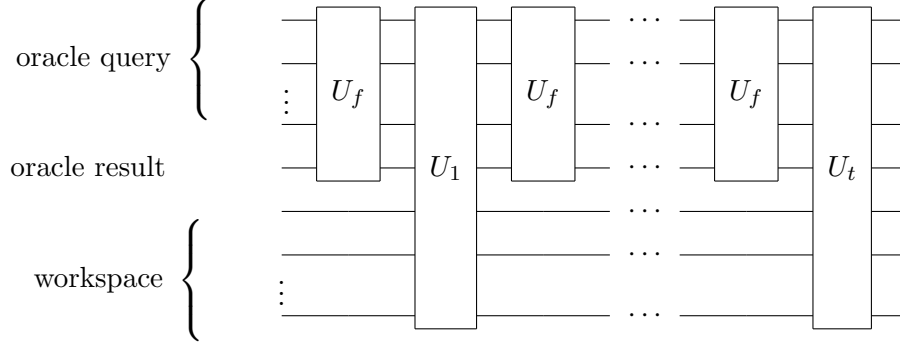


Figure 4: Quantum circuit for black-box query computation.

$$|\psi_j\rangle \stackrel{df}{=} U_j U_{j-1} \dots U_1 |\psi\rangle$$

(note that  $U_{X_0}$  is the identity operator).

We want to look at the distance  $\|\psi_j^x - |\psi_j\rangle\|$  (where  $\|\cdot\|$  stands for Euclidean norm). For  $j = 0$  this distance is 0. When  $j$  goes from 0 to  $t$ , the distance must change from 0 to 1% for any fixed  $x$  as our circuit must be able to distinguish between  $X_0$  and  $\mathbf{X}_x$ . We want to prove that the distance cannot change by more than a certain amount that depends on  $t$ . This will bound the number of times we have to apply the operator  $U_f$  in order to successfully solve the problem.

Let  $|\psi_j\rangle = \sum_y \alpha_{y,j} |y\rangle |\phi_y\rangle$ , where  $\sum_y |\alpha_{y,j}|^2 = 1$  and  $\|\phi_y\| = 1$ . On Figure 4,  $|y\rangle$  corresponds to the first register, and  $|\phi_y\rangle$  is the combination of oracle result and workspace. Also, let  $|\psi_j^x\rangle = \sum_y \alpha_{y,j}^x |y\rangle |\phi_y^x\rangle$ , and  $|\psi_{j+1}^x\rangle = \sum_y \alpha_{y,j+1}^x |y\rangle |\tilde{\phi}_y^x\rangle$ , where coefficients do not change, i.e.  $\alpha_{y,j+1}^x = \alpha_{y,j}^x$  for all  $y$ , and  $|\tilde{\phi}_y^x\rangle = |\phi_y^x\rangle$  unless  $y = x$ . For  $y \neq x$ , we have no control over what happens to  $|\tilde{\phi}_y^x\rangle$  when changing from the state  $|\psi_j^x\rangle$  to  $|\psi_{j+1}^x\rangle$ , except for the fact that the length is preserved (and it will be important). Thus the operator  $|\psi_j\rangle \rightarrow |\psi_{j+1}\rangle = |\psi_j\rangle$  acts identically, but  $|\psi_j^x\rangle \rightarrow |\psi_{j+1}^x\rangle$  does not.

The first obvious idea to try is the triangle inequality

$$\| |\psi_{j+1}\rangle - |\psi_{j+1}^x\rangle \| \leq \| |\psi_j\rangle - |\psi_j^x\rangle \| + \| |\psi_j^x\rangle - |\psi_{j+1}^x\rangle \| \leq \| |\psi_j\rangle - |\psi_j^x\rangle \| + 2|\alpha_{x,j}^x|.$$

That is good, but for certain reasons that will become clear later, we would like the above inequality to depend only on values  $|\alpha_{x,j}|$  (without the

superscript  $x$ ). For this purpose we split our Hilbert space  $\mathcal{L}$  into a direct sum of two subspaces

$$\mathcal{L} = \left( \bigoplus_{y \neq x} \mathcal{L}_y \right) \oplus \mathcal{L}_x.$$

In the subspace  $\bigoplus_{y \neq x} \mathcal{L}_y$  all operators act identically. In the subspace  $\mathcal{L}_x$  we have

$$\| \alpha_{x,j} |\phi_x\rangle - \alpha_{x,j}^x |\phi_x^x\rangle \| + \alpha_{x,j} \geq \alpha_{x,j}^x \geq \| \alpha_{x,j} |\phi_x\rangle - \alpha_{x,j+1}^x |\tilde{\phi}_x^x\rangle \| - \alpha_{x,j}$$

(We did use for this calculation that  $\alpha_{x,j+1}^x = \alpha_{x,j}^x$ !) This gives us the desired bound in  $L$

$$\| |\psi_{j+1}\rangle - |\psi_{j+1}^x\rangle \| \leq \| |\psi_j\rangle - |\psi_j^x\rangle \| + 2|\alpha_{x,j}| \quad (19)$$

(An exercise!)

The amplitudes of  $|\psi_0\rangle = |\psi\rangle, |\psi_1\rangle, \dots, |\psi_t\rangle$  form a  $t \times N$  matrix

$$\begin{pmatrix} |\alpha_{0,0}| & \dots & |\alpha_{x,0}| & \dots \\ |\alpha_{0,1}| & \dots & |\alpha_{x,1}| & \dots \\ \vdots & \ddots & \vdots & \vdots \\ |\alpha_{0,t-1}| & \dots & |\alpha_{x,t-1}| & \dots \end{pmatrix}$$

By (19), when going from step 0 to step  $t$  of the computation the distance  $\| |\psi_j\rangle - |\psi_j^x\rangle \|$  is growing by at most twice the sum of the amplitudes from the  $x$ -th column  $2 \sum_{j=0}^{t-1} |\alpha_{x,j}|$ . To successfully recognize  $\Omega(N)$  of the non-zero strings with bounded probability we must have

$$\sum_{x=1}^N \sum_{j=0}^{t-1} |\alpha_{x,j}| = \Omega(N)$$

Note that  $\sum_{x=1}^N |\alpha_{x,j}|^2 = 1$ , since it is a sum of the amplitudes of a quantum state. The Cauchy-Schwartz inequality implies  $\sum_{x=1}^N |\alpha_{x,j}| \leq \sqrt{N}$ . We have

$$\sum_{x=1}^N \sum_{j=0}^{t-1} |\alpha_{x,j}| \leq t\sqrt{N}.$$

Thus,

$$t\sqrt{N} = \Omega(N)$$

and

$$t = \Omega(\sqrt{N}).$$

This completes the proof.  $\square$

## Lectures 10 and 11

Scribe: Philip Reinhold, University of Chicago

Date: February 10 and 15, 2011

### 6.2 Quantum Query Complexity vs. Other Complexity Measures

While Simon's Algorithm demonstrates the feasibility of exponential quantum speedup in black-box query complexity for at least one problem, another approach has shown that for another class of problems, the best one can achieve is a polynomial speedup. Namely, Simon's problem dealt with a predicate which was only defined on certain inputs, specifically that the input function  $f$  satisfied  $\exists s \forall x \forall y (f(x) = f(y) \leftrightarrow x = y \vee x \oplus y = s)$ . In the case that the input to Simon's algorithm did not satisfy this promise, the output is not well defined. We will now see that when we forbid the last feature, the situation changes dramatically.

**Definition 20.** A property  $F : \{0, 1\}^N \rightarrow \{0, 1\}$  is *total* if its output is defined for all inputs  $\{0, 1\}^N$ .

**Definition 21.** For a property  $F$  under the black-box model,  $D(F)$  is the *deterministic complexity*, i.e. the number of calls to the black box that must be made (in the worst-case) with a deterministic classical algorithm to determine the property.

Note that in this definition we do not count the internal work of the algorithm, only the number of queries.

**Definition 22.** For a property  $F$  under the black-box model,  $Q_2(F)$  is the *bounded-error quantum complexity*, i.e. the number of calls to the black box that must be made with a quantum computer such that the probability of returning the correct answer is at least  $2/3$ .

We have the following theorem relating these two (and a few other) measures.

**Theorem 23.** For any total property  $F$ ,  $D(F) \leq O(Q_2(F)^6)$ .

*Proof.*

$$\begin{aligned}
D(F) &\leq O(C^{(1)}(F)\text{bs}(F)) \\
&\leq O(\text{s}(F)\text{bs}(F)^2) \\
&\leq O(\text{bs}(F)^3) \\
&\leq O(\widetilde{\text{deg}}(F)^6) \\
&\leq O(Q_2(F)^6).
\end{aligned}$$

□

Fleshing out the content of this proof will be the following supporting concepts and theorems; for historical attributions of all these pieces see e.g. the survey [7].

**Definition 24.** A *one-certificate* for  $F : \{0, 1\}^N \rightarrow \{0, 1\}$  is an assignment  $c : S \rightarrow \{0, 1\}$  for some  $S \subseteq [N]$  such that for all inputs  $X$  that are consistent with  $c$ ,  $F(X) = 1$ . An input to  $F$ ,  $X$  is consistent with  $c$  iff  $\forall i \in S (X_i = c_i)$ .

The *one-certificate complexity*,  $C^{(1)}(F)$  is the minimum value such that for all inputs  $X$  on which  $F(X)$  is true, there exists a certificate  $c$  such that  $X$  is consistent with  $c$  and  $|c| \leq C^{(1)}(F)$ .

**Definition 25.** A function  $F : \{0, 1\}^N \rightarrow \{0, 1\}$  is *sensitive* on  $B \subseteq [N]$  for input  $X$  iff flipping the values  $X_i$  for  $i \in B$  flips the output of  $F$ , i.e.  $F(X) \neq F(X \oplus B)$ . Let the *block sensitivity* of  $F$ ,  $\text{bs}(F)$  be the size of the largest set of **disjoint** blocks  $B_i$  such that for some input  $X$ ,  $F$  is sensitive on  $B_i$  for  $X$ , for  $i$  from 1 to  $\text{bs}(F)$ .

**Theorem 26.**  $D(F) \leq C^{(1)}(F)\text{bs}(F)$ .

*Proof.* We show this by showing an algorithm whose steps are based on a single certificate (that is, use at most  $C^{(1)}$  queries), which converges on an answer after at most  $\text{bs}(F)$  steps.

At each stage we pick a certificate  $c : S \rightarrow \{0, 1\}$  of size at most  $C^{(1)}(F)$  which is consistent with those  $X_i$  already queried (If there is no consistent  $c$ , output 0 and stop.) We then query  $X_i$  for all previously unqueried  $i \in S$ . If  $X$  is consistent with  $c$ , output 1 and stop. If we have not terminated after  $\text{bs}(F)$  steps, pick any input  $Y$  consistent with those  $X_i$  queried, and output  $F(Y)$ .

The claim is that for any two inputs  $Y, Y'$  as above we necessarily have  $F(Y) = F(Y')$  (and thus in particular  $F(Y) = F(X)$ ). If we assume not, we

can show that there are disjoint subsets of the input on which  $F$  is sensitive,  $B_1, B_2, \dots, B_{b+1}$ , where  $b = \text{bs}(F)$ .

Let  $c_i$  for  $i \in [b]$  be the certificates whose indices were queried in the algorithm. Let  $Y, Y'$  be as above; assume w.l.o.g. that  $F(Y) = 0, F(Y') = 1$ . Let the certificate for  $Y'$  be  $c_{b+1}$ . Let  $B_i$  for  $i \in [b+1]$  be the set of variables on which  $c_i$  and  $Y$  disagree. Then,  $\forall i F(Y \oplus B_i) = 1$ , which shows that  $F$  is sensitive on  $B_i$ . To show that these sets are disjoint, consider two certificates used,  $c_i$  and  $c_j$ , where  $i < j$ . For all variables  $v \in B_i$ ,  $X_v = Y_v \neq c_i(v)$ . However, having queried  $v$  in step  $i$ , the certificate  $C_j$  would be picked to be consistent on  $v$ , so  $X_v = C_j(v)$ . So  $v \notin B_j$ . Therefore,  $B_i$  form a disjoint set of  $b+1$  blocks on which  $F$  is sensitive, which is a contradiction.  $\square$

**Definition 27.** The *sensitivity* of  $F$ ,  $s(F)$  is the number of variables of the input  $X$  on which a flip guarantees a flip in  $F(X)$ . It is equivalent to the block sensitivity with the additional restriction that  $|B_i| = 1$ , so  $s(F) \leq \text{bs}(F)$ .

**Theorem 28.**  $C^{(1)}(F) \leq O(s(F)\text{bs}(F))$

*Proof.* Let  $\text{bs}_X(F) \leq \text{bs}(F)$  be the size of a maximal set of blocks  $B_i$  such that  $F$  is sensitive for  $X$  on all  $B_i$ . Furthermore, let these blocks be minimal in  $|B_i|$ . It follows that  $c : \bigcup_i B_i \rightarrow \{0, 1\}$ ,  $c(i) = X_i$ , is a certificate for  $X$ , since if not, there would be the  $B_{\text{bs}_X(F)+1}$  defined as those input variables not in  $\bigcup_i B_i$  which  $X$  is still sensitive on. Since these blocks are minimally sized,  $X \oplus B_i$  must be sensitive on  $v$  for all  $v \in B_i$ , so  $|B_i| \leq s_{X \oplus B_i}(F) \leq s(F)$ . Since  $B_i$  are disjoint,

$$|c| = \sum_{i=1}^{\text{bs}(F)} |B_i| \leq s(F)\text{bs}(F)$$

$\square$

It is a big open problem to determine whether  $s(F)$  and  $\text{bs}(F)$  are always polynomially related; for a comprehensive survey on this problem see [8].

The following theorem requires the symmetrization technique. Let  $p : \mathbf{R}^N \rightarrow \mathbf{R}$  be a multilinear polynomial. Let a permutation  $\pi \in S_N$  be a rearrangement of the variables composing the input to  $p$ , i.e.  $\pi(X) = (X_{\pi_1}, X_{\pi_2}, \dots, X_{\pi_N})$ . The *symmetrization* of  $p$  is the average of  $p$  over all permutations of the inputs, i.e.

$$p^{\text{sym}}(X) = \frac{\sum_{\pi \in S_n} p(\pi(X))}{N!}.$$

**Lemma 29.**  $p^{\text{sym}}(X)$  can be equivalently written as a single-variate polynomial  $q(|X|)$ .

*Proof.* Let  $p(X) : \mathbf{R}^N \rightarrow \mathbf{R}$  be a multilinear polynomial. Let  $V_j$  denote the sum of all products of  $j$  input variables  $X_i$ . There are  $\binom{N}{j}$  terms in this sum. Since  $p^{\text{sym}}$  is symmetrical, it can be written as

$$p^{\text{sym}}(X) = a_0 + \sum_{i=1}^N a_i V_i.$$

On inputs  $X \in \{0, 1\}^N$ , the only terms contributing to the sum  $V_i$  are those which are 1. With  $|X| \equiv \sum_i X_i$ , there are  $\binom{|X|}{i}$  such terms, leaving  $V_j$  with the same value. Thus

$$p^{\text{sym}}(X) = q(|X|) \equiv a_0 + \sum_i a_i \binom{|X|}{i}.$$

□

**Definition 30.** The *approximate degree* of  $F$ ,  $\widetilde{\text{deg}}(F)$  is the smallest degree of a multi-linear polynomial which approximates  $F$ . More formally

$$\widetilde{\text{deg}}(F) = \min_p \{ \text{deg}(p) \mid \forall x \in \{0, 1\}^N : |p(x) - F(x)| \leq \frac{1}{3} \}.$$

Furthermore, this theorem relies on a result of Ehlich, Zeller, Rivlin and Cheney.

**Lemma 31.** If polynomial  $p$  is bounded, i.e.  $\forall i \in [N] b_1 \leq p(i) \leq b_2$ , and  $\exists x \in \mathbf{R} \left| \frac{dp(x')}{dx'}(x) \right| \geq c$  then

$$\text{deg}(p) \geq \sqrt{\frac{cN}{c + b_2 - b_1}}.$$

**Theorem 32.**  $\text{bs}(F) \leq O(\widetilde{\text{deg}}(F))$ .

*Proof.* Let  $p$  be the polynomial that approximates  $F$ , and let  $B_i$  be the  $\text{bs}(F) = b$  blocks on which  $F$  is sensitive. Let  $Y = (Y_1, \dots, Y_b)$  be a  $b$ -variate variable. For some input  $X$  where  $F(X) = 0$ , define  $Z = (Z_1, \dots, Z_N)$  such that  $Z_i = X_i \oplus Y_j$  if  $i \in B_j$ , and  $Z_i = X_i$  if  $i \notin B_1 \cup \dots \cup B_b$  (thus, in particular, when  $Y = \vec{0}$ ,  $Z = X$ ). Define  $q(Y) = p(Z)$ , making  $q(Y)$  a  $b$ -variate polynomial of degree  $\text{deg}(p)$ .

Note that since  $p$  is bounded in  $\{0, 1\}^N$ , so is  $q$  (in  $[N]$ ). Furthermore

$$|q(\vec{0}) - 0| = |p(X) - F(X)| \leq 1/3$$

(by the definition of  $\widetilde{\deg}(F)$ ), and for any input  $Y$  with Hamming weight  $|Y| = 1$ ,

$$|q(Y) - 1| = |p(X \oplus B_i) - F(X \oplus B_i)| \leq 1/3,$$

since  $F(X)$  flips when flipping block  $B_i$ .

Let  $r(|Y|) = q^{\text{sym}}(Y)$  be the single-variate polynomial obtained from symmetrizing  $q$  as in Lemma 29. Since  $r$  is obtained as an average over the possible inputs of  $q$ , the aforementioned properties translate to  $r$ , namely  $r(n) \in [0, 1]$ ,  $r(0) \leq 1/3$ ,  $r(1) \geq 2/3$ . By the mean value theorem we have  $|\frac{dr(n')}{dn'}(n)| \geq 1/3$  for some  $n$  in  $[0, 1]$ .

Using this value as  $c$ , and the  $[0, 1]$  bound for  $r$  in lemma 31 we have  $\sqrt{\frac{\frac{2}{3}b}{\frac{1}{3}+1-0}} = \sqrt{\frac{b}{4}} \leq \deg(r) \leq \deg(p)$ . So  $\text{bs}(F) \leq O(\widetilde{\deg}(F))$ .  $\square$

**Theorem 33.**  $\widetilde{\deg}(F) \leq O(Q_2(F)^2)$ .

*Proof.* Write a quantum circuit as an alternating series of arbitrary unitary transformations ( $U_j$ ) and queries to the input  $X$  ( $U_X$ ), see Figure 4. The output of this circuit can be written as  $\sum_{k \in K} \alpha_k^X |k\rangle$  where  $K$  is the set of possible output strings. We first claim that for any fixed  $k$ ,  $\alpha_k^X$  can be written as a multilinear polynomial in the variables  $X = (X_1, \dots, X_N)$  with  $\deg(\alpha_k^X) \leq Q_2(F)$ .

Write the state of the circuit just after applying the  $j$ th query to  $X$  as  $|\psi_j\rangle$ . Then  $|\psi_{j+1}\rangle = U_j U_X |\psi_j\rangle$  (cf. the proof of Theorem 19). Write without loss of generality that the query  $U_X$  maps  $|k\rangle = |y, b, z\rangle$  to  $|y, b \oplus X_y, z\rangle$ . Then we have the formula

$$U_X \left( \sum_{y,b,z} \alpha_{y,b,z}^X |y, b, z\rangle \right) = \sum_{y,b,z} (\alpha_{y,b,z}^X (1 - X_y) + \alpha_{y,b \oplus 1,z}^X X_y) |y, b, z\rangle,$$

which implies that the degree of the polynomials  $\alpha_{y,b,z}^X$  can increase by at most 1 from applying the operator  $U_X$ . And the unitary transformation  $U_j$  does not depend on  $X$  at all, therefore it will output a polynomial of the same degree as its input.

It follows that the amplitudes  $\alpha_t^X$  after  $t$ th step will be a polynomial of degree at most  $t$ . The probability of observing any basis state is  $p_k = |\alpha_k^X|^2$ , a polynomial of degree at most  $2t$ .  $\square$

**Remark 1.** Whether the bound of Theorem 23 can be improved is a major open problem. Some of the intermediate steps in this chain of inequalities are known to be tight, and some are not, we again refer to the survey [7] for further details.

**Remark 2.** The bound in Theorem 33 is called *polynomial method*. This is our second technique for proving quantum lower bounds in the black-box model; we will review one more in the next lecture.

## Lectures 12 and 13

Scribe: Pooya Hatami, University of Chicago

Date: February 16 and 17, 2011

### 6.3 Ambainis's Adversary Method

Let  $F : \{0, 1\}^N \rightarrow \{0, 1\}$  be a Boolean function. Consider sets  $\mathcal{X}, \mathcal{Y} \subseteq \{0, 1\}^N$  such that:

- $\forall X \in \mathcal{X} : F(X) = 1$ , and
- $\forall Y \in \mathcal{Y} : F(Y) = 0$ .

Let  $R \subseteq X \times Y$  be a binary relation. Defining  $b$  and  $b'$  as follows

$$b = \max_{X \in \mathcal{X}, z \in [n]} |\{Y | R(X, Y) \wedge X_z \neq Y_z\}|,$$

$$b' = \max_{Y \in \mathcal{Y}, z \in [n]} |\{X | R(X, Y) \wedge X_z \neq Y_z\}|,$$

we have the following theorem

**Theorem 34** (Ambainis).

$$Q_2(F) \geq \Omega\left(\frac{|R|}{\sqrt{|\mathcal{X}||\mathcal{Y}|bb'}}\right).$$

*Proof.* Let  $|\psi_j^X\rangle$  be the state just after the  $j$ th call to the oracle when the computation is led by  $X$ . Similar to what we have seen in Hybrid method (Theorem 19) and Polynomial method (Theorem 33), we have

$$|\psi_j^X\rangle = \sum_z \alpha_{z,j}^X |z\rangle |\phi_{z,j}^X\rangle.$$

We will study the following sum

$$W_j := \sum_{(X,Y) \in R} |\langle \psi_j^X | \psi_j^Y \rangle|.$$

Notice that  $W_0 = |R|$ .

Suppose algorithm  $\mathcal{A}$  has the property that for any input  $Z$  the probability of guessing the correct answer is at least  $1 - \epsilon$ . This means that the final stage of  $\mathcal{A}$  can correctly distinguish  $|\psi_t^X\rangle$  from  $|\psi_t^Y\rangle$  for any  $X, Y$  with  $F(X) \neq F(Y)$  (in particular, when  $(X, Y) \in R$ ) with probability at least  $1 - \epsilon$ . Intuitively, it should be clear that it implies that the states  $|\psi_t^X\rangle$  and  $|\psi_t^Y\rangle$  can not be too close to each other; the following theorem makes this intuition precise.

**Theorem 35** ([4, Theorem 9.2.1]). *Any procedure that on input  $|\psi_Z\rangle$  guesses whether  $Z = X$  or  $Z = Y$  will guess correctly with probability at most  $1 - \epsilon = \frac{1}{2} + \frac{1}{2}\sqrt{1 - \delta^2}$ , where  $\delta = |\langle\psi_X|\psi_Y\rangle|$ . This probability is achievable by an optimal measurement.*

By the above theorem we know that we must have

$$|\langle\psi_t^X|\psi_t^Y\rangle| = \delta \leq 2\sqrt{\epsilon(1 - \epsilon)},$$

and thus  $W_t \leq 2\sqrt{\epsilon(1 - \epsilon)}|R|$ , which for  $\epsilon < 1/2$  is less than  $|R|$ . Therefore it suffices to prove that  $W_{j+1} \geq W_j - 2\sqrt{|\mathcal{X}||\mathcal{Y}|bb'}$ .

**Lemma 36.**  $W_{j+1} \geq W_j - 2\sqrt{|\mathcal{X}||\mathcal{Y}|bb'}$ .

*Proof.* From definition of  $W_j$  we know that

$$|W_{j+1} - W_j| \leq \sum_{(X,Y) \in R} |\langle\psi_{j+1}^X|\psi_{j+1}^Y\rangle - \langle\psi_j^X|\psi_j^Y\rangle|.$$

We also know that

$$|\psi_{j+1}^X\rangle = |\psi_j^X\rangle - 2 \sum_{z: X_z=1} \alpha_{z,j}^X |z\rangle |\phi_{z,j}^X\rangle,$$

and

$$|\psi_{j+1}^Y\rangle = |\psi_j^Y\rangle - 2 \sum_{z: Y_z=1} \alpha_{z,j}^Y |z\rangle |\phi_{z,j}^Y\rangle.$$

Note that unlike the Hybrid proof (see page 28), we do use here the specific form of the operators  $U_X, U_Y$ , although I suspect it can be avoided.

It follows that

$$\langle\psi_{j+1}^X|\psi_{j+1}^Y\rangle = \langle\psi_{j+1}^X|\psi_{j+1}^Y\rangle - 2 \sum_{z: X_z \neq Y_z} \alpha_{z,j}^X \alpha_{z,j}^Y \langle\phi_{z,j}^X|\phi_{z,j}^Y\rangle.$$

Thus it suffices to bound

$$2 \sum_{z: X_z \neq Y_z} |\alpha_{z,j}^X| |\alpha_{z,j}^Y| \leq 2\sqrt{|\mathcal{X}||\mathcal{Y}|bb'}.$$

We know that

$$2|\alpha_{z,j}^X| \cdot |\alpha_{z,j}^Y| \leq r|\alpha_{z,j}^X|^2 + \frac{1}{r}|\alpha_{z,j}^Y|^2,$$

thus providing

$$2 \sum_{z: X_z \neq Y_z} |\alpha_{z,j}^X| |\alpha_{z,j}^Y| \leq r \sum_{z: X_z \neq Y_z} |\alpha_{z,j}^X|^2 + \frac{1}{r} \sum_{z: X_z \neq Y_z} |\alpha_{z,j}^Y|^2 \leq rb|\mathcal{X}| + \frac{1}{r}b'|\mathcal{Y}|.$$

Finally choosing  $r = \sqrt{\frac{b'|\mathcal{Y}|}{b|\mathcal{X}|}}$  finishes the proof. □

□

## 6.4 Quantum Query Complexity and Formula Size

Let  $F : \{0, 1\}^N \rightarrow \{0, 1\}$  be a Boolean function. It is known that

$$L(F) \geq \Omega\left(\frac{|R|^2}{|\mathcal{X}||\mathcal{Y}|}\right), \quad (20)$$

where  $L(F)$  is the formula size of  $F$ , and  $X, Y$  and  $R$  are defined as previous. It is also easy to see that  $Q_2(F) \leq L(F)$  (by induction on  $L(F)$ ). Theorem 34 in the case when  $b = b' = 1$ , implies that

$$Q_2(F) \geq \Omega\left(\frac{|R|}{\sqrt{|\mathcal{X}||\mathcal{Y}|}}\right). \quad (21)$$

Inequalities (20) and (21) lead to the belief that  $Q_2(F) \leq O(L(F)^{1/2})$ . Grover's search algorithm corresponds to the case when  $F$  is a single OR function, therefore this conjecture can be viewed as a far far reaching generalization of his result.

In a recent breakthrough, Ambainis, et. al. [9] almost proved this by showing that

$$Q_2(F) \leq L(F)^{\frac{1}{2}+o(1)}.$$

## 7 Quantum Communication Complexity

Suppose we have two parties Alice and Bob and a Boolean function  $F : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$ . Consider a setting in which Alice has a Boolean string  $X \in \{0, 1\}^N$  and Bob has a Boolean string  $Y \in \{0, 1\}^N$ , and their goal is to compute the value of  $F(X, Y)$  by communicating as few bits as possible. Alice and Bob agree on a communication protocol beforehand. Having received inputs, they communicate in accordance with the protocol. At the end of the communication one of the parties declares the value of the function  $F$ . The cost of the protocol is the number of bits exchanged on the worst-case input.

**Definition 37.** The deterministic communication complexity of  $F$ , denoted by  $DC(F)$ , is the cost of an optimal communication protocol computing  $F$ .

The topic of classical communication complexity was introduced and first studied by Andrew Yao in 1979 [10]. The following is a simple observation which is implied by the definition of deterministic communication complexity.

**Observation 38.** Let  $F : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$  be a non-trivial Boolean function, meaning that it depends on all of the variables. Then we have

$$\log_2 N \leq C(F) \leq N.$$

**Definition 39.** For a Boolean function  $F : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$ , the communication matrix of  $F$  is a  $\{0, 1\}^N$  by  $\{0, 1\}^N$  matrix, denoted by  $M_F$ , where  $M_F(X, Y) = F(X, Y)$ .

We have the following rank lower bound for  $C(F)$  due to Mehlhorn and Schmidt [11].

**Theorem 40** (Mehlhorn and Schmidt [11]). *For any Boolean function  $F : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$  we have*

$$C(F) \geq \log_2 \text{rk}(M_F).$$

**Definition 41.** Define  $EQ_N : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$  to be the Boolean function where  $EQ_N(X, Y) = 1$  if  $X = Y$  and  $EQ_N(X, Y) = 0$  otherwise.

We have the following lower bound on the communication complexity of the  $EQ_N$  function immediately following from Theorem 40.

**Observation 42.**

$$C(EQ_N) \geq N.$$

## 7.1 Probabilistic Communication Complexity

For a Boolean function  $F : \{0, 1\}^N \times \{0, 1\}^N \rightarrow \{0, 1\}$ , let the two-way error probabilistic communication complexity of  $F$  be denoted by  $C_2(F)$ . Formally, it is defined similarly to  $D(F)$ , with the difference that the protocol is allowed to use random coins and is allowed to err with probability  $\leq 1/3$  for each input  $(X, Y)$ . Then we have the following upperbound on the probabilistic communication complexity of the function  $EQ_N$ .

**Theorem 43** (Rabin and Yao).

$$C_2(EQ_N) \leq O(\log N).$$

*Proof.* Let  $p \geq 3N$  be a prime number which is only slightly greater than  $3N$ . Let  $E$  be an encoding of Boolean strings by low-degree polynomials over  $\mathbb{F}_p$ . The exact choice of this encoding does not matter, so we simply let  $E(X) = \sum_{i=1}^N X_i \xi^i \in \mathbb{F}_p[\xi]$ . Consider the following probabilistic protocol:

1. Alice chooses  $z \in \mathbb{F}_p$  at random and sends  $(z, E(X)(z))$  to Bob, where  $E(X)(z) = \sum_{i=1}^N X_i z^i$ .
2. Bob checks if  $E(X)(z) = E(Y)(z)$ , and outputs 1 if and only if it is the case.

If  $X = Y$ , then Bob always computes the correct value of  $EQ(X, Y)$  at the last step. If  $X \neq Y$ , then  $E(X)$  and  $E(Y)$  differ for at least  $2p/3$  out of  $p$  possibilities since their difference is a non-zero polynomial of degree  $\leq N \leq p/3$  and thus can have at most  $p/3$  roots in  $\mathbb{F}_p$ . So the probability that  $E(X)$  and  $E(Y)$  differ in the  $z$ -th coordinate and hence Bob computes the value of  $EQ(X, Y)$  to Alice is at least  $\frac{2}{3}$ , as desired.

It is easy to see that the number of bits transmitted through this protocol is  $O(\log N)$ .  $\square$

## 7.2 Quantum Communication Complexity

Almost 15 years elapsed before the same pioneer, Andrew Yao, thought of asking how the situation of communication complexity might change in the quantum computation world [12].

Let  $\mathcal{X}$  and  $\mathcal{Y}$  be two sets, and  $F : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$  be a Boolean function. We will be working with  $\mathcal{H}_A \otimes \mathcal{C} \otimes \mathcal{H}_B$ , where  $\mathcal{H}_A, \mathcal{H}_B$ , and  $\mathcal{C}$  are Hilbert spaces, which represent Alice's work space, Bob's work space, and communication channel respectively. A quantum communication protocol is defined as follows:

$$(I_A \otimes U_{Y,t})(U_{X,t} \otimes I_B) \cdots (I_A \otimes U_{Y,1})(U_{X,1} \otimes I_B)|\phi_0\rangle, \quad (22)$$

where  $U_{X,i}$  are *arbitrary* unitary operators on  $\mathcal{H}_A \otimes \mathcal{C}$  that depend only on Alice's input  $X$ , and  $U_{Y,i}$  are described dually. The cost of the protocol is  $t \cdot \log_2 \dim(\mathcal{C})$ ; the idea behind this measure is that we have  $t$  rounds of communication, with  $\log_2 \dim(\mathcal{C})$  qubits sent in each of them. The quantum communication complexity of a function  $F$  (again, with error probability  $1/3$ ) is equal to the cost of the most efficient quantum protocol to compute  $F$  and is denoted by  $QC_2(F)$ .

There have been different models of quantum communication complexity. While almost all of them are essentially equivalent, one important distinction that does not have any analogue in the black-box model is that of *prior entanglement*, depending on whether the initial vector  $|\phi_0\rangle$  in (22) is arbitrarily entangled or simply has the form  $|0^a\rangle \otimes |0^b\rangle \otimes |0^c\rangle$ .

The straightforward analogue of Theorem 23 can not be true since Observation 42 and Theorem 43 already imply an exponential separation between  $C(F)$  and  $C_2(F)$ . Thus, a sensible thing to ask is if  $C_2(F)$  and  $QC_2(F)$  are polynomially related. For *partial* functions this is known to be not true [13], and for total functions this is a major open problem:

**Conjecture 44.**  $C_2(F) \leq QC_2(F)^{O(1)}$  for the class of all totally defined functions  $F$ .

In the rest of this block we will discuss this conjecture for a natural class of total functions  $F$  where the progress is being made, and that I generally feel to be more tractable than the general case.

**Definition 45.** For two functions  $F : \{0, 1\}^N \rightarrow \{0, 1\}$ , and  $g : \{0, 1\} \times \{0, 1\} \rightarrow \{0, 1\}$ , the function  $F \circ g^N$  is defined as follows

$$F \circ g^N := F(g(X_1, Y_1), g(X_2, Y_2), \dots, g(X_N, Y_N)).$$

The  $F \circ g^N$  functions are called *block-composed functions* and are widely studied. Many authors also consider even more general case, when each block consists not of a single (qu)bit, but of a constant number of them. For the purpose of our discussion, however, the one-qubit case suffices.

Following are well-studied examples of block-composed functions:

1.  $EQ(X, Y) = \bigwedge_z (X_z = Y_z)$ .
2.  $IP(X, Y) = \bigoplus_z (X_z \wedge Y_z)$ .

$$3. \text{ DISJ}(X, Y) = \bigvee_z (X_z \wedge Y_z).$$

The following simple theorem relates quantum communication complexity of block-composed functions to the quantum complexity measure  $Q_2$ .

**Theorem 46** (Buhrman, Cleve and Wigderson 98 [14]).

$$QC_2(F \circ g^N) \leq O(Q_2(F) \log N).$$

*Proof.* In this case we will have a communication channel of dimension  $O(N)$  (that is, representable by  $O(\log N)$  qubits), and only Alice will have a work space. We will use an efficient black box protocol to compute  $F$ , and during the process, for every query

$$U_g : |x, s, u\rangle \mapsto |x, s \oplus g(X_x, Y_x), u\rangle,$$

Alice will compute  $U_g$  in her work space after providing  $X_x$  to Bob and getting back  $g(X_x, Y_x)$  from Bob in the communication channel.

- $|x, s, u\rangle|0, 0, 0\rangle \mapsto$  (Alice)  $|x, s, u\rangle|X_x, x, 0\rangle$
- $|x, s, u\rangle|X_x, x, 0\rangle \mapsto$  (Bob)  $|x, s, u\rangle|X_x, x, g(X_x, Y_x)\rangle$
- $|x, s, u\rangle|X_x, x, g(X_x, Y_x)\rangle \mapsto$  (Alice)  $|x, s \oplus g(X_x, Y_x), u\rangle|X_x, x, g(X_x, Y_x)\rangle$
- $|x, s \oplus g(X_x, Y_x), u\rangle|X_x, x, g(X_x, Y_x)\rangle \mapsto |x, s \oplus g(X_x, Y_x), u\rangle|0, 0, 0\rangle$

Where the last step can be done by the Garbage Removal Lemma (Theorem 1). The cost of the protocol is  $O(Q_2(F) \log N)$ .  $\square$

The following conjecture (that we purposely state in a little bit loose form) states that we can not in fact do much better than that:

**Conjecture 47.** There is no better way to compute block-composed functions other than computing  $G$ 's in parallel and computing  $F$  of the outputs at the end.

Razborov confirmed the above conjecture for the case when  $F$  is a symmetric Boolean function [15]. Note that we also trivially have the classical analogue of Theorem 46:  $C_2(F \circ g^N) \leq O(R_2(F))$ , where  $R_2(F)$  is the *randomized* decision-tree complexity of the predicate  $F$ . Since  $R_2(F)$  and  $Q_2(F)$  are polynomially related by Theorem 23, Conjecture 47 does imply Conjecture 44 for block-composed functions.

## Lectures 14,15

Scribe: Pratik Worah, University of Chicago.

Date: 22, 23 February, 2011

Now we study a technique (discrepancy method) for lower bounds on  $QC_2$ . We conclude with a sketch of some ideas involved in more advanced proofs based on generalizing the discrepancy method.

Note that we will work under the context described in Section 7 where the input is simply  $|0^a, 0^b, 0^c\rangle$  and no prior entanglement is present<sup>1</sup>. The output is written to the channel in the end. As in Definition 39, given a function  $F : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$  denote by  $M_F$  ( $M_F(X, Y) = F(X, Y)$ ) the communication matrix of  $F$  in the quantum model<sup>2</sup>. More precisely,

**Definition 48.** The *quantum communication complexity with bounded-error probability* of a function  $F : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$  denoted by  $QC_2(F)$  is the cost of the most efficient quantum protocol (cf. Section 7) to compute  $F$  such that the probability of computing the correct answer is at least  $\frac{2}{3}$ .

We will denote by  $P_F$  the matrix of probabilities of acceptance for  $F$  (i.e.  $P_F(X, Y)$  is the probability that the protocol accepts). Then, in the matrix notation, our acceptance condition can be written as  $\ell_\infty(M_F - P_F) \leq \frac{1}{3}$ <sup>3</sup>

### 7.3 Decomposition of quantum protocols

We now assume that the channel has only 1 qubit. This simplifies the expressions in the next theorem, and it is well-known that it does not restrict the power of the model much. The following observation abstracts out the structure of  $P_F$ .

**Theorem 49.** *Given  $F : \mathcal{X} \times \mathcal{Y} \rightarrow \{0, 1\}$  and  $P_F$  as before*

$$P_F = \sum_{i=1}^{2^{2QC_2(F)}} R_i,$$

where  $R_i$  are real rank 1 matrices of the form  $R_i = C_i D_i^*$  (we write  $D^*$  instead of  $D^\dagger$  for real matrices) such that  $\ell_\infty(C), \ell_\infty(D) \leq 1$ .

---

<sup>1</sup>Most of the material, however, can be generalized to the case with prior entanglement as well – see [15, Remark 4] for details.

<sup>2</sup>Let  $|\mathcal{X}| = |\mathcal{Y}| = \mathcal{N}$ , and so  $N = \log_2 \mathcal{N}$ .

<sup>3</sup> $\ell_\infty(A) = \max_{i,j} |A(i, j)|$ .

*Proof.* We start by proving a lemma regarding the structure of the quantum state.

**Lemma 50.** *The final state of a  $t$  step quantum communication protocol (22) is expressible as*

$$\sum_{c \in \{0,1\}} \sum_{i=1}^{2^t} A_i(X) \otimes |c\rangle \otimes B_i(Y), \quad (23)$$

where  $A_i, B_i$  are (complex) vectors with  $\ell_2$  norm  $\leq 1$ .

*Proof.* The proof will be by induction on  $t$  - the length of the protocol. In the base case  $t = 0$  the vectors  $A_i, B_i$  are unit vectors so the lemma is true.

Suppose the lemma holds for  $t - 1$  steps and suppose that Alice applied  $(U_{X,t} \otimes I_B)$ . Since  $U_{X,t}$  is unitary it preserves  $\ell_2$  norm, so taking projections implies  $\exists A_{i0}, A_{i1}$  with  $\ell_2$  norm at most that of  $A_i$  such that

$$(U_{X,t} \otimes I_B)(|A_i\rangle|c\rangle)|B_i\rangle = (|A_{i0}\rangle|0\rangle + |A_{i1}\rangle|1\rangle)|B_i\rangle. \quad (24)$$

Therefore the number of terms in our sum increases by at most a factor of 2. Observe that this suffices to prove the lemma.  $\square$

Given Lemma 50, we can calculate  $P_F$  as follows (assuming  $|1\rangle$  is accepting).

$$P_F(X, Y) = \sum_{i,j=1}^{2^t} \langle A_i^\dagger(X), A_j(X) \rangle \langle B_i^\dagger(Y), B_j(Y) \rangle.$$

Therefore  $P_F$  has the form  $\sum_{i,j=1}^{2^t} C_{i,j} D_{i,j}^\dagger$ . Since  $\|A_i\|_2, \|B_i\|_2 \leq 1$  the corresponding vectors  $C, D$  will have  $\ell_\infty$  norm  $\leq 1$ . Moreover,  $\text{rk}(CD) \leq \min(\text{rk}(C), \text{rk}(D))$  implies  $\text{rk}(R_i) \leq 1$ . Hence the proof follows.  $\square$

As a corollary, we obtain the log rank bound for  $QC$  (zero-error version of  $QC_2$ ).

**Corollary 51.**  $QC(F) \geq \Omega(\log \text{rk}(M_F))$ .

In the following, we discuss methods for obtaining lower bounds on  $QC_2(F)$ .

## 7.4 Lower bound for $QC_2(IP_2)$

We now change  $M_F$  to a  $\pm 1$  valued matrix while keeping  $\ell_\infty(M_F - P_F)$  bounded by at most a small enough constant say  $\epsilon$ . This will also require replacing  $P_F$  with  $J_{\mathcal{N}} - 2P_F$ , where  $J_{\mathcal{N}}$  is an all-one matrix. But this linear transformation does not affect the validity of Theorem 49, up to a small multiplicative increase in the number of terms, and this is the only property of  $P_F$  we are going to use (in particular, we will not need that it is non-negative).

**Definition 52.** Define the *Frobenius product* of two matrices  $A$  and  $B$  by  $\langle A, B \rangle := \sum_{i,j} A_{ij}B_{ij}$ .

Let us write  $M_F$  as  $P_F + \Delta$  where  $\ell_\infty(\Delta) \leq \epsilon$ . Observe that

$$\mathcal{N}^2 = \langle M_F, M_F \rangle = \langle M_F, \Delta \rangle + \langle M_F, P_F \rangle. \quad (25)$$

Since the first term in the RHS is at most  $\epsilon\mathcal{N}^2$ , the second term has to be  $\Omega(\mathcal{N}^2)$ . Observation 49 gives (for  $QC_2(F) = k$ )

$$\langle M_F, P_F \rangle = \sum_{i=1}^{2^{2k}} \langle M_F, R_i \rangle = \sum_{i=1}^{2^{2k}} \langle M_F, C_i D_i^* \rangle. \quad (26)$$

Note that for a single term in the last sum

$$\langle M_F, CD^* \rangle = \sum_{j,k=1}^{\mathcal{N}} M_F(j,k)(CD^*)(j,k) = \sum_{j,k=1}^{\mathcal{N}} C_j M_F(j,k) D_k = C^* M_F D.$$

Since  $\ell_\infty(C), \ell_\infty(D) \leq 1$ , we have  $\|C\|, \|D\| \leq \sqrt{\mathcal{N}}$  (recall that  $\|\cdot\|$  stands for the  $\ell_2$ -norm). Therefore by definition of spectral norm (that we will also denote simply by  $\|\cdot\|$ ),  $\|C^* M_F D\| \leq \mathcal{N} \|M_F\|$ .

Hence returning to the original equation (26),

$$\sum_{i=1}^{2^{2k}} \langle M_F, R_i \rangle \leq \sum_{i=1}^{2^{2k}} \|C_i^* M D_i\| \leq 2^{2k} \mathcal{N} \|M_F\|.$$

Since, as we observed above,  $\langle M_F, P_F \rangle \geq \Omega(\mathcal{N}^2)$ , we conclude

$$2^{2k} \geq \Omega\left(\frac{\mathcal{N}}{\|M_F\|}\right). \quad (27)$$

The lower bound method above is known as the *discrepancy* method. In summary we showed that for a relation like (25) to hold with small  $\ell_\infty(\Delta)$ ,

$\langle M_F, P_F \rangle$  must be large. Using this fact and the properties of our quantum model we obtained the desired lower bound.

As an example consider the function  $F = IP_2$  i.e.  $F(X, Y) = \bigoplus_{z=1}^N (X_z \wedge Y_z)$ .

**Observation 53.**  $\|M_F\| = \sqrt{N}$ .

*Proof.* The inner product matrix  $M_F$  is an orthogonal matrix, up to a normalizing factor (specifically a Hadamard matrix). Therefore all its eigenvalues are  $\sqrt{N}$  in absolute value.  $\square$

The above discussion therefore implies:

**Theorem 54** ([16]).  $QC_2(IP_2) = \Omega(N)$ .

As an aside, the following is an open problem in this area (see [17, Section 8] for more details).

**Conjecture 55.** If  $F \in AC^0$  then  $\|M\|$  is large for any large submatrix  $M$  of  $M_F$ .

A consequence of this would be that the “naive” discrepancy bound (27) *provably* does not work for functions in  $AC^0$ . In the next two subsections we discuss its generalizations that can do the job.

## 7.5 Lower bound for $QC_2(DISJ)$

In this subsection the aim is to study lower bounds on  $QC_2(F \circ \wedge^N)$  for block-composed functions (cf. later half of Section 7) with symmetric  $F$ .

Tight estimates of the approximate degree (see (30)) of symmetric boolean functions were obtained by Paturi [18] in terms of  $\Gamma$ , a quantity which depends on whether  $F$  changes values near  $\frac{N}{2}$  or far from  $\frac{N}{2}$ . For brevity, we identify  $F$  with its univariate representation  $[0, 1 \dots, N] \rightarrow \{0, 1\}$  (cf. Lemma 29).

**Definition 56.**  $\Gamma_0(F)$  and  $\Gamma_1(F)$  are defined as follows:

$$\Gamma_0(F) := \max\{k \mid 1 \leq k \leq \frac{N}{2}, F(k) \neq F(k-1)\}$$

$$\Gamma_1(F) := \max\{n-k \mid \frac{N}{2} \leq k \leq N, F(k) \neq F(k+1)\}.$$

Razborov [15] proved the following bound for symmetric  $F$ .

**Theorem 57.**  $QC_2(F \circ \wedge^N) = \tilde{\Theta}(\sqrt{N}\Gamma_0(F) + \Gamma_1(F))$

In case of  $F$  being the disjointness predicate (i.e.  $F = \vee$ ) we have  $\Gamma_0 = 1$  and  $\Gamma_1 = 0$  so  $QC_2(DISJ) = \Omega(\sqrt{N})$  [15] (upto a logarithmic factor). The following discussion briefly gives the ideas involved in this proof.

**Definition 58.** The *trace norm* of a real symmetric matrix  $A$  is defined as  $\|A\|_{tr} = \sum_{i=1}^n |\lambda_i(A)|$ .

This can also be defined for general matrices; one would replace eigenvalues by singular values. But the alternative characterization of the above definition given below covers this case as well.

**Observation 59.** For an arbitrary matrix  $A$

$$\|A\|_{tr} = \max_B \{\langle A, B \rangle \mid \|B\| = 1\}.$$

*Proof.* (for symmetric matrices) Since trace of a matrix (denoted  $Tr$ ) and spectral norm are invariant under conjugate transforms and since  $\langle A, B \rangle = Tr(AB^*)$  we can diagonalize  $A$  (which is symmetric) to obtain

$$\langle A, B \rangle = Tr \begin{pmatrix} \lambda_1(A)B_{11} & \cdots & 0 \\ \vdots & \cdots & \vdots \\ 0 & \cdots & \lambda_N(A)B_{NN} \end{pmatrix}.$$

Now  $\|B\| \leq 1$  implies  $|B_{ii}| \leq 1$  in any orthogonal basis, in particular in the one chosen above that diagonalizes  $A$ . Therefore by Definition 58, LHS  $\geq$  RHS in the statement above. Note that LHS=RHS when  $B$  is a diagonal matrix (in our basis) with non-zero entries appropriately chosen from the set  $\{\pm 1\}$ . Hence the proof follows.  $\square$

As a by-side remark (included mostly for educational purposes), this is a partial case of the following general paradigm.

**Definition 60.** The *dual norm* denoted  $\|\cdot\|_*$  of a norm  $\|\cdot\|$  is defined as

$$\|A\|_* = \sup\{A^*B \mid \|B\| \leq 1\}.$$

The observation above implies that the spectral norm and trace norm are dual norms. In general  $Tr(A^*B) \leq \|A\|_* \|B\|$  so in particular we have

$$\langle M_F, P_F \rangle \leq \|P_F\|_{tr} \|M_F\|.$$

Hence if we upper bound the trace norm of  $P$  and spectral norm of  $F$  then we can derive a contradiction to the decomposition in (25).

However, instead of working directly with the trace norm, [15] introduced the *approximate trace norm*

$$\|A\|_{\tilde{tr}} := \min\{\|B\|_{tr} \mid \ell_\infty(A - B) \leq \epsilon\}.$$

The following is not hard to prove.

**Theorem 61** ([15]).

$$QC_2(F) = \Omega\left(\log \frac{\|M_F\|_{\tilde{tr}}}{\mathcal{N}}\right).$$

All that remains is to develop methods for bounding  $\|M_F\|_{\tilde{tr}}$  from below, and this innocent-looking task turned out to be rather difficult.

## 7.6 Generalizations of the discrepancy method

These ideas were gradually developed and used in [12, 16, 19]; the exposition below follows [15, Section 5.2].

Let  $\mu$  be a  $\mathcal{N} \times \mathcal{N}$  matrix such that

$$\langle M_F, \mu \rangle = \langle P, \mu \rangle + \langle \Delta, \mu \rangle$$

and  $\langle \Delta, \mu \rangle \leq l_\infty(\Delta)l_1(\mu) \leq c\mathcal{N}^2$ . Earlier we had  $\mu = M_F$  in the normal discrepancy method (so that  $\ell_1(M_F) = \mathcal{N}^2$ ), but now we are free to choose  $\mu$  subject to the following constraints (we normalize by a factor of  $\mathcal{N}^2$ ):

1.  $l_1(\mu) \leq 1$ .
2.  $\langle M_F, \mu \rangle \geq \frac{2}{3}$ .
3.  $\|\mu\|$  is as small as possible.

[15] erroneously claimed that no such  $\mu$  can exist when  $F = DISJ$  and developed instead another method of “multi-dimensional” discrepancy. We can not go into much details here, but the general idea is to test  $M_F$  not against a single matrix  $\mu$ , but against a whole (finite) family of such matrices.

Using his *pattern matrix method*, Sherstov [20] showed that in fact a single  $\mu$  with the desired properties exist, that resulted in another proof of Theorem 57. While simpler than the original one, it is still too complicated to be included here.

Note that even more proof methods using different norms with desirable properties are known. Linial and Shraibman [21] use the norm  $\gamma_2$  defined as follows.

**Definition 62.** Given matrix  $A$ , let

$$\gamma_2(A) = \min_{XY=A} \left( \max_{\|x\|_2=1} \ell_\infty(Xx) \cdot \max_{\|y\|_1=1} \|Yy\|_2 \right).$$

It can be shown that  $\gamma_2(A) \geq \|A\|_{tr}$  (therefore it may be easier to lower bound than the trace norm) but  $\gamma_2$  is not invariant under conjugation. [21] use  $\gamma_2$  norm (and its many variants) to obtain quantum communication complexity lower bounds (including weaker lower bounds for *DISJ*).

## 7.7 Direct products

Sometimes it is possible to save resources by solving many instances of a problem together as opposed to solving each instance naively. Multiplying two  $n \times n$  matrices provides an example - a matrix-vector multiplication takes  $n^2$  operations, and thus one might expect that multiplying a matrix by  $n$  independent vectors should take  $n^3$  operations. However, using fast matrix multiplication algorithms [22] one can solve the same problem in  $o(n^3)$  operations. Of course most of the times one is not so lucky and then it is a challenge to prove that naive solving of the separate instances is the best that can be done (cf. Conjecture 47). Such theorems are known as *direct product theorems*. The following is such an open question about  $QC_2$ .

**Question 63.** Given  $F_i : \mathcal{X}_i \times \mathcal{Y}_i \rightarrow \{0, 1\}$  for  $i = 1, \dots, t$  define in natural manner the function  $(F_1 \times \dots \times F_t)(\mathcal{X}_1 \times \dots \times \mathcal{Y}_t) \rightarrow \{0, 1\}^t$ . Is  $QC_2(F_1 \times \dots \times F_t) \simeq \sum_{i=1}^t QC_2(F_i)$ ?

Note that a similar direct product result is known for the  $\gamma_2$  norm [23].

## Lectures 16, 17

Scribe: Youlian Simidjyski, University of Chicago.

Date: Feb 24 2011, Mar 1 22011

## 8 Quantum Error-Correcting Codes

Goal: Given a single qubit, we wish to preserve it in the presence of noise. We introduce noise into our system by applying various superoperators to the density matrix corresponding to our input. Three such superoperators corresponding to particular noisy channels are given below (here  $\eta > 0$  is a real parameter called *noise rate*).

**Depolarizing channel**  $\mathcal{E}_\eta(\rho) = (1 - \eta)\rho + \frac{\eta}{2}I_2$ .

**Bit flip channel**  $\mathcal{E}_\eta(\rho) = (1 - \eta)\rho + \eta \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \rho \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ .

**Phase shift channel**  $\mathcal{E}_\eta(\rho) = (1 - \eta)\rho + \eta \begin{pmatrix} 1 & 0 \\ 0 & \alpha \end{pmatrix} \rho \begin{pmatrix} 1 & 0 \\ 0 & \alpha^* \end{pmatrix}$  for some  $\alpha \in S^1$ . We will be using the case where  $\alpha = -1$ , so that this becomes the phase flip channel.

## 8.1 Operator-sum representation of superoperators

Every physically realizable superoperator has the normal form:

$$T(\rho) = \sum_k E_k \rho E_k^\dagger \text{ satisfying } \sum_k E_k^\dagger E_k = I$$

(the three noisy channels considered above make a very good example). Operators  $\{E_k\}$  are called *operation elements*.

The condition that  $\sum_k E_k^\dagger E_k = I$  stems from our earlier requirements on superoperators. Namely from the requirement that  $\forall \rho, \text{tr}(\rho) = \text{tr}(T(\rho))$ . This requirement implies

$$\text{tr}(T(\rho)) = \text{tr}\left(\sum_k E_k \rho E_k^\dagger\right) = \sum_k \text{tr}(E_k \rho E_k^\dagger) = \sum_k \text{tr}(E_k^\dagger E_k \rho) = \text{tr}\left(\sum_k (E_k^\dagger E_k) \rho\right),$$

and  $\text{tr}(\rho) = \text{tr}(\sum_k (E_k^\dagger E_k) \rho)$  holds for all  $\rho$  if and only if  $\sum_k E_k^\dagger E_k = I$ .

## 8.2 Projective Measurements

Given a Hilbert space  $\mathcal{H} = \mathcal{C}_1 \oplus \mathcal{C}_2 \oplus \dots \oplus \mathcal{C}_l$ , consider projections  $P_1, P_2, \dots, P_l$  onto  $\mathcal{C}_i$ . Projections satisfy the properties

1.  $P_i^2 = P_i = P_i P_i^\dagger$
2.  $\sum_k P_k^\dagger P_k = \sum_k P_k = I$ .

Thus, we have a superoperator mapping  $\rho \rightarrow \sum_k P_k^\dagger \rho P_k$ , which is called *projective measurement*.

### 8.3 Quantum Information Theory

**Definition 64.** Given a state  $\rho$  and a noisy channel  $\mathcal{E}_\eta$ , a *Recovery Operator*,  $\mathcal{R}$ , is a superoperator satisfying  $\mathcal{R}(\mathcal{E}_\eta(\rho)) = \rho$  for all  $\rho$ .

Given a general noisy channel (that is, an arbitrary superoperator), it is not possible to find such an  $\mathcal{R}$ . One good reason is that the trace distance inequality (18) on the noise operator yields  $D(\mathcal{E}_\eta(\rho), \mathcal{E}_\eta(\rho')) \leq D(\rho, \rho')$ , and if the inequality is strict, then inversion is not possible by a superoperator. However, we can hope to recover against specific classes of errors.

#### 8.3.1 Error Correcting Codes in Classical Information Theory

**Repetition codes:** Suppose that a bit  $x$  gets flipped during transmission with probability  $p < .5$ , independently of all others. If we first map  $x \rightarrow (x, x, x)$  before transmission, and we reconstruct by majority vote, then the probability of failed reconstruction is  $3p^2 - 2p^3 (\ll p)$ . This cannot be trivially implemented in quantum computation due to the no cloning theorem (There is no superoperator  $T$  which sends  $|\phi\rangle\langle\phi|$  to  $|\phi\rangle\langle\phi| \otimes |\phi\rangle\langle\phi|$ . One of the many possible proofs is based on Theorem 17).

Most codes in classical information theory are linear codes. The setup for such schemes is as follows. Let  $n > m$ , and consider  $\mathcal{C} \subset \{0, 1\}^n$ . We encode  $\{0, 1\}^m$  in  $\mathcal{C}$  by an injective mapping, typically including introducing some redundancy for decoding. We use a similar scheme for the quantum case.

Let  $\mathcal{C} \subset \mathcal{H}$  be defined over a field of characteristic 0, and let  $\dim(\mathcal{C}) = 2$ , since we are only interested in single qubits. We will encode  $\rho$  in  $\mathcal{C}$ .

#### 8.3.2 Correcting Against Quantum Bit Flip

Suppose that we are given a pure state on one qubit,  $\phi = a|0\rangle + b|1\rangle$ . We first map  $\phi$  to a pure state on three qubits by sending  $\phi \rightarrow a|000\rangle + b|111\rangle$ . Note that since these states are entangled, we have not violated the no cloning theorem. We now send each qubit across the channel  $\mathcal{E}$ . This yields a mixed state,  $\rho$ , because of the possibility of bit flips.

Consider now the projection operators  $P_0, P_1, P_2$ , and  $P_3$ , which project onto the subspaces in  $\mathcal{H}$  that are generated by  $(|000\rangle, |111\rangle)$ ,  $(|100\rangle, |011\rangle)$ ,  $(|010\rangle, |101\rangle)$ , and  $(|001\rangle, |110\rangle)$  respectively. Then the map  $\rho \rightarrow \sum_{k=0}^3 (U_k P_k \rho P_k^\dagger U_k^\dagger)$  (where  $U_k$  is the bit flip operator corresponding to the error detected by a

given  $P_k$ ) defines a superoperator which decodes  $\rho$  with probability  $3p^2 - 2p^3$  as in the classical case.

Physically, we have applied the projective measurement (see Section 8.2) followed by taking so-called *syndrome* represented in our case by the operator  $U_k$  (Note that the whole point of this procedure is that the operators  $U_k$  are *different* for different subspaces). For these reasons, this decoding procedure is sometimes called *syndrome measurement*. We will see in the proof of Theorem 66 that it is actually universal.

### 8.3.3 Correcting Against Quantum Phase Flip

Once we can correct against quantum bit flip, correcting against quantum phase flip is easy by first transforming into the Hadamard basis, given by  $(\frac{1}{\sqrt{2}})(|0\rangle + |1\rangle)$  and  $(\frac{1}{\sqrt{2}})(|0\rangle - |1\rangle)$ . Phase shift in the standard basis is bit flip in the Hadamard basis, so there is no work to be done.

### 8.3.4 Correcting Against Simultaneous Bit and Phase Flip

With the bit flip and phase flip correcting techniques in hand, composing the two codes immediately yields a 9-qubit code that prevents against a channel that could perform both bit flips and phase flips. In addition to correcting bit and phase flip errors, Shor's code actually corrects against arbitrary errors on a single qubit. For more information about this particular code, consult [1, Section 10.2].

We will finish this course with developing a *general* mathematical technique (called *discretization of errors*) that allows us to reduce error-correcting of a huge (typically continuous) class of errors to correcting a finite set of errors, in many cases possessing a nice structure. Unfortunately, we do not have time to talk about *stabilizer codes* that make one of the most beautiful applications of this theory, please see [1, Section 10.5] for details.

## 8.4 Properties of the Recovery Operator

**Question 65.** *What are necessary and sufficient conditions for the existence of a recovery operator  $\mathcal{R}$ ?*

We will ultimately show that given a channel  $\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger$  and an encoding scheme mapping our input space onto  $\mathcal{C} \subset \mathcal{H}$  as above, then

**Theorem 66.** *If  $P$  is the projection operator onto  $\mathcal{C}$ , then a recovery operator,  $\mathcal{R}$ , exists iff  $PE_i^\dagger E_j P = \alpha_{ij} P$  for some Hermitian matrix with entries  $\alpha_{ij}$ .*

A consequence of this theorem is that once we have been able to recover against this particular channel  $\mathcal{E}$ , we will be able to recover also against any other channel whose operation elements are linear combinations of  $E_k$ . We again refer to [1, Section 10.3.1] for further details.

To show Theorem 66, we will require one additional theorem and the polar decomposition lemma. We give the theorem first.

**Theorem 67** (Unitary Invariants). *Two superoperators  $\rho \rightarrow \sum_{k=1}^n E_k \rho E_k^\dagger$  and  $\rho \rightarrow \sum_{l=1}^n F_l \rho F_l^\dagger$  in operator-sum form are equal iff there exists a unitary  $U$  such that  $E_k = \sum_l u_{kl} F_l$ .*

We will give one direction of the proof here, as this direction is all that we will make use of. For the other direction, see [1, Theorem 8.2].

*Proof.* Given  $U$  as above, then

$$\begin{aligned} \sum_l E_l \rho E_l^\dagger &= \sum_{k,l} u_{k,l} F_l \rho F_l^\dagger u_{k,l}^* \\ &= \sum_l F_l \rho F_l^\dagger \sum_k u_{k,l} u_{k,l}^* \\ &= \sum_l F_l \rho F_l^\dagger \end{aligned}$$

where  $u_{k,l}^*$ ,  $l$  is term  $k, l$  in  $U^\dagger$ , and  $\sum_k u_{k,l} u_{k,l}^* = 1$  because  $U$  is unitary.  $\square$

Let  $A$  be any square matrix. Consider  $A^\dagger A$ . This is clearly Hermitian, positive-semidefinite, and so we can diagonalize  $A^\dagger A$  and calculate  $\sqrt{A^\dagger A}$  writing

$$\sqrt{A^\dagger A} = \begin{pmatrix} \sqrt{\lambda_1} & 0 & \cdots & 0 \\ 0 & \sqrt{\lambda_2} & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & \sqrt{\lambda_N} \end{pmatrix}.$$

**Lemma 68** (Polar Decomposition). *Given a matrix  $A$ , there exists a unitary  $U$  such that  $A = U \sqrt{A^\dagger A}$ . Thus, any matrix can be decomposed as the product of a hermitian matrix and a unitary matrix.*

*Proof.* A proof of the Polar Decomposition Lemma can be found in a standard linear algebra text, or in [1, Theorem 2.3].  $\square$

*Proof of Theorem 66.* We will show sufficiency of the conditions. To see that they are also necessary, consult [1].

First apply Theorem 67 in order to see that we can choose an equivalent set of operation elements so that the matrix  $\alpha$  is diagonal. We will assume that  $\alpha$  is diagonal for the remainder of the proof.

Now observe that by the polar decomposition,  $E_k P = U_k \sqrt{PE_k^\dagger E_k P} = \sqrt{\alpha_{k,k}} U_k P$ , where we have applied the fact that  $P^2 = P$ .

Now consider  $P_k = U_k P U_k^\dagger$ , the projection onto  $U_k \mathcal{C}$ , the image subspace of  $E_k P$ . Using the fact that the matrix  $\alpha$  is diagonal, we can conclude that  $U_l \mathcal{C}$  and  $U_k \mathcal{C}$  are pairwise orthogonal, as  $P_k P_l = U_k P U_k^\dagger U_l P U_l^\dagger \approx U_k (P E_k^\dagger E_l P) U_l^\dagger = 0$ . So the image spaces of  $P_k$  and  $P_l$  are orthogonal because  $P_k$  and  $P_l$  are projectors of the same dimension, and so must have either identical image, or the image of one must be the subset of the image of the other.

Thus, we can decompose  $\mathcal{E}$  into  $\sum_k P_k + Q$ , where the  $Q$  portion projects onto parts of  $\mathcal{H}$  that  $\mathcal{C}$  does not map into under the error map  $\mathcal{E}$ .

We can now recover our original  $\rho$  by the syndrome measurement using operators  $P_k \rho$  and  $U_k^\dagger$ . Mathematically:

$$\begin{aligned}
\mathcal{R}(\mathcal{E}(\rho)) &= \sum_{j,k} U_k^\dagger P_k E_j P \rho P E_j^\dagger P_k U_k \\
&= \sum_{j,k} U_k^\dagger (U_k P U_k^\dagger) E_j P \rho P E_j^\dagger (U_k P U_k^\dagger) U_k \\
&= \sum_{j,k} P U_k^\dagger E_j P \rho P E_j^\dagger U_k P \\
&= \frac{\sum_{j,k} (P E_k^\dagger E_j P) \rho (P E_j^\dagger E_k P)}{\alpha_{k,k}} \\
&= \frac{\sum_{j,k} \alpha_{k,k} P \rho \alpha_{k,k} P}{\sqrt{\alpha_{k,k}}} \\
&= \sum_k \alpha_{k,k} P \rho P
\end{aligned}$$

and since  $\rho \in \mathcal{C}$ , and  $P$  is the projector onto  $\mathcal{C}$ , we reach our conclusion that  $\mathcal{R}(\mathcal{E}(\rho)) \approx \rho$ . This completes the proof that our conditions are sufficient for  $\mathcal{R}$  to exist.  $\square$

## References

- [1] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.
- [2] A. Kitaev, A. Shen, and M. Vyalyi. *Classical and quantum computation*. American Math. Society, 2002. Extended version of a book originally published in Russian.
- [3] R. Bhatia. *Matrix Analysis. Graduate texts in mathematics, 169*. Springer-Verlag, 1997.
- [4] P. Kaye, R. Laflamme, and M. Mosca. *An Introduction to Quantum Computing*. Oxford University Press, 2007.
- [5] Ajtai M and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the 29th ACM STOC*, pages 284–293, 1997.
- [6] O. Regev. Quantum computation and lattice problems. *SIAM Journal on Computing*, 33(3):738–760, 2004.
- [7] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288:21–43, 2002.
- [8] R. Kulkarni P. Hatami and D. Pankratov. Variations on the sensitivity conjecture. Manuscript, available at <http://arxiv.org/abs/1011.0354>, 2010.
- [9] Andris Ambainis, Andrew M. Childs, Ben W. Reichardt, Robert Spalek, and Shengyu Zhang. Any and-or formula of size  $n$  can be evaluated in time  $n^{1/2+o(1)}$  on a quantum computer. *Foundations of Computer Science, Annual IEEE Symposium on*, 0, 2007.
- [10] A. Yao. Some complexity questions related to distributive computing. In *Proceedings of the 11th ACM Symposium on the Theory of Computing*, pages 209–213, New York, 1979. ACM Press.
- [11] K. Mehlhorn and E. M. Schmidt. Las Vegas is better than determinism in VLSI and distributive computing. In *Proceedings of the 14th ACM Symposium on the Theory of Computing*, pages 330–337, New York, 1982. ACM Press.

- [12] A. Yao. Quantum circuit complexity. In *Proceedings of the 34th IEEE Symposium on Foundations of Computer Science*, pages 352–361, Los Alamitos, 1993. IEEE Computer Society.
- [13] R. Raz. Exponential separation of quantum and classical communication complexity. In *Proceedings of the 31st ACM STOC*, pages 358–367, 1999.
- [14] H. Buhrman, R. Cleve, and A. Wigderson. Quantum vs. classical communication and computation. In *Proceedings of the 30th ACM Symposium on the Theory of Computing*, pages 63–86, New York, 1998. ACM Press. Preliminary version available at [quant-ph/9802040](#).
- [15] A. Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya: Mathematics*, 67(1):145–159, 2003.
- [16] I. Kremer. Quantum communication. Master’s thesis, Hebrew University, Jerusalem, 1995.
- [17] A. Razborov and A. Sherstov. The sign-rank of  $AC^0$ . *SIAM Journal on Computing*, 39(5):1833–1855, 2010.
- [18] R. Paturi. On the degree of polynomials that approximate symmetric Boolean functions. In *Proceedings of the 24th ACM Symposium on the Theory of Computing*, pages 468–474, New York, 1992. ACM Press.
- [19] H. Klauck. Lower bounds for quantum communication complexity. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pages 288–297, Los Alamitos, 2001. IEEE Computer Society. Preliminary version available at [quant-ph/0106160](#).
- [20] A. A. Sherstov. The pattern matrix method for lower bounds on quantum communication. In *STOC*, pages 85–94, 2008.
- [21] N. Linial and A. Shraibman. Lower bounds in communication complexity based on factorization norms. *Random Struct. Algorithms*, 34(3):368–394, 2009.
- [22] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symb. Comput.*, 9(3):251–280, 1990.
- [23] Troy Lee, Adi Shraibman, and Robert Spalek. A direct product theorem for discrepancy. In *IEEE Conference on Computational Complexity*, pages 71–80, 2008.